CLAMP: A Mechanistic Probe of Regulatory Structure in Foundation Models under Single-Cell Perturbations

Amaya Gallagher-Syed [1 2] **Aaron Wenteler** [1] **Santiago A. Martinez** [3] **Gregory Slabaugh** [1]

Abstract

Do single-cell foundation models preserve regulatory structure in a form that supports perturbation prediction? Existing evaluations score these models by predictive performance, which cannot separate a failure of the embedding from a failure of the predictor reading it. We introduce CLAMP, which answers this mechanistically. Instead of predicting perturbed expression, CLAMP fits a differentiable dynamical model to single-cell perturbations and recovers an explicit gene regulatory network. We treat a sustained CRISPR perturbation as a clamp on a Hill-kinetics network: each perturbed gene is held at its observed value while the remaining genes relax to equilibrium. A single shared network is fit across all single-gene conditions, comparing predicted and observed states through a frozen encoder, with implicit differentiation through the fixed-point solver keeping the joint fit tractable. Because the network is recovered in gene space, the encoder can be swapped without changing the fitting procedure, so recovery quality through a given encoder measures how well its embedding preserves regulatory structure. Through the identity encoder, the recovered network validates against independent ChIP-seq and lineage signatures and predicts held-out double-gene perturbations with no combinatorial training data, confirming it recovers genuine regulatory structure. Reading recovery through four foundation models (scGPT, scPRINT, Stack, Tahoe-x1) layer by layer then gives the first mechanistic account of how each preserves regulatory structure. Code at `https://github.com/AmayaGS/CLAMP-Perturbseq`.

[1]Queen Mary University of London [2]Imperial College London [3]Independent Researcher. Correspondence to: Amaya Gallagher-Syed <a.gallaghersyed@imperial.ac.uk>.

Accepted at the 2026 Workshop on Generative and Agentic AI for Biology (ICML 2026)

1. Introduction

Single-cell foundation models (scFMs) are pretrained on large expression atlases and reused as general-purpose encoders, supporting tasks from cell-type annotation to *in-silico* target discovery and perturbation prediction (Theodoris et al., 2023; Cui et al., 2024). Transcriptomic perturbation effect prediction asks how a cell would respond to an intervention, which presupposes some knowledge of how genes regulate one another. While scFMs have been hypothesized to learn such regulatory structure, they underperform additive and mean baselines (Wenteler et al., 2025; Ahlmann-Eltze et al., 2025), and simple summary statistics match far more expensive learned models (Svensson et al., 2026), at odds with the scale of resources invested in pretraining them. While existing work highlights limitations in scFM performance, the absence of a mechanistic explanation leaves open whether these models preserve regulatory structure in a recoverable form.

This question motivates our framework. Prediction accuracy is confounded: it depends on both the embedding's information content and the predictor reading it. A weak predictor may miss present structure, a strong one may impose absent structure. We therefore invert the problem: rather than measuring prediction accuracy, we recover the regulatory mechanism that enables prediction. Recovering that mechanism requires interventional data, which Perturb-seq supplies: it profiles single cells under many single-gene perturbations against an unperturbed control, giving a steady-state snapshot for each condition. CLAMP treats each snapshot as a steady state of one shared regulatory network. We clamp each perturbed gene at its observed value and let the remaining genes relax to equilibrium, fitting the network in gene space. Predicted and observed responses are compared in the encoder's embedding, so the encoder alone governs what the fit can recover, without a downstream predictor. What we recover is an explicit network we can simulate, inspect, and read as regulatory structure. Recovering that structure depends on the encoder preserving regulatory geometry, defined as the directions genes move during a perturbation response. CLAMP can only fit the network along directions the encoder transmits, so recovery through a given scFM measures how well its embedding preserves this geometry.

1.1. Contributions

We introduce CLAMP, a framework that recovers an explicit gene regulatory network from single-cell perturbations and uses that recovery to probe what regulatory structure a frozen scFM preserves. It fits one shared network with arbitrary nonlinear kinetics directly on the steady-state snapshots Perturb-seq provides. Our contributions are:

A reframing of perturbation prediction as a coordinate clamp on a learned ODE. We clamp each perturbed gene at its observed value and solve for the steady state of the remaining genes.

A tractable differentiable steady-state framework. We fit one shared network across all single-gene conditions by matching predicted to observed per-condition steady states. Implicit differentiation through the fixed-point solver decouples gradient cost from solver depth, making the joint fit tractable at Perturb-seq gene-panel scale.

Recovery of a validated regulatory network from interventional data alone. Through the identity encoder, the adjacency recovered from single-gene perturbations validates against independent ChIP-seq and lineage signatures. Because the network is shared, the same fitted parameters predict held-out double-gene perturbations through additional clamps, with no combinatorial training data.

An encoder-agnostic interface that turns recovery into a mechanistic probe of scFMs. Predicted and observed states are compared only after projection through the encoder, so Identity, PCA, and frozen foundation models all substitute without changing the fitting procedure. Recovery quality through a given encoder then measures how well its embedding preserves regulatory structure, a property no predictor-based benchmark can isolate.

1.2. Related Work

Our question, whether an encoder transmits regulatory structure in a form a gene-space model can recover, draws on three bodies of work: methods that predict perturbation response, methods that recover explicit regulatory dynamics, and benchmarks that evaluate foundation models by prediction accuracy.

Regression-based perturbation prediction. A large body of recent work models perturbation response as a learned mapping from perturbation identifier and context to expression, $f_\theta : (p, x_{\mathrm{ctrl}}) \mapsto \hat{x}_p$. GEARS (Roohani et al., 2024) embeds this mapping in a graph neural network over a Gene Ontology relational graph. CPA (Lotfollahi et al., 2023) learns disentangled latent representations of perturbation, cell type, and covariate effects, predicting unseen combinations through latent-space arithmetic. AttentionPert (Bai et al., 2024) uses attention over perturbation embeddings and gene context, while fine-tuned single-cell foundation models (Cui et al., 2024; Theodoris et al., 2023) apply the same regression framing through pretrained embeddings. These methods produce predicted expression profiles but not a recovered regulatory model, so the structure that generates the response is never made explicit and cannot itself be examined.

Recovering regulatory structure from expression. A second body of work fits ordinary differential equations to single-cell expression data, the closest in spirit to ours, though designed for temporal trajectories rather than endpoint snapshots. Cell-MNN (von Bassewitz et al., 2026) fits a locally linearized ODE in PCA space, constraining it to linear latent dynamics. FLeCS (Bertin et al., 2025) retains nonlinear kinetics but requires pseudotime estimation to construct training trajectories and fits parameters per-condition rather than assembling a shared regulatory model. PerturbODE (Lin et al., 2025) uses a neural drift function where regulatory structure remains implicit in MLP weights rather than interpretable kinetic parameters, and PHOENIX (Hossain et al., 2024) fits Hill kinetics but targets bulk temporal forecasting rather than interventional structural recovery. Continuous Lyapunov models like Bicycle (Rohbeck et al., 2024) leverage steady-state data but rely on linear stochastic differential equations and do not scale to Perturb-seq gene panels. A related line of work treats the network as a static object assembled from auxiliary data and propagates perturbations through the fixed graph: CellOracle (Kamimoto et al., 2023) from scATAC-seq accessibility and TF motifs, scTenifoldKnk (Osorio et al., 2022) from control-expression manifolds. These methods establish that structure recovery is tractable, but each operates in a fixed coordinate space or takes the structure as input rather than learning it from perturbations. None recovers a shared network through an encoder that can itself be evaluated.

Evaluating foundation models by prediction. Single-cell foundation models have been evaluated systematically through regression-based benchmarks (Wenteler et al., 2025; Ahlmann-Eltze et al., 2025), which establish that models trained on observational expression underperform additive and mean baselines on perturbation prediction. Recent work identifies failure modes in these accuracy-based evaluations themselves (Viñas Torné et al., 2025; Miller et al., 2025), further weakening prediction accuracy as a probe of what a model has learned. These results report what foundation models fail at, but not why, leaving open whether the regulatory structure is present in the embedding at all.

2. Methods

We formulate a CRISPR perturbation as a steady-state constraint on a regulatory ODE (§2.1) and fit one network jointly across conditions, with a frozen encoder as the dif-

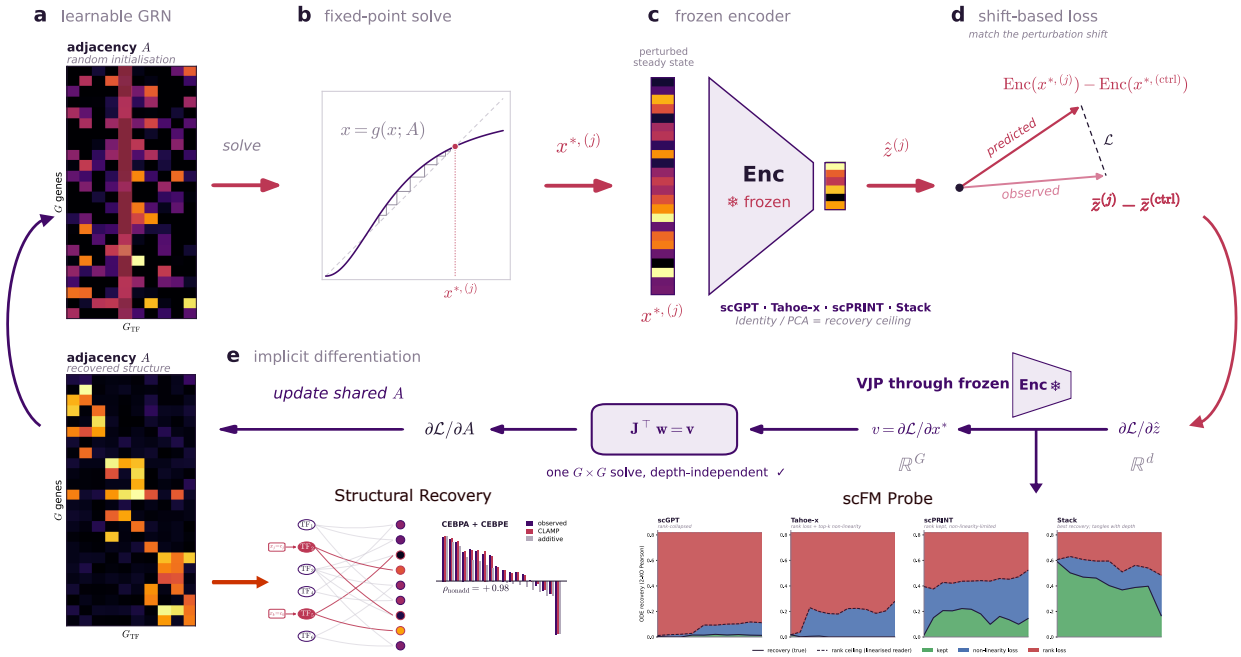

Figure 1. **Overview of the CLAMP pipeline.** A single optimization loop fits one shared, TF-restricted adjacency A. (**a**) Forward pass: a damped fixed-point solve maps A to each condition's clamped steady state. (**b**) A frozen differentiable encoder reads it to an embedding (**c**) swappable across scGPT, Tahoe-x1, scPRINT, Stack, with Identity/PCA setting the recovery ceiling. (**d**) A shift-based loss matches predicted to observed perturbation shifts. (**e**) Backward pass: the gradient is pulled back through the frozen encoder by a VJP and the fixed point is differentiated implicitly, yielding $\partial\mathcal{L}/\partial A$ in one depth-independent $G \times G$ solve that updates the shared A. The converged fit gives an explicit network (*structural recovery*; held-out CEBPA+CEBPE, $\rho^{\text{nonadd}} = +0.98$) and, read layer by layer, the *scFM probe* splitting recovery into kept, non-linearity loss, and rank loss against the linearized ceiling. Full walkthrough in Appendix B.

ferentiable metric comparing predicted and observed states (§2.2). Implicit differentiation through the solver makes the joint fit tractable (§2.3). Because the recovered network lives in gene space, any encoder can be swapped without changing the rest of the method, which turns recovery into a probe of the encoder. Section 2.4 defines the evaluation protocol and this probe. Figure 1 overviews the full pipeline.

2.1. Steady-state characterization of perturbation response

We model gene regulation as a deterministic ODE in gene space. For G genes with expression vector $\mathbf{x} \in \mathbb{R}^G$, the dynamics take the form

$$\frac{dx_i}{dt} = b_i + \sum_{j=1}^{G} A_{ij}\, h(x_j;\, n,\, K) - \gamma\, x_i, \tag{1}$$

where A_{ij} encodes the signed regulatory effect of gene j on gene i, b_i is the basal transcription rate, γ is a shared strictly positive mRNA decay rate, and $h(x;\, n,\, K) = x^n/(K^n + x^n)$ is the Hill function with cooperativity n and half-maximal constant K (Appendix D gives the biophysical derivation and discusses shared kinetic parameters). We collect the parameters as $\theta = (A,\, \mathbf{b},\, \gamma,\, n,\, K)$.

Steady-state recovery under coordinate constraints Two features of CRISPR Perturb-seq experiments justify Eq. 1 for recovery. CRISPR perturbations apply a sustained force rather than a transient pulse, driving target genes to stationary levels, and the long post-infection culture window (5–6 days in Norman et al. (2019) exceeds the protein-turnover timescale that rate-limits network equilibration (Zecha et al., 2018; Schwanhäusser et al., 2011). We therefore model the observed perturbed state not as a snapshot along a trajectory but as a constrained steady state, a fixed point where the perturbed coordinate is clamped at its observed value c_j while the remaining genes settle into a self-consistent equilibrium. This assumption transforms the dynamical recovery into an algebraic root-finding problem defined by the network's topology. Given the observed clamped value c_j, the steady state $x^{*,(j)}$ satisfies:

$$\begin{aligned} x_i^{*,(j)}(\theta) &= \frac{1}{\gamma}\left(b_i + \sum_k A_{ik}\, h(x_k^{*,(j)}; n, K)\right) \quad \text{for } i \neq j, \\ x_j^{*,(j)}(\theta) &= c_j, \end{aligned} \tag{2}$$

where the first expression enforces the equilibrium condition $dx_i/dt = 0$ for all unclamped genes, and the second represents the coordinate clamp. We resolve this system numerically using damped fixed-point iteration, reimposing the clamp c_j at each step to ensure the final state is consistent with the intervention. This approach generalizes naturally to multi-gene perturbations by applying multiple simultaneous coordinate constraints, which we exploit for compositional prediction in Section 2.4.

2.2. Joint multi-condition recovery via a differentiable encoder

Perturb-seq measures single-cell gene expression under a battery of single-gene CRISPR perturbations. Let $\mathcal{J}$ index the perturbed genes, with $\mathcal{X}^{(j)} \subset \mathbb{R}^G$ denoting cells receiving perturbation j and $\mathcal{X}^{(\text{ctrl})}$ the unperturbed controls. We assume one regulatory model governs the cell's response to every intervention, and therefore fit a single θ jointly across all conditions, rather than per-condition as in prior nonlinear-ODE work (Bertin et al., 2025). CLAMP compares predicted and observed states through a frozen encoder $\text{Enc} : \mathbb{R}^G \to \mathbb{R}^d$, which plays two roles. During recovery it is a differentiable metric: predicted and observed states are matched in its representation, and since $\mathbf{x}^{*,(j)}(\theta)$ depends on θ, the encoder must be differentiable in its input so that gradients reach θ through it. Because θ is nonetheless recovered in gene space, A stays interpretable and comparable across encoder choices: Identity ($\text{Enc}(\mathbf{x}) = \mathbf{x}$) recovers the unencoded baseline, and any other differentiable encoder substitutes without changing the fitting procedure. This is the encoder's second role, developed in Section 2.4: recovery quality then becomes a measurement of the encoder itself.

Shift-based steady-state loss. We fit θ by matching predicted steady states to observations through three loss components. First, we match the perturbation-induced *shift* between encoded representations rather than the encoded states themselves,

$$\mathcal{L}_{\text{pert}}(\theta) = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} \Big\| \big[\text{Enc}(\mathbf{x}^{*,(j)}(\theta)) - \text{Enc}(\mathbf{x}^{*,(\text{ctrl})}(\theta))\big] - \big[\bar{\mathbf{z}}^{(j)} - \bar{\mathbf{z}}^{(\text{ctrl})}\big] \Big\|_2^2. \tag{3}$$

where $\bar{\mathbf{z}}^{(j)}$ is the mean encoding of cells under perturbation j and $\bar{\mathbf{z}}^{(\text{ctrl})}$ is the control mean. The shift formulation removes the baseline component of $\text{Enc}(\mathbf{x}^{*,(\text{ctrl})})$ that is shared across conditions and would otherwise dilute the regulatory signal. Second, we anchor the predicted wild-type steady state to the observed control mean via $\mathcal{L}_{\text{ctrl}}(\theta) = \|\text{Enc}(\mathbf{x}^{*,(\text{ctrl})}(\theta)) - \bar{\mathbf{z}}^{(\text{ctrl})}\|_2^2$, fixing a reference point that the shift formulation alone leaves undetermined. Third, we add a weak L_1 sparsity penalty $\|A\|_1$ encoding the prior that biological regulatory networks are sparse. The full objective is

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{pert}}(\theta) + \lambda_{\text{ctrl}}\, \mathcal{L}_{\text{ctrl}}(\theta) + \lambda_{\text{sparse}}\, \|A\|_1, \tag{4}$$

with hyperparameters reported in Appendix E.

Adjacency parameterization.

We restrict the recovered adjacency A to fit edges only for TFs with direct interventional evidence. By structurally masking all unperturbed genes, we limit the network's effective degrees of freedom to the perturbed subset. This reduces the parameter count from G^2 to $G \cdot G_{\text{TF}}$, where $G_{\text{TF}} = |\mathcal{J}|$ ($G_{\text{TF}} \ll G$), bringing it to a favorable parameter-to-observation regime. We use a dense parameterization with the TF mask reapplied after each optimizer step to prevent AdamW second-moment statistics from drifting masked entries off zero.

