IDENTITY-EXACT MULTI-BIAS ACTIVATIONS: BREAKING OPTIMIZATION SYMMETRY WITHOUT CHANGING THE PRETRAINED FUNCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

A nominally three-branch activation can collapse to only two learned knot locations. In the $K = 3$ multi-bias parameterization studied here, identity initialization ties its two coefficient-$+1$ branches. Because these branches evaluate the same feature and receive equal gradients, AdamW's normalized update is identical for the two branches up to its ϵ term, preserving the tie throughout optimization. This optimizer-induced branch collapse helps explain why additional branches can match or underperform a same-site $K = 1$ control despite the model class's greater expressivity. A multi-bias unit replaces $\sigma(z + b)$ with $\sum_{k=1}^{K} a_k \sigma(z + b_k)$ along a shared preactivation. With learned real coefficients, a ReLU multi-bias unit is a one-dimensional spline, represents XOR with $K = 2$, and, after thresholding, is Boolean complete on finite hypercubes; an exhaustive exact-arithmetic certificate further shows that some five-input Boolean function requires at least eight knots, $K^*(5) \geq 8$. The deployed fixed-sign parameterization occupies a degenerate corner of this family: alternating ± 1 coefficients restrict the available slope increments, while tied biases create the optimizer symmetry. Separating the knots breaks this symmetry but, in a controlled audit, perturbs the pretrained activation by up to 53% in relative L^2. We resolve this conflict with identity-exact nesting, in which one branch starts with coefficient one and additional branches start with coefficient zero at distinct knots. The resulting network exactly preserves the pretrained function at initialization, while the added branches receive distinct coefficient gradients on the first optimization step. Across four frozen ResNet-50 transfer tasks, nested $K = 3$ significantly improves over its matched-site $K = 1$ control by 0.55–1.63 accuracy points over three seeds and achieves higher training accuracy in every case. On CIFAR-100, a matched fixed-sign replication is significantly worse than that control (-1.32 points), while nested is 0.78 points better: parameterization decides the effect's sign. Nested is never significantly worse than this control in any tested cell: ViT-B/16 is a four-seed statistical tie, and a 30-epoch sweep peaks at $K = 1$. These results show that expressivity and optimizer reachability can diverge even within a single activation family.

1 INTRODUCTION

The expressive power of a model class can be irrelevant if the parameterization and optimizer cannot reach it. This distinction is especially sharp for activation-level adaptation. A multi-bias neuron replaces one activation $\sigma(z + b)$ by several shifted copies along the same scalar preactivation z. With freely learned real coefficients, this small change makes one ReLU neuron a one-dimensional spline and even Boolean complete on finite hypercubes. Nevertheless, our earlier practical recipe often found that a matched-site $K = 1$ control equalled or beat $K > 1$.

We show that this is not a contradiction but a property of the parameterization. The theory uses real branch coefficients, whereas the deployed model fixed them to alternating ± 1. This freezes every slope increment. Moreover, identity initialization ties equal-sign branches at the same bias. Their gradients are parallel, and AdamW's coordinate-wise normalization preserves their equality. Thus increasing K adds nominal knot parameters while the trained unit can remain on a lower-dimensional, lower-knot manifold. Directly separating the biases breaks the lock only by changing the activation

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.

seen by the pretrained backbone: in our controlled audit the relative L^2 change grows from 0 to 0.096, 0.230, and 0.526 as separation and coefficient asymmetry increase.

Our repair separates *symmetry breaking* from *function perturbation*. NESTEDMULTIBIAS uses one coefficient-one identity branch and additional coefficient-zero branches at distinct knots. The model is exactly the pretrained model at initialization, but different branches already have different coefficient gradients. After this repair, nested $K = 3$ beats the matched-site $K = 1$ control on all four ResNet-50 PEFT datasets at two-sided Welch $p < .05$ and is never significantly worse in any tested matched-site transfer cell. On CIFAR-100, by contrast, the tied fixed-sign $K = 3$ arm is significantly worse than $K = 1$. The result is not a claim that larger K always helps: ViT-B/16 and long-budget Pets are ties, and the controlled Pets sweep peaks at $K = 1$. These neutral and negative results expose a second boundary: once optimization makes the extra degrees of freedom accessible, one-dimensional reshaping of a frozen representation still has a measurable ceiling.

