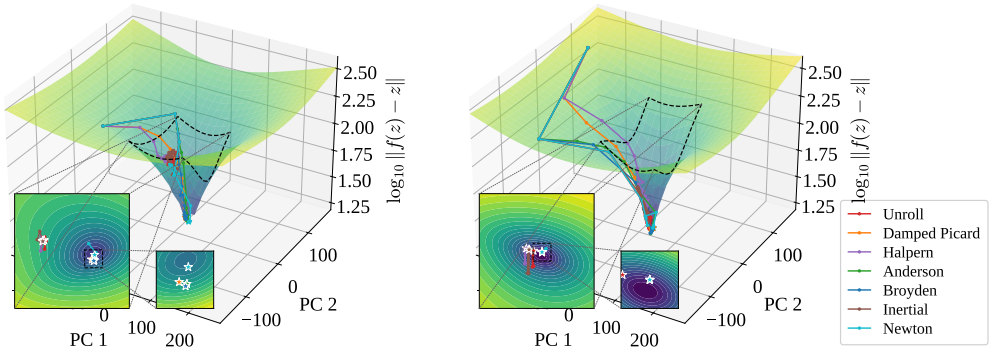
TOWARD SOLVER-INVARIANT DEEP EQUILIBRIUM MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep equilibrium models define their output through a fixed point of a weight-tied iteration. This formulation motivates solver invariance: predictions should be insensitive to the numerical procedure used for inference. This property need not hold at a finite computation budget. With model weights fixed, replacing the inference solver can reduce the accuracy of a trained MDEQ checkpoint to a fraction of its training-solver accuracy. We investigate the ungated identity path as a source of this sensitivity. FROST removes that bypass, scales the recurrent path with a single learned scalar, and initializes the otherwise free input scale by a self-similar rescaling rule. Without assuming a globally unique fixed point, we give sufficient conditions linking endpoint residuals, state proximity, and readout margins, and measure their premises on trained models. Under identical implicit training, FROST holds a substantially smaller accuracy spread across seven inference procedures than MDEQ. A constant input scale reproduces it, and the result transfers to a non-vision Poisson task.

1 INTRODUCTION



(a) MDEQ (b) FROST (ours)

Figure 1: Trajectories of seven solvers for one validation input, using one checkpoint per architecture. Each panel uses model-specific PCA coordinates; insets magnify the endpoint clusters.

Deep Equilibrium Models (DEQs) (Bai et al., 2019a) define their output as a fixed point of a weight-tied iteration. If solvers reach the same fixed point, the output is a property of the trained model rather than of the numerical procedure. Inference at a finite budget need not satisfy this premise. Swapping the training solver of MDEQ (Bai et al., 2020) for other inference procedures, with weights held fixed, can sharply reduce its accuracy (Figure 1, quantified in Table 2). We name this failure the *solver invariance problem*. The issue concerns predictions, not only representation differences; it does not contradict solver independence when a common fixed point is reached exactly.

Solver robustness would allow a model trained with an expensive solver to be served with a cheaper procedure. Implicit differentiation describes a selected equilibrium branch, whereas deployment returns a numerical approximation. Evaluating only the training solver does not establish that alternative procedures preserve predictions.

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

We design FROST, a DEQ parameterization with a single additional learnable scalar. Prediction agreement does not require a globally unique fixed point: distinct returned states may belong to the same prediction equivalence class. FROST pursues this target architecturally. The recurrent residual cell contains a direct identity path with no dedicated gain parameter. Other parts of the cell can still attenuate its effect. FROST removes this path and scales the learned recurrent path instead. The input scale is set from the scalar's initial value and a Hurst exponent, following a time axis self-similar rescaling with the input map treated as linear and absorbed into the stem, in the spirit of fractional Brownian motion (Mandelbrot & Van Ness, 1968). Given the initial scalar and exponent, this construction specifies the input prefactor without a separate prefactor search, though the initial scalar and the exponent themselves remain free design choices; their sensitivity is evaluated in the appendix.

Two conditional statements organize the analysis of prediction agreement. First, if two solver endpoints have small fixed point residuals and the averaged Jacobian of the residual map between them is nondegenerate, the endpoints are provably close. Second, any state within a margin-dependent radius of a reference state has the same prediction, provided the head sensitivity is bounded over that region. FROST has smaller residuals and endpoint distances in the sampled diagnostics, and prediction agreement holds whenever the sampled margin criterion is met. The measurements appear in §4 and §5.

Under identical implicit training, FROST has several times smaller accuracy spread than MDEQ across the seven evaluated procedures.

Our contributions are as follows.

- We quantify finite-budget solver sensitivity in MDEQ and investigate the role of the direct identity path through gain interventions and parameterization controls.
- We introduce FROST, a bypass-free parameterization with one additional scalar and a rescaling-based input prefactor, compatible with implicit training.
- We derive sufficient conditions for cross-solver prediction agreement without global contraction, and distinguish these conditions from their empirical diagnostics.
- We evaluate seven inference procedures, identical training regimes, input-scale controls, and a non-vision task.

2 Related Work

Implicit models and stabilization. Deep Equilibrium Models compute their output as the fixed point of a weight-tied cell and backpropagate through the equilibrium by implicit differentiation (Bai et al., 2019a), building on weight-tied precursors (Bai et al., 2019b) and scaling to large vision benchmarks through multiscale variants (Bai et al., 2020). Efforts to stabilize the forward iteration fall into two families. One family constrains the operator so that convergence is guaranteed by construction, as in monotone DEQs (Winston & Kolter, 2020) and positive concave DEQs (Gabor et al., 2024), and pays for the guarantee with architectural restrictions on the operator. The other family keeps the operator unconstrained and softly regularizes its sensitivity, as in Jacobian regularization (Bai et al., 2021), which offers no guarantee in return. Equilibrium-style architectures and stability mechanisms have also been studied in imaging and analysis settings (Gilton et al., 2021; Ling et al., 2024), and recent work revisits DEQ training and inference through learned solvers and reversibility (Bai et al., 2022; McCallum et al., 2025), supported by general training libraries (Geng & Kolter, 2023). These stabilization strategies address properties of the operator or training dynamics; such properties alone do not establish prediction agreement across finite-budget inference procedures. FROST occupies a different position. It leaves the operator weights unconstrained, analyzes conditional agreement with empirical diagnostics, and treats the inference procedure as an axis of evaluation. Table 1 summarizes this placement.

Fixed point solvers. Broyden's method (Broyden, 1965), Anderson acceleration (Anderson, 1965; Walker & Ni, 2011), Mann iteration (Mann, 1953), and Halpern iteration (Halpern, 1967) approach fixed-point problems with different convergence assumptions and trajectories. Implicit differentiation concerns the equilibrium reached, rather than the path to it (Pineda, 1987; Krantz & Parks, 2002; El Ghaoui et al., 2021; Gould et al., 2021); it does not ensure that finite-budget procedures return

Method	Guarantee	Constraint	Stability basis
MDEQ (Bai et al., 2020)	none	none	numerical solve
monDEQ (Winston & Kolter, 2020)	unique fixed point	monotone splitting	proof
JacReg (Bai et al., 2021)	none	none	regularized objective
pcDEQ (Gabor et al., 2024)	convergence	positive concave operator	proof
FROST (ours)	**none claimed**	**none**	**empirical diagnostics**

Table 1: Stabilization approaches. Constraints refer to operator weights or structure; guarantees require the assumptions in the cited work and do not imply equal predictions at a fixed finite budget.

equivalent predictions. We therefore treat the solver as an experimental variable rather than an implementation choice, and our protocol spans damped, anchored, accelerated, quasi-Newton, momentum, and Newton-Krylov schemes (Knoll & Keyes, 2004) together with truncated ones, so that agreement cannot be an artifact of a single solver family.