2.3. Implicit differentiation through the fixed-point solver

Backpropagating through the unrolled fixed-point iteration is impractical at the scale of this work: the activation trace must be retained for every perturbation condition until the backward pass, with both memory and time scaling linearly in iteration count. Following the approach established in Bai et al. (2019) and Blondel et al. (2022), we instead compute gradients analytically via the implicit function theorem. The forward iteration is run without an autograd graph and returns only the converged $\mathbf{x}^*$. Writing $g(\mathbf{x}, \theta) = (\mathbf{b} + A\, h(\mathbf{x}; n, K))/\gamma$, the converged state satisfies $F(\mathbf{x}^*, \theta) = \mathbf{x}^* - g(\mathbf{x}^*, \theta) = \mathbf{0}$. For a scalar loss $\mathcal{L}$ with upstream gradient $\mathbf{v} = \partial\mathcal{L}/\partial\mathbf{x}^*$, the parameter gradient is

$$\frac{\partial \mathcal{L}}{\partial \theta} = -\mathbf{w}^\top \frac{\partial F}{\partial \theta}, \quad J^\top \mathbf{w} = \mathbf{v}, \quad \text{with} \quad J = I - \tfrac{1}{\gamma} A\, \text{diag}(h'(\mathbf{x}^*)). \tag{5}$$

We solve the linear system $J^\top \mathbf{w} = \mathbf{v}$ once per condition and then evaluate the analytic vector-Jacobian product (VJP), which reads: $-\mathbf{w}^\top\, \partial F/\partial\theta$ for each parameter group. The Hill derivatives have closed forms, so all parameter gradients are evaluated exactly at the fixed point. This linear system is well-posed provided J is non-singular, a condition typically satisfied when the forward iteration converges to a stable attractor. Full closed-form expressions for $\partial\mathcal{L}/\partial A$, $\partial\mathcal{L}/\partial\mathbf{b}$, $\partial\mathcal{L}/\partial\gamma$, $\partial\mathcal{L}/\partial n$, and $\partial\mathcal{L}/\partial K$ are reported alongside a discussion of numerical stability in Appendix F. We enforce the clamp on the gradient side by zeroing the entry of $\mathbf{v}$ for clamped genes and replacing the corresponding row of J with $\mathbf{e}_j^\top$, so backpropagation contributes no parameter update from coordinates whose values are set by the intervention rather than predicted by θ. Backward cost then reduces to a single $G \times G$ solve per condition, independent of forward solver depth, and we wrap the entire fixed-point solve in a `torch.autograd.Function` so it composes seamlessly with gradients through the frozen encoder.

2.4. Evaluation

We evaluate two things: whether the recovered network reflects genuine regulatory structure, and what recovery reveals about the encoders. The first is tested by held-out compositional prediction and by structural validation against independent biological references. The second runs the same recovery through each encoder and reads it as a measurement. We evaluate two baselines and four foundation-model encoders. Identity is the unencoded baseline and PCA, fitted on control cells, a linear compression baseline. scGPT (Cui et al., 2024), scPRINT (Kalfon et al., 2025), Stack (Dong et al., 2026), and Tahoe-x1 (Gandhi

et al., 2025) are pretrained transformer-based scFMs, used as frozen wrappers mapping an expression vector to a cell-level embedding. Per-encoder vocabulary mapping, input format, and depth-token details are in Appendix G.

Held-out prediction. We evaluate on Norman 2019 (Norman et al., 2019), to our knowledge the only published Perturb-seq dataset with double-gene perturbations at the scale required for per-pair compositional evaluation (see details on data and preprocessing in Appendix H). The framework is fit on single-gene perturbations alone, with a subset held out for single-gene validation (SKI val) and all double-gene perturbations held out for compositional evaluation (DKI val). A k-gene perturbation imposes k simultaneous coordinate constraints on the same Hill ODE, with unclamped genes determined by Eq. 2. The same learned θ resolves unseen combinatorial perturbations through the same solver, requiring no additional training data or architectural extension.

For each held-out pair (j, k) we compute the Pearson correlation between predicted and observed shifts relative to control,

$$\rho_{j,k} = \text{Pearson}\Big(\mathbf{x}^{*,(j,k)}(\theta) - \mathbf{x}^{*,(\text{ctrl})}(\theta), \bar{\mathbf{x}}^{(j,k)} - \bar{\mathbf{x}}^{(\text{ctrl})}\Big), \tag{6}$$

with the perturbed coordinates excluded, and the same correlation restricted to the non-additive residual $\rho^{\text{nonadd}}_{j,k}$, obtained by replacing each shift in Eq. 6 with its difference from the additive prediction $\hat{\mathbf{s}}^{(j,k)}_{\text{add}} = \bar{\mathbf{x}}^{(j)} + \bar{\mathbf{x}}^{(k)} - 2\bar{\mathbf{x}}^{(\text{ctrl})}$. Aggregate $\rho_{j,k}$ measures overall fit to the predicted shift, the bulk of which on Perturb-seq is additive: $\hat{\mathbf{s}}_{\text{add}}$ alone outperforms fine-tuned scFMs on this metric (Ahlmann-Eltze et al., 2025). The non-additive correlation $\rho^{\text{nonadd}}_{j,k}$ isolates the interaction component (synergy, buffering, suppression), zero for any additive predictor and positive only when the recovered model captures structure beyond single-gene superposition. We additionally report Top-1/Top-5 retrieval and the Frobenius norm and effective rank of A as diagnostics.

Structural validation. We assess the recovered adjacency against three independent reference datasets: curated hematopoietic lineage signatures, ChEA 2022 ChIP-seq target sets (Lachmann et al., 2010), and ENCODE TF ChIP-seq 2015 target sets (Consortium et al., 2012). All non-lineage libraries were queried through Enrichr (Kuleshov et al., 2016), with hypergeometric enrichment of the top-25 recovered targets per TF (Appendix L). As these references are unused during fitting, agreement with the recovered A is independent evidence of structural recovery rather than a consistency check.

The encoder as a probe. Holding the gene-space ODE, the loss, and the fitting procedure fixed and varying only the encoder isolates the encoder's contribution to recovery. The encoder enters the recovery gradient through its input Jacobian: the upstream term $\mathbf{v} = \partial\mathcal{L}/\partial\mathbf{x}^*$ of Section 2.3 is the encoded loss pulled back through $\partial\text{Enc}/\partial\mathbf{x}$ at the predicted state, so the fit can move A only along directions the encoder transmits out of gene space. Recovery through a given encoder is therefore a measurement of how well its local geometry preserves the directions in which perturbations move cells. We characterize it two ways: local geometric readouts of the encoder, computed without any fit, and a recovery-units decomposition separating what the embedding discards from what our gene-space ODE cannot read.

Geometric readouts. For each single-gene condition j we form the observed shift $\Delta x_j = \mathbb{E}[x \mid \text{SKI}_j] - \bar{x}_{\text{ctrl}}$ in log1p HVG space, stack these into a matrix S, and take its top-K principal directions $\{v_k\}$ ($K = 48$), with regulatory weights $w_k = \sum_j (\Delta x_j \cdot v_k)^2$ recording how much perturbation energy lies along each. The signal that reaches the ODE at layer L is the directional derivative $D_{v_k}\text{Enc}_L(x_i)$ of that layer's readout along v_k, evaluated at control cells $\{x_i\}$. This is the same encoder Jacobian-vector product that enters the recovery gradient, taken forward rather than backward (Appendix J). From it we form two ratios. *Linearity* asks whether the layer acts as a consistent linear map on regulatory directions, the property the gene-space ODE requires, as the regulatory-weighted mean cross-cell consistency of the readout direction,

$$\Lambda_L = \frac{1}{\sum_k w_k} \sum_k w_k \left\| \frac{1}{n_a} \sum_{i=1}^{n_a} \frac{D_{v_k}\text{Enc}_L(x_i)}{\|D_{v_k}\text{Enc}_L(x_i)\|_2} \right\|_2, \tag{7}$$

with 1 a perfectly linear response and a chance floor of $1/\sqrt{n_a}$. *Regulatory rank* asks whether the K directions stay separable in the embedding, a necessary condition for carrying a gene-space network. We measure regulatory rank as the participation ratio of the readout vectors' singular values, normalized by K so that 1 marks an injective map and $1/K$ a full collapse. Comparing this against a random-subspace baseline isolates compression specific to the regulatory subspace from compression the encoder applies everywhere (Appendix J).

Calibration and decomposition. A near-zero recovery score remains ambiguous between two causes: the encoder may have discarded the regulatory directions, or kept them in a non-linear form that a single gene-space ODE cannot invert. The second is a limitation of our approach, not of the embedding, and therefore we seek to quantify it. We anchor the scale on Identity, which leaves gene space untouched: the network recovered there validates against the references above, so its recovery C reflects true regulatory structure and fixes the top of the scale. To split the remaining deficit, we replace each encoder, at each layer, by its first-order Taylor expansion at the control-mean cell $\mathbf{x}_0$,

$$\text{Enc}_{\text{lin}}(\mathbf{x}) = \text{Enc}(\mathbf{x}_0) + J\,(\mathbf{x} - \mathbf{x}_0), \qquad J = \left.\frac{\partial\text{Enc}}{\partial\mathbf{x}}\right|_{\mathbf{x}_0}, \tag{8}$$

and refit the same ODE through this fixed affine map, applied identically to predictions and targets (Appendix K).

Because $\mathrm{Enc}_{\mathrm{lin}}$ is linear it transmits gene-space shifts consistently, so its recovery $\mathrm{rec}_{\mathrm{lin}}$ is limited by the rank of J alone. Writing $\mathrm{rec}_{\mathrm{true}}$ for recovery through the unmodified encoder, the deficit partitions in recovery units as

$$\underbrace{C - \mathrm{rec}_{\mathrm{lin}}}_{\text{rank loss}} + \underbrace{\mathrm{rec}_{\mathrm{lin}} - \mathrm{rec}_{\mathrm{true}}}_{\text{non-linearity loss}} + \underbrace{\mathrm{rec}_{\mathrm{true}}}_{\text{recovered}} = C. \tag{9}$$

Rank loss is regulatory signal the encoder's linear action fails to keep separable, a limitation of the embedding rather than of how we read it. Non-linearity loss is signal the encoder retains but presents non-linearly, which our gene-space ODE cannot follow even though a more flexible model in principle could. Because $\mathrm{Enc}_{\mathrm{lin}}$ is an empirical refit pinned at $\mathbf{x}_0$ while the perturbed state drifts from it, the split is a calibrated estimate: rank loss is an upper bound and non-linearity loss a lower bound on the true quantities. A genuinely linear encoder gives $\mathrm{Enc}_{\mathrm{lin}} = \mathrm{Enc}$, so Identity and PCA register zero non-linearity loss, the sanity check on the split. Computing this in recovery units also catches non-smooth non-linearity, such as discrete tokenization, that a Jacobian-based linearity score can under-report. Identity and PCA fix the references, a linear full-rank encoder and a linear compression against which the foundation models are read.

3. Experimental Results

Following the evaluation protocol of §2.4, we test whether the recovered network is genuine regulatory structure (§3.1, §3.2) and what recovery reveals about each encoder (§3.3).

3.1. Held-out prediction

Table 1 reports prediction, retrieval, and learned structure for the two baselines, the four foundation-model encoders, and the additive baseline. Through Identity, CLAMP recovers a Hill-kinetics adjacency from single-KI data alone that predicts held-out double-KI shifts at 0.782 ± 0.008 aggregate Pearson, captures their non-additive component at $\rho^{\mathrm{nonadd}} = 0.467$ (95% bootstrap CI $[0.437, 0.496]$, $n = 129$ pairs), reaches 0.706 ± 0.011 on single-KI prediction, retrieves the matching observed shift as nearest neighbor for 77.3% of held-out pairs, and recovers an adjacency of Frobenius norm $\|A\|_F = 2.12$ at effective rank 69.4. Top-1 retrieval well above chance ($\approx 1\%$) shows predictions discriminate between conditions, and the moderate Frobenius norm at sub-maximal effective rank (bounded by the 102 TF columns) indicates the adjacency concentrates signal on a structured subset of regulatory directions.

The additive baseline is indistinguishable from Identity within error bars (0.799 ± 0.124 vs 0.782 ± 0.008) but has zero non-additive component by construction. Aggregate Pearson is dominated by the additive part of double-gene responses, so ρ^{nonadd} is the metric that isolates the interaction structure distinguishing mechanistic recovery from arithmetic composition. PCA tracks Identity at reduced magnitude with a denser, higher-rank adjacency ($\|A\|_F = 4.35$, rank 77.2). The foundation models, read at their final layer, separate sharply. We report this layer as the default embedding a downstream task would consume, though recovery varies with depth (see 3.3). scGPT and Tahoe-x1 collapse to near-zero aggregate recovery (DKI val -0.011 and -0.047), scGPT with a near-empty adjacency at near-maximal rank ($\|A\|_F = 0.23$, rank 98.4) and Tahoe-x1 with a denser low-rank one (3.02, rank 46.8). scPRINT and Stack place higher (0.171 and 0.157), Stack retrieving the correct pair for 64.5% of cases despite the low aggregate score. The positive ρ^{nonadd} values for scGPT and Tahoe-x1 (0.373, 0.361) sit beside aggregate recovery at chance, so they reflect residual direction alone and are not compositional success.

Stratifying held-out pairs by Norman 2019 functional category (Figure 2, left) shows a structured pattern. Recovery is strongest on Redundant, Suppressor, and Epistasis pairs (median projection ~ 0.5–0.6), intermediate on Approximately additive and Neomorphic pairs (~ 0.4), and weakest on both Strong synergy classes and Potentiation pairs (~ 0.1–0.25), where the joint phenotype is qualitatively new and not implicit in either TF's single-gene response. The ODE recovers non-additive structure cleanly on redundant interactions (CEBPA+CEBPE, $r = 0.98$, $\rho^{\mathrm{nonadd}} = +0.98$) and epistatic masking (KLF1+CEBPA, $r = 0.96$, $\rho^{\mathrm{nonadd}} = +0.79$), and weakens on dissimilar-phenotype synergy (DUSP9+SNAI1, $r = 0.71$, $\rho^{\mathrm{nonadd}} = -0.01$), where high aggregate Pearson hides limited residual recovery (Figure 2, right).

3.2. Structural validation

Predictive accuracy on held-out perturbations does not identify which adjacency from the consistency family has been recovered. We assess structural recovery against three independent reference types for the Identity encoder, on three master hematopoietic regulators chosen to span orthogonal axes of the K562 bipotential state. KLF1 and SPI1 anchor its two mutually exclusive developmental fates (erythroid and myeloid, respectively (Miller & Bieker, 1993; Scott et al., 1994)), while IRF1 provides an independent axis of external immune response (Fujita et al., 1989). Top-25 recovered targets per TF are shown as a combined subnetwork (Figure 3A).

Each TF's top-25 enriches strongly against its matched lineage signature: KLF1 against erythroid ($48.6\times$, $p_{\mathrm{bonf}} = 2.1\mathrm{e}{-15}$), SPI1 against myeloid monocyte ($44.4\times$, $p_{\mathrm{bonf}} = 6.6\mathrm{e}{-15}$), and IRF1 against the combined interferon response signature ($38.9\times$, $p_{\mathrm{bonf}} = 2.8\mathrm{e}{-11}$). Across all eight lineage signatures the matched lineage reaches $p_{\mathrm{bonf}} < 10^{-10}$ for every TF, while each wrong-lineage and negative-control signature (T cell, B cell, cell

Table 1. **Prediction, retrieval, and learned structure on Norman 2019.** SKI val, DKI val, retrieval, and learned-structure columns are mean ± SD across 5 seeds; ρ^{nonadd} is reported as per-pair bootstrap mean and 95% CI over $n = 129$ held-out double-gene pairs. Additive baseline included as reference: ρ^{nonadd} is zero by construction for additive predictors and isolates the interaction component. Foundation-model encoders are read at their final layer. Per-layer recovery is reported in §3.3 and Figure 4.

	Prediction			Retrieval (%)		Learned structure	
Condition	SKI val	DKI val	DKI val ρ^{nonadd}	Top-1	Top-5	$\|A\|_F$	Eff. rank
Identity	0.706 ± 0.011	0.782 ± 0.008	$0.467\ [0.437, 0.496]$	77.3 ± 1.3	85.7 ± 2.6	2.12 ± 0.06	69.4 ± 2.3
PCA	0.540 ± 0.069	0.582 ± 0.067	$0.312\ [0.284, 0.341]$	78.6 ± 1.9	83.7 ± 1.1	4.35 ± 0.97	77.2 ± 1.3
scGPT	-0.001 ± 0.006	-0.011 ± 0.019	$0.373\ [0.319, 0.425]$	7.6 ± 1.7	17.8 ± 3.0	0.23 ± 0.13	98.4 ± 1.5
scPRINT	0.090 ± 0.022	0.171 ± 0.032	$0.283\ [0.258, 0.308]$	41.0 ± 15.2	62.0 ± 12.3	3.56 ± 0.49	46.4 ± 1.7
Stack	0.132 ± 0.014	0.157 ± 0.015	$0.287\ [0.251, 0.323]$	64.5 ± 12.9	76.5 ± 9.2	13.17 ± 4.57	53.6 ± 1.7
Tahoe-x1	-0.015 ± 0.001	-0.047 ± 0.003	$0.361\ [0.310, 0.412]$	20.6 ± 1.4	35.0 ± 2.9	3.02 ± 0.03	46.8 ± 1.0
Additive	–	0.799 ± 0.124	–	–	–	–	–

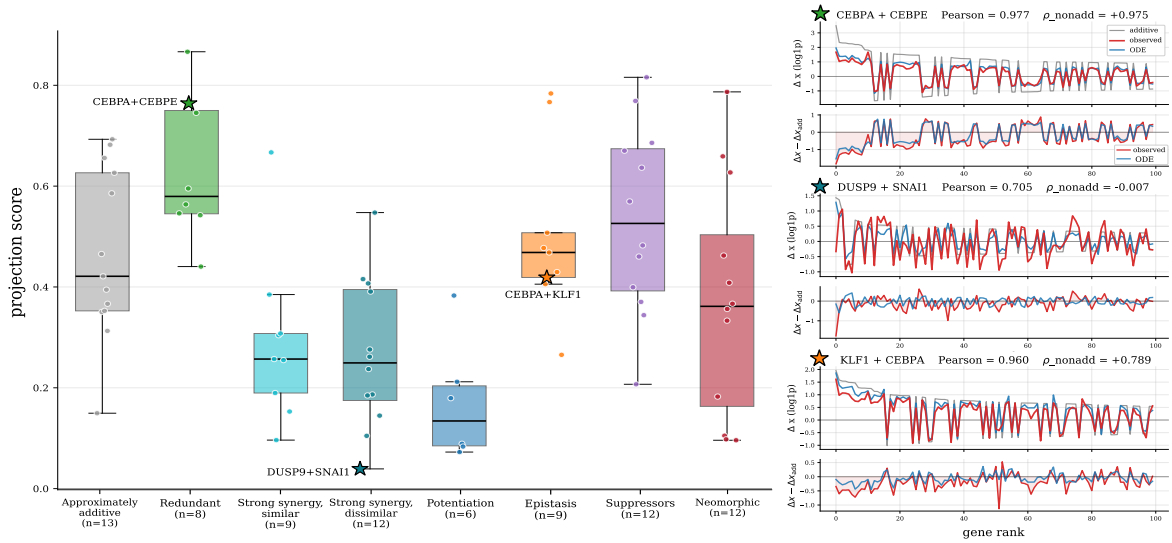

Figure 2. **Non-additive recovery on held-out double-KI pairs, stratified by Norman 2019 gene categories. Left:** Projection of the predicted residual onto the observed residual direction, normalized by $\|e_{\text{obs}}\|$. 1 = predicted residual matches observed in magnitude and direction; 0 = no component along observed; negative = anti-aligned. Boxes show median and IQR, circles are individual held-out pairs. Stars mark the three pairs shown on the right. **Right:** per-pair waterfall plots. Top sub-panel shows Δx from control across the top 100 genes ranked by $|\Delta x_{\text{add}}|$, comparing observed (red), ODE prediction (blue), and additive baseline (grey). Bottom sub-panel shows the residual $\Delta x - \Delta x_{\text{add}}$ for observed (red) and ODE (blue). Per-panel y-axis ranges differ.

cycle) stays above $p_{\text{bonf}} = 0.1$ (Figure 3B). Independently curated ChIP-seq references confirm these matches, each TF's top-25 enriching against same-TF target sets in ChEA 2022 and ENCODE 2015 (Figure 5, in Appendix L).