Our contributions are:

1. We identify two distinct degeneracies in the commonly deployed fixed-sign formulation: a finite slope palette and an AdamW symmetry lock. We formalize both and verify the optimizer mechanism directly.
2. We introduce an identity-exact nested parameterization whose coefficient-zero branches have distinct first-step gradients. It supports progressive growth in K as an exact no-op.
3. We connect the diagnosis to the expressivity theory. The rank-one two-layer view explains the symmetry, while the spline and XOR results identify precisely what fixed signs give up: arbitrary slope increments and the two-branch XOR certificate. Boolean completeness itself is weaker evidence, since alternating hinge pairs also reach it at a larger K.
4. We evaluate the repair and its limits: it is significantly better than matched-site $K = 1$ in all four ResNet-50 cells, never significantly worse in the tested transfer grid, and tied on ViT and long-budget Pets. A matched fixed-sign arm is significantly worse on CIFAR-100.

2 MULTI-BIAS NEURONS AND IDENTITY NESTING

Definition. For scalar preactivation $z = \langle w, x \rangle$, base activation σ, and channel c, a K-branch multi-bias unit is

$$f_c(z) = \sum_{k=1}^{K} a_{c,k}\, \sigma(z + b_{c,k}). \tag{1}$$

The biases set knot locations and the real coefficients set slope changes or branch amplitudes. The standard unit is $K = 1$, $a_{c,1} = 1$. In a layer with C channels, learning both arrays adds $2KC$ scalars; fixing a leaves only KC trainable. We use "fixed-sign" for $a_{c,k} = (-1)^{k+1}$ and "real-coefficient" when $a_{c,k}$ is learned independently for each channel.

Multi-bias is a rank-one-constrained two-layer block. For one output, Equation (1) is exactly a width-K two-layer network whose first-layer matrix and second-layer vector are

$$W_1 = \begin{bmatrix} w^\top \\ \vdots \\ w^\top \end{bmatrix} = \mathbf{1}_K w^\top, \qquad w_2 = (a_{c,1}, \ldots, a_{c,K})^\top.$$

Thus $\mathrm{rank}(W_1) = 1$: branches share a projection and differ only in bias. This view correctly limits what the unit can represent—it remains a ridge function—and also anticipates the optimization pathology: at tied biases every branch receives the same scalar feature.

Lemma 1 (Identity-exact nesting)**.** *Choose an identity branch k_0 with $b_{c,k_0} = b_c$ and $a_{c,k_0} = 1$. For all $k \neq k_0$, choose arbitrary, mutually distinct biases and set $a_{c,k} = 0$. Then $f_c(z) = \sigma(z + b_c)$ for every z. Consequently, replacing any collection of standard activations by these units leaves the entire network function unchanged, up to exact floating-point multiplication by zero.*

Proof. All added summands are zero, and the remaining summand is the original activation. □

Unlike tied alternating-sign initialization, Lemma 1 does not require extra branches to evaluate the same feature. At initialization,

$$\frac{\partial \mathcal{L}}{\partial a_{c,k}} = \sum_i \delta_{i,c}\, \sigma(z_{i,c} + b_{c,k}), \tag{2}$$

which generally differs across distinct $b_{c,k}$. Bias gradients of unused branches are initially zero because they are multiplied by $a_{c,k}$; once a coefficient grows, its knot can move. We place extra knots symmetrically with gap 0.5, update coefficients at the head learning rate, and update biases at $10\times$ that rate.

Corollary 1 (Progressive K)**.** *At any training iterate, appending a branch at any new bias with coefficient zero increases K without changing the represented function. Its coefficient gradient is given by Equation* (2)*, so optimization can activate the branch without an initialization shock.*

3 Real coefficients expose the full shape family

For ReLU, order the knots $t_k = -b_k$. On every interval between consecutive knots, f is affine, and crossing t_k changes its slope by a_k. Therefore real coefficients are exactly the degrees of freedom that choose an arbitrary piecewise-linear shape. Fixed alternating coefficients retain some finite-domain expressivity when more branches are allowed, but not this full K-knot family or the minimal two-branch XOR construction below.

