ADACELL: POPULATION-LEVEL STEERING FOR SINGLE-CELL PERTURBATION PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Single-cell perturbation prediction aims to estimate how biological interventions affect cellular states, which is essential for prioritizing targets and compounds in drug discovery and functional genomics. Existing diffusion-based virtual cell models can capture complex and heterogeneous response distributions, but fitting the overall perturbed distribution does not necessarily ensure accurate recovery of the perturbation response relative to matched controls. A prediction may reproduce broad expression patterns while misestimating which genes respond, the strength and direction of their responses, or the fraction of responding cells. To address these limitations, we propose **AdaCell**, a population-level test-time steering framework for diffusion-based perturbation prediction. Specifically, AdaCell represents each sampling particle as an entire candidate cell population and jointly evaluates its change relative to the corresponding control using criteria that capture differential expression programs, as well as the direction and geometry of population shifts. To handle unseen perturbations, AdaCell estimates their response priors from observed training perturbations and perturbation descriptors available before the experiment, without accessing any ground-truth responses from the test split. Finally, population-level reweighting and resampling steer reverse diffusion toward candidate populations that better match these priors, while the frozen diffusion model anchors the search to its learned distribution. Extensive experiments on representative single-cell perturbation benchmarks demonstrate that AdaCell improves the recovery of perturbation-specific population changes over frozen diffusion backbones and competitive test-time baselines while preserving cellular realism and population diversity. Our code is available at our website.

1 INTRODUCTION

In drug discovery and functional genomics, identifying the most promising genetic targets and candidate compounds for experimental validation from vast candidate spaces remains a major challenge (Bender et al., 2026; Park et al., 2025). Virtual cell models help address this challenge by **predicting population-level responses** to these interventions before experiments are performed, thereby narrowing the experimental search space and guiding candidate selection (Bunne et al., 2024; Roohani et al., 2025). These models learn from single-cell perturbation screens, which apply genetic or chemical interventions to cell populations and measure the resulting molecular states (Song et al., 2025). However, most single-cell measurements are destructive, preventing the same cell from being measured both before and after an intervention (Bunne et al., 2023). Models therefore learn from separately sampled control and perturbed populations rather than paired cellular trajectories. Accordingly, single-cell perturbation prediction is fundamentally a **population-level task**: learning from unpaired observations to predict how a specific intervention reshapes the distribution of cellular states relative to the control population.

Existing methods approach this population-level prediction problem from different perspectives, including representation learning, distributional transport, and conditional generation (Lotfollahi et al., 2023; Bunne et al., 2023; Klein et al., 2025). Among these approaches, diffusion models are particularly promising because iterative denoising provides a flexible way to capture nonlinear gene-expression patterns and heterogeneous cellular responses (He et al., 2026; Yuan et al., 2026). Recent models such as Squidiff and PerturbDiff leverage this capacity to predict high-dimensional transcriptomic states, with PerturbDiff further modeling distributions over cell populations con-

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

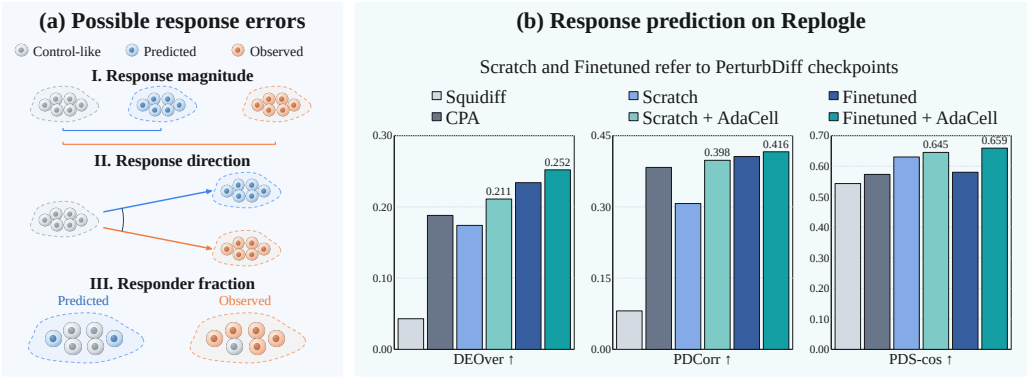

Figure 1: Motivation for population-level steering. (a) Schematic errors in response magnitude, direction, and responder fraction. (b) Response prediction on Replogle, comparing selected baselines and PerturbDiff checkpoints with and without AdaCell. Higher values indicate better performance.

ditioned on matched controls. These developments establish a strong foundation for generating heterogeneous perturbed populations. Using controls and modeling population distributions, however, are distinct from explicitly evaluating which candidate populations best recover biologically informative responses relative to those controls. This distinction motivates examining not only the overall quality of generated populations, but also the accuracy of their perturbation responses.

In practice, accurately fitting the overall perturbed distribution does not ensure the accurate recovery of the change caused by a perturbation relative to control (Viñas Torné et al., 2025). Statistically, a perturbation effect is defined as the contrast between the perturbed and control distributions, and exact recovery of these two distributions would imply exact recovery of the perturbation effect (Dong et al., 2023; Viñas Torné et al., 2025). However, standard diffusion training minimizes a denoising loss over the full distribution (Ho et al., 2020), but does not directly constrain the intervention contrast. Weak intervention signals may therefore be obscured by cellular heterogeneity, measurement noise, or model error, causing models with similar overall fits to predict different response magnitudes, directions, or population compositions. Recent benchmarks reinforce this concern. Simple mean or linear baselines can match or outperform complex models, while common metrics are sensitive to systematic variation (Ahlmann-Eltze et al., 2025; Viñas Torné et al., 2025). Overall prediction quality, therefore, cannot substitute for the direct evaluation of intervention effects. Broadly, existing approaches to generative alignment follow two routes. One updates model parameters through additional training objectives or reinforcement learning with evaluator feedback (Black et al., 2024; Wu et al., 2026). The other keeps the generative model fixed and adjusts its outputs during inference through guidance, candidate selection, or sequential sampling (Bansal et al., 2024; Kim et al., 2025). Yet key aspects of single-cell perturbation responses, including differential expression programs, the direction of population shifts, response heterogeneity, and the fraction of responding cells, can only be assessed by comparing the complete candidate population with its control population. How to use these population-level signals to steer generation at test time, without retraining the model or allowing the generated population to drift away from the biological distribution learned by the base model, remains largely unexplored.

Thus, a good fit to the overall perturbed distribution may still miss subtle changes relative to control (Viñas Torné et al., 2025). Standard diffusion training minimizes a denoising loss over the full distribution, without explicitly prioritizing these changes (Ho et al., 2020). Consequently, weak perturbation responses may be masked by cellular heterogeneity, noise, or model error. As illustrated schematically in Figure 1(a), models with similar overall fits may nevertheless misestimate response magnitude, shift direction, or the fraction of responding cells. Test-time alignment can guide generation without updating the base model (Bansal et al., 2024; Kim et al., 2025). For perturbation prediction, key response properties, such as differential expression and the direction of population shifts, require **population-level evaluation** relative to controls. This calls for jointly evaluating candidate cell populations rather than scoring cells independently. The challenge is to use these

population-level signals to steer diffusion sampling while remaining consistent with the distribution learned by the frozen model.

To address this gap, we introduce AdaCell, a population-level framework that steers diffusion-based virtual cells at test time without updating the underlying model. Each particle represents a complete candidate cell population, which is evaluated and selected as a unit. As outlined in Figure 2, AdaCell estimates response priors from training data, evaluates candidate populations relative to matched controls, and uses the resulting rewards to guide reweighting and resampling during denoising. The rewards assess expression-signature agreement, response-direction alignment, and distributional consistency with shifted controls. For unseen perturbations, the response priors are predicted from descriptors available before the experiment, without accessing test responses. The frozen diffusion model remains the source of candidate trajectories throughout this process. Figure 1(b) illustrates the resulting improvements on Replogle (Replogle et al., 2022): AdaCell improves DEOver, PDCorr, and PDS-cos for both the Scratch and Finetuned PerturbDiff checkpoints. We further evaluate response errors, population diversity, and computational cost to characterize the effects of steering beyond these aggregate scores. Our main contributions are as follows:

- We identify perturbation contrast under-specification as a practical failure mode of diffusion-based virtual cells, showing that accurate overall generation does not necessarily ensure the recovery of perturbation-specific population changes.
- We propose AdaCell, a training-free population-level test-time steering framework that jointly evaluates complete candidate cell populations using biological criteria and aligns diffusion sampling without accessing ground-truth responses for test perturbations.
- Extensive experiments across diverse perturbation settings show that AdaCell improves perturbation response fidelity while preserving cellular realism and population diversity.

2 Related Works

Perturbation Modeling and Prediction. Perturbation models use biological structure and cellular context to infer beyond observed conditions (Lotfollahi et al., 2023; Roohani et al., 2024; Adduri et al., 2025). Importantly, population modeling is already crucial to this topic. Transport and flow methods learn transformations between cell populations (Bunne et al., 2023; Klein et al., 2025), while diffusion approaches range from generating conditional cell states to modeling response distributions as random variables (He et al., 2026; Yuan et al., 2026). A remarkable question is whether the predicted changes capture the intended perturbation effects. CINEMA-OT addresses treatment-effect estimation by separating confounding variation from treatment-associated variation before constructing counterfactual cell matches (Dong et al., 2023). Systema addresses evaluation by using alternative references, including the perturbed centroid, to distinguish perturbation-specific effects from systematic shifts shared across perturbations (Viñas Torné et al., 2025). Critical benchmarks further establish additive and linear models as important comparisons for genetic perturbation prediction (Ahlmann-Eltze et al., 2025). These studies motivate careful effect estimation but are distinct from adapting a generator during sampling. AdaCell complements this problem by steering complete candidate populations using response priors derived from the training dataset.