Beyond matched-TF hits, each TF's top-25 also enriches against ChIP target sets of its known co-regulators in the matched cellular context (Appendix L). KLF1's recovered targets enrich against GATA1 and GATA2 K562 ChIP (canonical erythroid co-regulators), IRF1's against STAT2 K562 ChIP (the IRF1–STAT2 axis in type I interferon signaling), and SPI1's against TAL1, RUNX1, and LYL1 ChIP in mouse hematopoietic progenitors, three of seven members of the *heptad* TF complex that co-regulates hematopoietic stem and progenitor programs alongside SPI1 (Wilson et al., 2010; Subramanian et al., 2023). The framework fits each TF independently with no information about co-regulatory relationships, so these structures emerge from the single shared adjacency: fitting one A across all conditions lets information from each perturbation constrain entries beyond the directly perturbed gene.

3.3. Per-layer recovery decomposition

A foundation model that recovers little of the gene regulatory network leaves two possibilities. The regulatory directions may be absent from its embedding, or present in a geometry the gene-space ODE cannot read. Recovery alone cannot separate these, so in Figure 4 we read recovery at each layer against the linearized rank ceiling of §2.4, splitting each model's deficit into rank loss and non-linearity loss. The gap from ODE recovery (green) up to the ceiling is

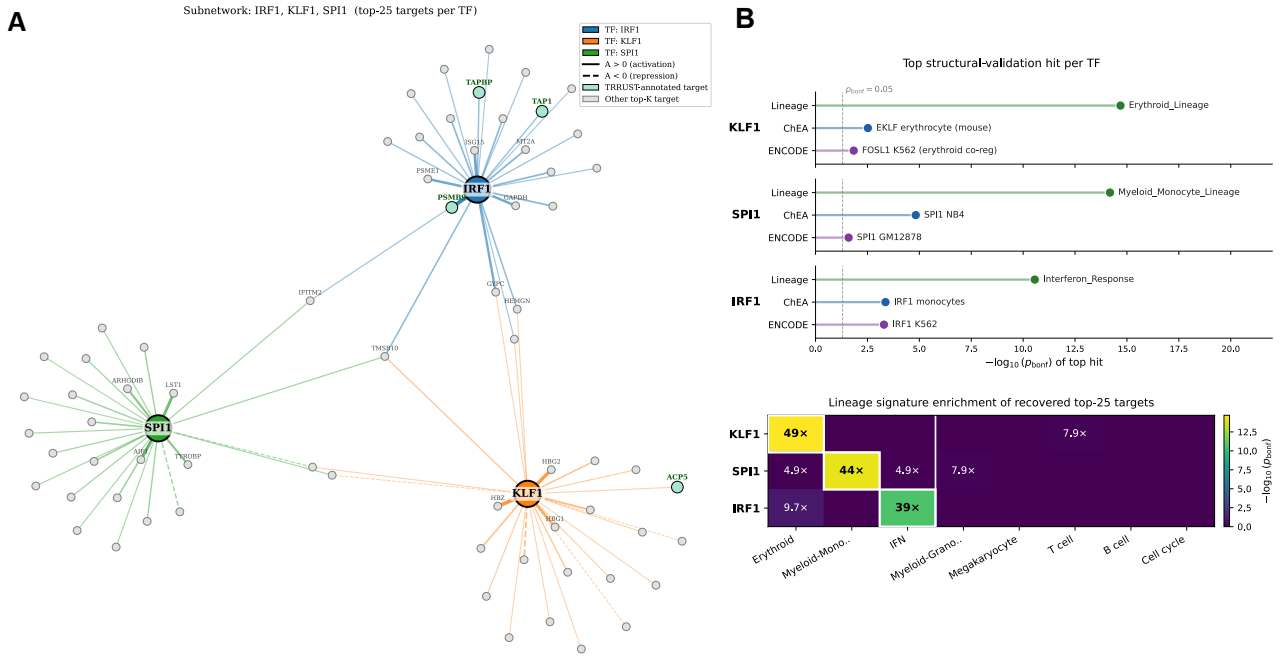

Figure 3. **Structural validation of the recovered adjacency. (A)** Combined subnetwork of the top-25 recovered targets per master TF (*IRF1*, *KLF1*, *SPI1*). Edges colored by source; solid = activation ($A > 0$), dashed = repression. TRRUST-annotated targets shown with colored fill. **(B)** Top hit per TF across three reference types (lineage signatures, ChEA 2022, ENCODE 2015), reported as $-\log_{10}(p_{\text{bonf}})$ of the strongest matched-TF or matched-lineage gene set. Dashed line: $p_{\text{bonf}} = 0.05$.

non-linearity loss (blue): directions the encoder retains but presents in a form the gene-space ODE cannot follow. The gap from the ceiling up to Identity recovery ($DKI = 0.78$) is rank loss (red): regulatory directions the embedding fails to keep separable. The two no-fit geometric readouts, computed from the encoder's input Jacobian on the top-K shift directions, corroborate these splits independent of an ODE fit (Figure 4, bottom): linearity Λ_L measures whether each regulatory direction's encoded image holds consistent across control cells, and regulatory rank measures whether the K directions stay separable in the embedding. The first tracks the non-linearity band, the second the rank loss.

The four models occupy distinct regimes. scGPT recovers nothing at any depth beneath a rank ceiling that stays near zero throughout: almost the entire deficit is rank loss, the regulatory directions are not linearly separable in the embedding at any layer (low regulatory rank at every depth, Figure 4d), even though its projection stays near-linear ($\Lambda_L \approx 0.93$). Tahoe-x1 also recovers near to nothing, but its ceiling rises to about 0.2, indicating a part of its deficit is non-linearity on top of a dominant rank loss. The directions are present, yet collapsed too far to carry a gene-space network. scPRINT recovers up to 0.22 at its strongest layers beneath a ceiling near 0.45, splitting its deficit across genuine recovery, a substantial non-linearity band, and rank loss. Stack recovers most, 0.60 at its first layer falling to 0.17 at depth. Its rank ceiling stays high, between 0.50 and 0.65 across all layers, so a substantial regulatory subspace remains separable (the highest regulatory rank of the four, Figure 4d). The growing gap between recovery and that ceiling is non-linearity loss, which widens with depth as the encoder de-linearizes (Λ_L falling from 0.95 to 0.45), alongside a steady rank loss above the ceiling.

Stack and Tahoe-x1 draw the cleanest contrast on whether the regulatory subspace stays separable. Stack keeps high rank at every depth, so a large regulatory subspace stays separable, and it loses recovery mainly as the encoder de-linearizes with depth (widening blue band), structure a more flexible reader could potentially still recover. Tahoe-x1 collapses rank toward zero from the first layer, so the directions are no longer separable in the embedding and no gene-space model can recover them, whatever its flexibility. scGPT and scPRINT make the complementary contrast in linearity. scGPT keeps the most linear projection of the four yet recovers nothing, its deficit due entirely to low rank. scPRINT is the least linear at its early layers yet recovers a real signal, because the rank it preserves leaves the fit something to follow. Linearity without rank recovers near nothing, and rank without linearity recovers only what the gene-space ODE can read through a single Jacobian.

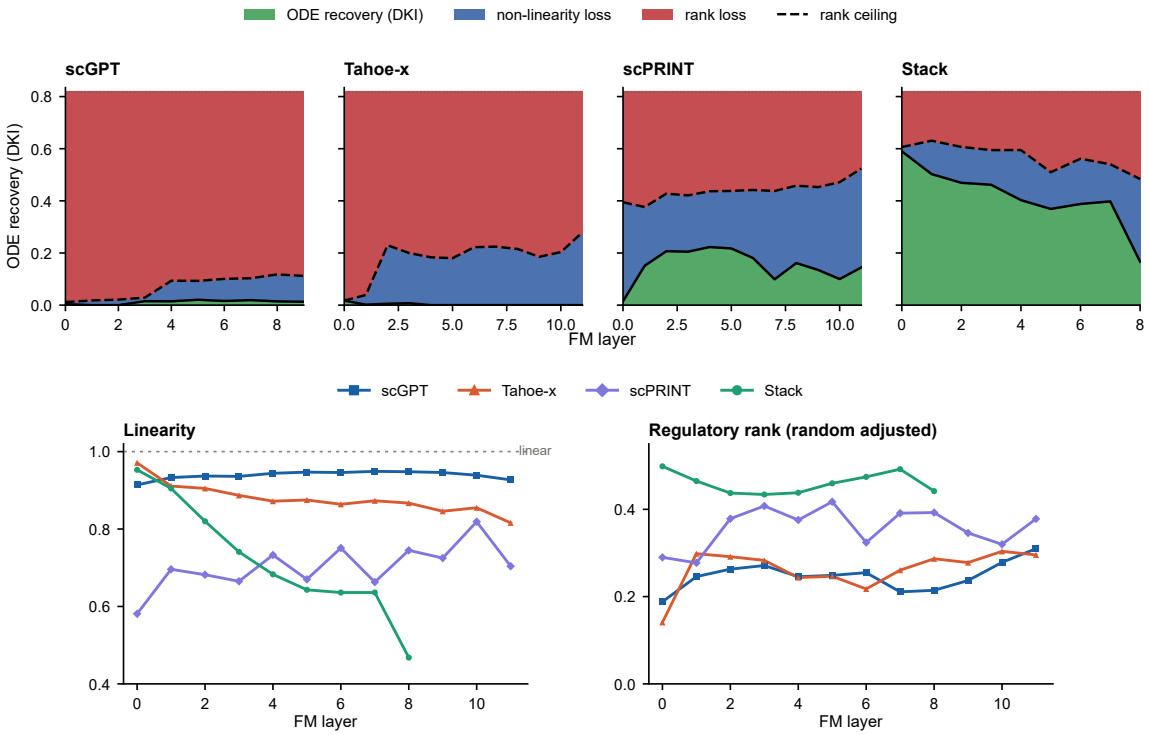

Figure 4. **Per-layer recovery decomposition and its geometric corroboration for the four foundation-model encoders.** Held-out double-KI recovery is read at every layer. **Top:** the solid line is recovery through the unmodified encoder ($\mathrm{rec}_{\mathrm{true}}$, green); the dashed line is the rank ceiling from refitting through the encoder's local linearization at the control-mean cell ($\mathrm{rec}_{\mathrm{lin}}$, Eq. 9). The blue band is non-linearity loss, signal the encoder retains but presents in a form a single local linearization cannot follow; the red band above the ceiling is rank loss, regulatory directions the embedding does not keep separable. The top of the scale is Identity recovery C. **Bottom:** the two no-fit readouts, from the encoder's input Jacobian on the top-K shift directions. *Linearity* (Λ_L, Eq. 7; 1 linear) is each direction's image consistency across control cells, and where it falls the blue band widens. *Regulatory rank* (random-adjusted participation ratio; 1 injective, $1/K$ collapsed) tracks rank loss, and where it drops the red band dominates. The two are independent.

4. Conclusion

CLAMP reframes a sustained CRISPR perturbation as a coordinate clamp on a shared regulatory ODE and fits one adjacency jointly across all single-gene conditions. Implicit differentiation through the solver makes this tractable, and because the adjacency is held in gene space, any frozen encoder substitutes as the metric. Through Identity, CLAMP predicts held-out double-KI shifts at 0.78 aggregate Pearson, captures their non-additive component at $\rho^{\mathrm{nonadd}} = 0.47$, aligns with curated lineage signatures and ChIP-seq references at $p_{\mathrm{bonf}} < 10^{-10}$, and respects co-regulatory groupings it was never given.

As a probe, CLAMP splits each encoder's deficit into rank loss and non-linearity loss. Whether rank or non-linearity dominates the deficit is set by the pretraining objective, not by scale or data volume. scGPT keeps almost none of the regulatory rank: masked reconstruction on observational cells carries no pressure to keep interventional directions separable. scPRINT recovers a real but modest signal split across both, consistent with a design geared toward gene-network inference. Stack preserves the most regulatory rank, consistent with its cross-cell attention, which denoises each cell's readout against its context and so keeps distinct regulatory directions separable. Its latent identifiability regularizer may also help, concentrating the condition signal into a per-cell-set constant that aligns with the shift between condition means that CLAMP targets. Tahoe-x1 is the instructive case: trained by masked reconstruction like scGPT but on the perturbative Tahoe-100M compendium, it collapses the regulatory rank from its first layer all the same. Here, interventional data does not rescue a reconstruction objective.

These results highlight that loss of regulatory information is objective-specific, not intrinsic to foundation models per se. They point pretraining toward a concrete target: keeping the regulatory subspace separable. A natural next step is therefore to train on an objective that explicitly preserves separability and test whether it yields more recoverable embeddings, a question this probe is now equipped to answer. We give a fuller treatment of scope and limitations in Appendix C.

Acknowledgment

We wish to thank Dr. Dimitris Volteras for discussion on the chemical master equation, mechanistic models and causality. A.G.S. receives funding from the Wellcome Trust. A.W. receives funding from the UKRI/BBSRC Collaborative Training Partnership in AI for Drug Discovery and Queen Mary University of London. This work was also supported by the Engineering and Physical Sciences Research Council [grant number EP/Y009800/1], through funding from Responsible Ai UK (KP0016) and acknowledges the support of the National Institute for Health Research Barts Biomedical Research Centre (NIHR203330). This research was conducted using Queen Mary's Apocrita HPC facility, supported by QMUL Research-IT (King et al., 2017).

References

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

Bai, D., Ellington, C. N., Mo, S., Song, L., and Xing, E. P. Attentionpert: accurately modeling multiplexed genetic perturbations with multi-scale effects. *Bioinformatics*, 40 (Supplement_1):i453–i461, 2024.

Bai, S., Kolter, J. Z., and Koltun, V. Deep equilibrium models. *Advances in Neural Information Processing Systems*, 32, 2019.

Bertin, P., Viviano, J. D., Tejada-Lapuerta, A., Wang, W., Bauer, S., Theis, F. J., and Bengio, Y. A scalable gene network model of regulatory dynamics in single cells. *arXiv preprint arXiv:2503.20027*, 2025.

Blondel, M., Berthet, Q., Cuturi, M., Frostig, R., Hoyer, S., Llinares-López, F., Pedregosa, F., and Vert, J.-P. Efficient and modular implicit differentiation. *Advances in Neural Information Processing Systems*, 35:5230–5242, 2022.

Consortium, E. P. et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57, 2012.

Cui, H., Wang, C., Maan, H., Pang, K., Luo, F., Duan, N., and Wang, B. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature methods*, 21(8):1470–1480, 2024.

Dong, M., Adduri, A., Gautam, D., Carpenter, C., Shah, R., Ricci-Tam, C., Kluger, Y., Burke, D. P., and Roohani, Y. H. Stack: In-context learning of single-cell biology. *bioRxiv*, 2026.

Fujita, T., Kimura, Y., Miyamoto, M., Barsoumian, E. L., and Taniguchi, T. Induction of endogenous ifn-α and ifn-β genes by a regulatory transcription factor, irf-1. *Nature*, 337(6204):270–272, 1989.

Gandhi, S., Javadi, F., Svensson, V., Khan, U., Jones, M. G., Yu, J., Merico, D., Goodarzi, H., and Alidoust, N. Tahoe-x1: Scaling perturbation-trained single-cell foundation models to 3 billion parameters. *bioRxiv*, 2025.

Han, H., Cho, J.-W., Lee, S., Yun, A., Kim, H., Bae, D., Yang, S., Kim, C. Y., Lee, M., Kim, E., et al. Trrust v2: an expanded reference database of human and mouse transcriptional regulatory interactions. *Nucleic acids research*, 46(D1):D380–D386, 2018.

Hay, S. B., Ferchen, K., Chetal, K., Grimes, H. L., and Salomonis, N. The human cell atlas bone marrow single-cell interactive web portal. *Experimental Hematology*, 68:51–61, 2018.

Heumos, L., Ji, Y., May, L., Green, T., Zhang, X., Wu, X., Ostner, J., Peidli, S., Schumacher, A., Hrovatin, K., Müller, M., Chong, F., Sturm, G., Tejada, A., Dann, E., Dong, M., Bahrami, M., Gold, I., Rybakov, S., Namsaraeva, A., Moinfar, A., Zheng, Z., Roellin, E., Mekki, I., Sander, C., Lotfollahi, M., Schiller, H. B., and Theis, F. J. Pertpy: an end-to-end framework for perturbation analysis, August 2024. URL https://www.biorxiv.org/content/10.1101/2024.08.04.606516v1.

Hossain, I., Fanfani, V., Fischer, J., Quackenbush, J., and Burkholz, R. Biologically informed neuralodes for genome-wide regulatory dynamics. *Genome Biology*, 25(1):127, 2024.

Jassal, B., Matthews, L., Viteri, G., Gong, C., Lorente, P., Fabregat, A., Sidiropoulos, K., Cook, J., Gillespie, M., Haw, R., et al. The reactome pathway knowledgebase. *Nucleic acids research*, 48(D1):D498–D503, 2020.

Kalfon, J., Samaran, J., Peyré, G., and Cantini, L. scprint: pre-training on 50 million cells allows robust gene network predictions. *Nature Communications*, 16(1):3607, 2025.

Kamimoto, K., Stringa, B., Hoffmann, C. M., Jindal, K., Solnica-Krezel, L., and Morris, S. A. Dissecting cell identity via network inference and in silico gene perturbation. *Nature*, 614(7949):742–751, 2023.

King, T., Butcher, S., and Zalewski, L. *Apocrita - High Performance Computing Cluster for Queen Mary University of London*. March 2017. doi: 10.5281/zenodo.438045. URL https://doi.org/10.5281/zenodo.438045.

Kuleshov, M. V., Jones, M. R., Rouillard, A. D., Fernandez, N. F., Duan, Q., Wang, Z., Koplev, S., Jenkins, S. L., Jagodnik, K. M., Lachmann, A., et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.

Lachmann, A., Xu, H., Krishnan, J., Berger, S. I., Mazloom, A. R., and Ma'ayan, A. Chea: transcription factor regulation inferred from integrating genome-wide chip-x experiments. *Bioinformatics*, 26(19):2438–2444, 2010.

Liberzon, A., Birger, C., Thorvaldsdóttir, H., Ghandi, M., Mesirov, J. P., and Tamayo, P. The molecular signatures database hallmark gene set collection. *Cell systems*, 1(6): 417–425, 2015.

Lin, Z., Chang, S., Zweig, A., Kang, M., Azizi, E., and Knowles, D. A. Interpretable neural odes for gene regulatory network discovery under perturbations. *arXiv preprint arXiv:2501.02409*, 2025.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Lotfollahi, M., Klimovskaia Susmelj, A., De Donno, C., Hetzel, L., Ji, Y., Ibarra, I. L., Srivatsan, S. R., Naghipourfar, M., Daza, R. M., Martin, B., et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular systems biology*, 19(6): MSB202211517, 2023.