Proposition 1 (One-dimensional shaping)**.** *A K-branch ReLU multi-bias unit realizes exactly the continuous piecewise-linear functions with at most K knots whose value and slope vanish to the far left. On a compact interval, one extra hinge to the left of that interval and one downstream additive constant lift this to any continuous piecewise-linear function with K knots. Hence multi-bias units approximate arbitrary continuous univariate functions on compact intervals.*

Corollary 2 (XOR at $K = 2$)**.** *For any input pair i, j, a single real-coefficient ReLU multi-bias neuron represents $x_i \oplus x_j$ on $\{-1, +1\}^n$.*

Proof. Set $z = x_i + x_j \in \{-2, 0, 2\}$ and $f(z) = \mathrm{ReLU}(z + 1) - 2\,\mathrm{ReLU}(z)$. Then $f(-2) = 0$, $f(0) = 1$, and $f(2) = -1$; thresholding at $1/2$ selects exactly the two inputs with unequal signs. □

The coefficient vector $(1, -2)$ is important: with two fixed coefficients $(1, -1)$, the response is a monotone clipped ramp and cannot isolate the middle projected value.

Theorem 1 (Finite-domain completeness and a verified lower bound)**.** *Let $K^*(n)$ be the largest, over Boolean targets $T : \{-1, +1\}^n \to \{0, 1\}$, of the minimum branch count of a thresholded real-coefficient ReLU multi-bias unit representing T. Then $K^*(n) \le 2^n - 1$. Moreover, $K^*(5) \ge 8$.*

The upper bound orders the 2^n projected vertices and interpolates their labels with a univariate piecewise-linear function. The lower bound is computer verified by exhaustive enumeration in exact integer arithmetic: the shipped truth table has minimum boundary count eight over all $R(5) = 1{,}981{,}440$ generic signed orderings of the five-cube. Appendix A gives the proof, witness, exhaustive chamber enumeration, and verifier. As a complementary count, along one fixed generic ordering the exact number of indicators with at most K changes is

$$M(K, n) = 2 \sum_{j=0}^{\min(K, 2^n - 1)} \binom{2^n - 1}{j},$$

so $M(5, 5) = 412{,}736$. This is a fixed-ordering count, not a claim that $K = 5$ represents all five-bit functions.

These statements are deliberately separated from deployment. They establish what Equation (1) can express with free real coefficients; they do not apply unchanged to the fixed alternating-sign corner used by the earlier experiments. Alternating hinge pairs can still prove Boolean completeness with a looser branch count; our claim is that arbitrary slope increments and the $K = 2$ XOR certificate require coefficients outside the deployed pattern (Figure 1a).