Diffusion Alignment and Steering. Generative alignment introduces downstream objectives that are not originally emphasized by the training loss. DDPO treats denoising as a sequential decision process and updates diffusion models through policy optimization (Black et al., 2024). In perturbation biology, PerturbCellRL post-trains a pretrained flow-matching generator using biological verifiers defined at the cell level (Wu et al., 2026). An alternative is to retain the generator and modify its sampling procedure. Universal Guidance applies external guidance functions to denoised predictions without retraining the diffusion model (Bansal et al., 2024). DAS formulates alignment as sampling from a reward-tilted base distribution, using sequential Monte Carlo and tempering to balance reward optimization with diversity and generalization (Kim et al., 2025). These methods provide general steering mechanisms without specifying how to evaluate a biological response jointly across cells and relative to controls. Moreover, a collection of sampling particles is not the same as a biological cell population. In AdaCell, each particle represents an entire empirical cell population, and its evaluation depends jointly on the generated cells and the corresponding control population rather than on independent cell scores. AdaCell therefore builds on established reweighting and resampling tools while defining the inference object and biological criteria at the population level.

3 PRELIMINARIES

3.1 SINGLE-CELL PERTURBATION PREDICTION

Let $\mathcal{D}_{\mathrm{tr}} = \{(x_j, a_j, h_j)\}$ denote training observations, where $x_j \in \mathbb{R}^d$ contains expression levels for d selected genes, a_j identifies a perturbation or control condition, and h_j denotes the cellular and experimental context. Given a perturbation a and control population $C = \{c_i\}_{i=1}^m$ from a target context h, the task is to predict a response population $B = \{b_i\}_{i=1}^m$. Control and response observations are unpaired, so the prediction concerns a distribution of cellular states rather than matched cell trajectories (Bunne et al., 2023). We represent a population by its empirical distribution and summarize its mean response relative to control as

$$\widehat{P}_B = \frac{1}{m}\sum_{i=1}^{m} \delta_{b_i}, \qquad \Delta(B, C) = \bar{b} - \bar{c}, \tag{1}$$

where δ_{b_i} is a point mass and $\bar{b}, \bar{c}$ are population means. Matching the context does not imply pairing individual cells, and the mean contrast does not fully describe response heterogeneity.

3.2 CONDITIONAL DIFFUSION SAMPLING

We build on a pretrained conditional diffusion model that induces a population distribution $p_\theta(B \mid y)$, with $y = (C, a, h)$ (Yuan et al., 2026). Its existing reverse sampler starts from noise Z_0 and, at operation s, produces a next state and an estimate of clean expression,

$$\begin{aligned} Z_s &\sim M_{\theta,s}(\cdot \mid Z_{s-1}, y, H_{s-1}), \\ \widehat{B}_s &= D_{\theta,s}(Z_{s-1}, y, H_{s-1}). \end{aligned} \tag{2}$$

Here $s = 1, \ldots, T$ indexes operations in the prescribed denoising schedule, H_{s-1} is any self-conditioning memory, and Z_T is the final clean population. Memory is initialized and updated by the native sampler's unchanged rule. Deterministic samplers correspond to point-mass transitions. AdaCell preserves this sampling interface and keeps θ fixed; it changes how candidate populations are evaluated and selected.

4 METHODOLOGY

AdaCell steers a frozen diffusion model by evaluating candidate cell populations relative to their matched controls. The method has three components, illustrated in Figure 2: response priors estimated from training data, biological rewards defined over populations, and a sampling procedure that uses these rewards to reweight and resample candidates. Each particle represents a population of m cells, which is evaluated and selected as a unit. Expression vectors and response shifts use the same ordered gene panel and expression scale throughout.

4.1 RESPONSE PRIORS

To guide an intervention without observing its test response, we first estimate a reference expression shift from the training data (Figure 2(a)). For a perturbation observed in training, we subtract the control mean in each cell's context before pooling its responses.

$$\widehat{\Delta}_a^{\mathrm{obs}} = \frac{1}{n_a}\sum_{j \in \mathcal{I}_a^{\mathrm{tr}}} \left(x_j - \widehat{\mu}_{0,h_j}^{\mathrm{tr}} \right), \tag{3}$$

where $\mathcal{I}_a^{\mathrm{tr}}$ contains the n_a training cells receiving perturbation a, and $\widehat{\mu}_{0,h_j}^{\mathrm{tr}}$ is the training control mean for context h_j. This centering accounts for differences in baseline means and produces a pooled response reference. For an unseen perturbation, we predict its shift from a fixed descriptor available before the experiment. Let $\mathcal{A}_{\mathrm{tr}}$ contain the training perturbations and $\mathcal{A}_{\mathrm{obs}} \subseteq \mathcal{A}_{\mathrm{tr}}$ those with available descriptors. We fit centered ridge regression,

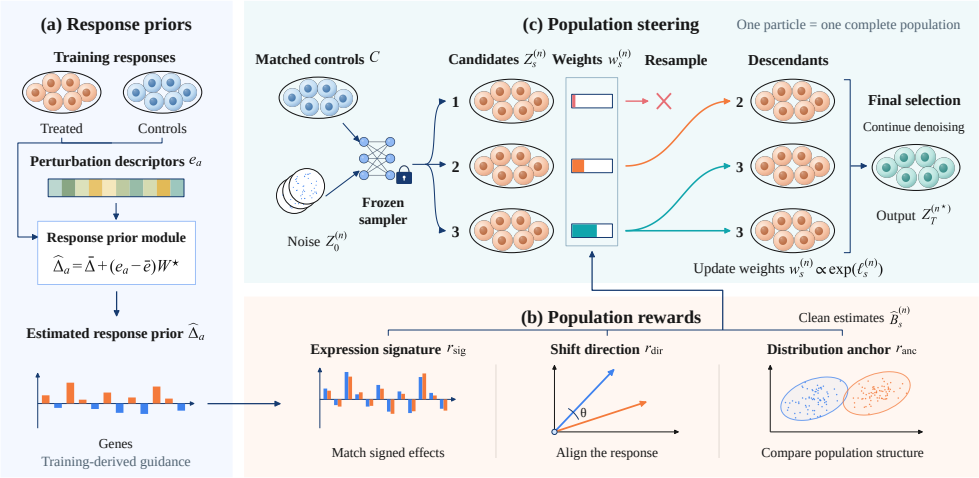

Figure 2: Overview of AdaCell. (a) Response priors are estimated from context-centered training responses and extended to unseen perturbations through descriptor-based regression. (b) Clean population estimates are evaluated using expression-signature, shift-direction, and distribution-anchor rewards. Axes in the direction and distribution panels denote schematic expression-space dimensions. (c) These rewards guide particle reweighting and resampling during frozen-model denoising.

$$
\begin{aligned}
W^\star &= \arg\min_W \sum_{a \in \mathcal{A}_{\text{obs}}} \left\| (e_a - \bar{e})W - (\widehat{\Delta}_a^{\text{obs}} - \bar{\Delta}) \right\|_2^2 + \lambda \|W\|_F^2, \\
\widehat{\Delta}_a &= \bar{\Delta} + (e_a - \bar{e})W^\star, \qquad a \notin \mathcal{A}_{\text{tr}},
\end{aligned}
\tag{4}
$$

where descriptors are unit-normalized row vectors $e_a \in \mathbb{R}^{1\times p}$; response shifts follow the same row vector convention, and $W \in \mathbb{R}^{p\times d}$. The means $\bar{e}$ and $\bar{\Delta}$ are computed over $\mathcal{A}_{\text{obs}}$, and $\lambda > 0$ controls regularization. Observed perturbations use $\widehat{\Delta}_a = \widehat{\Delta}_a^{\text{obs}}$. Prior estimation is performed offline using training responses; test responses and their derived statistics are excluded. The target control population remains an input to prediction, and the diffusion parameters are unchanged. Descriptor sources and fitting details are provided in Appendix A.2.

4.2 Biological Population Rewards

A useful candidate should reproduce the estimated response without forcing every cell toward the same expression profile. We therefore evaluate its mean shift, response direction, and distributional structure jointly, as illustrated in Figure 2(b). Let S_a contain the L genes with the largest $|\widehat{\Delta}_{a,g}|$, and let $A_a = \{c_i + \widehat{\Delta}_a\}_{i=1}^m$ be the control population translated by the estimated shift. Using the contrast $\Delta(B, C)$ defined in equation 1, we construct

$$
\begin{aligned}
r_{\text{sig}}(B; y) &= -\frac{1}{|S_a|} \left\| \Delta(B, C)_{S_a} - \widehat{\Delta}_{a,S_a} \right\|_2^2, \\
r_{\text{dir}}(B; y) &= \frac{\langle \Delta(B, C), \widehat{\Delta}_a \rangle}{\max(\|\Delta(B, C)\|_2, \epsilon) \max(\|\widehat{\Delta}_a\|_2, \epsilon)}, \\
r_{\text{anc}}(B; y) &= -\,\text{MMD}_{k_\sigma}^2 \left(\widehat{P}_B, \widehat{P}_{A_a} \right).
\end{aligned}
\tag{5}
$$

The signature reward measures signed response errors on genes prioritized by estimated effect size. The direction reward measures alignment across all genes, with $\epsilon > 0$ ensuring numerical stability. The anchor uses maximum mean discrepancy (Gretton et al., 2012) with the RBF kernel $k_\sigma(u, v) = \exp[-\|u - v\|_2^2/(2d\sigma^2)]$, where $\sigma > 0$. It compares distributional structure beyond the mean against a reference that retains the variation present in the controls. This reference is a soft

constraint, not an estimate of every perturbation-induced change in population composition. These rewards evaluate population summaries and interactions between cells rather than independent cell scores. In particular, matching the population mean does not directly penalize variation around that mean, unlike matching each cell to a common target. We use the all-pairs empirical MMD, so all three criteria are invariant to cell ordering. The estimator and numerical conventions are specified in Appendix A.2.