Path independence and test-time computation. Closest to our question, Anil et al. (2022) study path independence: whether an equilibrium model converges to the same attractor regardless of initialization given enough computation. They show that more path-independent models exploit additional test-time iterations better, and that per-example path independence on out-of-distribution inputs tracks accuracy. Their axis of variation is the initialization and the iteration count under a single solver in the convergence regime. We vary a different axis: the root-finding algorithm itself, at a fixed and often truncated budget, including procedures that do not converge. The two are related views of computation-invariant prediction, yet solver invariance does not follow from path independence, since a path-independent model can still disagree across solvers before convergence. Our per-input and distribution-shift results (§5) parallel their per-example analysis.

Self-similarity as a design prior. Statistical self-similarity is a long-standing prior for natural signals and stochastic processes, formalized by fractional Brownian motion and fractal geometry (Mandelbrot & Van Ness, 1968; Mandelbrot, 1983; Hutchinson, 1981). FROST uses this prior in a narrow and checkable role. The time axis rescaling of fractional Brownian motion enters the derivation of the input scale (§3).

Computation leaking into model semantics. The failure we study has a counterpart in continuous depth models, where reported behavior is entangled with the discretization and tolerances of the ODE solver (Chen et al., 2018; Dupont et al., 2019), and trained neural ODEs can depend on the specific numerical solver, so that evaluating with a finer solver than the one used in training degrades accuracy (Ott et al., 2021). State space models face a related question in the discretization of their recurrence (Gu et al., 2022; Smith et al., 2023; Gu & Dao, 2024). Our results study this concern in DEQs through an architectural change and explicit solver-swap evaluation.

3 FROST

3.1 Formulation

We formulate general DEQs as a weight-tied residual iteration of a neural network:

$$z_{t+1} \;=\; f_\theta(z_t, x), \qquad z^\star \;=\; f_\theta(z^\star, x), \tag{1}$$

where x denotes the stem features of the input, held fixed across iterations, and f_θ is a weight-tied cell whose standard instantiations wrap an unconstrained block together with an additive identity path (Bai et al., 2019a; 2020). Inference locates $z^\star$ with a numerical root solver, typically Broyden's method (Broyden, 1965; Bai et al., 2019a), and a classification head $\hat{y}$ reads out the state the solver returns. Training either backpropagates through the equilibrium by implicit differentiation or unrolls a fixed number of iterations (Bai et al., 2019a).

In the main multiscale implementation, A denotes the learned recurrent path, $\lambda \in (0, 1]$ is the additional scalar, H is a fixed rescaling exponent, and T is the iteration budget. Solvers share $z_0 = 0$ within each main comparison. The separate CIFAR parameterization controls use an input-dependent initialization, as specified in Appendix C.

3.2 Problem Definition

The claim under study is operational, so we fix its measurement before the architecture. Let S be a specified set of inference procedures applied to one trained model, with weights fixed and each procedure's budget reported. We include a truncated forward unroll in S because deployment may return a finite iterate rather than a numerically converged root. We define the solver spread as the gap between the best and worst validation accuracy over S. Accordingly, our empirical use of "solver invariance" means a small finite-budget accuracy spread, not exact equality of states, guaranteed convergence, or mathematical independence from every possible solver. Small aggregate accuracy spread does not imply agreement on each example; the state and margin diagnostics address that separate question. Our protocol instantiates S with seven procedures spanning truncated, damped, anchored, accelerated, quasi-Newton, momentum, and Newton-Krylov schemes (Table 2).

We investigate a structural source of this sensitivity. A schematic residual cell adds the state to the learned path,

$$f_\theta(z, x) \;=\; \sigma\big(F_\theta(z) + z + x\big), \tag{2}$$

so the step Jacobian factorizes by the chain rule as

$$\frac{\partial z_{t+1}}{\partial z_t} \;=\; D_\sigma\big(J_F + I\big), \tag{3}$$

where D_σ is the activation derivative and J_F is the block Jacobian. This schematic isolates the identity path; the implemented multiscale cells also contain normalization and branch fusion (Appendix C). The additive identity acts as a direct bypass around every learnable operator. In particular, the architecture has no parameter that directly gates this branch; attenuation must instead arise through cancellation by the learned path, the surrounding activation, or the numerical solver. Quasi-Newton solvers can provide such damping through their low-rank corrections. Training under such a solver can therefore couple the learned representation to its numerical trajectory. A swapped or truncated procedure changes that trajectory and exposes the ungated reentry path. Gain interventions and Jacobian measurements test this interpretation (§5.3).

3.3 Derivation

The diagnosis specifies what the cell must provide: a damping path that lives inside the learnable map rather than in the solver. We derive the FROST update in two steps, a learned gain on the loop and a rescaling principle that fixes the input scale.

Following the state space formulation (Gu et al., 2022; Smith et al., 2023; Gu & Dao, 2024), consider the continuous dynamics

$$\dot{h}(t) \;=\; A(h(t)) \;+\; B(x(t)), \tag{4}$$

where A is the state transition operator and B the input mapping. An explicit Euler step of size λ gives $h_{t+1} = h_t + \lambda\big(A(h_t) + B(x_t)\big)$. The leading h_t is a unit-gain state passthrough, analogous to the residual bypass in equation 3. FROST removes it and keeps only the flow field under the learned gain,

$$z_{t+1} \;=\; \lambda\, A(z_t) \;+\; \lambda\, B(x). \tag{5}$$

This is an architectural construction, not an Euler approximation preserving the original ODE. The learned path remains unconstrained and can learn self-coupling; we do not assert an identical function class at fixed width. For the bare map equation 5, the step Jacobian is λJ_A. Surrounding normalization in the implementation contributes additional Jacobian factors, so $\lambda \le 1$ alone does not imply contraction.

The gain alone leaves the input coupling underdetermined, since any constant prefactor on B preserves the structure of equation 5 while the balance it sets between injection and state can decide whether the iteration trains (Appendix D). To specify the input prefactor, we impose a joint rescaling of time and input. For $H \in (0, 1)$, define

$$x_\lambda(t) \;\coloneqq\; \lambda^{-H} x(\lambda t), \qquad h_\lambda(t) \;\coloneqq\; h(\lambda t) \tag{6}$$

and require covariance under transformed operators, not identical dynamics at fixed weights. Substituting into equation 4 identifies the scaled operators $A_\lambda = \lambda A$ and $B_\lambda(x) = \lambda B(\lambda^H x)$, which for linear B reduces to $B_\lambda = \lambda^{1+H} B$. The rescaling equation 6 is the self-similarity symmetry

of fractional Brownian motion (Mandelbrot & Van Ness, 1968; Falconer, 2013), and Appendix A develops the derivation in full, starting from the definition of fBm. We use this symmetry as a design prior; the static image input and the learned trajectory are not assumed to be fBm.

Absorbing B into the stem features x, the FROST update reads

$$z_{t+1} \;=\; \lambda\, A(z_t) \;+\; \lambda^{1+H} x. \tag{7}$$

Equation equation 7 specifies the gain and injection before the surrounding normalization of the main implementation (Appendix C). Its scalar stays in $(0, 1]$, and the readout applies λ^{-H} before the unchanged classifier, adding no further parameter. The parameter-count increase over the main multiscale baseline is one scalar. The following schematic maps contrast recurrent paths, omitting implementation-specific normalization and fusion:

$$z_{t+1}^{\text{MDEQ}} \;=\; \sigma\big(F_\theta(z_t) \;+\; z_t \;+\; x\big), \tag{8}$$

$$z_{t+1}^{\text{pcDEQ}} \;=\; \sigma_c\big(F_\theta^{+}(z_t) \;+\; z_t \;+\; x\big), \tag{9}$$

$$z_{t+1}^{\text{FROST}} \;=\; \lambda\, A(z_t) \;+\; \lambda^{1+H} x, \tag{10}$$

where pcDEQ (Gabor et al., 2024) constrains the weights to be nonnegative under a concave activation σ_c while retaining the identity path.

Given H and λ_{init}, the derivation specifies a candidate constant $\lambda_{\text{init}}^{1+H}$ without a separate prefactor search. Experiments test this constant (§5.4), and Appendix C reports its sensitivity to the design choices.