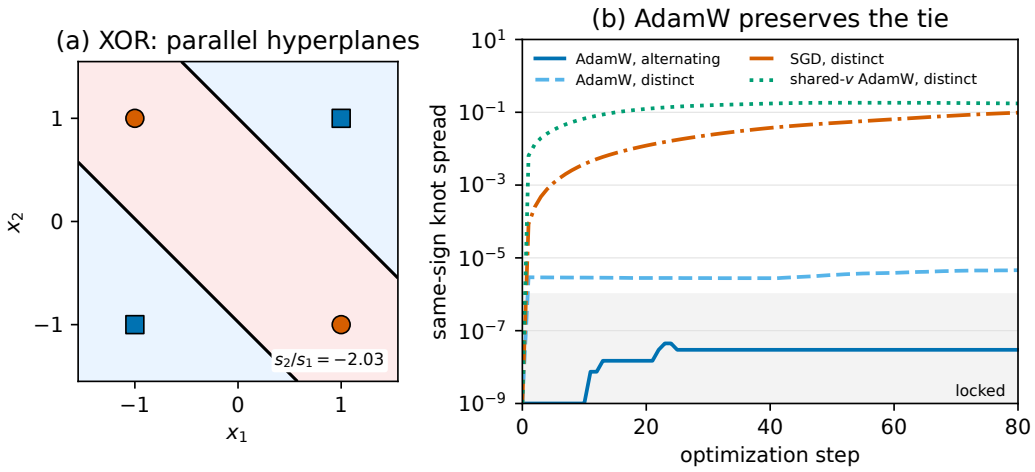

Figure 1: Expressivity versus reachability. (a) A trained real-coefficient $K = 2$ unit solves XOR with parallel decision hyperplanes; its learned ratio $1 : -2.03$ recovers Corollary 2's analytic $1 : -2$, while fixed $(1, -1)$ coefficients cannot isolate the middle projection. (b) In an 80-step TinyNet, coordinate-wise AdamW holds tied alternating and distinct-magnitude branches below 5×10^{-6}; SGD and shared-second- moment AdamW reach $0.097/0.175$ spread (zero is plotted at 10^{-9}). NESTEDMULTIBIAS starts from distinct knots with zero coefficients and has no tie to escape; Table 3 gives its full-network separation.

4 TWO DEGENERACIES OF THE FIXED-SIGN UNIT

4.1 DEGENERACY I: A FROZEN SLOPE PALETTE

Proposition 2 (Fixed coefficients freeze slope increments)**.** *Let π be the knot order of a ReLU multi-bias unit with fixed coefficient multiset $\{a_1, \ldots, a_K\}$. Its consecutive slopes are*

$$0, \quad a_{\pi(1)}, \quad a_{\pi(1)} + a_{\pi(2)}, \quad \ldots, \quad \sum_{k=1}^{K} a_k.$$

Changing the biases can reorder or move the knots but cannot change these increments. Across all knot orders, every slope lies in the finite set of subset sums of the fixed coefficients.

For the deployed $K = 3$ pattern $(1, -1, 1)$, the palette is $\{-1, 0, 1, 2\}$. A trained channel selects one sequence from that palette; it does not learn three independent slope changes. Thus the advertised $2KC$ shape parameters become KC trainable knot positions plus KC frozen coefficients. The Boolean results in Section 3, including the two-hinge XOR certificate $(1, -2)$, rely on precisely those missing real degrees of freedom.

4.2 DEGENERACY II: ADAMW SYMMETRY LOCK

At tied biases, write the loss derivative shared by all branches of a channel as $q_{t,c}$. Then

$$g_{t,c,k} := \frac{\partial \mathcal{L}_t}{\partial b_{c,k}} = a_{c,k} q_{t,c}. \tag{3}$$

The gradients are parallel to the coefficient vector. This observation alone does not imply a lock: SGD retains differences in coefficient magnitude. AdamW (Loshchilov & Hutter, 2019), using Adam's coordinate-wise first and second moments (Kingma & Ba, 2015), changes the conclusion.

Proposition 3 (Adam normalization preserves same-sign ties)**.** *Consider AdamW with zero-initialized moments, as in its standard form. If two tied branches have fixed coefficients of the same sign, Equation* (3) *gives $m_{t,k} = a_k \bar{m}_t$ and $v_{t,k} = a_k^2 \bar{v}_t$. Ignoring the standard ϵ term,*

$$\frac{m_{t,k}}{\sqrt{v_{t,k}}} = \operatorname{sign}(a_k) \frac{\bar{m}_t}{\sqrt{\bar{v}_t}},$$

Table 1: Breaking tied-bias symmetry perturbs the starting function. Relative L^2 is measured against ReLU; nested branches use distinct knots at gap 0.5.

Parameterization	Fixed coefficients	Gap	Relative L^2
Tied	alternating	0	0
Separated	alternating	0.5	0.096
Separated	distinct magnitudes	0.5	0.230
Separated	distinct magnitudes	1.0	0.526
Nested	**learned, extras zero**	0.5	**0**

so same-sign branches receive identical normalized updates regardless of coefficient magnitude. Equal weight decay preserves equality. With $\epsilon > 0$, the equality is approximate and converges as moments grow. For equal coefficients it is exact for any coordinate-wise optimizer.