4.3 Tempered Population Diffusion Steering

Building on test-time diffusion alignment (Kim et al., 2025), AdaCell uses these rewards to favor promising populations while retaining the frozen model as the source of candidate trajectories, as shown in Figure 2(c). We initialize K independent noise populations $Z_0^{(n)} \in \mathbb{R}^{m\times d}$ and advance each through the frozen sampler in equation 2. At each step, the clean population estimate $\widehat{B}_s^{(n)}$ is evaluated using the three biological rewards, with the actual output used at the terminal step. To combine rewards with different numerical scales, we standardize each component across the current candidates,

$$\widetilde{r}_{j,s}^{(n)} = \frac{r_j(\widehat{B}_s^{(n)}; y) - \mu_{j,s}}{\max(\sigma_{j,s}, \epsilon)}, \qquad R_s^{(n)} = \sum_j \omega_j \widetilde{r}_{j,s}^{(n)}, \tag{6}$$

where $j \in \{\mathrm{sig}, \mathrm{dir}, \mathrm{anc}\}$, $\mu_{j,s}$ and $\sigma_{j,s}$ are the mean and standard deviation across candidates, and $\omega_j \geq 0$ weights each component. Early predictions are uncertain, so we give the rewards less influence at the beginning of sampling. We use the tempering schedule $\beta_s = (s-1)/(T-1)$ for $s = 1, \ldots, T$ and $T > 1$, which increases from zero to one. The log weights $\ell_s^{(n)}$ are updated and normalized as

$$\begin{aligned} \ell_s^{(n)} &= \ell_{s-1}^{(n)} + \frac{\beta_s R_s^{(n)} - \beta_{s-1} R_{s-1}^{(n)}}{\tau}, \\ w_s^{(n)} &= \frac{\exp(\ell_s^{(n)})}{\sum_{q=1}^K \exp(\ell_s^{(q)})}, \end{aligned} \tag{7}$$

with $\ell_0^{(n)} = R_0^{(n)} = \beta_0 = 0$ and temperature $\tau > 0$. This update uses the change in reward preference instead of accumulating the full reward at every step. When weights become concentrated on a few candidates, we resample whole populations according to their weights. Specifically, we apply systematic resampling if $\mathrm{ESS}_s = (\sum_n (w_s^{(n)})^2)^{-1} < \rho K$, where $\rho \in (0, 1]$. Each selected population retains its sampling state and current tempered reward for the next update, while its log weight is reset to zero. No resampling is performed at the final step. We then return $Z_T^{(n^\star)}$, where $n^\star = \arg\max_n w_T^{(n)}$, preserving one complete candidate rather than mixing cells from different populations. The entire procedure leaves the diffusion model unchanged and requires no reward gradients. Algorithm 1 and Appendix A provide the full algorithm and implementation details.

5 Experiments

5.1 Experimental Setup

We evaluate AdaCell on Replogle, PBMC, and Tahoe100M, covering genetic perturbations, cytokine stimulation, and chemical treatments, respectively. We follow the benchmark tasks, data splits, and evaluation protocol released with PerturbDiff (Yuan et al., 2026), and apply AdaCell to its frozen *Scratch* and *Finetuned* checkpoints. Comparisons include non-diffusion baselines, namely Mean and its variants, Linear (Ahlmann-Eltze et al., 2025), CPA (Lotfollahi et al., 2023), GEARS (Roohani et al., 2024), STATE (Adduri et al., 2025), and CellFlow (Klein et al., 2025), together with Squidiff (He et al., 2026) and the unsteered PerturbDiff checkpoints. GEARS is evaluated only on Replogle. Steering priors use training responses and perturbation descriptors available before the experiment, without access to test responses. We report differential expression overlap (DEOver), Pearson delta correlation (PDCorr), perturbation discrimination score with cosine distance (PDS-cos), and mean squared error (MSE), averaged across test perturbations. Main results and component ablations cover all three datasets; the remaining analyses use Replogle with the Scratch checkpoint. Appendix B provides dataset, implementation, and metric details.

5.2 Main Results

Table 1 shows that AdaCell improves its corresponding frozen checkpoint in 23 out of 24 metric comparisons across three datasets and two training regimes. It achieves the highest DEOver among all compared methods on each dataset, reaching 0.252 on Replogle, 0.624 on PBMC, and 0.594 on Tahoe100M. These gains extend beyond differential expression recovery to response correlation and prediction error: with the Scratch checkpoint, Replogle PDCorr increases from 0.307 to 0.398, while Tahoe100M MSE decreases from 6.831×10^{-4} to 3.347×10^{-4}, a 51.0% reduction. Steering also benefits checkpoints that have already been fine-tuned. On PBMC, the Finetuned checkpoint gains 0.044 in DEOver and reduces MSE by 24.4%; on Replogle, its PDS-cos rises from 0.580 to 0.659. The only decrease is 0.002 in Finetuned PDS-cos on Tahoe100M. The broad improvements across both training regimes demonstrate that population steering can further improve perturbation predictions without updating the diffusion model.

Table 1: Perturbation prediction across Replogle, PBMC, and Tahoe100M. Metrics are averaged across test perturbations within each dataset. MSE values for PBMC and Tahoe100M are reported in units of 10^{-4}. Shaded rows indicate AdaCell with the corresponding frozen PerturbDiff checkpoint. Arrows indicate the preferred direction. Best and second-best results in each column are bold and underlined, respectively.

Model	Replogle				PBMC				Tahoe100M			
	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE ↓	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE 10^{-4} ↓	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE 10^{-4} ↓
Non-diffusion baselines												
Mean	0.116	0.052	0.556	0.107	0.518	0.579	0.722	9.247	0.470	0.188	0.552	28.31
Mean (cell type)	0.194	0.412	0.457	0.009	0.547	0.433	0.562	5.754	0.553	0.416	0.823	7.523
Mean (batch)	0.121	0.002	0.458	0.064	0.603	0.563	0.713	5.091	0.465	0.201	0.544	28.42
Mean (overall)	0.121	-0.001	0.547	0.077	0.604	0.447	0.460	9.163	0.392	0.210	0.542	27.63
Linear	0.074	0.064	0.730	0.012	0.600	0.699	0.830	4.854	0.553	**0.784**	0.836	3.775
CPA	0.188	0.383	0.573	0.082	0.448	0.164	0.649	120.0	0.455	0.383	0.531	8.144
STATE	0.177	**0.476**	**0.733**	**0.007**	0.557	0.731	**0.898**	**1.528**	0.548	0.487	0.777	23.18
CellFlow	0.100	-0.003	0.542	0.086	0.293	0.687	0.787	3.880	0.200	0.247	0.502	28.52
GEARS	0.068	0.206	0.502	0.012	–	–	–	–	–	–	–	–
Diffusion baselines and steering												
Squidiff	0.043	0.081	0.543	4.255	0.324	0.030	0.458	40270	0.383	0.010	0.458	61370
PerturbDiff (Scratch)	0.174	0.307	0.630	0.013	0.618	0.887	0.889	1.757	0.473	0.621	0.886	6.831
+ AdaCell (ours)	0.211	0.398	0.645	0.011	**0.624**	**0.896**	0.895	1.740	0.516	0.651	0.905	**3.347**
PerturbDiff (Finetuned)	0.234	0.406	0.580	0.016	0.489	0.643	0.877	3.435	0.531	0.592	**0.984**	11.40
+ AdaCell (ours)	**0.252**	0.416	0.659	0.011	0.533	0.691	0.889	2.598	**0.594**	0.640	0.982	9.920

5.3 Diagnosing Perturbation Response Errors

We examine perturbation contrast under-specification through three complementary aspects of the response: magnitude, direction, and responder fraction. For a perturbation in a given cellular context, the difference between the perturbed and matched control means defines the population response vector. Comparing its length and orientation separates errors in response strength from errors in the transcriptional changes being predicted. The responder fraction provides a complementary view of how extensively the intervention affects the population. Figure 3 places the PerturbDiff and AdaCell errors on the horizontal and vertical axes, respectively, so that points below the diagonal indicate lower errors with steering. This decomposition distinguishes correcting a weak response from correcting its direction or the proportion of responding cells, making the evaluation more informative than an aggregate expression score alone.