4 Conditions for Solver Agreement

4.1 Solver Agreement

Solver agreement asks that the states different solvers return be read the same way, which is weaker than requiring a single fixed point that every solver reaches. Two conditional results formalize this distinction: a residual-to-distance bound and a distance-to-prediction bound. Neither depends on the algorithm producing the states.

Proposition 1 (State agreement)**.** *Let $J(z) = \partial f_\theta(z, x)/\partial z$, and let z_a and z_b be two solver endpoints with residuals $\delta_a = \|f_\theta(z_a, x) - z_a\|$ and $\delta_b = \|f_\theta(z_b, x) - z_b\|$. Assume the segment integral of J represents the difference of f_θ along the segment, as holds for a continuously differentiable map. If the segment-averaged $I - J$ has smallest singular value at least $m > 0$, then*

$$\|z_a - z_b\| \;\le\; \frac{\delta_a + \delta_b}{m}. \tag{11}$$

Proof. Write $g(z) = f_\theta(z, x) - z$. Along the segment $z_s = (1-s)z_a + s z_b$, the fundamental theorem of calculus gives $g(z_b) - g(z_a) = -\bar{M}(z_b - z_a)$ for the segment-averaged $\bar{M} = \int_0^1 (I - J(z_s))\, ds$. Since $\sigma_{\min}(\bar{M}) \ge m$, we have $\|z_b - z_a\| = \|\bar{M}^{-1}(g(z_b) - g(z_a))\| \le (\delta_a + \delta_b)/m$. □

The empirical diagnostic uses a different quantity: the minimum of estimated pointwise singular values of $I - J$ at five points on the segment. Denote it by $\hat{m}$. It is not a lower bound on the singular value of the averaged matrix. Across 32 images and the three pairs among Broyden, Anderson, and damped Picard, $\hat{m}$ is approximately 0.08–0.09 for both models, while mean residuals are 0.17 for FROST and 7.38 for MDEQ. The proxy inequality holds in all 96 comparisons for each model, so what separates the two is not whether the bound holds but the residual and distance magnitude it acts on. A separate 128-image diagnostic gives mean relative distances of 0.3–0.4 percent for FROST and 13–17 percent for MDEQ across these three pairs, over an order of magnitude apart. Thus proximity is directly observed.

4.2 Agreement Condition

State agreement places two solver endpoints close together, and the prediction is invariant once they are close enough.

Proposition 2 (Prediction preservation). *Let r be a reference state, $\hat{y}(z) = \arg\max_k h_k(z)$, and $\gamma_r = h_{(1)}(r) - h_{(2)}(r) > 0$. Assume $\|h(z) - h(r)\|_\infty \le L_h \|z - r\|_2$ throughout the region considered, with $L_h > 0$. Any state z with*

$$\|z - r\| \; < \; \frac{\gamma_r}{2L_h} \tag{12}$$

satisfies $\hat{y}(z) = \hat{y}(r)$.

Proof. Let $k^\star = \hat{y}(r)$. Each logit changes by less than $\gamma_r/2$, so for every $j \neq k^\star$, $h_{k^\star}(z) - h_j(z) > h_{k^\star}(r) - h_j(r) - \gamma_r \ge 0$. □

Taking $r = z_a$ combines the propositions: $(\delta_a + \delta_b)/m < \gamma_{z_a}/(2L_h)$ suffices for pairwise prediction agreement when both premises hold. No exact fixed point is needed as the reference.

The diagnostic instead uses a 200-step forward state as r and the maximum head slope from eight random perturbations at each of three scales as $\hat{L}_h$. Over 32 images and five 26-step procedures, $40/160$ comparisons satisfy the sampled radius criterion for FROST (25.0 percent), versus $5/160$ for MDEQ (3.1 percent), and all such predictions match the reference. Outside the criterion, agreement is 65.8 versus 45.2 percent. These are solver-to-reference comparisons, not all pairwise solver comparisons or accuracy guarantees. Because the sampled radius is not certified, the firing rate indicates agreement rather than bounding it, and it is not a prediction-disagreement rate, since comparisons that do not fire can still agree, consistent with the small spread of §5.2. Mean reference margins are 2.77 and 0.99, and the radius also depends on head sensitivity. Appendix C details both diagnostics.

Appendix F analyses the increments of the bare iteration, where the input scale sets the initial increment and the learned scalar sets its asymptotic decay rate, together with a Broyden measurement on ImageNet. That linear geometry describes how an iterate approaches its fixed point, whereas cross-solver agreement rests on the two conditions above.

5 Experiments

5.1 Setup

We evaluate on ImageNet-1K (Deng et al., 2009) and CIFAR-100 (Krizhevsky et al., 2009). The main comparisons share the multiscale backbone, channel widths, and convolutional parameter shapes of MDEQ (Bai et al., 2020). FROST changes recurrent and input coupling, their placement relative to normalization, and the scalar readout scale, adding one learnable parameter. Separate CIFAR parameterization and input-scale controls are identified below. Our baselines are the unconstrained MDEQ and MDEQ with Jacobian regularization (Bai et al., 2021), the standard soft stabilizer, with the contraction-constrained pcDEQ (Gabor et al., 2024) added on CIFAR.

The main protocol evaluates seven procedures: five-step forward unroll, damped Picard (Mann, 1953), Halpern (Halpern, 1967), Anderson acceleration (Anderson, 1965), Broyden (Broyden, 1965), inertial iteration, and Jacobian-free Newton-Krylov (Knoll & Keyes, 2004). Each starts at $z_0 = 0$, with a nominal solver budget of 26, except for the five-step unroll. Newton-Krylov counts Jacobian-vector products within its budget; equal nominal budgets do not imply equal wall-clock cost (Appendix C). The solver spread of §3.2 is the gap between the best and worst validation accuracy over this set, read from the same weights with no fine-tuning.

Figure 1 visualizes the solver trajectories on a residual landscape. We project the iterates of all solvers onto the top two principal components of their pooled trajectory and read the residual $\|f_\theta(z, x) - z\|$ on that plane. The plane captures 0.26 of the variance for MDEQ and 0.60 for FROST, and since it is a two dimensional slice the lowest residual reachable on it exceeds the full space minimum, so its floor sits higher. Both panels use the same validation image and initialization $z_0 = 0$. Their PCA coordinates are fitted separately and should not be interpreted as a shared state-space coordinate system.

Model	Train	Unroll	Picard	Halpern	Anderson	Broyden	Inertial	Newton	Spread
FROST	Unroll	63.15	57.51	56.25	57.05	56.87	53.45	57.48	9.70
FROST	Broyden	55.37	60.08	56.06	60.07	60.10	54.74	60.05	**5.36**
MDEQ	Broyden	20.57	55.90	23.45	56.40	56.29	19.56	55.71	36.84
MDEQ + Bai et al. (2021)	Broyden	49.30	57.42	41.94	57.43	58.28	36.09	57.39	22.19

Table 2: ImageNet-1K Top-1 under seven solvers on the full validation set. Spread is best minus worst, and underline marks the training solver. Least spread is bold.

5.2 Solver Invariance on ImageNet

Table 2 reports Top-1 across the solvers for both architectures. Trained with Broyden, FROST holds within 5.36 points across the seven solvers while MDEQ swings by 36.84, collapsing to 20.57 under the five-step unroll, to 23.45 under Halpern, and to 19.56 under the inertial iteration from a converged 56.29. FROST is read with the same cheap and truncated solvers and does not collapse, so the swing reflects the architecture rather than a train-inference mismatch that would penalize both models. Jacobian regularization narrows the swing but does not close it. Including Newton-Krylov and inertial iteration preserves the spread advantage, so it is not confined to one solver family. Because these rows hold the training solver fixed at Broyden, they compare the two architectures under one training regime. Table 2 also lists an unroll-trained FROST: a truncated-budget operating point whose wider 9.70-point spread and higher five-step accuracy are those of a non-converged model. Its role is to exhibit, within a single trained model, the finite-budget behavior that equilibrium models are subject to; the equilibrium result is the Broyden-trained row, and Appendix G traces the depth profile of each.