Miller, H. E., Mejia, G. M., Leblanc, F. J., Wang, B., Swain, B., and Camillo, L. P. d. L. Deep learning-based genetic perturbation models do outperform uninformative baselines on well-calibrated metrics. *bioRxiv*, pp. 2025–10, 2025.

Miller, I. J. and Bieker, J. J. A novel, erythroid cell-specific murine transcription factor that binds to the caccc element and is related to the krüppel family of nuclear proteins. *Molecular and cellular biology*, 13(5):2776–2786, 1993.

Norman, T. M., Horlbeck, M. A., Replogle, J. M., Ge, A. Y., Xu, A., Jost, M., Gilbert, L. A., and Weissman, J. S. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793, 2019.

Novershtern, N., Subramanian, A., Lawton, L. N., Mak, R. H., Haining, W. N., McConkey, M. E., Habib, N., Yosef, N., Chang, C. Y., Shay, T., et al. Densely interconnected transcriptional circuits control cell states in human hematopoiesis. *Cell*, 144(2):296–309, 2011.

Osorio, D., Zhong, Y., Li, G., Xu, Q., Yang, Y., Tian, Y., Chapkin, R. S., Huang, J. Z., and Cai, J. J. sctenifoldknk: An efficient virtual knockout tool for gene function predictions via single-cell gene regulatory network perturbation. *Patterns*, 3(3), 2022.

Peidli, S., Green, T. D., Shen, C., Gross, T., Min, J., Garda, S., Yuan, B., Schumacher, L. J., Taylor-King, J. P., Marks, D. S., Luna, A., Blüthgen, N., and Sander, C. scPerturb: harmonized single-cell perturbation data. *Nature Methods*, 21(3):531–540, March 2024. ISSN 1548-7105. doi: 10.1038/s41592-023-02144-y. URL https://www.nature.com/articles/s41592-023-02144-y. Publisher: Nature Publishing Group.

Rohbeck, M., Clarke, B., Mikulik, K., Pettet, A., Stegle, O., and Ueltzhöffer, K. Bicycle: Intervention-based causal discovery with cycles. In *Causal Learning and Reasoning*, pp. 209–242. PMLR, 2024.

Roohani, Y., Huang, K., and Leskovec, J. Predicting transcriptional outcomes of novel multigene perturbations with gears. *Nature Biotechnology*, 42(6):927–935, 2024.

Schoenfelder, S. and Fraser, P. Long-range enhancer–promoter contacts in gene expression control. *Nature Reviews Genetics*, 20(8):437–455, 2019.

Schwanhäusser, B., Busse, D., Li, N., Dittmar, G., Schuchhardt, J., Wolf, J., Chen, W., and Selbach, M. Global quantification of mammalian gene expression control. *Nature*, 473(7347):337–342, 2011.

Scott, E. W., Simon, M. C., Anastasi, J., and Singh, H. Requirement of transcription factor pu. 1 in the development of multiple hematopoietic lineages. *Science*, 265(5178): 1573–1577, 1994.

Subramanian, S., Thoms, J. A., Huang, Y., Cornejo-Páramo, P., Koch, F. C., Jacquelin, S., Shen, S., Song, E., Joshi, S., Brownlee, C., et al. Genome-wide transcription factor–binding maps reveal cell-specific changes in the regulatory architecture of human hspcs. *Blood*, 142(17):1448–1462, 2023.

Svensson, V., Khan, U., Heydari, H., Ubas, A. A., Thomas, N., Merico, D., Goodarzi, H., Yu, J., Alidoust, N., and Gandhi, S. Back to basics: Observed statistics are sufficient to predict drug responses. *bioRxiv*, 2026.

Tenen, D. G. Disruption of differentiation in human cancer: Aml shows the way. *Nature Reviews Cancer*, 3(2):89–101, 2003.

Theodoris, C. V., Xiao, L., Chopra, A., Chaffin, M. D., Al Sayed, Z. R., Hill, M. C., Mantineo, H., Brydon, E. M., Zeng, Z., Liu, X. S., et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

Viñas Torné, R., Wiatrak, M., Piran, Z., Fan, S., Jiang, L., Teichmann, S. A., Nitzan, M., and Brbić, M. Systema: a framework for evaluating genetic perturbation response

prediction beyond systematic variation. *Nature Biotechnology*, pp. 1–10, 2025.

von Bassewitz, J.-P., Pervez, A., Fumero, M., Robinson, M., Karaletsos, T., and Locatello, F. Learning explicit single-cell dynamics using ode representations. *International Conference on Learning Representation*, 2026.

Wenteler, A., Occhetta, M., Branson, N., Curean, V., Huebner, M., Dee, W., Connell, W., Chung, S. P., Hawkins-Hooker, A., Ektefaie, Y., Córdova, C. M. V., and Gallagher-Syed, A. Perteval-scFM: Benchmarking single-cell foundation models for perturbation effect prediction. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=t04D9bkKUq.

Wilson, N. K., Foster, S. D., Wang, X., Knezevic, K., Schütte, J., Kaimakis, P., Chilarska, P. M., Kinston, S., Ouwehand, W. H., Dzierzak, E., et al. Combinatorial transcriptional control in blood stem/progenitor cells: genome-wide analysis of ten major transcriptional regulators. *Cell stem cell*, 7(4):532–544, 2010.

Wolf, F. A., Angerer, P., and Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19(1):15, February 2018. ISSN 1474-760X. doi: 10.1186/s13059-017-1382-0. URL https://doi.org/10.1186/s13059-017-1382-0.

Zecha, J., Meng, C., Zolg, D. P., Samaras, P., Wilhelm, M., and Kuster, B. Peptide level turnover measurements enable the study of proteoform dynamics. *Molecular & Cellular Proteomics*, 17(5):974–992, 2018.

Supplementary Material

This supplementary material provides methodological details deferred from the main text.

- Appendix A (supporting Section 1.2) gives an overview of various pre-existing methods that use dynamical systems approaches to model gene regulatory networks and how our CLAMP fits into this.
- Appendix B (supporting Section 2) gives an overview of the CLAMP framework and how it is optimized.
- Appendix C contains a detailed description of the scope, assumptions and limitations of CLAMP.
- Appendix D (supporting Section 2.1) derives the Hill function from cooperative TF binding and discusses our use of shared kinetic parameters.
- Appendix E (supporting Section 2.2) reports the optimizer, loss-weight, solver, and training-schedule values used across all experiments.
- Appendix F (supporting Section 2.3) gives the implicit-gradient derivation, closed-form Jacobian and Hill derivatives, per-parameter gradient expressions, and numerical-stability and verification details.
- Appendix G (supporting Section 2.2) documents the per-encoder wrappers, including vocabulary mapping, input format, depth-token handling, and the diagnostic checks each wrapper passes before being used in experiments.
- Appendix H describes the Norman 2019 dataset and the preprocessing pipeline used to prepare it for recovery.
- Appendix I details the dense adjacency parameterization, including the transcription-factor restriction, the reapplication of the TF mask under AdamW, and the initialization scheme.
- Appendix L reports the full structural validation results against curated hematopoietic lineage signatures, ChEA 2022 ChIP-seq target sets, ENCODE TF ChIP-seq 2015 target sets, MSigDB Hallmark, Reactome 2022, and TRRUST literature-curated targets, alongside a discussion of the recovered local regulatory neighborhood and methodological notes on multiple-testing correction and background gene-set choice.
- Appendix J (supporting Section 2.4) gives the formal definitions of the encoder-geometry readouts: the regulatory-shift basis and weights, the layerwise Jacobian-vector product, the linearity score Λ_L, and the participation-ratio regulatory rank with its random-subspace baseline.
- Appendix K (supporting Section 2.4) details the linearized-reader refit $\mathrm{Enc}_{\mathrm{lin}}$ and the decomposition of each encoder's recovery deficit into rank loss and non-linearity loss against the linearized rank ceiling.
- Appendix M documents the curated hematopoietic lineage signature library used as the primary structural validation reference, including per-signature gene lists and curation sources, and specifies the disease-context and generic catch-all denylist used to filter MSigDB Hallmark and Reactome 2022 prior to enrichment testing.

A. Overview of dynamical-system approaches for gene regulatory network modeling

Table 2. Comparison of dynamical systems methods for gene regulatory network inference

Method	**Dynamical System Type**	**Temporal Scope**	**Latent Space Constraint**	**Interpretability Mechanism**	**Unseen Perturbations**
Cell-MNN	Locally Linearized ODE	Trajectory / Interpolation	PCA Subspace	Linear Transition Matrix	×
FLECS	Message-Passing ODE	Pseudotime / Trajectory	None (Gene Space)	Sparse Prior Network Weights	×
PerturbODE	Neural Drift ODE	Trajectory / Steady-State	None (Gene Space)	Post-hoc Jacobian Extraction	×
PHOENIX	Hill-Langmuir Neural ODE	Bulk Temporal Microarray	None (Gene Space)	Imposed Functional Forms	×
Bicycle	Linear SDE (Ornstein-Uhlenbeck)	Steady-State Snapshot	Unconstrained Latent Space	Explicit Causal Matrix via Lyapunov	✓
CLAMP	Hill-Kinetics Fixed-Point ODE	Steady-State Snapshot	Encoder-Agnostic	Explicit Kinetic Parameters	✓

B. Full Pipeline

Figure 1 summarizes the full CLAMP pipeline as a single optimization loop wrapped around three moving parts: a shared regulatory adjacency A, an inner solver that maps A to a constrained steady state for each perturbation condition, and a frozen encoder through which predicted and observed states are compared. The loop has a forward pass (panels a–d), which carries the current adjacency through the solver and the encoder to a scalar loss, and a backward pass (panel e), which returns the gradient of that loss to the adjacency by implicit differentiation through the fixed-point solver. The two panels beneath the loop show what the converged fit produces: the recovered network structure and the per-layer scFM probe. The schema is a roadmap for the formal treatment in Section 2, and we walk through it in the order information flows at fitting time, then describe inference.

Panel a (learnable GRN). The parameter object is the shared adjacency $A \in \mathbb{R}^{G \times G_{\mathrm{TF}}}$, restricted to the $G_{\mathrm{TF}} = 102$ columns of perturbed transcription factors (Section 2.2). It is initialized at random (top) and is the only structural quantity the outer loop updates; the recovered structure (bottom) is the fixed point of the optimization, and the purple loop around the panel denotes the repeated update of this single shared A across all conditions.

Panel b (fixed-point solve). For each condition j, the inner solver maps A to the constrained steady state $\mathbf{x}^{*,(j)}(\theta)$ by damped fixed-point iteration of the steady-state map $\mathbf{x} = g(\mathbf{x}; \theta)$ (Eq. 2), reimposing the clamp $x_j^{*,(j)} = c_j$ at every iteration so the converged state lies exactly on the clamp plane (cobweb inset). A k-gene perturbation is resolved by imposing k simultaneous clamps on the same equations. The iteration runs without an autograd graph and returns only the converged $\mathbf{x}^*$, decoupling memory cost from solver depth.

Panel c (frozen encoder). The converged steady state is read through a frozen, differentiable encoder Enc, which maps the gene-space state $\mathbf{x}^{*,(j)}$ to an embedding $\hat{\mathbf{z}}^{(j)}$. The encoder is the only interchangeable component: the four foundation models probed here (scGPT, Tahoe-x1, scPRINT, Stack) and the linear Identity and PCA baselines that set the recovery ceiling all substitute at this position without touching the rest of the pipeline.

Panel d (shift-based loss). The loss matches the perturbation-induced shift rather than the absolute encoded state: the predicted shift $\mathrm{Enc}(\mathbf{x}^{*,(j)}) - \mathrm{Enc}(\mathbf{x}^{*,(\mathrm{ctrl})})$ is compared to the observed shift $\bar{\mathbf{z}}^{(j)} - \bar{\mathbf{z}}^{(\mathrm{ctrl})}$ (Eq. 3), with the control-anchor and L_1 sparsity terms of Eq. 4 completing the objective. The shift formulation removes the condition-shared baseline that would otherwise dilute the regulatory signal.

Panel e (implicit differentiation). The backward pass closes the loop. The upstream gradient $\partial\mathcal{L}/\partial\hat{\mathbf{z}} \in \mathbb{R}^d$ is pulled back through the frozen encoder by a vector-Jacobian product to $\mathbf{v} = \partial\mathcal{L}/\partial\mathbf{x}^* \in \mathbb{R}^G$, after which the implicit-function-theorem system $J^\top \mathbf{w} = \mathbf{v}$ (Eq. 5) yields $\partial\mathcal{L}/\partial A$ in a single $G \times G$ solve per condition, independent of how many iterations the forward solver took. The resulting gradient flows back to panel a and updates the shared A. The encoder enters only at the loss and in this pullback, and is held fixed throughout, so swapping it changes which projection of the steady state the loss sees but leaves the solver and the gradient machinery untouched.

Structural recovery and the scFM probe. The two panels beneath the loop show the outcomes the converged fit is evaluated on. *Structural recovery* (lower left) reads the recovered A as an explicit network: the subnetwork around the master regulators, and the held-out double-gene pair CEBPA+CEBPE, whose predicted (CLAMP) and observed shifts agree on both the additive and the non-additive component ($\rho^{\mathrm{nonadd}} = +0.98$). No double-gene data enters the fit; the recovered θ comes from single-gene conditions alone (Section 3.1, Section 3.2). The *scFM probe* (lower right) runs the same loop through each foundation model layer by layer and decomposes its recovery deficit into kept recovery, non-linearity loss, and rank loss against the linearized rank ceiling (Section 3.3), the comparative readout that is the central result of the paper.

Inference. At inference only the inner loop (panel b) is exercised. A new perturbation, possibly on multiple target genes, is imposed as a set of simultaneous clamps on the fitted θ, and the predicted response is read from the converged steady state. This is the mechanism behind the held-out compositional results of Section 3.1: a k-gene perturbation is resolved by the same solver that produced the single-gene training states, with no architectural extension and no additional training data.

C. Scope, assumptions and limitations

The framework rests on four assumptions that delimit where it applies and where it does not.

1. **Steady-state response.** The steady-state assumption (Section 2.1) requires that the cellular response to perturbation has equilibrated by the time of measurement. This is well-supported for the 5–6 day post-infection sampling window of Norman 2019 given the protein turnover times of human cell lines, but would not hold for short-timescale perturbations such as drug treatments sampled within a few hours, where the observed state lies along a trajectory rather than at a fixed point.

2. **Single deterministic attractor per condition.** The framework treats each condition as a single deterministic attractor and fits to the condition mean. It does not model cell-to-cell variability within a condition, multistability, or the case where a single perturbation drives the population towards two or more distinct fates.

3. **Adjacency support restricted to perturbed TFs.** The recovered adjacency is restricted to the columns of perturbed transcription factors, so A_{ij} is identified only for $j \in \mathcal{J}$. This restriction is both a methodological choice and a reflection of what the data supports: only the perturbed TFs carry direct interventional evidence, and fitting columns for unperturbed genes would require the model to extrapolate regulatory effects from observational variation alone. The restriction also keeps the parameter count tractable, reducing the adjacency from $G^2 \approx 6.5 \times 10^6$ to $G \cdot G_{\mathrm{TF}} \approx 2.6 \times 10^5$ and yielding the roughly one-to-one parameter-to-observation regime discussed in Section 2.2. Edges between genes that were never directly perturbed are not estimated, and the framework is silent on regulators outside the intervention battery.

4. **Hill kinetics as the instantiation choice.** The kinetic form used in this work is fixed to Hill activation with a shared cooperativity (n), decay rate (γ) and half-maximal constant (K) across all regulators (Appendix D). The framework itself does not require this: any differentiable kinetic form admitting a fixed point (mass action, Michaelis–Menten, or learned neural drift functions, among others) can be substituted without changing the fitting procedure, and Hill activation is the choice made in this work for biophysical interpretability and parameter parsimony. The Hill choice is a deliberate compression of biological reality: real transcription factors vary in their cooperativity and binding affinities, and individual transcripts exhibit heterogeneous decay rates. The signed scalar A_{ij} and shared kinetic scalars collapse these into a single effective coupling. This compression is what makes joint multi-condition fitting tractable at the parameter counts of Section 2.2, and the structural and predictive results of Section 3.1 and Section 3.2 indicate it is empirically adequate at this scale, but we do not claim that the recovered A disentangles direct from indirect regulation or recovers the underlying biophysical kinetics of any individual edge.

The fourth assumption is the source of the identifiability concern noted in Section 3.2, and the structural validation results presented there are the response: independent ChIP-seq and lineage signatures provide external evidence that the recovered A aligns with the true regulatory structure rather than merely a steady-state-equivalent surrogate.

D. Hill kinetics: biophysical derivation and parameter choices

This appendix derives the Hill function from cooperative TF-promoter binding equilibria, justifies our use of shared scalar parameters in the synthetic benchmark, and notes the framework's generality over kinetic forms beyond Hill.

D.1. Derivation of the Hill function

Consider a promoter with n binding sites for a transcription factor (TF) at intracellular concentration x. Under the cooperative-binding assumption, the promoter is transcriptionally active only when all n sites are simultaneously occupied by the TF. Writing the cooperative binding reaction as $\mathrm{P} + n\,\mathrm{TF} \rightleftharpoons \mathrm{P}_{\text{bound}}$, the equilibrium dissociation constant is

$$K_d = \frac{[\mathrm{P}]\,x^n}{[\mathrm{P}_{\text{bound}}]}. \tag{10}$$

The fraction of time the promoter spends in the active state is therefore

$$\frac{[\mathrm{P}_{\text{bound}}]}{[\mathrm{P}] + [\mathrm{P}_{\text{bound}}]} = \frac{x^n/K_d}{1 + x^n/K_d} = \frac{x^n}{K^n + x^n} = h(x;\, n,\, K), \tag{11}$$

where $K = K_d^{1/n}$ is the concentration at which the promoter is half-occupied. The Hill function is sigmoidal in x, with steepness controlled by n: at $n = 1$ it reduces to Michaelis-Menten saturation, and as $n \to \infty$ it approaches a hard threshold at $x = K$. We use $n = 2$ as a default, corresponding to dimerized cooperative binding, the most common regime in eukaryotic transcriptional regulation.

D.2. Per-regulator versus multi-regulator nonlinearity

When a target gene i is regulated by multiple transcription factors $\{x_j\}_{j \in \text{regs}(i)}$, two natural functional forms are available. The *per-regulator* form, used in this work,

$$\frac{dx_i}{dt} \propto \sum_j A_{ij}\, h(x_j;\, n,\, K), \tag{12}$$

applies the Hill nonlinearity to each regulator independently before summing. The *multi-regulator* form,

$$\frac{dx_i}{dt} \propto h\left(\sum_j A_{ij}\, x_j;\, n,\, K\right), \tag{13}$$

applies a single Hill activation to a linear combination of regulators. Per-regulator nonlinearity is consistent with how cooperativity is modeled in deterministic systems biology when each TF binds independently at distinct sites, and it makes the individual edge coefficients A_{ij} interpretable as marginal regulatory contributions. The multi-regulator form, used by FLeCS (Bertin et al., 2025), is a coarser approximation that compresses regulator interactions into a sigmoidal threshold on a weighted sum and obscures the contribution of any single TF to the response. Both forms are admissible in our framework without changing the recovery procedure. We adopt the per-regulator form for interpretability.