5.4 Why Population Steering?

We compare variants of the same frozen PerturbDiff backbone to examine when and how candidate populations should be evaluated. Random-16 generates 16 candidates without reward-based selection, while Best-of-16 selects the highest-scoring population only after denoising. Cellwise-16 averages individual-cell rewards instead of evaluating the population jointly, but retains population-level resampling. AdaCell-16 and AdaCell-8 apply our steering procedure with 16 and eight particles,

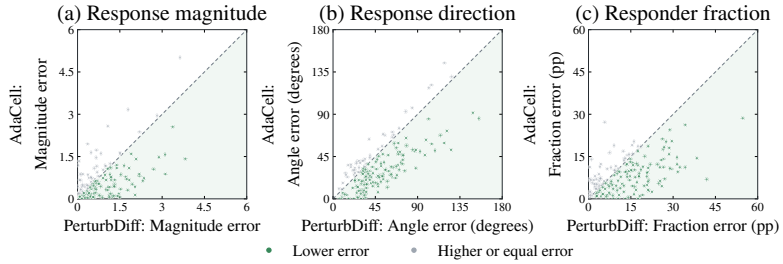

Figure 3: Paired comparisons of perturbation response errors for PerturbDiff and AdaCell. The horizontal and vertical axes show PerturbDiff and AdaCell errors, respectively. Panels measure (a) the absolute difference between predicted and reference response-vector norms, (b) the angle between predicted and reference response vectors in degrees, and (c) the absolute difference between predicted and reference responder fractions in percentage points. Green points indicate lower errors with AdaCell; gray points indicate higher or equal errors.

respectively. Mean correction adjusts the Random-16 output toward training-derived mean responses under a nonnegativity constraint. In the DEOver comparison in Figure 4, AdaCell-16 reaches 0.211, compared with 0.188 for Random-16, 0.184 for Best-of-16, and 0.193 for Cellwise-16. AdaCell-8 retains a score of 0.206 with fewer particles, whereas mean correction yields 0.141. This ranking favors population steering over the tested alternatives for differential expression recovery and is consistent with the benefit of jointly evaluating populations during denoising.

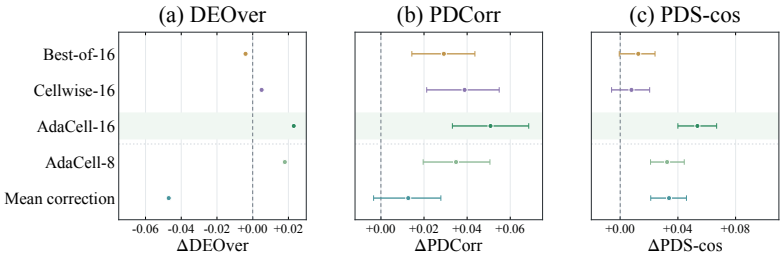

Figure 4: Population steering versus alternative sampling and correction strategies on Replogle using the same frozen PerturbDiff checkpoint. Panels show differences relative to Random-16 in (a) DEOver, (b) PDCorr, and (c) PDS-cos.

5.5 Ablation Study

We remove each reward component or disable reward standardization while keeping the remaining settings fixed for the corresponding Scratch checkpoint. Full AdaCell achieves the best result in every column of Table 2. Removing the shift-direction reward produces the largest PDCorr decrease on all three datasets, including a drop from 0.398 to 0.341 on Replogle, supporting its role in recovering the response pattern across genes. The expression signature provides additional information on prioritized genes: without it, Tahoe100M DEOver decreases from 0.516 to 0.493. The distribution anchor contributes to perturbation discrimination, with Replogle PDS-cos falling from 0.645 to 0.586 when it is removed. Reward standardization also matters: using raw reward values raises Replogle MSE from 0.011 to 0.018 and lowers PBMC PDS-cos from 0.895 to 0.879. The different effects of these removals support combining the biological criteria with reward standardization for population selection.

5.6 Response Accuracy, Diversity, and Computational Cost

The response improvements in Table 1 motivate a closer examination of population diversity and sampling cost. Figure 5(a,b) separates two aspects of diversity: total expression variance measures

Table 2: Ablation study across Replogle, PBMC, and Tahoe100M using the corresponding frozen PerturbDiff (Scratch) checkpoints. Metrics are averaged over test perturbations within each dataset. MSE values for PBMC and Tahoe100M are reported in units of 10^{-4}. The shaded row denotes full AdaCell. Best and second-best results in each column are bold and underlined, respectively.

Variant	Replogle				PBMC				Tahoe100M			
	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE ↓	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE 10^{-4} ↓	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE 10^{-4} ↓
AdaCell (full)	**0.211**	**0.398**	**0.645**	**0.011**	**0.624**	**0.896**	**0.895**	**1.740**	**0.516**	**0.651**	**0.905**	**3.347**
w/o Expression signature	0.191	0.370	0.592	0.013	0.617	0.891	0.890	1.821	0.493	0.638	0.900	4.225
w/o Shift direction	0.176	0.341	0.597	0.012	0.610	0.883	0.886	2.054	0.502	0.623	0.897	3.858
w/o Distribution anchor	0.190	0.368	0.586	0.014	0.623	0.894	0.888	1.766	0.511	0.645	0.894	3.601
w/o Reward standardization	0.192	0.376	0.585	0.018	0.615	0.887	0.879	1.914	0.492	0.633	0.891	4.803

the amount of variation, while effective rank measures how broadly this variation is distributed across expression directions. Relative to Random-16, the updated AdaCell-16 ratios are 1.100 and 1.140, respectively. These summaries indicate increased aggregate spread and effective dimensionality relative to the unselected reference. They do not, on their own, establish closer agreement with the observed biological distribution. Figure 5(c) reports the earlier matched timing comparison across all six sampling configurations. AdaCell-16 requires 726.5 seconds per 1,000 cells, comparable to Random-16, Best-of-16, and Cellwise-16 at 730.7, 744.3, and 738.0 seconds, respectively. Reducing the particle count to eight or four lowers the measured cost to 376.6 or 198.3 seconds. These measurements characterize the shared implementation at the indicated candidate budgets, not its overhead relative to a single-candidate sampler. The updated diversity summaries and the earlier timing measurements are reported separately and do not imply a jointly verified accuracy–diversity–cost trade-off for one run.

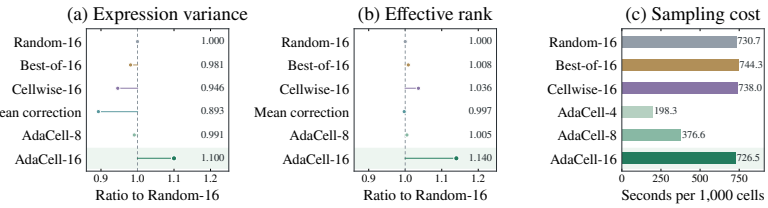

Figure 5: Population diversity and sampling cost on Replogle. (a) Total expression variance and (b) effective rank, expressed as ratios to Random-16. (c) All six configurations evaluated on the same 11 groups containing 230 cells.

6 Conclusion

We presented AdaCell, a test-time steering framework that makes perturbation responses relative to control an explicit criterion for diffusion sampling. By treating each particle as a candidate cell population, AdaCell uses training-derived response priors to evaluate and select populations jointly, without updating the diffusion model. Experiments on Replogle, PBMC, and Tahoe100M show improvements over both Scratch and Finetuned checkpoints, while component ablations support combining the biological rewards with reward standardization. These results demonstrate the value of evaluating responses at the population level during generation, complementing advances in diffusion training with biological information at inference time.

AI use statement

We used generative AI tools to support and polish manuscript writing and to provide suggestions for figure organization and visual design. The authors reviewed and revised the AI-assisted text and figure designs for accuracy, clarity, and consistency with the methods and experimental results. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with AI assistance.

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, 2025.

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.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Roni Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In International Conference on Learning Representations, volume 2024, pp. 51304–51323, 2024.

Andreas Bender, Morgan C Thomas, Jack W Scannell, David A Shaywitz, Gian Marco Ghiandoni, Joe G Greener, Lavinia-Lorena Pruteanu, Rachel DeVay Jacobson, Koichi Handa, Mariko Hirano, et al. Artificial intelligence in drug discovery—what it is, where we stand and the path forward. Nature Reviews Drug Discovery, pp. 1–22, 2026.

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.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In International Conference on Learning Representations, volume 2024, pp. 4965–4987, 2024.

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.

Fei Xavier Chen, Ashley R Woodfin, Alessandro Gardini, Ryan A Rickels, Stacy A Marshall, Edwin R Smith, Ramin Shiekhattar, and Ali Shilatifard. Paf1, a molecular regulator of promoter-proximal pausing by rna polymerase ii. Cell, 162(5):1003–1015, 2015.

Fei Xavier Chen, Peng Xie, Clayton K Collings, Kaixiang Cao, Yuki Aoi, Stacy A Marshall, Emily J Rendleman, Michal Ugarenko, Patrick A Ozark, Anda Zhang, et al. Paf1 regulation of promoter-proximal pause release via enhancer activation. Science, 357(6357):1294–1298, 2017.

Yiqun Chen and James Zou. Genept: a simple but effective foundation model for genes and cells built from chatgpt. BioRxiv, pp. 2023–10, 2024.

Mingze Dong, Bao Wang, Jessica Wei, Antonio H de O. Fonseca, Curtis J Perry, Alexander Frey, Feriel Ouerghi, Ellen F Foxman, Jeffrey J Ishizuka, Rahul M Dhodapkar, et al. Causal identification of single-cell experimental perturbation effects with cinema-ot. Nature methods, 20(11):1769–1779, 2023.