A second run with a longer schedule reproduces the gap. Trained for 100 epochs and evaluated under the same seven solvers, FROST spreads 4.25 against MDEQ's 38.68, so the separation is not confined to the checkpoints in Table 2.

The MDEQ optimum of 56.29 sits below its single-solver benchmark, which raises whether the spread is an undertraining artifact rather than a property of the operator. Four observations place it with the operator. The longer schedule above leaves the gap intact, so additional training does not close it. The four converging solvers, Picard, Anderson, Broyden, and Newton-Krylov, reach comparable MDEQ accuracy within a 0.7-point range near 56, so the readout is internally consistent and the collapse is confined to the truncated and anchored procedures. At the endpoints used in Section 4, the MDEQ fixed-point residual is 7.38 against 0.17 for FROST, so its solved state stays far from a fixed point even under its own training solver. Finally, the released MDEQ-SMALL checkpoint, which reaches the accuracy reported for this architecture, shows the same pattern under our protocol with a spread of 16.05 (Appendix H). The spread therefore tracks the operator, not an incompletely trained checkpoint.

An unroll-only control separates architecture from training objective: over unroll, damped Picard, Halpern, Anderson, and Broyden, MDEQ and FROST have spreads of 4.30 and 6.90 points and training-procedure Top-1 scores of 61.78 and 63.15. Thus both have relatively small spread after unroll-only training; the distinguishing result is that, under the shared implicit curriculum, FROST retains a 4.73-point spread while MDEQ widens to 35.83 over this same five-procedure subset. The FROST advantage is therefore specific to implicit training: under unroll-only training both are already robust, while under the shared implicit curriculum only FROST stays so.

Spread is an aggregate over the validation set, so it need not reflect agreement on individual inputs. We therefore measure per-input disagreement directly, as the fraction of $10{,}000$ validation images on which a solver pair assigns different top-1 labels. Among the four converging solvers, Picard, Anderson, Broyden, and Newton-Krylov, FROST predictions differ on at most 0.99 percent of images, against 6.9 to 13.1 percent for MDEQ. The truncated five-step unroll and the halted Halpern and inertial runs disagree more for both models, and across all seven solvers the mean pairwise disagreement is 15.9 percent for FROST versus 53.3 for MDEQ. Per-input agreement therefore follows the aggregate spread rather than masking disagreement beneath it.

The same ordering carries to the two evaluated distribution shifts. On ImageNet-R and ImageNet-V2, the Broyden-trained FROST leads the Broyden-trained MDEQ by 4.01 and 4.35 points under Broyden, with no retraining. Switching these checkpoints to five-step unroll loses 3.25 and 5.03 points for FROST, versus 15.89 and 28.86 for MDEQ (Appendix E).

5.3 Investigating the Mechanism

We vary the FROST scalar and the baseline's convolutional-path gain at fixed weights and measure the secant amplification $\|f(z_a) - f(z_b)\|/\|z_a - z_b\|$ between simultaneous unroll and Halpern states, averaged over the trajectory. Under the intervention FROST stays near 0.01 while MDEQ stays in the 0.7 to 1.0 range, a contrast of 0.0104 versus 0.9569 along the tested state differences.

Power-iteration estimates of the full cell Jacobian norm $\|J_f\|_2$, including normalization, exceed one over 128 sampled images: approximately 9.7 for FROST and 11.4–13.8 for MDEQ, with a mid-trajectory comparison of 9.5 versus 13.8. Large worst-direction sensitivity is compatible with small secant gains along realized trajectories. That along-trajectory gentleness, rather than any global contraction, is what lets the truncated and anchored solvers agree, so faster forward convergence and cross-solver agreement are two views of one property rather than one reducing the other. The measurements therefore do not establish global Euclidean contraction or isolate bypass removal as the sole cause of robustness.

5.4 The Self-Similarity Prior

The derivation of §3.3 supplies the input scale as a fixed constant, 0.287 at the initial scalar, so we can fix it directly rather than couple it to the learned λ. The scalar is genuinely trained: from its 0.5 initialization it settles near 0.02 in the main checkpoints, so it is a learned parameter, not a fixed value, and the coupled scale it implies differs by orders of magnitude from the 0.287 constant, making the constant-scale variant a distinct model rather than a reparameterization of the dynamic one. At the trained scalar the coupled injection scale λ^{1+H} is near 10^{-3}, so the derived constant acts at initialization rather than as a maintained operating value, and training moves the coupled scale downward from it. We train a variant that holds the input prefactor at this constant and is otherwise identical to FROST. On ImageNet it spreads 1.51 across seven solvers and reaches 59.32 Top-1 under its training solver, compared with 60.10 for the dynamic variant. Solver invariance is preserved at a constant input scale, so the self-similar rescaling is a principled way to choose an otherwise free input-scale initialization, not the source of the robustness.

A separate CIFAR scale sweep, using a gated residual implementation, reaches 54–57 percent at the derived value and larger tested constants, while 0.05 yields only a few percent (Appendix D). This tests a working magnitude in that implementation, not uniqueness of the derived value or an architecture-independent threshold.

5.5 Comparison on CIFAR

Model	Constraint	CIFAR-10	CIFAR-100
MDEQ (Bai et al., 2020)	none	86.40	52.21
pcDEQ-NC (Gabor et al., 2024)	$W \geq 0$, softsign	82.75	48.38
pcDEQ-PC (Gabor et al., 2024)	$W \geq 0$, sigmoid	83.27	47.30
FROST	$0 < \lambda \leq 1$	**86.74**	**57.10**

Table 3: Comparison on CIFAR. pcDEQ constrains the operator, whereas FROST applies a scalar loop gain and leaves the operator weights unconstrained.

pcDEQ (Gabor et al., 2024) reaches convergence by constraining the operator to nonnegative weights under a concave activation, which separates how a model is made convergent from whether it is. On CIFAR-10, FROST is within 0.34 points of the unconstrained baseline while pcDEQ is lower by three to four points; on CIFAR-100, FROST is higher by 4.89 points (Table 3). These results compare the tested parameterizations, not their function classes or the best achievable accuracy of either method.

pcDEQ is convergent by construction, so it tests whether a guaranteed fixed point alone removes the solver spread. Its accuracy still depends on the solver. Read under its Broyden training solver it reaches 48.4 percent on CIFAR-100, and under a five-step unroll it falls to 39.0, a loss of 9.4 points. The ImageNet solver-swap losses in Table 2 are separate comparisons, not like-for-like CIFAR controls. A convergence guarantee alone therefore does not remove the solver spread, so a guaranteed fixed point is not by itself sufficient for the agreement we study.

5.6 Component Ablation

FROST differs from the multiscale baseline in four places at once: it removes the identity bypass, adds the scalar λ, relocates the normalization, and rescales the readout. To isolate the bypass we vary only its gain on the main FROST cell: we insert a scalar γ into the fixed-point map, $f_\gamma(z) = \gamma z + \lambda A(z) + \lambda^{1+H} q$, retrain at each γ on CIFAR-100, and read the seven-solver spread, where $\gamma = 0$ is FROST and $\gamma = 1$ the ungated residual identity.

γ	0.00	0.25	0.50	0.75	0.90	0.99	1.00	1.50
Spread	0.12	0.18	0.24	0.24	0.32	1.78	49.0	51.5
Top-1	53.0	53.1	53.2	53.2	53.2	52.8	42.0	29.5

Table 4: Identity-gain sweep on the main FROST cell, CIFAR-100, varying only γ. Spread is the seven-solver best minus worst and Top-1 is the mean across solvers. Grid points use ten seeds and near-unit points use three.