D.3. Generality over kinetic forms

The recovery framework treats the production term in Equation 1 as a black-box differentiable function. Hill kinetics is one instantiation, chosen for biological interpretability. Other admissible parameterizations include the following.

- **Michaelis-Menten** kinetics, the $n = 1$ special case of Hill, suited to non-cooperative regulation.
- **Mass action** kinetics, $\frac{dx_i}{dt} \propto \sum_j A_{ij} x_j$, the linear special case in which each regulator contributes proportionally to its concentration.
- **Generalized Hill** with regulator-specific scalars $h(x_j;\, n_j,\, K_j)$, relaxing the shared-scalar simplification at the cost of additional parameters.

- **Neural drift**, $\frac{dx_i}{dt} \propto f_\phi(\mathbf{x})$ with f_ϕ a differentiable neural network, abandoning interpretability of individual edges in exchange for representational flexibility (cf. PerturbODE, (Lin et al., 2025)).

The implicit-differentiation backward of Section 2.3 requires only that $\partial F/\partial \mathbf{x}$ and $\partial F/\partial \theta$ be computable, either in closed form (Hill, Michaelis-Menten, mass action) or via autograd through the production term (neural drift). The shift-based loss of Equation 3 and the joint multi-condition objective are independent of the kinetic form. Choosing among parameterizations is therefore a modeling decision rather than a methodological constraint.

E. Training hyperparameters

This appendix lists the hyperparameter values used in our experiments. Values that vary by encoder or by gene-panel size are noted. The exact values used in the camera-ready experiments are stored in `configs/perturbseq.yaml` and `configs/synthetic_data.yaml`; what follows reflects the settings used throughout development and held stable across reported runs.

E.1. Optimizer

We use AdamW (Loshchilov & Hutter, 2017) with two parameter groups. The adjacency parameters (the active entries of A) receive weight decay; the basal rates $\mathbf{b}$ and the log-parameterized scalars $\log \gamma$, $\log n$, $\log K$ receive zero weight decay. This avoids drifting the kinetic scalars away from biologically reasonable values and avoids drifting $\mathbf{b}$ away from its initialization at the control mean.

Parameter	Value
Learning rate	1×10^{-3} (Tahoe-x1: 1×10^{-4})
β_1, β_2	$0.9, 0.999$
Weight decay (adjacency parameters)	1×10^{-4}
Weight decay (other parameters)	0
Gradient clipping (max norm)	5.0

The learning rate is 1×10^{-3} for all encoders except Tahoe-x1, which is fit at 1×10^{-4}. Tahoe-x1 recovers slightly better at 1×10^{-4} than at the shared 1×10^{-3}, so we report it at the better-performing rate rather than handicap it at a setting it underperforms at. Its recovery is near zero even at this rate (Table 1), so the Tahoe-x1 outcome reflects the geometry of its embedding rather than an under-tuned optimizer.

E.2. Loss weights

The total objective $\mathcal{L} = \mathcal{L}_{\text{pert}} + \lambda_{\text{ctrl}} \mathcal{L}_{\text{ctrl}} + \lambda_{\text{sparse}} \|A\|_1$ uses weights given below. The L_1 sparsity weight is held constant from epoch zero; we do not use a warmup schedule.

Weight	Value	Notes
λ_{ctrl}	1.0	Control anchor
λ_{sparse}	1×10^{-5}	Weak L_1 on A, applied from epoch 0

E.3. Fixed-point solver settings

The forward solver uses damped fixed-point iteration with the constraints reimposed at every step (Section 2.1). The backward pass uses the implicit gradient with Tikhonov regularization on the linear solve (Appendix F).

Parameter	Value	Notes
Damping factor α	0.5	In $x_{k+1} = \alpha\, g(x_k) + (1-\alpha)\, x_k$
Max iterations (cold start)	200	First 20 epochs
Max iterations (warm start)	50	Subsequent epochs
Convergence tolerance	1×10^{-5}	On $\|x_{k+1} - x_k\|_\infty$
Clamp bounds	$[10^{-6},\ 50.0]$	Numerical stability of $h(x)$
Backward Tikhonov ϵ	1×10^{-8}	See Equation 31

The forward solve for each perturbation condition is warm-started from its converged state at the previous epoch. The wild-type solve uses a separate cache.

E.4. Training procedure

We split cells within each perturbation condition into train and validation subsets, with the same fraction across all conditions. Validation perturbation loss is the early-stopping criterion and the best-checkpoint metric.

Setting	Value	Notes
Train/val split	80%/20%	Stratified within each condition
Random seed	42	Used consistently for all splits
Maximum epochs	1000	
Early stopping patience	100 epochs	On validation $\mathcal{L}_{\text{pert}}$
Best-checkpoint metric	validation $\mathcal{L}_{\text{pert}}$	

For dense parameterizations under TF restriction, the TF mask is reapplied after each optimizer step to prevent AdamW's second-moment statistics from drifting masked entries off zero.

E.5. Encoder output dimensions

The encoder output dimension d is inherited from the encoder's pretraining configuration. Per-encoder values used in this work are:

Encoder	Output dimension d
Identity	G (gene-panel size)
PCA	512
scGPT (Cui et al., 2024)	512
scPRINT (Kalfon et al., 2025)	512
Stack (Dong et al., 2026)	1600
Tahoe-x1 (Gandhi et al., 2025)	512

Encoder-internal settings (vocabulary mapping, raw-count handling, depth tokens, soft-binning temperature) are reported in Appendix G.

E.6. Compute requirements

All experiments were run on a single NVIDIA H200 NVL (141 GB HBM3e), 16 CPU cores, and approximately 320 GB of system memory. No multi-GPU training was used. Foundation model forward passes, fixed-point ODE rollouts, and adjacency optimization all fit within a single device. Encoder forward and backward passes dominate wall-clock time for the transformer encoders, because the loss requires evaluating the frozen differentiable encoder on both the predicted and target states at every gradient step.

Encoder batch sizes are set per-encoder by VRAM constraints. Identity, PCA, and scGPT run at the global batch size of 64 cells. scPRINT requires per-cell batching with batch size 1 and the internal context length restricted to `ncell`$= 100$ in order to fit a forward and backward pass through the full encoder stack on the H200, due to its larger working memory footprint at the soft-binning and gene-vocabulary embedding stages. Stack and Tahoe-x1 retain a full attention matrix per layer for the backward pass, so they evaluate the encoder in micro-batches under gradient checkpointing (query/cell micro-batches of 64 and 32 respectively): because conditions are independent sequences, this micro-batching is exact. Live-encoder embeddings for control cells are cached on first use and reused across seeds within a condition.

Encoder	Cell batch size	Notes
Identity	64	No encoder forward
PCA	64	Linear projection, fixed weights
scGPT	64	Differentiable continuous-value encoder
scPRINT	1	`ncell`$= 100$ to fit H200 VRAM
Stack	64	Query micro-batch vs. a 256-cell context; gradient-checkpointed
Tahoe-x1	32	Micro-batch, `torch` attention; gradient-checkpointed

The headline real-data run on Norman 2019 (102 single-perturbation conditions, 129 double-perturbation conditions, 2,553 genes after Highly Variable Gene (HVG) filtering, 11,855 control cells) covers the six encoders plus the additive baseline, fit over five random seeds each, for thirty independent ODE fits in total. Per-condition wall-clock aggregated across seeds is reported below.

Condition	Seeds	Total wall-clock	Mean per seed
Identity	5	1.11 h	13.4 min
PCA	5	0.50 h	6.1 min
scGPT	5	4.08 h	49.0 min
scPRINT	5	1.81 h	21.7 min
Stack	5	1.33 h	16.0 min
Tahoe-x1	4	1.57 h	23.5 min

Adjacency optimization uses a dense $G \times G$ parameterization with a TF column mask that holds $260{,}406$ active entries (the 102 perturbed-TF columns) and zeros the remaining $G \times (G - G_{\mathrm{TF}})$ entries at every step (Appendix I.1). Memory cost is dominated by the encoder activations rather than by A itself.

We estimate total project compute, including the synthetic benchmark, encoder ablations, structural validation, the six single-layer encoder fits above, and the per-layer recovery decomposition (Section 3.3) with its linearized refits (Appendix K), at roughly 80 to 100 H200 GPU-hours.

F. Implicit-differentiation gradients: derivation and closed forms

This appendix derives the implicit-gradient form used in Section 2.3, gives closed-form expressions for the Jacobian J and the per-parameter gradients of the Hill ODE, and discusses numerical stability of the linear solve.

F.1. Implicit function theorem at the fixed point

The forward solver returns the converged steady state $\mathbf{x}^*(\theta)$ defined implicitly by $F(\mathbf{x}^*, \theta) = \mathbf{x}^* - g(\mathbf{x}^*, \theta) = \mathbf{0}$, where $g(\mathbf{x}, \theta) = (\mathbf{b} + A\, h(\mathbf{x}; n, K))/\gamma$ is the right-hand side of Equation 2. Differentiating the implicit equation with respect to θ yields

$$\left.\frac{\partial F}{\partial \mathbf{x}}\right|_{\mathbf{x}^*} \frac{\partial \mathbf{x}^*}{\partial \theta} + \left.\frac{\partial F}{\partial \theta}\right|_{\mathbf{x}^*} = \mathbf{0}, \tag{14}$$

so that

$$\frac{\partial \mathbf{x}^*}{\partial \theta} = -J^{-1} \left.\frac{\partial F}{\partial \theta}\right|_{\mathbf{x}^*}, \qquad J = \left.\frac{\partial F}{\partial \mathbf{x}}\right|_{\mathbf{x}^*}. \tag{15}$$

For a scalar loss $\mathcal{L}$ that depends on θ only through $\mathbf{x}^*$, with upstream gradient $\mathbf{v} = \partial \mathcal{L}/\partial \mathbf{x}^*$, the parameter gradient is

$$\frac{\partial \mathcal{L}}{\partial \theta} = \mathbf{v}^\top \frac{\partial \mathbf{x}^*}{\partial \theta} = -\mathbf{v}^\top J^{-1} \frac{\partial F}{\partial \theta} = -\mathbf{w}^\top \frac{\partial F}{\partial \theta}, \tag{16}$$

where $\mathbf{w} = J^{-\top}\mathbf{v}$ is obtained by solving the linear system

$$J^\top \mathbf{w} = \mathbf{v}. \tag{17}$$

This is a vector-Jacobian product. One $G \times G$ linear solve provides the backward signal for all parameter groups simultaneously, after which the per-parameter gradient is read off from the analytic form of $\partial F/\partial \theta$.

F.2. Closed-form Jacobian

For the Hill production term used in this work, the components of F are

$$F_i(\mathbf{x}, \theta) = x_i - \frac{1}{\gamma}\left(b_i + \sum_k A_{ik}\, h(x_k;\, n,\, K)\right). \tag{18}$$

Differentiating with respect to x_l and using $\partial h(x_k)/\partial x_l = \delta_{kl}\, h'(x_l)$,

$$J_{il} = \frac{\partial F_i}{\partial x_l} = \delta_{il} - \frac{A_{il}\, h'(x_l)}{\gamma}, \tag{19}$$

or equivalently in matrix form

$$J = I - \frac{1}{\gamma} A \operatorname{diag}\big(h'(\mathbf{x}^*)\big). \tag{20}$$

The construction of J requires only the Hill derivative h' at $\mathbf{x}^*$ and the current adjacency A, both available without additional computation at backward time.

F.3. Closed-form Hill derivatives

The Hill function $h(x;\, n,\, K) = x^n/(K^n + x^n)$ admits closed-form derivatives with respect to its argument and its parameters. Direct differentiation yields

$$h'(x) = \frac{n\, K^n\, x^{n-1}}{(K^n + x^n)^2} = \frac{n}{x}\, h(x)\big(1 - h(x)\big), \tag{21}$$

$$\frac{\partial h}{\partial n} = \frac{K^n\, x^n\, \ln(x/K)}{(K^n + x^n)^2} = h(x)\big(1 - h(x)\big)\, \ln(x/K), \tag{22}$$

$$\frac{\partial h}{\partial K} = -\frac{n\, K^{n-1}\, x^n}{(K^n + x^n)^2} = -\frac{n}{K}\, h(x)\big(1 - h(x)\big). \tag{23}$$

The right-hand-side identities are convenient at runtime: the saturation factor $h(x)(1 - h(x))$ is computed once and reused across all three derivative quantities.

F.4. Per-parameter gradients

Each parameter gradient follows from the VJP formula $\partial\mathcal{L}/\partial\theta = -\mathbf{w}^\top \, \partial F/\partial\theta$ with $\mathbf{w}$ obtained from Equation 17. The relevant partial derivatives of F are

$$\frac{\partial F_i}{\partial A_{kl}} = -\frac{\delta_{ik}\, h(x_l)}{\gamma}, \qquad \frac{\partial F_i}{\partial b_k} = -\frac{\delta_{ik}}{\gamma}, \tag{24}$$

$$\left.\frac{\partial F_i}{\partial \gamma}\right|_{\mathbf{x}^*} = \frac{x_i^*}{\gamma}, \qquad \frac{\partial F_i}{\partial \alpha} = -\frac{1}{\gamma}\sum_k A_{ik}\,\frac{\partial h(x_k)}{\partial \alpha} \quad \text{for} \;\; \alpha \in \{n, K\}, \tag{25}$$

where the γ identity uses the fixed-point relation $b_i + \sum_k A_{ik}\, h(x_k^*) = \gamma\, x_i^*$ to simplify. Substituting into the VJP formula and simplifying yields the closed-form parameter gradients

$$\frac{\partial \mathcal{L}}{\partial A_{ij}} = \frac{w_i\, h(x_j^*)}{\gamma}, \tag{26}$$

$$\frac{\partial \mathcal{L}}{\partial b_i} = \frac{w_i}{\gamma}, \tag{27}$$

$$\frac{\partial \mathcal{L}}{\partial \gamma} = -\frac{1}{\gamma}\sum_i w_i\, x_i^*, \tag{28}$$

$$\frac{\partial \mathcal{L}}{\partial n} = \frac{1}{\gamma}\,\mathbf{w}^\top A\,\left.\frac{\partial \mathbf{h}}{\partial n}\right|_{\mathbf{x}^*}, \tag{29}$$

$$\frac{\partial \mathcal{L}}{\partial K} = \frac{1}{\gamma}\,\mathbf{w}^\top A\,\left.\frac{\partial \mathbf{h}}{\partial K}\right|_{\mathbf{x}^*}, \tag{30}$$

where $\partial\mathbf{h}/\partial n$ and $\partial\mathbf{h}/\partial K$ are vectors whose k-th entries are given by Equations 22 and 23 evaluated at x_k^*.

Clamped genes constraints on gradients. Clamped genes have $x_j^{*,(j)} = c_j$ independent of θ, so their contributions to all parameter gradients should vanish. As described in Section 2.3, this is enforced on the gradient side by zeroing the corresponding entry of $\mathbf{v}$ and replacing the j-th row of J with $\mathbf{e}_j^\top$ before solving Equation 17, which forces $w_j = 0$. With $w_j = 0$, the contributions of clamped indices to Equations 26-30 vanish automatically, and no further masking of the analytic VJPs is required.

F.5. Numerical stability of the linear solve

The system in Equation 17 is well-posed when J is non-singular. By Equation 20, $J = I - (A\,\mathrm{diag}(h'))/\gamma$, and J approaches singularity precisely when the spectral radius of $(A\,\mathrm{diag}(h'))/\gamma$ approaches one, which is the same condition under which the forward fixed-point iteration becomes slow or marginally stable. In our experiments J remains well-conditioned throughout training, but to guard against numerical instability we solve a Tikhonov-regularized system

$$(J^\top + \epsilon I)\,\mathbf{w} = \mathbf{v}, \qquad \epsilon = 10^{-8}, \tag{31}$$

which adds a perturbation that vanishes in the well-conditioned limit and bounds the gradient norm when J is near-singular. The introduced bias is negligible at this ϵ scale relative to the floating-point precision of the inputs. Persistent ill-conditioning indicates that the forward solver is operating near the stability boundary, in which case the appropriate response is to revisit the solver damping factor or iteration depth rather than to increase ϵ.

G. Encoder wrapper details

This appendix documents the per-encoder implementation details for the differentiable wrappers used in our experiments. All wrappers conform to the abstract interface $\text{Enc} : \mathbb{R}^G \to \mathbb{R}^d$ described in Section 2.2, return embeddings differentiable in their input expression vector, and pass a common suite of diagnostic checks documented at the end of this appendix.

G.1. Identity and PCA

The Identity encoder returns its input unchanged, with $d = G$. Differentiability is trivial. The shift-based loss in this case reduces to gene-space matching of perturbation shifts.

The PCA encoder is fit on control cells from the data being analyzed and projects onto the top d principal components, $\text{Enc}(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu})\, V$, where $\boldsymbol{\mu}$ is the control mean and $V \in \mathbb{R}^{G \times d}$ collects the top right singular vectors of the centered control matrix. We use $d = 512$ throughout, which captures the bulk of expression variance while leaving a sufficient compression to make PCA a meaningful baseline against the foundation-model encoders.

G.2. scGPT

We use scGPT as a frozen wrapper that maps an expression vector to a cell-level embedding via the published transformer architecture and pretrained weights. Two design points are important for differentiability and correctness.

Continuous value encoder. scGPT supports both continuous and categorical encodings of expression values. Categorical encoding bins values into discrete tokens and is non-differentiable in the input. We require checkpoints trained with the continuous variant, in which expression values pass through a learned MLP before being added to gene-token embeddings. The wrapper asserts this at initialization and refuses checkpoints that use the categorical path.

Gene vocabulary mapping. scGPT operates on a fixed gene vocabulary tied to the pretrained checkpoint. We map each input gene name in the dataset to its corresponding vocabulary index via case-aware matching (exact match preferred, then uppercase, lowercase, capitalized variants). Genes that fail to map are excluded from the input sequence rather than mapped to a placeholder token.

Cell embedding extraction. The wrapper returns the cell embedding from the model's CLS token.

G.3. scPRINT

We use scPRINT as a frozen wrapper. Its interface differs from scGPT in two important ways.

Raw counts and depth token. scPRINT applies its own internal $\log_2(1 + \text{count})$ transformation and conditions on a depth token derived from the per-cell library size. The wrapper therefore expects raw integer counts, not log-normalized expression, and we pass the full-transcriptome library size (sum over all genes prior to highly-variable-gene filtering) as the depth token. Failing to provide the full-transcriptome library size, for example by passing the sum over the highly-variable subset, produces a depth token that systematically underestimates the cell's true sequencing depth and degrades the embedding quality.

Gene identifiers. scPRINT's vocabulary indexes genes by Ensembl ID. The wrapper reads the dataset's Ensembl IDs directly when available and falls back to symbol-to-Ensembl lookup via mygene only for genes without an Ensembl annotation. Direct vocabulary lookup is faster and avoids ambiguity introduced by symbol aliases.