Consequently, tied alternating $K = 3$ branches split into at most two groups: the two positive branches remain coincident, so the nominal three-knot unit has at most two distinct knots. Making the fixed coefficients $(1, -1.5, 2)$ does not materially help under AdamW because normalization removes the intended magnitude asymmetry. In an 80-step deterministic TinyNet, the same-sign spread is 3.0×10^{-8} for alternating coefficients and 4.6×10^{-6} for distinct magnitudes. SGD retains a 0.097 spread for distinct magnitudes, while an AdamW variant sharing the second moment across the K branches retains 0.175. This isolates the mechanism; Section 6 tests whether optimizer changes alone repair end-task performance.

4.3 The escape–damage dilemma

One can escape Proposition 3 by initializing biases at different locations, but the same biases determine the forward function. Table 1 measures the resulting per-site change from ReLU on a fixed 257-point grid.

The damage compounds over replaced sites in a frozen pretrained network. On Flowers-102, increasing the distinct-sign gap from 0.5 to 1.0 lowers final test accuracy from 78.98% to 58.58%, and that arm never recovers: its final training accuracy reaches only 76.6%, against 99.83% for $K = 1$. Longer training repays most of the damage on Pets, whose final training accuracy reaches 97%, but not on Flowers. Within the fixed-coefficient family there is therefore no separation that both breaks the tie and preserves the initial function.

5 The identity-exact recipe

NestedMultiBias applies Lemma 1 independently per channel. The identity knot is fixed initially at the original bias, extra knots are placed at distinct offsets, and all coefficients are trainable. This makes the theoretical and deployed parameterizations agree: every channel has the real coefficients used by Section 3. More importantly, it changes the first optimization step. Equation (2) exposes different branch features to AdamW before any forward perturbation; only after coefficients leave zero do their bias gradients turn on.

Our default uses a residual gap of 0.5, coefficient learning-rate multiplier 1, bias multiplier 10, no bias weight decay, and ordinary AdamW. These choices are not tuned per dataset. Coefficient multipliers $0.1/1/10$ give $91.88/92.40/91.72$ on Pets, making 1 the robust middle choice; bias weight decay from 0 through 10^{-3} has no measurable effect. Progressive growth follows Corollary 1: a new zero-coefficient branch can be appended when validation improvement stalls, without restarting or changing the current predictor.

6 Experiments

Protocol. We replace every activation in a frozen ImageNet-pretrained ResNet-50 (He et al., 2016) or ViT-B/16 (Dosovitskiy et al., 2021) and train the classifier plus multi-bias parameters for ten ResNet epochs or 15 ViT epochs. The $K = 1$ control uses the same sites, data, optimizer, and seeds;

Table 2: Frozen-backbone PEFT (mean ± population standard deviation). Fixed-sign tied $K = 3$ was replicated on CIFAR-100 only. The last column reports nested minus $K = 1$ and a two-sided Welch p-value. ResNet uses three seeds and 10 epochs; ViT uses four seeds and 15 epochs.

Backbone	Dataset	$K = 1$	Fixed-sign tied $K = 3$	Nested $K = 3$	Nested Δ (p)
ResNet-50	CIFAR-100	81.53±0.18	80.21±0.21	**82.31±0.06**	+0.78 (.0165)
	Food-101	81.41±0.03	–	**81.96±0.07**	+0.55 (.0040)
	Pets-37	91.75±0.08	–	**92.42±0.10**	+0.67 (.0023)
	Flowers-102	84.14±0.22	–	**85.77±0.14**	+1.63 (.0016)
ViT-B/16	CIFAR-100	86.90±0.10	–	87.05±0.13	+0.15 (.182)

Table 3: CIFAR-100 mechanism check at ResNet-50 scale (three-seed means; final-epoch diagnostics). Eff. K is the mean in-range distinct-knot count per channel; Palette is the mean number of realized slope levels, including the far-left zero. The tied arm never develops same-sign separation in any epoch or seed: its 3.00 slope levels arise from only two distinct knots despite having three branches.