Solver spread stays near a tenth of a point at $\gamma = 0$ and rises smoothly as γ approaches one, reaching 0.28, 0.57, and 2.39 at $\gamma = 0.95$, 0.97, and 0.995, in step with the $1/(1-\gamma)$ scale of the fixed point $z = g(z)/(1-\gamma)$, while Top-1 holds near 53. At $\gamma = 1$ the coefficient of z vanishes and the spread jumps to about 49, with the loss on Broyden and Newton-Krylov, the procedures that solve the root exactly. At $\gamma = 1.5$ the map is expansive and the spread stays near 51 with accuracy at 29.5, so the failure is the non-contractive regime $\gamma \geq 1$, of which the unit gain is the standard residual bypass. Only γ varies, so the solver sensitivity is attributable to the identity path.

Promoting path independence through training is a separate route to invariance: following Anil et al. (2022), training the baseline with noise-initialized fixed points and a stochastic forward budget reaches spread 0.07 but 51.6 Top-1, below the 53.0 of FROST here, so like the constrained baselines of §5.5 it trades accuracy for invariance. A separate-backbone ablation (Appendix C.1) agrees.

6 Discussion

Beyond vision. The iteration carries no vision-specific structure, so we test it away from vision on a two dimensional Poisson problem cast as a DEQ fixed point, with the source field as input and the solution as the readout. Both models use eight-step unroll-only training in this control. Under that inference procedure, median relative error is 0.128 for FROST and 0.343 for MDEQ. Under the solver swap the median cross-solver spread is 0.45 for FROST against 10.97 for MDEQ, and FROST is smaller in all ten paired seeds. This supports lower solver sensitivity on the tested Poisson distribution; it is not an implicit-training replication or a general PDE guarantee. Appendix C specifies the error metric and budgets.

Limitations. Solver coverage and repeated training runs are limited. Experiments use convolutional models, and transformer generalization remains untested.

7 Conclusion

We showed that fixed-point modeling does not by itself ensure prediction robustness to finite-budget inference procedures. FROST removes the direct identity path, scales the recurrent path with one learned scalar, and uses a rescaling-based input prefactor. It reduces accuracy spread under identical implicit training across the evaluated solvers. Conditional theory, empirical diagnostics, parameterization controls, and a Poisson experiment support this result.

References

Donald G. Anderson. Iterative procedures for nonlinear integral equations. *J. ACM*, 12(4):547–560, October 1965. ISSN 0004-5411. doi: 10.1145/321296.321305.

Cem Anil, Ashwini Pokle, Kaiqu Liang, Johannes Treutlein, Yuhuai Wu, Shaojie Bai, J. Zico Kolter, and Roger B Grosse. Path independent equilibrium models can better exploit test-time computation. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in*

Neural Information Processing Systems, volume 35, pp. 7796–7809. Curran Associates, Inc., 2022. doi: 10.52202/068431-0566.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019a.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. Trellis networks for sequence modeling. In *International Conference on Learning Representations*, 2019b.

Shaojie Bai, Vladlen Koltun, and J. Zico Kolter. Multiscale deep equilibrium models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5238–5250. Curran Associates, Inc., 2020.

Shaojie Bai, Vladlen Koltun, and Zico Kolter. Stabilizing equilibrium models by jacobian regularization. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 554–565. PMLR, 18–24 Jul 2021.

Shaojie Bai, Vladlen Koltun, and J Zico Kolter. Neural deep equilibrium solvers. In *International Conference on Learning Representations*, 2022.

C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19:577–593, 1965.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Laurent El Ghaoui, Fangda Gu, Bertrand Travacca, Armin Askari, and Alicia Tsai. Implicit deep learning. *SIAM Journal on Mathematics of Data Science*, 3(3):930–958, 2021. doi: 10.1137/20M1358517.

Kenneth Falconer. *Fractal geometry: mathematical foundations and applications*. John Wiley & Sons, 2013.

Mateusz Gabor, Tomasz Piotrowski, and Renato L. G. Cavalcante. Positive concave deep equilibrium models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 14365–14381. PMLR, 21–27 Jul 2024.

Zhengyang Geng and J Zico Kolter. Torchdeq: A library for deep equilibrium models. *arXiv preprint arXiv:2310.18605*, 2023.

Davis Gilton, Gregory Ongie, and Rebecca Willett. Deep equilibrium architectures for inverse problems in imaging. *IEEE Transactions on Computational Imaging*, 7:1123–1133, 2021. doi: 10.1109/TCI.2021.3118944.

Stephen Gould, Richard Hartley, and Dylan Campbell. Deep declarative networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):3988–4004, 2021. doi: 10.1109/TPAMI.2021.3059462.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024.

Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Benjamin Halpern. Fixed points of nonexpanding maps. *Bulletin of the American Mathematical Society*, 73(6):957–961, 1967.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8340–8349, October 2021.

John Hutchinson. Fractals and self-similarity. *Indiana Univ. Math. J.*, 30:713–747, 1981. ISSN 0022-2518.

D.A. Knoll and D.E. Keyes. Jacobian-free newton-krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193(2):357–397, 2004. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2003.08.010.

Steven George Krantz and Harold R Parks. *The implicit function theorem: history, theory, and applications*. Number 11 in 202. Springer, 2002.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. *Technical Report*, 2009.

Zenan Ling, Longbo Li, Zhanbo Feng, Yixuan Zhang, Feng Zhou, Robert C Qiu, and Zhenyu Liao. Deep equilibrium models are almost equivalent to not-so-deep explicit models for high-dimensional Gaussian mixtures. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 30585–30609. PMLR, 21–27 Jul 2024.

Benoit B Mandelbrot. The fractal geometry of nature/revised and enlarged edition. *New York*, 1983.

Benoit B. Mandelbrot and John W. Van Ness. Fractional brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, 1968. doi: 10.1137/1010093.

W Robert Mann. Mean value methods in iteration. *Proceedings of the American Mathematical Society*, 4(3):506–510, 1953.

Sam McCallum, Kamran Arora, and James Foster. Reversible deep equilibrium models, 2025.

Katharina Ott, Prateek Katiyar, Philipp Hennig, and Michael Tiemann. Resnet after all: Neural {ode}s and their numerical solution. In *International Conference on Learning Representations*, 2021.

Fernando Pineda. Generalization of back propagation to recurrent and higher order neural networks. In D. Anderson (ed.), *Neural Information Processing Systems*, volume 0. American Institute of Physics, 1987.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 09–15 Jun 2019.

Jimmy T.H. Smith, Andrew Warrington, and Scott Linderman. Simplified state space layers for sequence modeling. In *The Eleventh International Conference on Learning Representations*, 2023.

Homer F. Walker and Peng Ni. Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis*, 49(4):1715–1735, 2011. doi: 10.1137/10078356X.

Ezra Winston and J. Zico Kolter. Monotone operator equilibrium networks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 10718–10728. Curran Associates, Inc., 2020.

TOWARD SOLVER-INVARIANT DEEP EQUILIBRIUM MODELS

APPENDIX

A FRACTIONAL BROWNIAN MOTION AND THE FULL DERIVATION

Fractional Brownian motion. Fractional Brownian motion (fBm) with Hurst exponent H obeys the self-similarity relation (Mandelbrot & Van Ness, 1968)

$$B_H(at) \overset{d}{=} a^H B_H(t), \qquad a > 0. \tag{13}$$

The derivation below uses equation 13 only as a deterministic scaling ansatz, stretching time by a while shrinking amplitude by a^{-H}, and draws on no other property of the process.