Numerical safety. scPRINT's internal $\log_2(1 + x)$ transformation produces `NaN` on negative inputs. Because Equation 3 fits a steady-state predictor whose intermediate states can transiently violate non-negativity during optimization, we clamp predicted expression to $x \geq 0$ before passing to scPRINT. For real-data raw-count inputs the clamp is a no-op.

G.4. Stack

Stack (Dong et al., 2026) is an in-context tabular-attention foundation model, used as a frozen wrapper at the Stack-Large size. It differs from the other encoders in that its forward pass attends across a set of cells rather than treating each cell

independently, so a cell's embedding depends on the reference cells it is read against; the wrapper presents this as a per-cell map by fixing that reference.

Continuous value encoder. Stack embeds expression through a linear gene-reduction layer ($\mathrm{Linear}(G_{\mathrm{panel}}, n_{\mathrm{hidden}} \cdot \mathrm{token_dim})$ followed by GELU) applied to log1p expression, with no binning, ranking, or discretization, so the embedding is differentiable with respect to every input gene. For the Stack-Large checkpoint the panel holds $15{,}012$ genes, the stack has 9 attention layers, and the cell embedding has dimension $n_{\mathrm{hidden}} \cdot \mathrm{token_dim} = 1600$; inference dropout is zero, so the embedding is deterministic.

Gene-panel alignment. Unlike the gene-token models, Stack consumes a dense vector over its own fixed $15{,}012$-gene panel. The wrapper aligns columns by symbol (exact match, then uppercase): each input gene is scattered into its panel column, unmapped panel columns are held at zero, and input genes absent from the panel are dropped, so gradients reach only the mapped columns.

In-context reference. Because cells cross-attend, $\mathrm{Enc}(\mathbf{x})$ is defined as embedding the query cell against a fixed, detached subsample of control cells, set once through `set_context` (the runner passes the control population; we use 256 reference cells, matching the checkpoint's trained set size). A cross-cell attention mask keeps the reference self-contained and lets each query attend only to the reference and to itself, never to other query cells, so each condition is encoded independently and the per-cell embedding matches the solo forward to floating-point round-off. The reference is detached, so the input Jacobian used during recovery and in the geometry probe of Appendix J is taken with respect to the query cell alone.

Input scale and cell embedding. The wrapper expects log1p-normalized expression and applies `expm1` to recover the linear-count scale Stack re-applies log1p to internally, so it is not a raw-count encoder. The returned embedding is the query rows of the final attention layer's output, with the reference rows discarded. The cross-cell attention retains an (B, H, S, S) matrix per layer for the backward pass, so the wrapper encodes queries in micro-batches under gradient checkpointing to bound peak memory.

G.5. Tahoe-x1

Tahoe-x1 (Gandhi et al., 2025) is a gene-token transformer pretrained on the Tahoe-100M perturbation compendium, used as a frozen wrapper at the 70M size (12 transformer blocks).

Continuous value encoder. The released checkpoints embed expression with a continuous value encoder (two linear layers and a normalization applied to the continuous value), differentiable with respect to the input expression as scPRINT's is. The training collator bins expression for the masked-language-model objective, but inference feeds continuous values, so no soft-binning approximation is needed.

Gene vocabulary mapping. Tahoe-x1 indexes genes by Ensembl ID. The wrapper uses the dataset's Ensembl IDs directly when available, which keeps a run fully offline, and falls back to a cached symbol-to-Ensembl lookup via mygene only for genes without an annotation, retrying the version-stripped identifier on a miss. Genes that fail to map are excluded from the input sequence rather than mapped to a placeholder.

Sequence-length cap. The model is scGPT-style with no positional encoding, so the sequence length (the CLS token plus the mapped genes) is a memory bound rather than an architectural one, capped at 2048. When the mapped panel exceeds the cap the top-value genes are kept; this top-value selection is non-differentiable for the dropped genes and is the non-smooth gate referred to in the tokenization caveat of Appendix J.

Attention backend. The shipped configuration requests a flash-attention kernel that needs flash-attn and an Ampere-or-newer GPU; the wrapper forces the pure-PyTorch attention kernel at load time, which leaves the weights unchanged and swaps only the kernel, so the encoder runs without flash-attn (including on CPU). As with the other transformers the kernel retains the full attention matrix per layer for backward, so `encode` micro-batches the conditions under gradient checkpointing; conditions are independent sequences and never cross-attend, so the chunking is exact.

Input scale and cell embedding. The wrapper expects log1p-normalized expression and applies `expm1` to recover the scale the expression encoder was trained on, with the CLS token (value zero) prepended at position 0, so it is not a raw-count encoder. The returned embedding is the position-0 (CLS) cell embedding.

G.6. Embedding caching

For encoders that perform substantial compute on each forward pass (the foundation models), we cache encoded observations to disk and reuse them across runs. The cache is keyed by a fingerprint hash that includes the encoder identity and wrapper version, the dataset and gene-panel hash, the input format flag (raw counts versus log-normalized), and the wrapper's vocabulary mapping. Any change that alters the encoded output invalidates the cache, including code changes to the wrapper itself.

G.7. Diagnostic validation

Each wrapper is validated against a common suite of checks: gradient flow (autograd produces non-zero gradients with respect to a perturbed input), perturbation sensitivity (encoded shifts have non-trivial magnitude under realistic input perturbations), discriminability (encoded shifts under different perturbations are distinguishable in cosine similarity), and reference-result reproducibility (encoded outputs match published embeddings on a small held-out set, where applicable). Wrappers that fail any check are not used in experiments.

H. Dataset

H.1. The Norman Dataset

The dataset from Norman et al. (2019) is one of the most comprehensive Perturb-seq resources available. Using pooled CRISPRa screening with scRNA-seq, it profiles the transcriptional responses of over 100,000 K562 (human leukemia) cells to roughly 105 single-gene and 131 double-gene perturbations, alongside unperturbed controls. Its scale of *combinatorial* (double-gene) perturbations is, to our knowledge, unique among published Perturb-seq datasets, making it the natural testbed for CLAMP: the single-gene conditions fit the network and the held-out double-gene pairs evaluate compositional prediction at single-cell resolution (Section 2.4).

Table 3. Overview of the Norman 2019 dataset and the subset retained after preprocessing.

Characteristic	**Description**
Cell line	K562 (human leukemia)
Perturbation method	CRISPRa (overexpression)
Sequencing platform	10x Genomics Chromium
Assay	Single-cell RNA-seq
Reference	Norman et al. (2019)
As profiled	
Cells profiled	∼110,000
Genes measured	20,000+
Retained for this work (post-preprocessing)	
Single-gene perturbation conditions	102
Double-gene perturbation conditions (held out)	129
Control cells	11,855
Genes after HVG + perturbed-gene protection	2,553

H.2. Pre-Processing

We start from the raw Norman 2019 count matrix and follow the preprocessing convention of PertEval (Wenteler et al., 2025). All operations use `ScanPy` (Wolf et al., 2018), with the dataset retrieved through `ScPerturb` (Peidli et al., 2024) and `PertPy` (Heumos et al., 2024).

Cell and gene filtering. We discard genes with five or fewer total counts across all cells, parse the guide identities into conditions (empty guide = control), and split single- from double-gene perturbations. The single-gene conditions and controls are used for fitting; the double-gene conditions are held out for compositional evaluation (Section 2.4). We then drop cells whose perturbed gene is absent from the measured panel and require at least three cells per condition. This yields 102 single-gene perturbation conditions, 129 held-out double-gene conditions, and 11,855 control cells.

Normalization. Counts are library-size normalized to the median library size (`scanpy.pp.normalize_total`) and $\log(1 + x)$-transformed (`scanpy.pp.log1p`); this log1p space is the ODE state space. For the raw-count encoders (scPRINT, Tahoe-x1) we additionally retain the unnormalized integer counts and the full-transcriptome per-cell library size (summed over *all* genes before HVG filtering), which supply each model's depth token (Appendix G).

Highly variable genes and perturbed-gene protection. We select the top 2,500 highly variable genes (`scanpy.pp.highly_variable_genes(n_top_genes=2500)`) on the log1p control-plus-perturbation matrix. Because Norman is a CRISPRa screen, every over-expressed perturbed gene must remain in the panel to serve as a clamp coordinate, so we force-include all perturbed genes irrespective of their HVG status. This adds the 53 perturbed genes not already selected, giving a final panel of 2,553 genes.

Transcription-factor restriction. The 102 perturbed genes in Norman are all transcription factors. We restrict the learnable adjacency columns to these 102 TFs (Section 2.2, Appendix I), so $A \in \mathbb{R}^{2553\times 102}$.

Condition-level aggregation and the clamp. CLAMP fits per-condition *mean* responses rather than individual cells: for each condition the encoded cells are averaged into a single profile $\bar{\mathbf{z}}^{(j)}$, and the loss matches the mean shift $\bar{\mathbf{z}}^{(j)} - \bar{\mathbf{z}}^{(\mathrm{ctrl})}$ (Eq. 3). The clamp value c_j for each perturbed gene is its observed condition-mean log1p expression, the elevated level CRISPRa drives it to, held fixed while the remaining genes relax to equilibrium (Eq. 2). Within each condition, 20% of cells are held out for validation (Appendix E).

I. Adjacency parameterization details

This appendix gives details for the dense parameterization of the regulatory adjacency matrix A introduced in Section 2.2, including the transcription-factor restriction and the initialization scheme.

I.1. Transcription-factor restriction

We restrict the recovered adjacency $A \in \mathbb{R}^{G \times G_{\mathrm{TF}}}$ so that only the perturbed transcription factors are admissible as regulators, fitting edges only for TFs for which we have interventional evidence. This reflects the identifiability constraint that a column of A can only be directly constrained by perturbing the corresponding regulator: TFs whose perturbations are not observed contribute no direct interventional signal and would have their columns determined only through indirect downstream effects.

Because we use a dense parameterization for A, this restriction is implemented as a strict multiplicative mask on the parameter matrix. Entries corresponding to non-perturbed columns are set to exactly zero at initialization. Crucially, this mask is reapplied and the non-perturbed entries are zeroed again after every optimizer step. Reapplication is necessary because we optimize the network using AdamW (Loshchilov & Hutter, 2017): even if the gradient for a masked entry is zero, AdamW's second-moment running statistics for that parameter can become non-zero by virtue of past momentum during the run. Without continuous reapplication of the mask, these optimizer statistics would cause the masked, non-perturbed entries to slowly drift away from zero over time.

I.2. Initialization scheme

The active, learnable entries of the dense adjacency matrix A (corresponding to the G_{TF} columns) are initialized by sampling independently from a zero-mean Gaussian distribution, $A_{ij} \sim \mathcal{N}(0, \sigma^2)$. The standard deviation σ is kept small to calibrate the expected magnitude of the initial regulatory inputs. This ensures that the initial interaction weights provide a stable, well-conditioned starting point for the forward fixed-point solver, preventing the Hill activation functions from being driven into hard saturation prematurely during the earliest epochs of training. All non-TF columns remain strictly zero throughout initialization and training.

J. Encoder geometry probes: linearity and regulatory rank

This appendix gives the formal definitions of the two no-fit geometric readouts introduced in Section 2.4 and plotted in the bottom row of Figure 4: the linearity score Λ_L (Eq. 7) and the regulatory rank. Both are read off the encoder's local geometry per layer at real control cells and require no ODE fit. We specify in turn the regulatory-shift basis and its weights, the layerwise directional derivative the two readouts share with the recovery gradient, the linearity score, the participation-ratio regulatory rank and its random-subspace baseline, and the implementation choices, sanity gates, and caveats.

J.1. Regulatory-shift basis and weights

The probe is computed in the directions along which the perturbations actually move cells, not in directions of natural expression variation. For each single-gene condition $j \in \mathcal{J}$ we form the observed shift in log1p HVG space,

$$\Delta \mathbf{x}_j = \mathbb{E}[\mathbf{x} \mid \mathrm{SKI}_j] - \bar{\mathbf{x}}_{\mathrm{ctrl}} \in \mathbb{R}^G, \tag{32}$$

and stack the $|\mathcal{J}|$ shifts as the rows of a matrix $S \in \mathbb{R}^{|\mathcal{J}| \times G}$. We center S along the condition axis and take the right singular vectors of the centered matrix as an orthonormal basis $\{v_k\}$ of gene space, ordered by decreasing singular value (the PCA of the perturbation shifts). The energy each component carries is its *regulatory weight*

$$w_k = \sum_{j \in \mathcal{J}} \left(\Delta \mathbf{x}_j \cdot v_k\right)^2, \tag{33}$$

the summed squared projection of all observed shifts onto v_k. We rank the components by w_k and retain the top $K = 48$; for the shift basis this ordering essentially coincides with the variance ordering, and the retained K carry the large majority of the total regulatory weight (the first few components alone account for over half of it). The K directions $\{v_k\}_{k=1}^K$ and their weights $\{w_k\}_{k=1}^K$ are the fixed regulatory subspace in which both geometric readouts below are evaluated. The same construction supplies the basis used by the finite-injection transmission diagnostic, so the two analyses probe identical directions.

J.2. Layerwise directional derivative

Write $\mathrm{Enc}_L : \mathbb{R}^G \to \mathbb{R}^{d_L}$ for the readout at transformer layer L (the pooled block-L representation; see Appendix G), and let $J_L(\mathbf{x}) = \partial \,\mathrm{Enc}_L / \partial \mathbf{x}$ be its input Jacobian. The signal that reaches the ODE along regulatory direction v_k at a cell $\mathbf{x}_i$ is the directional derivative

$$D_{v_k} \mathrm{Enc}_L(\mathbf{x}_i) \;=\; J_L(\mathbf{x}_i)\, v_k, \tag{34}$$

the Jacobian-vector product (JVP) of the layer readout. This is exactly the map the recovery gradient uses, read in the opposite direction: in Section 2.3 the upstream term $\mathbf{v} = \partial \mathcal{L} / \partial \mathbf{x}^*$ is the encoded loss pulled *back* through $\partial \mathrm{Enc} / \partial \mathbf{x}$ (a vector-Jacobian product), so the fit can only move A along directions the encoder's local Jacobian transmits out of gene space. Here we take the same Jacobian *forward* along the regulatory directions v_k and read it layer by layer, where depth is resolved.

We evaluate Eq. 34 at n_a control cells sampled as Jacobian anchor points ($n_a = 32$ by default). Differentiating the all-layers readout once per direction v_k yields the JVP at every layer in a single pass. The JVP is computed by forward-mode automatic differentiation where available and by double-backward otherwise; both run under the math scaled-dot-product-attention backend, because the fused efficient/flash attention kernels implement neither forward-mode AD nor double-backward, while the math backend (plain matmul-softmax) supports both and is numerically equivalent. For in-context encoders the control context is set and detached before any readout, so the JVP is taken with respect to the query input alone. For raw-count encoders the $\mathrm{log1p} \to \mathrm{raw}$ denormalization is wrapped *inside* the differentiated path, so the tangent stays in log1p units and the readouts remain comparable across encoders; at control cells $\mathrm{expm1}(\mathbf{x}) \geq 0$, so the non-negativity clamp is inactive and gradients flow.

J.3. Linearity Λ_L

Linearity asks whether the layer acts as a *consistent* linear map on the regulatory directions, the property the gene-space ODE requires: the ODE matches a single adjacency across all conditions and control states, so it can only follow a regulatory direction whose encoded image points the same way regardless of which cell it is read at. For direction v_k we normalize the

per-cell image to a unit vector and take the length of its average over the n_a control cells,

$$c_{k,L} \;=\; \left\| \frac{1}{n_a} \sum_{i=1}^{n_a} \frac{D_{v_k}\mathrm{Enc}_L(\mathbf{x}_i)}{\left\| D_{v_k}\mathrm{Enc}_L(\mathbf{x}_i) \right\|_2} \right\|_2, \tag{35}$$

the resultant length of the n_a image directions. If the layer is a consistent linear projection on v_k, every $D_{v_k}\mathrm{Enc}_L(\mathbf{x}_i)$ points the same way and $c_{k,L} = 1$; if the image direction drifts cell to cell because the Jacobian is state-dependent (the signature of a non-linear projection), the unit vectors partially cancel and $c_{k,L}$ falls toward the chance floor. The linearity score is the regulatory-weighted mean of $c_{k,L}$ over the subspace,

$$\Lambda_L = \frac{1}{\sum_k w_k} \sum_{k=1}^{K} w_k \, c_{k,L}, \tag{36}$$

which is Eq. 7 of the main text. The chance floor is $1/\sqrt{n_a}$, the expected resultant length of n_a independent uniformly random unit vectors in high dimension, so Λ_L should be read on $[1/\sqrt{n_a}, 1]$ (at $n_a = 32$ the floor is ≈ 0.18). A genuinely linear encoder has a constant Jacobian and gives $c_{k,L} = 1$ for every k, hence $\Lambda_L = 1$: this is the sanity gate satisfied by Identity ($J = I$) and, up to sampling noise, by PCA.

J.4. Regulatory rank and the random-subspace baseline

Regulatory rank asks whether the K directions stay *separable* after encoding, a necessary condition for the embedding to carry a gene-space network: if several regulatory directions collapse onto a single embedding direction, their distinct downstream effects can no longer be told apart and no reader, linear or not, can recover the network along them. At each layer L and cell $\mathbf{x}_i$ we collect the K image vectors as the columns of

$$M_{L,i} = \left[\, D_{v_1}\mathrm{Enc}_L(\mathbf{x}_i) \;\cdots\; D_{v_K}\mathrm{Enc}_L(\mathbf{x}_i) \,\right] \in \mathbb{R}^{d_L \times K}, \tag{37}$$

and read its singular values $\sigma_1 \geq \cdots \geq \sigma_K$, equivalently the eigenvalues $\lambda_k = \sigma_k^2$ of the $K \times K$ Gram matrix $M_{L,i}^\top M_{L,i}$. The spread of the λ_k measures how many independent embedding directions the regulatory subspace occupies. We summarize it with the participation ratio, normalized by K,

$$\mathrm{rank}_{\mathrm{reg}}(L) \;=\; \frac{1}{K} \left\langle \mathrm{PR}(M_{L,i}) \right\rangle_i, \qquad \mathrm{PR}(M) = \frac{\left(\sum_k \lambda_k\right)^2}{\sum_k \lambda_k^2}, \tag{38}$$

averaged over the control cells. The normalized rank runs from 1 when the K regulatory directions map to K embedding directions of equal magnitude (fully injective image) down to $1/K$ when they collapse onto a single embedding direction (maximal loss). Being a ratio of singular-value moments, it is scale-invariant and therefore immune to the growth of residual-stream norm with depth. As a complementary conditioning diagnostic we also record $\sigma_{\min}/\sigma_{\max} = \sqrt{\lambda_{\min}/\lambda_{\max}}$, which goes to zero when at least one regulatory direction is fully collapsed; the figures report the participation-ratio rank.