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.

Siyu He, Yuefei Zhu, Daniel Naveed Tavakol, Haotian Ye, Yeh-Hsing Lao, Zixian Zhu, Cong Xu, Shradha Chauhan, Guy Garty, Raju Tomer, et al. Squidiff: predicting cellular development and responses to perturbations using a diffusion model. Nature methods, 23(1):65–77, 2026.

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

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In International Conference on Learning Representations, volume 2025, pp. 87238–87272, 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.

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.

Ajay Nadig, Joseph M Replogle, Angela N Pogson, Mukundh Murthy, Steven A McCarroll, Jonathan S Weissman, Elise B Robinson, and Luke J O'Connor. Transcriptome-wide analysis of differential expression in perturbation atlases. Nature Genetics, 57(5):1228–1237, 2025.

Lukas Oesinghaus, Sören Becker, Larsen Vornholz, Efthymia Papalexi, Joey Pangallo, Amir Ali Moinfar, Jenni Liu, Alyssa La Fleur, Maiia Shulman, Simone Marrujo, et al. A single-cell cytokine dictionary of human peripheral blood. BioRXiv, 2025.

Byung-Sun Park, Mieun Lee, Jaeyeol Kim, and Tackhoon Kim. Perturbomics: Crispr–cas screening-based functional genomics approach for drug target discovery. Experimental & Molecular Medicine, 57(7):1443–1454, 2025.

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.

Yusuf H Roohani, Tony J Hua, Po-Yuan Tung, Lexi R Bounds, Feiqiao B Yu, Alexander Dobin, Noam Teyssier, Abhinav Adduri, Alden Woodrow, Brian S Plosky, et al. Virtual cell challenge: Toward a turing test for the virtual cell. Cell, 188(13):3370–3374, 2025.

Bicna Song, Dingyu Liu, Weiwei Dai, Natalie F McMyn, Qingyang Wang, Dapeng Yang, Adam Krejci, Anatoly Vasilyev, Nicole Untermoser, Anke Loregger, et al. Decoding heterogeneous single-cell perturbation responses. Nature cell biology, 27(3):493–504, 2025.

Ramon Viñas Torné, Maciej Wiatrak, Zoe Piran, Shuyang Fan, Liangze Jiang, Sarah A Teichmann, Mor Nitzan, and Maria Brbić. Systema: a framework for evaluating genetic perturbation response prediction beyond systematic variation. Nature Biotechnology, pp. 1–10, 2025.

Seychelle M Vos, Lucas Farnung, Marc Boehning, Christoph Wigge, Andreas Linden, Henning Urlaub, and Patrick Cramer. Structure of activated transcription complex pol ii–dsif–paf–spt6. Nature, 560(7720):607–612, 2018.

Dongxia Wu, Mingyu Li, Yuhui Zhang, Anurendra Kumar, Emma Lundberg, Serena Yeung-Levy, and Emily B Fox. Perturbcellrl: Verifier-guided reinforcement learning for single-cell perturbation prediction. arXiv preprint arXiv:2606.27752, 2026.

Ming Yu, Wenjing Yang, Ting Ni, Zhanyun Tang, Tomoyoshi Nakadai, Jun Zhu, and Robert G Roeder. Rna polymerase ii–associated factor 1 regulates the release and phosphorylation of paused rna polymerase ii. Science, 350(6266):1383–1386, 2015.

Xinyu Yuan, Xixian Liu, Ya Shi Zhang, Zuobai Zhang, Hongyu Guo, and Jian Tang. Perturbdiff: Functional diffusion for single-cell perturbation modeling. arXiv preprint arXiv:2602.19685, 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.

A Additional Method Details

A.1 Notation and Algorithm

The reverse-process index s increases from 1 to T and need not equal the diffusion model's native time coordinate. A particle $Z_s^{(n)}$ contains a complete population, not a single cell. Its clean estimate is $\widehat{B}_s^{(n)}$, and $H_s^{(n)}$ denotes the sampler's auxiliary state, including self-conditioning when used. All particles for a target share the condition $y = (C, a, h)$ and the same response prior. Padding, when required by the backbone, is excluded from reward calculations and exported predictions. Algorithm 1 makes the ancestry bookkeeping explicit. The stored potential $F^{(n)}$ follows the selected ancestor, while the importance log weight is reset after resampling. Copying the potential is necessary: subtracting a potential from a different ancestor would change the incremental update. Resampling always copies an intact population and its auxiliary state; it never assembles a population from cells belonging to different particles.

Algorithm 1 AdaCell population steering

Require: Frozen sampler, condition $y = (C, a, h)$, prior $\widehat{\Delta}_a$, particles K, steps T, temperature τ, ESS fraction ρ

Draw independent initial populations $Z_0^{(1:K)}$ from the sampler's noise law
Initialize native auxiliary states $H_0^{(1:K)}$; set $F^{(1:K)} \leftarrow 0$, $\ell^{(1:K)} \leftarrow 0$
for $s = 1, \ldots, T$ **do**
    Advance each $(Z_{s-1}^{(n)}, H_{s-1}^{(n)})$ using equation 2
    Obtain $Z_s^{(n)}$, $H_s^{(n)}$, and clean estimate $\widehat{B}_s^{(n)}$
    if $s = T$ **then**
        Set $\widehat{B}_s^{(n)} \leftarrow Z_T^{(n)}$ for scoring
    end if
    Evaluate and standardize the three rewards; combine them into $R_s^{(n)}$
    Set $\beta_s \leftarrow (s-1)/(T-1)$ if $T > 1$, otherwise $\beta_s \leftarrow 1$
    $F_{\text{new}}^{(n)} \leftarrow \beta_s R_s^{(n)}/\tau$ for all n
    $\ell^{(n)} \leftarrow \ell^{(n)} + F_{\text{new}}^{(n)} - F^{(n)}$; $F^{(n)} \leftarrow F_{\text{new}}^{(n)}$
    $w^{(1:K)} \leftarrow \mathrm{softmax}(\ell^{(1:K)})$
    if $s < T$ and $1/\sum_n (w^{(n)})^2 < \rho K$ **then**
        Draw ancestor indices $q_{1:K}$ by systematic resampling with weights $w^{(1:K)}$
        $(Z_s^{(n)}, H_s^{(n)}, F^{(n)}) \leftarrow (Z_s^{(q_n)}, H_s^{(q_n)}, F^{(q_n)})$ simultaneously
        Reset $\ell^{(1:K)} \leftarrow 0$
    end if
end for
$n^\star \leftarrow \arg\max_n w^{(n)}$
return $Z_T^{(n^\star)}$

Systematic resampling draws $u \sim \mathcal{U}(0, 1/K)$ and assigns the points $u + (n-1)/K$, $n = 1, \ldots, K$, to the cumulative weight intervals. We do not resample at the terminal step. Final selection therefore uses the weights accumulated since the most recent reset, which need not give the same ranking as the terminal rewards alone. If the native transition is deterministic, replicated particles with identical auxiliary states remain identical unless the sampler introduces subsequent randomness. Resampling does not itself create new trajectories.

A.2 Response Priors and Population Rewards

Observed priors pool context-centered training responses as in equation 3. Pooling is cell-weighted: a context with more training cells for a perturbation contributes more to its reference. This removes baseline mean differences but does not assume that a perturbation has identical effects in every context. A target control population is an allowed prediction input; held-out treated responses and statistics computed from them are not used to construct priors, signature genes, or anchors.

For descriptor regression, descriptors are unit-normalized before centering. Let E_c and D_c collect their centered rows and the centered observed-prior rows over $\mathcal{A}_{\text{obs}}$. The solution to equation 4 can be written as

$$W^\star = E_c^\top (E_c E_c^\top + \lambda I)^{-1} D_c. \tag{8}$$

The implementation solves the linear system rather than explicitly forming an inverse. Replogle uses fixed GenePT descriptors (Chen & Zou, 2024). Perturbations observed in any eligible training context use their empirical priors; descriptor regression is needed only when no training response is available. The signature set S_a is selected by estimated absolute effect size, not by a significance test on held-out cells. The direction reward compares the full response vectors. For the anchor, write $A_a = \{a_i\}_{i=1}^m$ with $a_i = c_i + \widehat{\Delta}_a$. The all-pairs empirical estimator is

$$\begin{aligned}\widehat{\mathrm{MMD}}^2(B, A_a) = \frac{1}{m^2}\sum_{i,j=1}^{m} k_\sigma(b_i, b_j) + \frac{1}{m^2}\sum_{i,j=1}^{m} k_\sigma(a_i, a_j) \\ - \frac{2}{m^2}\sum_{i,j=1}^{m} k_\sigma(b_i, a_j),\end{aligned} \tag{9}$$

using the RBF kernel in the main text (Gretton et al., 2012). This biased V-statistic includes diagonal terms and is invariant to cell ordering. Small negative values caused by roundoff are clipped to zero before negation. Translating controls preserves their within-population variation, but does not model every intervention-induced change in covariance or composition; the anchor is therefore a soft reference rather than a ground-truth perturbed distribution.