Rescaling equivariance for the state dynamics. The derivation of §3.3 transplants this joint rescaling onto the dynamics equation 4. Demand that the rescaled pair $x_\lambda(t) = \lambda^{-H} x(\lambda t)$ and $h_\lambda(t) = h(\lambda t)$ of equation 6 obey the same dynamical equations as (x, h). Differentiating h_λ and applying the chain rule,

$$\frac{d}{dt} h_\lambda(t) = \lambda \dot{h}(\lambda t) \tag{14}$$

$$= \lambda A\big(h(\lambda t)\big) + \lambda B\big(x(\lambda t)\big) \tag{15}$$

$$= \lambda A\big(h_\lambda(t)\big) + \lambda B\big(\lambda^H x_\lambda(t)\big), \tag{16}$$

where the last line substitutes $x(\lambda t) = \lambda^H x_\lambda(t)$ from the definition of x_λ. The right hand side is the original dynamics applied to the scaled operators

$$A_\lambda(h) = \lambda A(h), \qquad B_\lambda(x) = \lambda B(\lambda^H x), \tag{17}$$

and for linear B the input operator reduces to $B_\lambda = \lambda^{1+H} B$. With linear B, the adopted rescaling construction determines the prefactor λ^{1+H}. This is covariance under transformed operators, not invariance of a trained map or an assertion that the neural trajectory is fBm. Removing the state passthrough is a separate architectural choice, not a consequence of fBm symmetry. Absorbing B into the stem features then gives the bare update equation 7.

The working constant. At initialization, $c = \lambda_0^{1+H}$. Once λ_0 and H are chosen, no separate search over c is needed to instantiate this rule. This does not make those design choices parameter-free or optimal. The ImageNet constant-scale control retains low solver spread, showing that dynamic coupling to λ is not necessary in that experiment.

B PROOFS

Propositions 1 and 2 are proved in the main text.

Proof of Proposition 3. For linear A the update is $z_{t+1} = \lambda A z_t + \lambda^{1+H} x$, so $\Delta_{t+1} = \lambda A(z_{t+1} - z_t) = \lambda A \Delta_t$ and $\Delta_t = (\lambda A)^t \Delta_0$ with $\Delta_0 = \lambda^{1+H} x$ from $z_0 = 0$. Let μ_1 be the simple real eigenvalue strictly dominant in modulus, with $|\mu_1| = \rho(A) > 0$, right eigenvector v_1, and left eigenvector w_1 normalized by $w_1^\top v_1 = 1$. Suppose $w_1^\top \Delta_0 \neq 0$. The spectral projection onto this mode gives

$$(\lambda A)^t \Delta_0 = (\lambda \mu_1)^t \big((w_1^\top \Delta_0)\, v_1 + o(1)\big), \tag{18}$$

since every other mode decays by a strictly smaller factor relative to $(\lambda \rho(A))^t$. Taking norms gives $\|\Delta_t\| = (\lambda \rho(A))^t (c + o(1)) \|\Delta_0\|$ with $c = |w_1^\top \Delta_0| \, \|v_1\| / \|\Delta_0\|$. Subdominant Jordan terms are absorbed by $o(1)$. A nontrivial dominant Jordan block or several equally dominant eigenvalues falls outside the proposition: polynomial factors or oscillations may prevent its stated prefactor from converging. Non-normality can produce transient growth even under the stated assumptions. □

Proof of Proposition 4. The increment obeys $\Delta_{t+1} = \lambda(A(z_{t+1}) - A(z_t))$. Along the segment from z_t to $z_{t+1} = z_t + \Delta_t$,

$$A(z_{t+1}) - A(z_t) = \left(\int_0^1 J(z_t + s\Delta_t)\, ds \right) \Delta_t = J^\star \Delta_t + E_t \Delta_t, \tag{19}$$

where $J^\star = \partial A/\partial z|_{z^\star}$ and $E_t = \int_0^1 (J(z_t + s\Delta_t) - J^\star)\, ds$. The bounded Hessian makes J Lipschitz with a constant M, so $\|E_t\| \le M(\|z_t - z^\star\| + \|\Delta_t\|)$. Therefore

$$\varepsilon_t = \frac{\|\Delta_{t+1} - \lambda J^\star \Delta_t\|}{\|\Delta_t\|} = \frac{\|\lambda E_t \Delta_t\|}{\|\Delta_t\|} \le \lambda M \big(\|z_t - z^\star\| + \|\Delta_t\| \big). \tag{20}$$

Near a locally attracting fixed point with convergence rate $\kappa < 1$, both $\|z_t - z^\star\|$ and $\|\Delta_t\|$ are $O(\kappa^t)$, so $\varepsilon_t = O(\kappa^t)$. □

Non-smooth operators. For a ReLU network, the smooth proof applies if the iterates and connecting segments eventually remain in a smooth activation region with bounded second derivatives. Lipschitz continuity of the network alone does not make its Jacobian Lipschitz across activation boundaries. We therefore do not extend the fixed-Jacobian rate to trajectories that continue crossing such boundaries.

C Implementation and Experimental Details

In the main multiscale implementation, $\lambda = \exp(\min(\log \hat{\lambda}, 0)) \in (0, 1]$. Defaults are $H = 0.8$ and $\lambda_{\text{init}} = 0.5$, except in the stated sensitivity controls. The separate CIFAR parameterization controls below use a positivity-only scalar and can exceed one.

Architecture. The backbone is the multiscale MDEQ-SMALL configuration, a four branch multiresolution cell with channels $[32, 64, 128, 256]$, expansion factor five, GroupNorm, weight normalization, and a bottleneck classification head, about 25.5M parameters. FROST replaces the cell with its single scalar variant, adding one parameter $\log \hat{\lambda}$, and pcDEQ clamps the convolution weights to be nonnegative under a concave activation. The main CIFAR comparison uses MDEQ-TINY, a two-branch cell with channels $[24, 24]$, about 288K parameters. It is distinct from the separate controls below.

Normalized cell and readout. The main FROST convolutional branch computes $\mathrm{GN}_3(\mathrm{ReLU}(\lambda A(z) + bx))$, where A includes two convolutions, internal activation, and normalization, and $b = \lambda^{1+H}$ (or the fixed constant in its control). The MDEQ branch injects x before its second normalization and adds z afterward, before the final activation and normalization. Branch fusion is retained. Consequently, the bare recurrence in the derivation is not the complete implemented cell, and the scalar cannot be factored out of the full map through normalization. The main classification head applies λ^{-H} to pooled features before the classifier.

Separate CIFAR parameterization controls.

C.1 Component ablation on a second backbone

A separate parameterization ablation on the FROSTResNet backbone varies the identity bypass jointly with the input scale, complementing the single-edge sweep of §5.6.

Variant	No bypass	Input	Unroll	Picard	Halpern	Anderson	Broyden	Inertial	Newton	Δ_7	Δ_5
Residual, $H = 0$		λ	10.02	53.78	56.82	3.43	5.07	29.79	13.53	53.39	53.39
No identity	✓	λ	41.94	51.32	51.40	51.31	51.33	46.54	51.30	9.46	0.10
Residual, $H = .8$		λ^{1+H}	13.10	54.98	58.28	4.29	4.65	43.66	5.36	53.99	53.99
FROST	✓	λ^{1+H}	38.15	56.75	57.01	56.72	56.73	44.52	56.65	18.86	0.36

Table 5: CIFAR-100 parameterization controls on a separate backbone. Δ_7 includes all seven procedures and Δ_5 excludes unroll and inertial iteration. The subset is not a convergence-certified group.

We compare $H = 0$ and $H = 0.8$ with residual and bypass-free maps (Table 5). Bypass-free variants reduce full-set spread from approximately 53 to 54 points down to 9.46 to 18.86 points. Their spread is below one point over Picard, Halpern, Anderson, Broyden, and Newton, but unroll and inertial accuracy remain lower. With no bypass, $H = 0.8$ raises peak accuracy from 51.40 to 57.01, while worsening five-step accuracy and full-set spread. Its benefit therefore depends on the inference procedure.

Table 5 uses the FROSTResNet implementation with 16-step unrolled training, not the main MDEQ-TINY cell, so its full-set spreads are not comparable to the ImageNet unroll-only control of §5.2. Write $q = bB(x)$ and $u(z, x) = \mathrm{sigmoid}(G_x(x) + G_h(z))$. Its bypass-free update is $f(z, x) = \lambda A(z) + q$; its residual alternative is $f(z, x) = z + u(z, x)\mathrm{ReLU}(\lambda A(z) + q)$. The four variants use $b = \lambda$ or λ^{1+H}. All evaluation procedures start from $z_0 = q$ within a checkpoint, use the same readout $\mathrm{head}(\lambda^{-H}C(z) + D(x))$, and have nominal budget 26 except for five-step unroll. Inertial iteration is not shorter than the training depth. The scalar is $\lambda = \exp(\log \hat{\lambda}) > 0$, without an upper clamp. The CIFAR constant-scale sweep uses this implementation's gated residual update; it is not the bypass-free ImageNet constant-scale experiment.