A low absolute rank can mean two different things: the encoder may specifically crush the *regulatory* subspace, or it may simply produce a generically low-dimensional embedding in which any K-dimensional subspace looks compressed. To separate these, we recompute Eq. 38 on a random K-dimensional orthonormal subspace of gene space (the columns of the Q factor of a QR decomposition of a Gaussian matrix), giving a baseline $\mathrm{rank}_{\mathrm{rand}}(L)$. The relation between the two is the informative quantity: $\mathrm{rank}_{\mathrm{reg}} < \mathrm{rank}_{\mathrm{rand}}$ flags compression *specific* to the regulatory subspace, while $\mathrm{rank}_{\mathrm{reg}} \approx \mathrm{rank}_{\mathrm{rand}}$ indicates a generic low-rank embedding. The figures plot the random-adjusted regulatory rank

$$\widetilde{\mathrm{rank}}_{\mathrm{reg}}(L) = \frac{\mathrm{rank}_{\mathrm{reg}}(L)}{\sqrt{\mathrm{rank}_{\mathrm{rand}}(L)}}, \tag{39}$$

which discounts the generic low-rank floor while preserving the absolute injectivity gap between encoders (so that, e.g., Stack remains visibly separated from the collapsed encoders rather than being normalized onto them, as a plain ratio would do). It is read comparatively across encoders rather than on the absolute $[1/K, 1]$ scale of $\mathrm{rank}_{\mathrm{reg}}$. The baseline calibrates as expected: Identity gives $\mathrm{rank}_{\mathrm{reg}} = \mathrm{rank}_{\mathrm{rand}} = 1$, and PCA gives a high regulatory rank above its random baseline.

J.5. Implementation, sanity gates, and caveats

The readouts are computed on the live encoders, since the JVP needs the model and not just cached embeddings, with default settings $n_a = 32$ anchor cells, $K = 48$ regulatory directions, and a K-dimensional random baseline. The linear encoders provide the sanity gates that must hold for every run: Identity gives $\Lambda_L = 1$ and $\mathrm{rank}_{\mathrm{reg}} = 1$, and PCA gives $\Lambda_L \approx 1$ with $\mathrm{rank}_{\mathrm{reg}} \approx 0.98$. Three caveats bound the interpretation.

1. **The probe is local.** Both readouts use the Jacobian at control cells, so they characterize the encoder's behavior in the neighborhood the perturbed steady states drift away from. They therefore test *necessary* conditions for recovery (a consistent projection and a separable subspace) rather than sufficient ones: where linearity or rank fails, the loss is genuine; where they hold, recovery is limited by the complexity of the readout rather than by missing structure.

2. **Linearity has a non-zero floor.** Λ_L lives on $[1/\sqrt{n_a}, 1]$, not $[0, 1]$; values near the floor mean the image directions are no more consistent across cells than random. Larger n_a sharpens the separation at higher memory cost.

3. **Discrete tokenization gates the JVP.** Encoders that select genes through a non-smooth operation (for example a top-k token gate) zero the gradient through the unselected coordinates, which can depress their absolute regulatory rank relative to non-gating encoders. The recovery-unit decomposition of Appendix K is the complementary measurement that does not rely on a smooth Jacobian and so catches non-linearity that a JVP-based readout under-reports.

K. Linearized rank ceiling and the rank/non-linearity decomposition

This appendix details the linearized-reader refit introduced in Section 2.4 (Eq. 8) and the decomposition of each encoder's recovery deficit into rank loss and non-linearity loss (Eq. 9), which together produce the bands of Figure 4 (top). The geometric readouts of Appendix J characterize the encoder without an ODE in the loop; the decomposition here measures the same two failure axes *in recovery units*, by refitting the network through a controlled linear surrogate of each encoder and comparing to recovery through the unmodified encoder.

K.1. The linearized reader

For each encoder, at each layer, we replace the readout Enc_L by its own first-order Taylor expansion at the control-mean cell $\mathbf{x}_0$,

$$\mathrm{Enc}_{\mathrm{lin}}(\mathbf{x}) = \mathrm{Enc}(\mathbf{x}_0) + J\,(\mathbf{x} - \mathbf{x}_0), \qquad J = \left.\frac{\partial \mathrm{Enc}}{\partial \mathbf{x}}\right|_{\mathbf{x}_0} \in \mathbb{R}^{d_L \times G}, \tag{40}$$

which is Eq. 8 of the main text. The anchor $\mathbf{x}_0$ is the mean control cell in log1p (model-facing) units. The dense Jacobian J is formed once per layer by chunked reverse-mode automatic differentiation: one backward pass per output coordinate, seeded with a unit cotangent and accumulated in chunks to trade memory for speed. This uses only first-order AD, unlike the JVP probe of Appendix J, and is robust to encoders whose forward pass breaks vectorized (vmap) differentiation. For raw-count encoders the $\mathrm{log1p} \to \mathrm{raw}$ step is again wrapped inside the differentiated path so that J is expressed in log1p units, matching the ODE state space and the geometry probe; at the control anchor the non-negativity clamp is inactive.

Once J is formed, $\mathrm{Enc}_{\mathrm{lin}}$ is a fixed affine map, so evaluating it is a single matmul and the ODE fit through this reader runs without any foundation-model forward pass. The expensive step is forming J; the refit itself is cheap.

K.2. Why recovery through $\mathrm{Enc}_{\mathrm{lin}}$ is a pure rank ceiling

We refit the same steady-state ODE through this fixed affine map, with the loss, solver, hyperparameters (Appendix E), and random seeds all unchanged, applying $\mathrm{Enc}_{\mathrm{lin}}$ identically to predictions and targets. The shift-based loss of Eq. 3 compares *differences* of encoded states, and for an affine map the constant term and the $\mathbf{x}_0$ offset cancel,

$$\mathrm{Enc}_{\mathrm{lin}}(\mathbf{a}) - \mathrm{Enc}_{\mathrm{lin}}(\mathbf{b}) = J\,(\mathbf{a} - \mathbf{b}), \tag{41}$$

so every shift is read through the single linear map J. The projection is therefore perfectly consistent across cells and conditions ($\Lambda = 1$ by construction), and the only thing that can limit recovery is the rank and conditioning of J: the fit can move the adjacency A along a gene-space shift direction exactly to the extent that J keeps that direction separable in the embedding. Recovery through $\mathrm{Enc}_{\mathrm{lin}}$, written $\mathrm{rec}_{\mathrm{lin}}$, is thus the recovery a perfectly faithful (linear) reader of *this encoder's own rank* achieves: the rank-limited ceiling. It is the dashed line in Figure 4 (top).

K.3. The decomposition in recovery units

We anchor the top of the scale on Identity recovery C, which leaves gene space untouched and whose recovered network validates against the independent references of Appendix L. Writing $\mathrm{rec}_{\mathrm{true}}$ for recovery through the unmodified encoder, the deficit relative to C partitions as

$$\underbrace{C - \mathrm{rec}_{\mathrm{lin}}}_{\text{rank loss}} + \underbrace{\mathrm{rec}_{\mathrm{lin}} - \mathrm{rec}_{\mathrm{true}}}_{\text{non-linearity loss}} + \underbrace{\mathrm{rec}_{\mathrm{true}}}_{\text{recovered}} = C, \tag{42}$$

which is Eq. 9. Rank loss, $C - \mathrm{rec}_{\mathrm{lin}}$, is regulatory signal that even a perfectly linear reader of the encoder cannot keep separable: a limitation of the embedding, not of how we read it. Non-linearity loss, $\mathrm{rec}_{\mathrm{lin}} - \mathrm{rec}_{\mathrm{true}}$, is signal the encoder retains (the linear reader recovers it) but presents in a state-dependent, non-linear form that the single gene-space ODE cannot follow, even though a more flexible reader in principle could. The recovery metric throughout is the held-out double-gene Pearson correlation of Eq. 6 (the DKI val column of Table 1 and the recovery axis of Figure 4); the true sweep (Section 3.3) and the linearized sweep are run identically and differ only in the encoder.

In the figure the bands are drawn with $\mathrm{rec}_{\mathrm{true}}$ clipped to $[0, C]$ and the ceiling drawn as $\max(\mathrm{rec}_{\mathrm{lin}}, \mathrm{rec}_{\mathrm{true}})$ clipped to $[0, C]$. The $\max$ floors the non-linearity band at zero: when the linearization slightly under-recovers the true encoder because it is pinned at $\mathbf{x}_0$ while the fit explores states away from it, that case is reported as zero non-linearity loss rather than a negative band. This is the conservative convention behind the bounds stated below.

K.4. Sanity gate, bounds, and what the recovery-unit split adds

Sanity gate. A genuinely linear encoder has $\mathrm{Enc}_{\mathrm{lin}} = \mathrm{Enc}$, so $\mathrm{rec}_{\mathrm{lin}} = \mathrm{rec}_{\mathrm{true}}$ and the non-linearity loss is zero. Identity and PCA therefore register zero non-linearity loss, the check that the split attributes nothing to non-linearity for an encoder that has none.

Bounds. Because $\mathrm{Enc}_{\mathrm{lin}}$ is an empirical refit pinned at $\mathbf{x}_0$ while the perturbed steady states drift away from the anchor, the linear reader is evaluated slightly off the point where it equals the true encoder, so $\mathrm{rec}_{\mathrm{lin}}$ is a conservative (lower) estimate of the true linear-reader ceiling. The split is therefore a calibrated estimate rather than an exact identity: rank loss $C - \mathrm{rec}_{\mathrm{lin}}$ is an *upper* bound and non-linearity loss $\mathrm{rec}_{\mathrm{lin}} - \mathrm{rec}_{\mathrm{true}}$ a *lower* bound on the true quantities, as stated in Section 2.4.

What the recovery-unit split adds. Measuring non-linearity loss in recovery units, by refitting, rather than from the Jacobian-based linearity score Λ_L, catches non-smooth non-linearity that a local Jacobian under-reports, in particular discrete tokenization (e.g. a top-k gene gate), whose effect on recovery is real even though it is invisible to a first-order linearity readout. The two probes are thus complementary: Λ_L and the regulatory rank of Appendix J corroborate the bands of Eq. 42 from outside the ODE, and the recovery-unit split captures the part of non-linearity that a smooth readout misses.

K.5. Compute

The Jacobian J is formed once per (encoder, layer); thereafter the affine refit requires no foundation-model forward or backward pass, so the linearized sweep is substantially cheaper than the true per-layer sweep of Section 3.3, whose cost is dominated by encoder forward and backward passes (Appendix E). The dominant cost of the linearized sweep is the one-time per-layer Jacobian, which is chunkable in the embedding dimension to fit available memory.

L. Structural validation details

This appendix reports the full structural validation results summarized in Figure 3 of the main text. The recovered adjacency for the Identity encoder is tested against three independent reference types: curated hematopoietic lineage signatures, ChEA 2022 ChIP-seq target sets, and ENCODE TF ChIP-seq 2015 target sets. Pathway enrichment against MSigDB Hallmark (Liberzon et al., 2015) and Reactome 2022 (Jassal et al., 2020) is also reported as a fourth tier for completeness.

For each TF, we take the top-25 recovered targets ranked by $|A[:, \mathrm{TF}]|$ (Figure 5 panel C), compute hypergeometric enrichment against every gene set in each library after restricting to the gene panel, and apply Bonferroni correction over the size of each library. Background size is the panel target count ($n = 2553$ measured genes). Self-loops are excluded from the top-25 ranking.

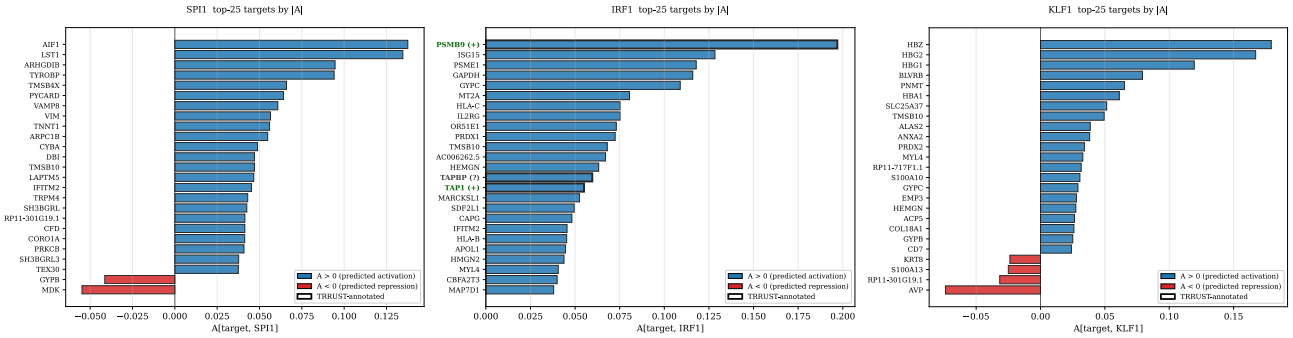

Figure 5. Fold enrichment of each TF's top-25 against eight curated hematopoietic lineage signatures, colored by $-\log_{10}(p_{\mathrm{bonf}})$. Expected hits boxed.

L.1. Lineage signature enrichment

The lineage signature library (Appendix M) consists of eight curated hematopoietic gene sets: three matched-lineage signatures (Erythroid, Myeloid Monocyte, Interferon Response) and five negative or alternative controls (Myeloid Granulocyte, Megakaryocyte, T cell, B cell, Cell cycle).

Each TF's matched-lineage hit reaches $p_{\mathrm{bonf}} < 10^{-10}$ with fold enrichment between $38.9\times$ and $48.6\times$. Every wrong-lineage and negative-control signature has $p_{\mathrm{bonf}} > 0.1$. The closest off-diagonal hit is IRF1 against Erythroid Lineage at $9.7\times$ ($p_{\mathrm{bonf}} = 0.10$), consistent with the K562 hematopoietic context where interferon-responsive genes overlap with erythroid co-regulated genes (notably *HEMGN* and *GYPC*, which appear in both the IRF1 top-25 and the broader erythroid program).

L.2. ChEA 2022 ChIP-Seq target enrichment

ChEA 2022 contains 1546 gene sets, each representing the targets of one TF in one ChIP-Seq experiment with cell-line and reference annotated. Bonferroni correction uses $m = 685$ (size after restricting sets to the gene panel and applying minimum size threshold of 5; full library spans 1546 entries before panel restriction).

Each TF's top-25 recovered targets enrich significantly ($p_{\mathrm{bonf}} < 10^{-3}$) against same-TF ChIP target sets in ChEA. The IRF1 top hit is IRF8, the closest IRF-family paralog that shares many target genes with IRF1; the IRF1 same-TF entry is the second-strongest. The KLF1 top hit uses the gene synonym EKLF (Erythroid Krüppel-Like Factor), which is the same gene as KLF1.

The next-strongest hits in each TF's table reflect the local regulatory neighborhood. KLF1's recovered targets enrich against GATA2 and GATA1 ChIP in K562, the canonical erythroid co-regulators that share targets with KLF1 in erythroid commitment. SPI1's top targets enrich against TAL1, RUNX1, and LYL1 ChIP in mouse hematopoietic progenitors, which together with SPI1 form the heptad transcription factor complex characterized in early myeloid specification (Wilson et al., 2010). IRF1 enriches against MAF in Th1 cells, consistent with the IRF1-MAF interplay in interferon-responsive immune programs.

Table 4. Fold enrichment and Bonferroni-corrected hypergeometric p-values of top-25 recovered targets against each lineage signature, per TF. Bold entries highlight the matched-lineage hit per TF. $m = 8$ for Bonferroni.

TF	Lineage signature	Overlap	Fold	p_{bonf}
KLF1	**Erythroid Lineage**	**10/21**	**48.6×**	**2.1e-15**
KLF1	Lymphoid T cell Lineage	1/13	7.9×	7.2e-1
KLF1	Myeloid Granulocyte Lineage	0/13	0.0×	1.0
KLF1	Myeloid Monocyte Lineage	0/23	0.0×	1.0
KLF1	Interferon Response	0/21	0.0×	1.0
KLF1	Megakaryocyte Lineage	0/–	0.0×	1.0
KLF1	Lymphoid B cell Lineage	0/–	0.0×	1.0
KLF1	Cell Cycle	0/–	0.0×	1.0
SPI1	**Myeloid Monocyte Lineage**	**10/23**	**44.4×**	**6.6e-15**
SPI1	Myeloid Granulocyte Lineage	1/13	7.9×	7.2e-1
SPI1	Erythroid Lineage	1/21	4.9×	1.0
SPI1	Interferon Response	1/21	4.9×	1.0
SPI1	Lymphoid T cell Lineage	0/13	0.0×	1.0
SPI1	Megakaryocyte Lineage	0/–	0.0×	1.0
SPI1	Lymphoid B cell Lineage	0/–	0.0×	1.0
SPI1	Cell Cycle	0/–	0.0×	1.0
IRF1	**Interferon Response**	**8/21**	**38.9×**	**2.8e-11**
IRF1	Erythroid Lineage	2/21	9.7×	1.0e-1
IRF1	Myeloid Granulocyte Lineage	0/13	0.0×	1.0
IRF1	Myeloid Monocyte Lineage	0/23	0.0×	1.0
IRF1	Lymphoid T cell Lineage	0/13	0.0×	1.0
IRF1	Megakaryocyte Lineage	0/–	0.0×	1.0
IRF1	Lymphoid B cell Lineage	0/–	0.0×	1.0
IRF1	Cell Cycle	0/–	0.0×	1.0

L.3. ENCODE TF ChIP-Seq 2015 target enrichment

ENCODE TF ChIP-Seq 2015 contains 816 gene sets, each representing one TF binding event in one cell line under the ENCODE consortium. K562 is the most-represented human cell line in this library, making it directly relevant to the Norman 2019 dataset.

KLF1 and IRF1 enrich strongly against ENCODE entries with K562 specificity. KLF1's top ENCODE hits reflect erythroid co-regulation: FOSL1 in K562, GATA1 in erythroblast, and TAL1 in two erythroid contexts (G1E-ER4 is a GATA1-rescuable erythroid line, erythroblast is primary cell). IRF1's IRF1 K562 entry recovers 13 of 291 ENCODE-defined IRF1 targets in the matched cell line at $p_{\text{bonf}} = 5.0 \times 10^{-4}$. The IRF1 STAT2 K562 hit ($14.2\times$ fold) reflects the canonical IRF1-STAT2 axis in type I interferon signaling.

SPI1 ENCODE enrichment is weaker and does not pass Bonferroni correction at the 0.05 level. SPI1 same-TF entries (GM12878, HL-60) do appear in the top 5 but the available ENCODE SPI1 ChIP experiments are in cell lines (lymphoblast and APL respectively) less directly matched to K562 than the ChEA SPI1 NB4 entry (Table 5). The SPI1 lineage signature enrichment (Table 4, $44.4\times$, $p_{\text{bonf}} = 6.6 \times 10^{-15}$) is the strongest evidence of SPI1 recovery.

L.4. MSigDB Hallmark and Reactome enrichment

For completeness, we report top hits against the broader pathway databases used as initial validation references. Bonferroni correction over $m = 49$ (Hallmark, after panel restriction) and $m = 575$ (Reactome 2022, after panel restriction and exclusion of disease categories per Appendix M).