A.3 Numerical Conventions and Interpretation

At each step, component scores are standardized using the population standard deviation across the K candidates,

$$\mu_{j,s} = \frac{1}{K}\sum_n r_{j,s}^{(n)}, \qquad \sigma_{j,s}^2 = \frac{1}{K}\sum_n (r_{j,s}^{(n)} - \mu_{j,s})^2. \tag{10}$$

The denominator is clamped below by 10^{-6} in the inspected implementation. A constant component then contributes zero preference. We normalize log weights with log-sum-exp and reject non-finite rewards. The norm floor for the direction reward is kept separate from the standardization floor in the run record. For a fixed reward, reward-based diffusion alignment motivates a reference of the form $\pi(B \mid y) \propto p_\theta(B \mid y)\exp(R(B; y)/\tau)$ (Kim et al., 2025). Without resampling, the incremental potentials telescope to the terminal potential. AdaCell additionally standardizes rewards using the current candidates and returns the maximum-weight population. It is therefore an adaptive population-search procedure, not an exact sampler from a fixed reward-tilted distribution. Neither this formulation nor the control-based anchor guarantees diversity preservation. All diffusion calls are forward-only, and neither reward gradients nor updates to the diffusion parameters are required.

B Experimental Details

B.1 Datasets, Preprocessing, and Data Splits

We evaluate three complementary perturbation datasets. Replogle combines CRISPRi Perturb-seq data from K562, RPE1, Jurkat, and HepG2 cells (Replogle et al., 2022; Nadig et al., 2025). The Parse PBMC atlas contains approximately 9.7 million cells spanning 18 immune cell types from 12 donors treated with 90 cytokines or PBS (Oesinghaus et al., 2025). Tahoe100M profiles approximately 100 million cells from 50 cancer cell lines exposed to 379 compounds at three concentrations, with DMSO controls (Zhang et al., 2025).

We use the processed expression matrices and fixed dataset-specific panels of 2,000 highly variable genes. Expression values are normalized and log-transformed; Replogle additionally uses filtering based on on-target knockdown efficacy. To match the frozen diffusion checkpoints, predictions are returned to the log-normalized scale for evaluation. Table 3 summarizes the fixed splits. Validation and test sets contain specified perturbations within the designated cell lines or donors. All remaining treated observations form the training set, including other perturbations in these same contexts.

Table 3: Dataset composition and fixed evaluation splits. Treated-cell partitions are disjoint; controls are counted separately.

Property	Replogle	PBMC	Tahoe100M
Perturbation unit	Target gene	Cytokine	Drug–dose pair
Control condition	Non-targeting	PBS	DMSO
Holdout axis	Cell line	Donor	Cell line
Designated evaluation contexts	HepG2	Donor1, Donor4, Donor9, Donor12	Five cell lines
Configured validation labels	104	4	54
Configured test labels	624	62	735
Training treated cells	572,545	6,657,336	89,495,239
Validation treated cells	4,825	150,484	553,076
Test treated cells	26,878	2,260,453	8,790,272
Control cells	39,165	629,701	2,330,156
Evaluation genes	2,000	2,000	2,000

B.2 Baseline Implementations

Mean averages training condition means with equal condition weights in the inspected adapter, including the control condition. Mean (overall) instead averages treated training cells. Mean (cell type) and Mean (batch) restrict this cell-weighted estimate to the relevant training context or batch. These variants differ when condition sizes are unequal. Linear is a regularized prediction baseline (Ahlmann-Eltze et al., 2025); its descriptor source, target parameterization, and penalty must match the recorded implementation.

The baselines are CPA (Lotfollahi et al., 2023), GEARS (Roohani et al., 2024), STATE (Adduri et al., 2025), CellFlow (Klein et al., 2025), and Squidiff (He et al., 2026). GEARS is evaluated only on Replogle. A missing entry denotes an unevaluated setting, not a score of zero. PerturbDiff Scratch and Finetuned (Yuan et al., 2026) provide two frozen backbone checkpoints. AdaCell uses the corresponding checkpoint and its native denoising schedule without parameter updates.

B.3 Evaluation Metrics

We follow the evaluation protocol of Cell-Eval 0.6.6 (Adduri et al., 2025) and the population-mean R^2 metric from CellFlow (Klein et al., 2025). All methods are evaluated on the same 2,000 genes and log-normalized expression scale, with identical observed controls and matched cell counts per perturbation. Except for ES, metrics are computed per perturbation and averaged equally within each dataset; NaNs are excluded and valid counts are recorded. Higher values are better except for MSE and MAE. For each evaluation split, let M be the number of test perturbations and $\mathcal{G}$ the evaluation gene set, with $G = |\mathcal{G}|$. For perturbation p, $\boldsymbol{\mu}_p$, $\hat{\boldsymbol{\mu}}_p$, and $\boldsymbol{\mu}_0$ denote observed, predicted, and control population means, respectively. Hats consistently indicate predicted quantities, and response vectors are defined as

$$\boldsymbol{\delta}_p = \boldsymbol{\mu}_p - \boldsymbol{\mu}_0, \qquad \hat{\boldsymbol{\delta}}_p = \hat{\boldsymbol{\mu}}_p - \boldsymbol{\mu}_0. \tag{11}$$

Coefficient of Determination (R^2). This metric measures agreement between predicted and observed mean expression across genes,

$$R^2_p = 1 - \frac{\sum_{g\in\mathcal{G}}(\hat{\mu}_{p,g} - \mu_{p,g})^2}{\sum_{g\in\mathcal{G}}(\mu_{p,g} - \bar{\mu}_p)^2}, \qquad \bar{\mu}_p = \frac{1}{G}\sum_{g\in\mathcal{G}} \mu_{p,g}. \tag{12}$$

It is a coefficient of determination, not squared Pearson correlation.

Differential Expression Overlap (DEOver). DEOver measures recovery of observed differentially expressed genes (DEGs). DE analysis uses Wilcoxon tests against the same controls and Benjamini–Hochberg adjustment (Benjamini & Hochberg, 1995). Let $S_p = \{g \in \mathcal{G} : q_{p,g} < 0.05\}$, where $q_{p,g}$ is the adjusted p-value, and define $K_p = |S_p|$; hatted quantities are obtained from predicted cells. Let $T_p(k)$ contain the first k genes in S_p ranked by absolute log fold change, or all genes in S_p if fewer than k are available. Then

$$\mathrm{DEOver}_p = \frac{|T_p(K_p) \cap \hat{T}_p(K_p)|}{K_p}. \tag{13}$$

Following Cell-Eval, the score is zero when $K_p = 0$.

Differential Expression Precision (DEPrec). DEPrec instead sets the comparison size to the number of predicted DEGs, $\hat{K}_p = |\hat{S}_p|$,

$$\mathrm{DEPrec}_p = \frac{|T_p(\hat{K}_p) \cap \hat{T}_p(\hat{K}_p)|}{\hat{K}_p}. \tag{14}$$

The score is zero when $\hat{K}_p = 0$.

Effect Size (ES). ES assesses whether predicted DEG counts preserve the relative strength of different perturbations,

$$\mathrm{ES} = \rho_{\mathrm{S}}\Big((K_p)_{p\in\mathcal{P}_+}, (\hat{K}_p)_{p\in\mathcal{P}_+}\Big), \qquad \mathcal{P}_+ = \{p : K_p > 0\}, \tag{15}$$

where ρ_{S} denotes Spearman correlation. Following Cell-Eval 0.6.6, perturbations without observed DEGs are excluded, missing predicted DEG counts are set to zero, and an empty $\mathcal{P}_+$ returns one. Unlike the other metrics, ES is computed across perturbations rather than averaged over them.

Direction Agreement (DirAgr). DirAgr measures the fraction of observed DEGs with correctly predicted response signs,

$$\mathrm{DirAgr}_p = \frac{1}{K_p} \sum_{g\in S_p} \mathbf{1}\Big[\mathrm{sgn}(\hat{\ell}_{p,g}) = \mathrm{sgn}(\ell_{p,g})\Big], \tag{16}$$

where $\ell_{p,g}$ denotes the observed $\log_2$ fold change relative to control. Genes need not be significant in the predictions to enter this comparison.

Log Fold Change Spearman Correlation (LFCSpear). LFCSpear evaluates the ordering of gene effects over observed DEGs. Cell-Eval computes the correlation on fold changes $f_{p,g}$, whose ranks equal those of $\ell_{p,g} = \log_2 f_{p,g}$ for positive finite values,

$$\mathrm{LFCSpear}_p = \rho_{\mathrm{S}}\Big((\hat{f}_{p,g})_{g\in S_p}, (f_{p,g})_{g\in S_p}\Big). \tag{17}$$

Area Under the Precision–Recall Curve (AUPRC). AUPRC evaluates the prioritization of observed DEGs using labels $y_{p,g} = \mathbf{1}[g \in S_p]$ and scores $s_{p,g} = -\log_{10} \hat{q}_{p,g}$. Zero adjusted p-values are replaced by 10^{-10} before taking logarithms. Following Cell-Eval, we use average precision,

$$\mathrm{AUPRC}_p = \sum_{j=1}^{J_p} (\mathrm{Rec}_{p,j} - \mathrm{Rec}_{p,j-1})\,\mathrm{Prec}_{p,j}, \tag{18}$$

where j indexes successive groups of tied scores in descending order, $\mathrm{Prec}_{p,j}$ and $\mathrm{Rec}_{p,j}$ are the resulting precision and recall, and $\mathrm{Rec}_{p,0} = 0$.