Training. On ImageNet-1K at 224×224 we use SGD with Nesterov momentum 0.9, a learning rate of 0.2 scaled linearly from a base 0.05, weight decay 5×10^{-5}, a cosine schedule to zero, mixed precision, and an effective batch of 448 over 80 epochs. On CIFAR-100 at 32×32 we use Adam with learning rate 10^{-3}, weight decay 2.5×10^{-6}, a cosine schedule, and batch 64 over 30 epochs. The FROST scalar is trained with a learning rate scaled by 0.001 relative to the base rate.

Training regimes. Table 2 contains two separately trained FROST checkpoints. The row labeled "Unroll" uses five explicitly unrolled steps throughout training. The row labeled "Broyden" uses the native MDEQ curriculum: an unrolled warmup followed by fixed-point solution with Broyden and implicit differentiation. The MDEQ and Jacobian-regularized baselines use the latter curriculum. Thus, comparisons within the Broyden-trained rows hold the training regime fixed, while the two FROST rows show that the parameterization supports both regimes. Algorithm 1 states both alternatives.

Solvers. The main protocol uses $z_0 = 0$ and a nominal budget of 26, apart from five-step unroll. For Broyden this is an update threshold, with an initial residual evaluation in addition; the other procedures use their implementation-specific iteration or evaluation counters. It is not an exact equal-FLOP protocol. Damped Picard uses $\tau = 0.5$, Halpern anchors at z_0, Anderson uses history $m = 5$, and Broyden uses the standard low rank quasi-Newton update. The inertial iteration adds momentum $y_k = z_k + \beta(z_k - z_{k-1})$ with $\beta = 0.05$. Newton-Krylov starts with six Picard steps, builds a Krylov subspace using forward-mode Jacobian-vector products, solves the local Newton system by GMRES with subspace size four, and backtracks for residual decrease. Each function evaluation and Jacobian-vector product counts as one against its 26-unit budget. It uses Jacobian information for root finding, not an exact Hessian or a second-order optimization method.

State and margin diagnostics. The state and margin diagnostics use the 100-epoch checkpoints. The state diagnostic evaluates 32 validation images and three pairs among Broyden, Anderson, and damped Picard. At segment fractions $0, 1/4, 1/2, 3/4, 1$, a matrix-free Lanczos calculation estimates the smallest singular value of $I - J_f$; the reported $\hat{m}$ is the minimum of these estimates. Neither finite Lanczos estimates nor the minimum of pointwise values certify $\sigma_{\min}(\int (I - J_f)\, ds)$. In particular, pointwise nonsingularity does not prevent cancellation in the matrix average. The separate 128-image endpoint diagnostic computes $\|z_a - z_b\|_2$ over the same three converging-solver pairs (Broyden, Anderson, damped Picard) and normalizes it by the mean norm of all five returned states (unroll, damped Picard, Halpern, Anderson, Broyden) for that image, then averages over images. The margin diagnostic uses a 200-step direct-unroll reference and 26-step endpoints from unroll, damped Picard, Halpern, Anderson, and Broyden. This diagnostic's unroll depth differs from the five-step main-table operating point. At the reference, eight random perturbations at each relative scale $10^{-3}, 10^{-2}, 10^{-1}$ yield 24 finite-difference head slopes; their maximum is $\hat{L}_h$. The reference need not be an exact root or a correct prediction, and the sampled radius is not certified.

Poisson experiment. The task maps smooth random source fields to solutions of the discrete Poisson equation on a 64×64 grid with homogeneous Dirichlet boundaries. Targets use a discrete

sine-transform solve. Each paired seed uses 4096 training and 256 test fields, 32 hidden channels, batch 64, and 3000 training steps with learning rate 0.002. The reported ten-seed comparison (seeds 42–51) trains both models by eight-step unrolling throughout. Evaluation uses unroll at depths 8 and 64, and damped Picard, Anderson, and Halpern at 32 iterations. For each procedure, error is the test mean of $\|\hat{u} - u\|_2 / \|u\|_2$; spread is the maximum minus minimum of these five errors, and the discussion reports medians over seeds. These units are relative error, not classification percentage points.

Algorithm 1 The two FROST training regimes and shared inference protocol

Require: implemented cell f_θ with scalar λ and exponent H, regime R, warmup steps N, unroll depth T

for training step $t = 1, 2, \ldots$ **do**
  $x \leftarrow \mathrm{stem}(\text{input})$
  if R is unroll-only or (R is implicit and $t < N$) **then**
    initialize $z \leftarrow 0$ and apply $z \leftarrow f_\theta(z, x)$ for T steps
    backpropagate the task loss through the T unrolled steps
  else
    approximately solve $z^\star = f_\theta(z^\star, x)$ with Broyden
    backpropagate the task loss by implicit differentiation at $z^\star$
  end if
end for
at inference, apply S to the same cell from $z_0 = 0$ and return the scaled classification readout at z_S

Exponent sensitivity. The exponent is fixed at 0.8. On a grid over $H \in \{0.2, 0.4, 0.6, 0.8, 1.0\}$ and the initial scalar $\lambda_{\text{init}} \in \{0.3, 0.5, 0.7, 0.9\}$, the response is non-monotone in both axes and the strongest configuration falls at $H = 0.8$, while making H learnable gives no improvement over the fixed value. We therefore treat $H = 0.8$ as a fixed design choice supported by this sweep, not as an estimated property of ImageNet.

D Constant Input Scale

Section 5.4 shows that fixing the input prefactor at the derived constant reproduces solver invariance. Here we locate the working range of the input constant. The derivation fixes the prefactor at $c = \lambda_{\text{init}}^{1+H} = 0.5^{1.8} \approx 0.287$ for the initial scalar, and the ImageNet variant holds it at 0.2871, otherwise identical to FROST and trained for 80 epochs at an effective batch of 448.

Which constants train. A sweep over the input constant on CIFAR shows a broad working range around the derived value. A constant of 0.05 collapses to 5 to 6 percent over three seeds, while 0.287 reaches 55.64 over ten seeds and 0.6 and 1.0 reach 54 to 57 percent over three seeds each. These results concern the separate gated residual CIFAR implementation described in Appendix C, whose scalar is positive but not bounded by one. They identify a tested working range, not a universal lower threshold or a unique optimum at the derived value. The default H and initial scalar determine that value before the constant sweep.

E Out-of-Distribution Generalization

We re-run the depth sweep on two distribution-shifted validation sets with no retraining (Tables 6 and 7). ImageNet-R (Hendrycks et al., 2021) contains stylistic renditions of a 200 class subset, evaluated with the logits masked to those classes, and ImageNet-V2 (Recht et al., 2019) is a newly collected natural set aligned to the original classes.

Two properties carry over. FROST's solver-swap loss stays small under the shift, at most about 5 points between the five-step unroll and Broyden on either set, whereas MDEQ loses 16 points on ImageNet-R and 29 on ImageNet-V2. Thus the in-distribution solver-robustness pattern persists on both evaluated shifts; we do not claim invariance under arbitrary distribution shift. Comparing unroll-trained FROST at five steps with Broyden-trained MDEQ under Broyden gives gains of 7.48 points on both shifted sets. This compares different training regimes. Under identical Broyden

Model	Train	T=2	T=5	T=10	T=20	T=50	T=100	Broyden
FROST	Unroll	11.53	**30.38**	27.18	26.27	24.79	24.05	26.50
FROST	Broyden	6.57	23.66	24.83	24.64	24.58	24.35	**26.91**
MDEQ	Broyden	4.47	7.01	9.22	8.73	8.44	8.38	**22.90**

Table 6: Accuracy at each forward depth on ImageNet-R, masked to the 200 class subset.