KLF1's Hallmark Heme Metabolism hit recovers 9 of 42 panel-resident heme metabolism genes in the top-25, which is consistent with the lineage signature result (10/21 erythroid hit, Table 4). IRF1's Hallmark Interferon Gamma Response enrichment is the expected primary signal. SPI1 does not reach Bonferroni significance in either Hallmark or Reactome at top-25, because neither library contains a coarse-grained myeloid lineage category that captures the recovered target program. This finding motivated the curated lineage signature library used as the primary structural validation in Figure 3 of the main text.

Table 5. Top 5 ChEA hits per TF for top-25 recovered targets. Cell line and species are part of the gene set name. Same-TF entries are bold.

TF	ChEA hit (top 5)	Overlap	Fold	p_{bonf}
KLF1	**EKLF 21900194 ChIP-Seq erythrocyte (mouse)**	**8/107**	**7.6×**	**3.0e-3**
KLF1	GATA2 19941826 ChIP-Seq K562 (human)	9/238	3.9×	1.7e-1
KLF1	GATA1 19941826 ChIP-Seq K562 (human)	6/110	5.6×	3.5e-1
KLF1	**KLF1 20508144 ChIP-Seq fetal liver erythroid (mouse)**	**6/112**	**5.5×**	**3.8e-1**
KLF1	CLOCK 20551151 ChIP-Seq 293T (human)	5/73	7.0×	3.9e-1
SPI1	**SPI1 23547873 ChIP-Seq NB4 (human)**	**16/377**	**4.3×**	**1.5e-5**
SPI1	TAL1 20887958 ChIP-Seq HPC-7 (mouse)	9/191	4.8×	3.0e-2
SPI1	RUNX1 20887958 ChIP-Seq HPC-7 (mouse)	7/123	5.8×	8.1e-2
SPI1	KLF6 26769127 ChIP-Seq PDAC cell line (human)	8/174	4.7×	1.1e-1
SPI1	LYL1 20887958 ChIP-Seq HPC-7 (mouse)	6/89	6.9×	1.1e-1
IRF1	IRF8 22096565 ChIP-ChIP GC-B (human)	7/32	22.3×	6.8e-6
IRF1	**IRF1 21803131 ChIP-Seq monocytes (human)**	**6/35**	**17.5×**	**4.3e-4**
IRF1	MAF 26560356 ChIP-Seq Th1 (human)	11/231	4.9×	2.6e-3
IRF1	SA1 27219007 ChIP-Seq B cells (human)	9/200	4.6×	4.3e-2
IRF1	UTX 26944678 ChIP-Seq Jurkat (human)	9/219	4.2×	8.9e-2

Table 6. Top 5 ENCODE hits per TF for top-25 recovered targets. Same-TF entries are bold.

TF	ENCODE hit (top 5)	Overlap	Fold	p_{bonf}
KLF1	FOSL1 K562 hg19	7/92	7.8×	1.5e-2
KLF1	GATA1 erythroblast hg19	9/177	5.2×	1.9e-2
KLF1	TAL1 G1E-ER4 mm9	6/80	7.7×	7.0e-2
KLF1	TAL1 erythroblast mm9	6/81	7.6×	7.5e-2
KLF1	ZNF274 K562 hg19	6/88	7.0×	1.2e-1
SPI1	TFAP2C HeLa-S3 hg19	7/145	4.9×	2.7e-1
SPI1	**SPI1 GM12878 hg19**	**8/215**	**3.8×**	**2.55e-2**
SPI1	EP300 CH12.LX mm9	8/258	3.2×	1.0
SPI1	**SPI1 HL-60 hg19**	**6/148**	**4.1×**	**1.0**
SPI1	ZNF384 ES-E14 mm9	6/150	4.1×	1.0
IRF1	RCOR1 CH12.LX mm9	10/143	7.1×	2.9e-4
IRF1	MYC CH12.LX mm9	9/112	8.2×	4.0e-4
IRF1	**IRF1 K562 hg19**	**13/291**	**4.6×**	**5.0e-4**
IRF1	POLR2A K562 hg19	14/420	3.4×	5.5e-3
IRF1	STAT2 K562 hg19	5/36	14.2×	1.5e-2

L.5. TRRUST literature-curated reference

TRRUST v2 contains 9396 human TF-target pairs curated from literature (Han et al., 2018). After restriction to the gene panel and the perturbed TF set, TRRUST annotates 16 IRF1 targets, 23 SPI1 targets, and 1 KLF1 target. Recovered top-25 overlap with TRRUST: IRF1 3/25 (*PSMB9*, *TAPBP*, *TAP1*, with *PSMB9* and *TAP1* marked as activations and *TAPBP* as unsigned), SPI1 0/25, KLF1 0/25.

The IRF1 3/25 TRRUST overlap is statistically significant against the random baseline of 16/2553 expected overlap ($p_{\mathrm{hypergeom}} < 10^{-3}$). The SPI1 and KLF1 TRRUST results are non-significant, but the recovered targets are strongly biologically interpretable in the lineage signature, ChEA, and ENCODE references (Tables 4–6). This pattern reflects TRRUST's known sparsity for K562-context lineage regulation rather than failure of the recovered adjacency: KLF1 has only one TRRUST-annotated target in the panel, and SPI1's TRRUST annotations weight toward immune contexts not well-captured in K562 erythromyeloid biology.

We retain TRRUST as a tertiary validation tier because it provides a positive control for IRF1 (a well-curated TF in TRRUST) and because the TRRUST annotation shown in Figure 3 panel A (TRRUST-annotated targets highlighted with colored fill) helps reviewers familiar with literature-curated GRNs orient to the recovered subnetworks.

Table 7. Top hit per TF in MSigDB Hallmark and Reactome 2022.

TF	Library	Top hit	Overlap	Fold	p_{bonf}
KLF1	Hallmark	Heme Metabolism	9/42	21.9×	3.0e-9
KLF1	Reactome	Metabolism of Porphyrins	2/5	40.8×	5.2e-1
SPI1	Hallmark	(none significant at top-25, $m = 49$)	–	–	n.s.
SPI1	Reactome	Signaling by Rho GTPases	5/66	7.7×	2.0e-1
IRF1	Hallmark	Interferon Gamma Response	8/60	13.6×	2.2e-6
IRF1	Reactome	ER-Phagosome Pathway	5/23	22.2×	1.0e-3

L.6. Co-regulator recovery and the local regulatory neighborhood

A consistent pattern across Tables 5 and 6 is that each TF's recovered top-25 enriches not only against its own ChIP targets but against the ChIP targets of its known co-regulators in the matched cellular context.

For KLF1, the co-regulator hits are GATA1, GATA2, FOSL1, and TAL1, all canonical erythroid TFs that share regulatory targets with KLF1 during erythroid maturation. GATA1 in particular is the master erythroid TF upstream of KLF1, and the recovered KLF1 adjacency captures targets that GATA1 binds in K562 chromatin.

For SPI1, the co-regulator hits in ChEA mouse hematopoietic progenitors are TAL1, RUNX1, and LYL1, three of the seven members of the heptad transcription factor complex (FLI1, ERG, GATA2, RUNX1, LYL1, LMO2, TAL1) that co-regulates myeloid commitment. The SPI1 top-25 recovers shared heptad targets even though our model fits SPI1 individually.

For IRF1, the co-regulator hit is STAT2 in K562 (Table 6, $14.2\times$ fold), reflecting the IRF1-STAT2 axis in canonical interferon signaling, plus IRF8 in B cells (Table 5, $22.3\times$ fold) reflecting the IRF1-IRF8 paralog target overlap.

This local-regulatory-neighborhood pattern is consistent with the single shared regulatory adjacency formulation (Section 2.2 of the main text): the model fits one A across all conditions, and the implicit constraint that targets co-regulated by a related TF should retain similar A entries propagates information beyond the directly perturbed TF. We do not claim the framework explicitly recovers co-regulator structure, but the structural validation tables show that the recovered targets respect known co-regulatory groupings.

L.7. Methodological notes

Multiple-testing correction. Bonferroni correction is applied within each library independently. We do not jointly correct across libraries because they are independent reference types being reported separately, not a single multiple-comparison test. Within-library correction uses the panel-restricted library size (after minimum-size and disease/generic filtering for Hallmark and Reactome; no filtering applied to ChEA, ENCODE, or lineage signatures, where pathway names do not match the disease/generic substring lists).

Background gene set. All hypergeometric tests use the gene panel ($n = 2553$ measured genes) as the background, not the full transcriptome. This is the conditional null appropriate for enrichment tests where pathway membership outside the panel cannot affect ranking.

Self-loops. Self-loops (the perturbed gene's own row in its column of A) are excluded from the top-25 ranking before enrichment, because the perturbed gene is clamped to its observed value and does not represent a learned regulatory edge.

M. Lineage signature library and pathway library filtering

This appendix documents the curated hematopoietic lineage signature library used as the primary structural validation reference in Section 3.2, Figure 3 and Table 4 and the denylist used to filter disease-context and generic catch-all pathways from MSigDB Hallmark and Reactome 2022 before enrichment testing. Both are referenced from Appendix L.

M.1. Lineage signature library

The lineage signature library consists of eight curated hematopoietic gene sets, three of which are matched-lineage signatures expected to enrich for the master transcription factors evaluated in Section 3.2 and five of which are negative or alternative

controls. The sets are intentionally compact (20–47 genes each) to focus on canonical lineage markers rather than broader expression programs, since the goal is structural validation against the most strongly lineage-defining genes rather than against the full transcriptional response of each lineage.

The eight signatures and their biological role in the validation are as follows.

ERYTHROID_LINEAGE (44 genes). Primary expected hit for KLF1. Includes the human globin cluster (HBA1, HBA2, HBB, HBG1, HBG2, HBD, HBE1, HBM, HBQ1, HBZ), heme biosynthesis enzymes (ALAS2, FECH, HMBS, ALAD, UROD, UROS, CPOX, PPOX), iron metabolism and heme catabolism genes (SLC25A37, TFRC, FTH1, FTL, BLVRB, HMOX1), erythroid transcription factors (KLF1, GATA1, NFE2, TAL1, LDB1, LMO2, EPOR), red blood cell membrane and cytoskeletal proteins (GYPA, GYPB, GYPC, EPB42, EPB41, SPTA1, SPTB, ANK1, SLC4A1, RHCE, RHD), and erythroid maturation markers (AHSP, HEMGN).

MYELOID_GRANULOCYTE_LINEAGE (34 genes). Alternative myeloid signature for SPI1. Includes primary granule proteases and antimicrobials (ELANE, MPO, PRTN3, CTSG, AZU1, BPI), secondary granule contents (LCN2, LTF, CAMP, MMP8, MMP9), neutrophil-enriched calcium-binding proteins (S100A8, S100A9, S100A12), defensins (DEFA1, DEFA3, DEFA4), granulocyte receptors and surface markers (CSF3R, FCGR3B, CEACAM3, CEACAM8, FPR1, FPR2, ITGAM), the NADPH oxidase complex (CYBA, CYBB, NCF1, NCF2, NCF4), granulocytic transcription factors (CEBPA, CEBPE, GFI1), and other markers (ALOX5, PGLYRP1).

MYELOID_MONOCYTE_LINEAGE (39 genes). Primary expected hit for SPI1. Includes canonical monocyte and macrophage markers (CD14, CD68, CD163, CD86, MRC1, MARCO, MERTK), Fc receptors (FCGR1A, FCGR2A, FCGR2B, FCGR3A), macrophage and dendritic cell transcription factors (SPI1, IRF8, MAFB, KLF4), lysozyme and lysosomal enzymes (LYZ, CTSS, CTSB, CTSD, LAPTM5), the CSF1 axis (CSF1R, CSF2RA), monocyte-enriched cytoskeletal proteins (AIF1, LST1, TYROBP, CORO1A, ARPC1B, ARHGDIB), complement components (CFD, C1QA, C1QB, C1QC), inflammation effectors (TLR2, TLR4, NLRP3, FCN1), and other myeloid markers (CD33, VAMP8, PYCARD).

MEGAKARYOCYTE_LINEAGE (22 genes). Discriminative control. Shares GATA1 and NFE2 with the erythroid lineage, so a recovered KLF1 adjacency that reflects erythroid commitment specifically (rather than the generic erythroid–megakaryocyte progenitor program) should hit ERYTHROID_LINEAGE more strongly than MEGAKARYOCYTE_LINEAGE. Includes platelet integrins and glycoproteins (ITGA2B, ITGB3, GP1BA, GP1BB, GP9, GP5, GP6), alpha granule contents (PF4, PPBP, VWF, FGA, FGB), the thrombopoietin receptor (MPL), megakaryocytic transcription factors (FLI1, RUNX1, MEIS1), P-selectin and adhesion (SELP, TLN1), shared erythroid–megakaryocyte transcription factors (GATA1, NFE2), and other markers (TUBB1, PROS1).

LYMPHOID_T_CELL_LINEAGE (33 genes). Negative control. K562 lacks T cell lineage commitment and no enrichment is expected for any of KLF1, SPI1, or IRF1. Includes the T cell receptor complex (CD3D, CD3E, CD3G, CD247, TRAC, TRBC1, TRBC2), coreceptors and accessory molecules (CD2, CD4, CD7, CD8A, CD8B, CD27, CD28), T cell transcription factors (TCF7, LEF1, TBX21, GATA3, FOXP3, RUNX3, BCL11B), effector molecules (GZMA, GZMB, GZMK, PRF1, IFNG), interleukin receptors (IL7R, IL2RA, IL2RB), and T-specific signaling components (ZAP70, LCK, ITK, THEMIS).

LYMPHOID_B_CELL_LINEAGE (28 genes). Negative control. Includes the B cell receptor complex (CD19, MS4A1, CD22, CD79A, CD79B), immunoglobulin chains (IGHM, IGHD, IGLL1, IGKC), B cell transcription factors (PAX5, EBF1, POU2AF1, TCF3, BCL6), plasma cell terminal differentiation transcription factors (PRDM1, XBP1, IRF4), surface and signaling molecules (CD72, BANK1, BLNK, MZB1, TNFRSF13B, TNFRSF13C), and other B cell markers (VPREB1, VPREB3, FCRL1, FCRL2, FCRL5).

INTERFERON_RESPONSE (47 genes). Primary expected hit for IRF1. Combines type I and type II interferon-stimulated genes into a single signature to avoid splitting power across two highly overlapping sets. Includes canonical interferon-stimulated genes (ISG15, ISG20, IFIT1, IFIT2, IFIT3, IFIT5), the OAS family (OAS1, OAS2, OAS3, OASL), MX dynamins (MX1, MX2), antiviral IFITM proteins (IFITM1, IFITM2, IFITM3), other ISGs (RSAD2, IFI6, IFI27, IFI44, IFI44L, USP18, BST2, HERC5, HERC6, EIF2AK2), the GBP family (GBP1, GBP2, GBP4, GBP5), JAK–STAT signaling components (STAT1, STAT2, IRF1, IRF7, IRF9), antigen presentation machinery (HLA-A, HLA-B, HLA-C, TAP1, TAP2, TAPBP, PSMB8, PSMB9, PSME1, PSME2), and type II effectors (CIITA, IDO1, WARS1).

CELL_CYCLE (38 genes). Negative control. No enrichment is expected for KLF1, SPI1, or IRF1 if the recovered adjacency captures lineage-specific regulation rather than generic proliferation. Includes cyclins (CCNA1, CCNA2, CCNB1, CCNB2, CCND1, CCND2, CCND3, CCNE1, CCNE2), cyclin-dependent kinases (CDK1, CDK2, CDK4, CDK6, CDK7), CDK inhibitors (CDKN1A, CDKN1B, CDKN2A, CDKN2B), proliferation markers (MKI67, PCNA, TOP2A), replication licensing factors (MCM2–MCM7), spindle and mitotic regulators (AURKA, AURKB, PLK1, BUB1, BUB1B, MAD2L1), and E2F transcription factors with RB1 and TFDP1.

Sources used to curate the gene lists are in (Hay et al., 2018), (Novershtern et al., 2011), the MSigDB C8 cell type signatures collection, (Schoenfelder & Fraser, 2019) for erythroid biology, and (Tenen, 2003) for myeloid transcription factor circuitry. The total number of unique genes across all eight signatures is 283. The library is provided as a tab-separated file in Enrichr format (one set per line: `<set_name>\t\t<gene1>\t<gene2>\t...`) so that it is consumed directly by the same enrichment infrastructure used for MSigDB Hallmark and Reactome 2022. Bonferroni multiple-testing correction over $m = 8$ provides modest correction with substantial statistical power for the matched-lineage hits expected to dominate.

M.2. Pathway library filtering for Hallmark and Reactome

For the MSigDB Hallmark 2020 and Reactome 2022 enrichment runs reported in Appendix L (Table 7), we apply a denylist filter before testing to exclude two categories of pathways that are statistically real annotations but biologically inappropriate for evaluating a regulatory network recovered from K562 perturbation data.

The first category is disease and pathogen response pathways. These pathways describe the cellular response to infection or pathology rather than physiological regulation, and pathways such as "Tuberculosis Infection Response" or "Influenza Infection" are statistically enrichable for any general inflammatory or interferon program but do not describe normal regulatory biology. We exclude pathways whose names contain any of the following substrings (case-insensitive): `infection`, `infectious`, `disease`, `diseases`, `tuberculosis`, `mtb`, `mycobacterium`, `hiv`, `influenza`, `sars`, `coronavirus`, `covid`, `hepatitis`, `leishmania`, `malaria`, `plasmodium`, `carcinoma`, `neoplasm`, `leukem`, `tumor`, `tumour`, `cancer`, `syndrome`, `disorder`.

The second category is generic catch-all pathways at the top of the Reactome hierarchy. These pathways are broad superset annotations such as "Generic Transcription Pathway" or "Signal Transduction" that aggregate large numbers of genes across many specific functions. They are consistently enriched in any sufficiently broad gene list and provide no biological discrimination. We exclude pathways whose names match the following exact strings: `Generic Transcription Pathway`, `Developmental Biology`, `Signal Transduction`, `Metabolism`, `Metabolism of proteins`, `Metabolism of RNA`, `Gene expression (Transcription)`, `Gene expression`, `RNA Polymerase II Transcription`, `Cellular responses to stimuli`, `Cellular responses to stress`, `Cellular responses to external stimuli`, `Hemostasis`, `Programmed Cell Death`, `Cell Cycle`, `Cell Cycle, Mitotic`, `Mitotic Cell Cycle`, `Vesicle-mediated transport`, `Membrane Trafficking`, `Post-translational protein modification`.

The exact-name match is applied after stripping the trailing Reactome stable identifier (e.g., `R-HSA-NNNNNN`) that Enrichr appends to Reactome pathway names. We do not pattern-match on the substring `Metabolism`, because specific subpathways such as "Heme Metabolism" or "Metabolism of Lipids" are legitimate and biologically informative annotations that should be tested. The exact-string filter targets only the top-level catch-all entries.

The filter is not applied to the lineage signature library (Appendix M.1), to ChEA 2022, or to ENCODE TF ChIP-Seq 2015. The lineage signatures are curated specifically for this evaluation and contain no disease or generic-catch-all entries by construction. The ChEA and ENCODE libraries name gene sets by transcription factor and cell line rather than by pathway concept, so neither the disease substring list nor the generic-catch-all exact match list captures any entries in those libraries.