Area Under the Receiver Operating Characteristic Curve (AUROC). Using the same labels and scores, AUROC measures separation between observed DEGs and other genes,

$$\mathrm{AUROC}_p = \int_0^1 \mathrm{TPR}_p(u)\,\mathrm{d}u, \tag{19}$$

where u is the false positive rate and $\mathrm{TPR}_p(u)$ is the linearly interpolated true positive rate. Both AUPRC and AUROC are undefined when all evaluation genes belong to the same class.

Pearson Delta Correlation (PDCorr). PDCorr measures agreement in expression changes relative to control,

$$\mathrm{PDCorr}_p = \rho_{\mathrm{P}}\Big(\hat{\boldsymbol{\delta}}_p, \boldsymbol{\delta}_p\Big), \tag{20}$$

where ρ_{P} denotes Pearson correlation across genes. Because it is insensitive to positive rescaling, PDCorr does not independently assess response magnitude.

Mean Squared Error (MSE). MSE measures the average squared discrepancy between predicted and observed population means,

$$\mathrm{MSE}_p = \frac{1}{G} \sum_{g\in\mathcal{G}} (\hat{\mu}_{p,g} - \mu_{p,g})^2. \tag{21}$$

With identical controls, this also equals the mean squared error between the response vectors.

Mean Absolute Error (MAE). MAE measures the average absolute discrepancy between population means,

$$\mathrm{MAE}_p = \frac{1}{G} \sum_{g \in \mathcal{G}} |\hat{\mu}_{p,g} - \mu_{p,g}|. \tag{22}$$

Neither MSE nor MAE assumes pairing between individual cells.

Perturbation Discrimination Score with ℓ_1 Distance (PDS-L1). PDS measures whether a predicted response distinguishes its corresponding perturbation from the others. Let $r_{p,d} \in \{0, \ldots, M-1\}$ be the zero-based rank of the correct observed response among all M observed responses, ordered by increasing distance from $\hat{\boldsymbol{\delta}}_p$ under d. Using ℓ_1 distance,

$$d_{\ell_1}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_1, \qquad \mathrm{PDS}_{p,\ell_1} = 1 - \frac{r_{p,\ell_1}}{M}. \tag{23}$$

All PDS variants retain Cell-Eval's tie handling and exclude the query perturbation's targeted gene from distance calculations when that gene is measured.

Perturbation Discrimination Score with ℓ_2 Distance (PDS-L2). This variant uses Euclidean distance to rank the observed responses,

$$d_{\ell_2}(\mathbf{a}, \mathbf{b}) = \|\mathbf{a} - \mathbf{b}\|_2, \qquad \mathrm{PDS}_{p,\ell_2} = 1 - \frac{r_{p,\ell_2}}{M}. \tag{24}$$

Perturbation Discrimination Score with Cosine Distance (PDS-cos). This variant ranks observed responses by their angular agreement with the predicted response,

$$d_{\cos}(\mathbf{a}, \mathbf{b}) = 1 - \frac{\mathbf{a}^\mathsf{T}\mathbf{b}}{\|\mathbf{a}\|_2 \|\mathbf{b}\|_2}, \qquad \mathrm{PDS}_{p,\cos} = 1 - \frac{r_{p,\cos}}{M}. \tag{25}$$

Zero-vector cases follow the evaluator's distance implementation. PDS-cos is a normalized ranking score, not a cosine similarity.

B.4 Default Steering Configuration

We use $K = 16$ population particles and reward temperature $\tau = 1.0$ as the default steering configuration. Table 4 lists the settings of the Replogle reference run. The three reward components are standardized across particles before being combined with equal weights. Systematic resampling is triggered when $\mathrm{ESS}_s < 0.5K$ at a nonterminal step. At the final step, we return the intact population with the largest weight. AdaCell-8 denotes the reduced-budget variant with $K = 8$; temperature and particle-count variations are reported in Appendix D.1.

Table 4: Default steering settings for the Replogle reference configuration. The diffusion model remains frozen throughout sampling.

Setting	Symbol	Default
Population particles	K	16
Reward temperature	τ	1.0
ESS threshold fraction	ρ	0.5
Reward weights	$(\omega_{\mathrm{sig}}, \omega_{\mathrm{dir}}, \omega_{\mathrm{anc}})$	$(1.0, 1.0, 1.0)$
Signature genes	L	20
Prior ridge regularization	λ	1.0
RBF anchor bandwidth	σ	1.0
Reward normalization	–	Across-particle z-score
Standard-deviation floor	–	10^{-6}
Tempering schedule	β_s	$(s-1)/(T-1), T > 1$
Final selection	$n^\star$	$\arg\max_n w_T^{(n)}$

Table 5: Changes from each frozen checkpoint to AdaCell, calculated from the displayed main-table values. The first three columns are absolute score differences; the last is $100(\mathrm{MSE}_{\mathrm{base}} - \mathrm{MSE}_{\mathrm{AdaCell}})/\mathrm{MSE}_{\mathrm{base}}$. Positive values denote improvement. Rounding in the main table limits the precision of these derived changes.

Dataset	Checkpoint	ΔDEOver	ΔPDCorr	ΔPDS-cos	MSE reduction
Replogle	Scratch	+0.037	+0.091	+0.015	15.4%
	Finetuned	+0.018	+0.010	+0.079	31.3%
PBMC	Scratch	+0.006	+0.009	+0.006	1.0%
	Finetuned	+0.044	+0.048	+0.012	24.4%
Tahoe100M	Scratch	+0.043	+0.030	+0.019	51.0%
	Finetuned	+0.063	+0.048	-0.002	13.0%

C Supplementary Results

C.1 Paired Improvements Across Frozen Checkpoints

Table 5 isolates the effect of steering by subtracting each frozen checkpoint's result from its AdaCell counterpart in Table 1. AdaCell improves 23 of the 24 dataset–checkpoint–metric comparisons. DEOver improves in all six settings, with absolute gains from 0.006 to 0.063. The largest relative MSE reduction is 51.0% for Scratch on Tahoe100M. The only decrease is Tahoe100M Finetuned PDS-cos, from 0.984 to 0.982. These paired comparisons establish improvement relative to the corresponding generator, without implying that AdaCell is best among every baseline on every metric.

C.2 Full Metric Results

The primary metrics for all baselines appear in Table 1. Additional evaluator outputs should use the identical prediction artifacts, gene panel, and aggregation rule. Context-level averages and improved-perturbation fractions cannot be recovered from dataset-level means.

Table 6: Available metrics on Replogle. Metrics are averaged across test perturbations. Shaded rows indicate AdaCell with the corresponding frozen PerturbDiff checkpoint. Arrows indicate the preferred direction. Best and second-best results in each column are bold and underlined, respectively.

Method	R^2 ↑	DEOver ↑	DEPrec ↑	ES ↑	DirAgr ↑	LFCSpear ↑	AUPRC ↑	AUROC ↑	PDCorr ↑	MSE ↓	MAE ↓	PDS-L1 ↑	PDS-L2 ↑	PDS-cos ↑
Non-diffusion baselines														
Mean	0.702	0.116	0.088	0.389	0.505	0.073	0.085	0.443	0.052	0.107	0.217	0.536	0.558	0.556
Mean (cell type)	0.939	0.194	0.081	0.388	0.700	0.453	0.074	0.374	0.412	0.009	0.061	0.467	0.472	0.457
Mean (batch)	0.739	0.121	0.086	0.388	0.448	-0.026	0.082	0.423	0.002	0.064	0.186	0.476	0.468	0.458
Mean (overall)	0.734	0.121	0.088	0.389	0.446	-0.028	0.082	0.425	-0.001	0.077	0.185	0.467	0.475	0.547
Linear	0.915	0.074	0.070	0.335	0.505	0.053	0.080	0.401	0.064	0.012	0.079	0.488	0.491	0.730
CPA	0.945	0.188	0.080	0.596	0.693	0.459	0.077	0.381	0.383	0.082	0.057	0.538	0.542	0.573
STATE	0.994	0.177	0.192	**0.816**	**0.775**	**0.504**	0.238	0.631	**0.476**	**0.007**	**0.055**	**0.785**	**0.799**	**0.733**
CellFlow	0.716	0.100	0.086	0.417	0.448	-0.035	0.079	0.412	-0.003	0.086	0.220	0.479	0.480	0.542
GEARS	0.911	0.068	0.075	0.433	0.581	0.210	0.072	0.369	0.206	0.012	0.076	0.459	0.464	0.502
Diffusion baselines and steering														
Squidiff	-10.80	0.043	0.085	0.394	0.401	-0.000	0.075	0.343	0.081	4.255	2.006	0.462	0.465	0.543
PerturbDiff (Scratch)	0.919	0.174	0.162	0.579	0.650	0.315	0.199	0.582	0.307	0.013	0.085	0.646	0.648	0.630
+ AdaCell (ours)	0.992	0.211	0.190	0.740	0.740	0.450	0.250	0.640	0.398	0.011	0.069	0.755	0.760	0.645
PerturbDiff (Finetuned)	0.912	0.234	0.178	0.723	0.679	0.367	0.236	0.617	0.406	0.016	0.084	0.712	0.723	0.580
+ AdaCell (ours)	**0.995**	**0.252**	**0.210**	0.770	0.750	0.470	**0.270**	**0.650**	0.416	0.011	0.069	0.770	0.775	0.659

D Additional Analyses

D.1 Hyperparameter Sensitivity

We examine the reward temperature τ and particle count K on Replogle with the Scratch checkpoint. We vary $\tau \in \{0.5, 1, 2\}$ at $K = 16$ and compare $K \in \{8, 16\}$ at $\tau = 1$. As shown in Table 9, the main configuration, $(\tau, K) = (1, 16)$, achieves the highest DEOver, PDCorr, and PDS-cos among

Table 7: Available metrics on PBMC. Metrics are averaged across test perturbations. MSE values are reported in units of 10^{-4}. Shaded rows indicate AdaCell with the corresponding frozen PerturbDiff checkpoint. Arrows indicate the preferred direction. Best and second-best results in each column are bold and underlined, respectively.