Arm	Trainable	Test	Train	Eff. K	Same-sign	Palette
$K = 1$	229,668	81.53±0.18	93.89	0.93	0.000	2.00
Fixed-sign tied $K = 3$	279,204	80.21±0.21	92.19	1.77	0.000	3.00
Nested $K = 3$	353,508	**82.31±0.06**	96.94	2.60	0.080	4.00

it differs only in the added branches and is therefore a placement control, not a parameter-matched model. We report mean and population standard deviation over three seeds for ResNet and four seeds for ViT. The four transfer datasets are CIFAR-100 (Krizhevsky, 2009), Food-101 (Bossard et al., 2014), Oxford-IIIT Pet (Parkhi et al., 2012), and Flowers-102 (Nilsback & Zisserman, 2008). Appendix C specifies preprocessing, hyperparameters, per-seed results, and compute.

6.1 Identity-exact PEFT transfers across datasets

The four ResNet-50 deltas answer the optimization question: once initialization is identity exact, $K > 1$ no longer loses anywhere in this grid, including Flowers-102, where the separated-bias recipe fell 4.8 points below its own $K = 1$ control. The result is not explained by uncontrolled overfitting. Last-epoch train/test accuracy is $93.89/81.53$ versus $96.94/82.31$ on CIFAR-100, $88.39/81.41$ versus $92.23/81.96$ on Food-101, $98.85/91.75$ versus $99.35/92.42$ on Pets, and $99.83/84.14$ versus $100.00/85.77$ on Flowers.

Table 3 connects the toy diagnosis to the full network. The fixed-sign arm has more trainable parameters and occupies 1.77 in-range knots, but its same-sign branches remain exactly coincident; it exposes only three measured slopes, underfits even $K = 1$, and finishes 1.32 points below it. Nested separates the knots, exposes nearly four slopes, and raises both train and test accuracy. The fixed arm is actually *more* often non-monotone than nested (39.2% versus 30.1% of channels), so non-monotonicity alone is not the mechanism.

Train–test gap and regularization headroom. Nested reaches higher final training accuracy in every ResNet cell, by 3.05, 3.84, 0.50, and 0.17 points respectively. On CIFAR-100 and Food-101, the train–test gap widens from 12.36 to 14.63 and from 6.98 to 10.27 points even as test accuracy improves; the extra branches therefore provide usable fitting capacity that the single-bias baseline does not have. The fixed-sign CIFAR arm supplies the opposite diagnostic: its $92.19/80.21$ train/test result underfits even $K = 1$'s $93.89/81.53$, so adding branches and parameters alone does not create the wider nested gap. Reachable coefficient degrees of freedom do.

ViT is the neutral transfer check, and we report it as such. Its $K = 1$ frozen features reach 86.90%, and nested reaches 87.05% over four seeds ($p = .182$); we read the 0.15-point delta as a statistical tie rather than as a smaller win. Its train/test gap is also nearly unchanged ($99.13/86.90$ versus $99.36/87.05$). On Pets and Flowers the nested gap narrows, Flowers nearly saturates training accuracy in both arms, the noisy label advantage survives, and decaying knot biases is ineffective. These are counterweights to the CIFAR/Food pattern. Regularizing learned coefficients or whole branches—rather than knot locations—is therefore a specific untested lever that might convert

accessible fit into generalization, but it could also suppress the useful branches; no benefit is claimed here.

6.2 MATCHED-DEPTH BINARY NETWORKS

We next vary activation shape at fixed architecture and binary-weight count in MNIST MLPs (LeCun et al., 1998; Hubara et al., 2016). These models learn real branch coefficients, the native analogue of nesting, and are used as a mechanism check rather than as an efficiency claim.

Table 4: MNIST test accuracy (%, three seeds) at matched depth and binary weights. Learned-coefficient $K = 2$ wins at every tested depth; $K = 4$ adds no further benefit.