Model	Train	T=2	T=5	T=10	T=20	T=50	T=100	Broyden
FROST	Unroll	21.24	**50.64**	46.57	45.25	43.28	42.32	45.23
FROST	Broyden	9.41	42.48	43.72	43.77	43.84	44.05	**47.51**
MDEQ	Broyden	5.40	14.30	17.16	16.77	16.85	16.81	**43.16**

Table 7: Accuracy at each forward depth on ImageNet-V2.

training and inference, the gains are 4.01 and 4.35 points, respectively. The depth profile carries over as well, since the unroll-trained FROST peaks at five steps on both sets and the Broyden-trained one plateaus from ten steps onward, while the baseline stays under 10 percent on ImageNet-R and under 18 on ImageNet-V2 at every unroll depth and recovers only under Broyden.

F Increment Dynamics

We analyze increments $\Delta_t = z_{t+1} - z_t$ of the bare recurrence equation 7. For linear A and $z_0 = 0$, $\Delta_0 = \lambda^{1+H} x$. In a nonlinear map, H may also affect the equilibrium and its local Jacobian through the input scale.

Proposition 3 (Asymptotic increment decay)**.** *For linear A the increments satisfy $\Delta_{t+1} = \lambda A \Delta_t$ exactly. If A has a simple real eigenvalue strictly dominant in modulus, Δ_0 has nonzero projection onto its mode, and $0 < \lambda\rho(A) < 1$, then*

$$\|\Delta_t\| \;=\; (\lambda\rho(A))^t \,(c + o(1))\, \|\Delta_0\|, \tag{21}$$

where c depends on the projection of Δ_0 onto the leading eigenspace of A.

For non-normal A, transient growth may precede decay; equation 21 is not a per-step bound.

For nonlinear $A \in C^2$ with bounded Hessian, one can instead bound the local linearization error along a convergent trajectory. Write $J^\star$ for the Jacobian of A at $z^\star$.

Proposition 4 (Asymptotic linearization)**.** *For $A \in C^2$ with bounded Hessian near $z^\star$, suppose $\|z_t - z^\star\| = O(\kappa^t)$ with $0 < \kappa < 1$. For nonzero increments,*

$$\varepsilon_t \;=\; \frac{\|\Delta_{t+1} - \lambda J^\star \Delta_t\|}{\|\Delta_t\|} \;=\; O\big(\kappa^t\big)\,. \tag{22}$$

Here κ is an assumed geometric convergence rate, not necessarily the spectral radius itself. Appendix B gives the proof and the additional smooth-region qualification for ReLU networks. These local results motivate diagnostics; they do not prove convergence of the normalized implementation.

Over 26 Broyden updates on ImageNet, the FROST residual decreases by a factor of 179 without a backstep, compared with a factor of 6.26 and two backsteps for MDEQ. This Broyden measurement is a complementary numerical observation, not a direct test of Propositions 3 and 4, which describe the FROST iteration itself. The per-step dynamics show the same difference at finer granularity (Table 8).

Quantity	FROST	MDEQ
Residual decay over 26 steps	**179**×	6.26×
Non-monotone backsteps	0/26	2/26

Table 8: Step-level Broyden dynamics on ImageNet-1K over the trajectory pool.

FROST's residual decays by 179 times over 26 steps and does so monotonically, while the baseline decays by only 6.26 times with two backsteps along the way. Because Broyden includes quasi-Newton corrections, this table does not instantiate the direct recurrence in Proposition 3. It instead shows that the favorable residual behavior is also visible under the training solver.

The scaling is temporal. The rescaling of §3.3 acts on the iteration index rather than on image space. An exploratory spatial-spectrum diagnostic yields slopes of 0.63, 0.98, and 1.01. The temporal rescaling rule does not prescribe a two-dimensional spatial spectral slope; in particular, it does not imply $2H + 1 = 2.6$ for these images. The spatial measurements therefore neither validate nor refute the temporal construction.

G Anytime Prediction

Reading the classifier head at every forward depth shows how usable each truncation of the iteration is (Table 9).

Model	Training	T=2	T=5	T=10	T=20	T=50	T=100	Broyden
FROST	Unroll	29.77	**63.15**	58.18	56.66	53.83	52.68	56.87
FROST	Broyden	12.50	55.37	56.71	56.60	56.27	56.39	**60.10**
MDEQ	Broyden	7.88	20.57	24.17	23.11	22.89	23.05	**56.29**

Table 9: Anytime prediction on ImageNet-1K across forward depths, with the training-time solver in bold.

The unroll-trained FROST peaks at five steps and loses 10.47 points by 100 steps; its accuracy is not monotone in depth. The Broyden-trained FROST varies from 55.37 to 56.71 over the evaluated unroll depths 5–100, below its 60.10 Broyden accuracy. Both perform poorly at two steps. The baseline remains below 25 percent at all evaluated unroll depths, versus 56.29 under Broyden. The unroll-trained checkpoint is thus a truncated-budget operating point rather than a converged equilibrium, and its non-monotone depth profile is the finite-budget behavior this paper studies, exhibited within a single model. The equilibrium result is the Broyden-trained FROST, whose accuracy is stable across depth and whose solver spread is the smallest in Table 2. The five-step peak therefore belongs specifically to the unroll-trained checkpoint, not to every FROST model or arbitrary truncation depth.

H Released Checkpoint

The MDEQ comparison in Section 5.2 uses a checkpoint trained under our own budget, which reaches a lower training-solver accuracy than the released MDEQ-SMALL model. To separate the solver spread from that budget, we apply the protocol of Section 5.1 unchanged to the publicly released checkpoint, with no retraining, on the full validation set.

Unroll	Picard	Halpern	Anderson	Broyden	Inertial	Newton	Spread
59.38	75.44	66.73	75.43	75.42	61.55	75.38	16.05

Table 10: The released MDEQ-SMALL checkpoint read under the seven inference procedures, without retraining.

The four converging procedures agree within 0.06 points at 75.4, while the five-step unroll, the inertial iteration, and Halpern lose 16.05, 13.89, and 8.71 points. The spread is therefore present in a checkpoint that reaches the accuracy reported for this architecture, and the pattern matches Table 2: the converging procedures agree and the truncated and anchored ones fall away. Continuing to train the released checkpoint for five further epochs leaves the spread between 14 and 20 points, so additional training does not remove it. Its severity differs from the 36.84 of Table 2, so training quality modulates how large the spread is. This is why the architecture comparison of Section 5.2 holds the training budget fixed for both models rather than comparing across budgets.

I Use of large language models

We used a large language model as a coding assistant and for language polishing. On the code side it helped implement and refactor the measurement and figure code, and the authors read, ran, and verified that code themselves. On the writing side it assisted with language editing, literature search, and checks of consistency among claims, proofs, and experimental descriptions. Reported empirical measurements originate from experiment outputs, not language-model generation. The authors are

responsible for verifying the final claims, citations, numbers, and implementation descriptions and for the content of the manuscript.

J ETHICS STATEMENT

This work studies the numerical behavior of deep equilibrium models using existing ImageNet-1K and CIFAR benchmarks and a synthetic Poisson problem. We collect no new human-subject data; use of existing image datasets remains subject to their access and licensing conditions. The contribution is a change to how a fixed point is parameterized and read out, and we see no application of it that creates a capability for harm beyond those already present in image classification.

K REPRODUCIBILITY STATEMENT

The FROST iteration is defined in Section 3.3 and the two conditions in Section 4, with full proofs in Appendix B. The architecture, the training recipe, the solver settings, and the training procedure are given in Appendix C, the measurement details for the constant scale and the out-of-distribution generalization are in Appendices D and E, and the CIFAR comparison is in Section 5.5. We release the code, the trained checkpoints, and the table of numbers behind the paper, so that every result can be recomputed.