Method	R^2 ↑	DEOver ↑	DEPrec ↑	ES ↑	DirAgr ↑	LFCSpear ↑	AUPRC ↑	AUROC ↑	PDCorr ↑	MSE 10^{-4} ↓	MAE ↓	PDS-L1 ↑	PDS-L2 ↑	PDS-cos ↑
Non-diffusion baselines														
Mean	0.905	0.518	0.511	0.010	0.692	0.449	0.512	0.500	0.579	9.247	0.014	0.671	0.713	0.722
Mean (cell type)	0.992	0.547	0.509	-0.115	0.597	0.232	0.509	0.500	0.433	5.754	0.009	0.575	0.548	0.562
Mean (batch)	**0.999**	0.603	0.509	0.018	0.683	0.460	0.510	0.500	0.563	5.091	0.007	0.666	0.640	0.713
Mean (overall)	0.973	0.604	0.509	NaN	0.617	0.235	0.510	0.500	0.447	9.163	0.012	0.460	0.460	0.460
Linear	0.995	0.600	0.546	0.181	0.626	0.344	0.556	0.523	0.699	4.854	0.009	0.606	0.617	0.830
CPA	0.864	0.448	0.484	-0.164	0.507	0.347	0.484	0.500	0.164	120.0	0.054	0.626	0.666	0.649
STATE	0.997	0.557	0.545	-0.364	**0.786**	**0.599**	0.545	0.511	0.731	**1.528**	**0.005**	**0.950**	**0.939**	**0.898**
CellFlow	0.992	0.293	0.329	0.039	0.613	0.354	0.333	0.500	0.687	3.880	0.009	0.694	0.748	0.787
Diffusion baselines and steering														
Squidiff	-200.05	0.324	0.514	NaN	0.356	0.021	0.514	0.500	0.030	40270	1.976	0.458	0.458	0.458
PerturbDiff (Scratch)	0.997	0.618	0.552	0.273	0.713	0.493	0.577	0.573	0.887	1.757	0.006	0.874	0.879	0.889
+ AdaCell (ours)	0.997	**0.624**	0.582	0.210	0.764	0.520	0.612	0.598	**0.896**	1.740	0.006	0.876	0.883	0.895
PerturbDiff (Finetuned)	0.989	0.489	0.553	**0.364**	0.659	0.405	0.629	0.608	0.643	3.435	0.008	0.849	0.861	0.877
+ AdaCell (ours)	0.994	0.533	**0.585**	0.290	0.720	0.445	**0.635**	**0.610**	0.691	2.598	0.007	0.856	0.863	0.889

Table 8: Full metric suite on Tahoe100M. Metrics are averaged across test perturbations. MSE values are reported in units of 10^{-4}. Shaded rows indicate AdaCell with the corresponding frozen PerturbDiff checkpoint. Arrows indicate the preferred direction. Best and second-best results in each column are bold and underlined, respectively.

Method	R^2 ↑	DEOver ↑	DEPrec ↑	ES ↑	DirAgr ↑	LFCSpear ↑	AUPRC ↑	AUROC ↑	PDCorr ↑	MSE 10^{-4} ↓	MAE ↓	PDS-L1 ↑	PDS-L2 ↑	PDS-cos ↑
Non-diffusion baselines														
Mean	0.657	0.470	0.476	0.325	0.557	0.259	0.476	0.475	0.188	28.31	0.028	0.483	0.475	0.552
Mean (cell type)	0.922	0.553	0.477	0.021	0.648	0.542	0.479	0.476	0.416	7.523	0.011	0.644	0.652	0.823
Mean (batch)	0.661	0.465	0.475	0.445	0.549	0.250	0.474	0.471	0.201	28.42	0.028	0.461	0.466	0.544
Mean (overall)	0.665	0.392	0.476	0.438	0.555	0.255	0.475	0.472	0.210	27.63	0.028	0.470	0.469	0.542
Linear	0.943	0.553	0.544	0.356	0.716	0.481	0.569	0.597	**0.784**	3.775	0.010	0.836	0.827	0.836
CPA	0.894	0.455	0.475	0.176	0.666	0.435	0.474	0.470	0.383	8.144	0.012	0.615	0.556	0.531
STATE	0.804	0.548	0.520	0.299	0.662	0.641	0.518	0.504	0.487	23.18	0.021	0.786	0.842	0.777
CellFlow	0.640	0.200	0.292	0.322	0.599	0.277	0.299	0.471	0.247	28.52	0.029	0.572	0.529	0.502
Diffusion baselines and steering														
Squidiff	-654.0	0.383	0.548	NaN	0.471	0.256	0.546	0.471	0.010	61370	2.379	0.474	0.476	0.458
PerturbDiff (Scratch)	0.913	0.473	0.539	0.496	0.690	0.415	0.549	0.586	0.621	6.831	0.013	0.917	0.926	0.886
+ AdaCell (ours)	**0.995**	0.516	0.588	**0.505**	**0.754**	0.476	0.604	0.642	0.651	**3.347**	**0.009**	0.816	0.907	0.905
PerturbDiff (Finetuned)	0.843	0.531	0.564	0.447	0.602	0.650	0.582	0.621	0.592	11.40	0.018	0.737	0.833	**0.984**
+ AdaCell (ours)	0.932	**0.594**	**0.621**	0.490	0.663	**0.692**	**0.640**	**0.678**	0.640	9.920	0.014	**0.993**	**0.995**	0.982

the tested settings. Both lower and higher temperatures reduce these three response scores, although $\tau = 2$ lowers MSE from 0.011 to 0.010. Using eight particles retains DEOver of 0.206, PDCorr of 0.382, and PDS-cos of 0.624, compared with 0.211, 0.398, and 0.645 for 16 particles, while yielding a lower MSE of 0.009. These results support the main configuration for the three response metrics, while showing that the preferred setting depends on whether response recovery or mean squared error is prioritized.

Table 9: Sensitivity to reward temperature τ and particle count K on Replogle with the Scratch checkpoint. Metrics are averaged over test perturbations. The setting $(\tau, K) = (1, 16)$ matches the main-table AdaCell result. Arrows indicate the preferred direction.

τ	K	DEOver ↑	PDCorr ↑	PDS-cos ↑	MSE ↓
1.0	16	0.211	0.398	0.645	0.011
1.0	8	0.206	0.382	0.624	0.009
0.5	16	0.199	0.385	0.622	0.012
2.0	16	0.208	0.391	0.635	0.010

D.2 Biological Case Studies

We examine *PAF1* knockdown in HepG2, selected retrospectively for the largest absolute DEOver gain among the 380 test perturbations in the auxiliary analysis. PAF1 is a core component of the PAF1 complex (PAF1C), which regulates the transition of RNA polymerase II (Pol II) from promoter-proximal pausing to productive elongation. Biochemical studies connect PAF1C recruitment to P-TEFb activity and subsequent CDK12 recruitment, linking the complex to Pol II phosphorylation (Yu et al., 2015). Structural work further shows how PAF1C displaces the negative elongation factor NELF as an active elongation complex assembles (Vos et al., 2018). This regulatory role makes PAF1 perturbation informative beyond changes in the abundance of a single transcript. Cellular studies show that PAF1 depletion can increase pause release and activate a subset of enhancers, altering transcription at associated genes (Chen et al., 2015; 2017). The direction of its effect on pausing also differs across cellular contexts (Yu et al., 2015). These findings motivate evaluating whether a prediction recovers the identities and signs of the observed expression changes, rather than assuming that PAF1 loss uniformly increases or decreases transcription. Using the same frozen Scratch checkpoint and a 16-candidate budget, AdaCell improves DEOver from 0.2115 to 0.4956 and PDCorr from 0.4289 to 0.6258 relative to the reference. Differential-expression direction agreement increases from 0.7137 to 0.8612, and LFC Spearman correlation rises from 0.3224 to 0.7178, alongside a 19.8% reduction in MSE. The gains extend from identifying affected genes to recovering their response signs and relative effect ordering, illustrating improved prediction of a multigene response to the disruption of a transcriptional regulator. This retrospectively selected example supports expression-level agreement, not direct validation of pause-release or enhancer mechanisms in HepG2.

D.3 Limitations

AdaCell depends on the candidate populations produced by the frozen generator and the quality of training-derived response priors. Steering may be less effective when relevant response modes are poorly represented or when estimated priors do not transfer across cellular contexts. The translated-control anchor provides a useful distributional reference but may not capture perturbation-induced changes in covariance or cell-state composition. Repeated resampling can also reduce trajectory diversity, particularly under deterministic denoising, while maintaining multiple candidates increases inference costs. The reported improvements, therefore, do not imply uniformly better performance or guaranteed preservation of biological heterogeneity. Further evaluation across additional diffusion

architectures and cellular contexts is needed to establish the broader applicability of population steering.