Depth	$K = 1$	$K = 2$	$K = 4$
1	97.02±0.07	**97.78±0.04**	97.64±0.10
2	97.23±0.03	**97.72±0.09**	97.68±0.04
3	97.17±0.02	**97.58±0.09**	97.54±0.06

6.3 PARAMETERIZATION AND OPTIMIZER ABLATIONS

Changing alternating coefficients to distinct fixed magnitudes under AdamW moves Pets by $+0.22$ points and Flowers by -0.11; this is a practical no-op consistent with Proposition 3. Grouping AdamW's second moment produces the predicted toy spread, but does not rescue the damaged end-task recipe: on Pets, separated AdamW/GroupAdamW score $91.22/91.14$, whereas nested AdamW/GroupAdamW score $92.39/92.30$. Flowers gives the same conclusion ($78.91/78.54$ separated and $85.83/85.91$ nested). Nested also needs no explicit anti-coalescence penalty, which reproduces it to within noise ($92.40/85.78$). The tied arm's small Pets and Flowers gains in Table 5 are a benign corner, not a broken lock: coincident same-sign branches still leave a restricted two-knot unit that can help on some tasks, while the matched CIFAR-100 replication in Table 3 shows that the same constraint can reverse the effect.

Two rows qualify the method. A per-layer gated residual over shared signs is the better arm on Flowers (85.91 against 85.77) while being a no-op on Pets ($+0.01$), so nested is the more reliable default rather than the uniformly best parameterization. BiasNorm, a per-site rescaling of the knots by the preactivation scale, is safe only when the identity knot is pinned after the affine map: pinned it costs 0.13 points on Pets against nested, and unpinned it displaces the identity branch and falls 10.5 points below even the $K = 1$ control. Both rows support the paper's thesis rather than the specific unit: what matters is whether a parameterization separates symmetry breaking from function perturbation.

Three further checks constrain alternative explanations. First, bias weight decay from 0 to 10^{-3} is a complete accuracy no-op on both datasets. Second, with 20% label noise nested remains ahead: 92.10 versus 91.74 on Pets and 84.34 versus 82.52 on Flowers. Third, re-monotonizing trained CIFAR-100 channels drops accuracy by 7.35 points, compared with 5.49 for a matched monotone-channel control; non-monotone shapes are used, although only 18% of channels are non-monotone.

7 A MEASURED CEILING

Making degrees of freedom accessible does not make them universally useful. Table 6 trains the same frozen ResNet-50 Pets cell for 30 epochs. Effective K counts occupied distinct knots within the empirical 1–99% preactivation range.

Theorem 1 and this table answer different questions, but their scales are informative: a worst-case five-bit target requires at least eight knots, while the $K = 11$ transfer model actually occupies 9.69 knots and nevertheless loses accuracy. The lower bound demonstrates representational demand, not that comparable knot counts improve a frozen vision representation.

This is saturation, not a hidden lock: utilization stays above 0.90 through $K = 11$, yet accuracy declines. Doubling $K = 16$ training to 60 epochs gives 91.74%, so more optimization does not recover the gap. External comparators locate the cell, with the caveat that they use the ten-epoch

Table 5: The parameterization, not merely $K = 3$, determines performance (test accuracy, three seeds). "Separated" is the old distinct-knot recipe. All non-$K = 1$ rows have the same trainable-parameter count, yet Flowers spans 79.31–85.91 (6.60 points): parameters are necessary but insufficient. The sweep reruns its own $K = 1$ control, which is why Pets reads 91.77 here and 91.75 in Table 2.

Parameterization	Pets	Flowers
$K = 1$	91.77	84.14
Fixed coefficients, tied	91.86	85.12
Fixed coefficients, separated	92.17	79.31
Sorted separated knots	91.67	82.67
Nested, learned coefficients	**92.42**	85.77
Nested, BiasNorm, identity pinned	92.29	85.57
Nested, BiasNorm, identity unpinned	81.28	85.56
Gated residual, shared signs	91.78	**85.91**

Table 6: Increasing K occupies more knots but does not raise the frozen-backbone ceiling on Pets (three seeds except $K = 32$, two seeds).

K	Test accuracy	Effective K	Knot utilization
1	**92.58±0.24**	0.95	0.95
3	92.54±0.16	2.72	0.95
7	92.44±0.15	6.36	0.93
11	92.22±0.16	9.69	0.90
16	91.80±0.19	12.41	0.81
32	91.73±0.09	17.03	0.55

budget of Section 6 rather than the thirty epochs of Table 6: full fine-tuning reaches 92.85% (two seeds), a comparable-parameter LoRA layer-4 cell reaches 92.00% (145k trainable parameters, versus 224k for nested $K = 3$), and a linear probe reaches 89.42%. The remaining limitation is the frozen representation: a per-channel ridge function can reshape existing features but cannot create new projection directions.

The complete negative-result ledger is consistent with this boundary. Nested's Pets advantage shrinks to a tie at 60 epochs (92.31 versus 92.24), while its Flowers advantage remains 0.80 points through 60 epochs (Table 9); the ViT result is a tie; GroupAdamW adds nothing over AdamW; sorted minimum-gap knots lose on both datasets; the gated residual is the better Flowers arm; unpinned BiasNorm costs 10.5 points on Pets; and Food-101's overall peak remains the old, non-identity-exact minimum-gap recipe (82.33 versus nested 81.96). Nested is the clean test of accessible expressivity, not the winner of every cell.

8 Related work

Learned activations. PReLU learns a channel slope (He et al., 2015); Maxout uses independent affine branches (Goodfellow et al., 2013); APL learns hinge locations and slopes (Agostinelli et al., 2015); rational units fit a quotient of polynomials (Molina et al., 2020); and KANs learn edge-wise spline functions (Liu et al., 2024). Multi-bias instead shares one projection across branches, making it a rank-one block. This constraint is cheaper than independent affine branches but creates the symmetry studied here. Li et al. (2016) also shift one activation by several biases, but emit the responses as separate channels; we sum them into one output and study identity nesting. Matched Maxout, APL, and KAN comparisons are reported in Appendix C.6.

Parameter-efficient tuning and binary networks. BitFit tunes existing biases (Ben Zaken et al., 2022); LoRA learns low-rank weight updates (Hu et al., 2022); adapters insert bottleneck modules (Houlsby et al., 2019); prompt tuning learns input embeddings (Lester et al., 2021); and prefix tuning inserts learned key–value prefixes throughout a transformer (Li & Liang, 2021). Our question is orthogonal: when activations themselves are the trainable interface, which parameterization makes their nominal shape degrees of freedom reachable? Function-preserving zero initialization also has

close precedents: LoRA zero-initializes one low-rank factor, Fixup zero-initializes the last layer of each residual branch (Zhang et al., 2019), and ReZero starts each residual gate at zero (Bachlechner et al., 2021). Nesting transfers this principle to activation branches while placing their knots at distinct locations so their coefficient gradients differ immediately. The reduction is not otherwise direct: those constructions gate or zero branches with their own transform or projection directions, whereas all of our branches share one projection. Here the knot is both the symmetry-breaking coordinate and part of the forward function, so separating it directly causes the measured 0.096–0.526 relative-L^2 initialization damage. Binary networks (Hubara et al., 2016; Rastegari et al., 2016; Hubara et al., 2018) provide a controlled matched-depth setting, but we make no hardware, compression, or lookup-table efficiency claim.

Logic and lookup-table models. Differentiable logic-gate networks and differentiable weightless networks learn discrete logic or interconnected lookup-table structures (Petersen et al., 2022; Bacellar et al., 2024). FINN maps binarized networks to FPGA inference engines (Umuroglu et al., 2017), while LUTNet, LogicNets, PolyLUT, and NeuraLUT map quantized neurons or subnetworks into FPGA soft logic (Wang et al., 2019; Umuroglu et al., 2020; Andronic & Constantinides, 2023; 2024). These are deployment architectures, whereas our unit is a width-preserving activation inside an otherwise conventional network. We use FINN-CNV only for the coefficient ablation in Appendix C.5; because its summed output is not re-binarized, we make no cross-system accuracy or hardware claim.

Threshold logic and optimization symmetry. Our finite-domain analysis follows the classical connection between linear threshold functions and hyperplane arrangements (Muroga, 1971; Winder, 1965; Ojha, 2000; Stanley, 2007; Zaslavsky, 1975). More generally, parameter symmetries constrain neural optimization dynamics (Kunin et al., 2021). Our result isolates a specific interaction of shared rank-one features, tied biases, and coordinate-wise adaptive normalization. It complements expressivity results by showing that representation and reachability can disagree within the same nominal model class.

9 Conclusion

Multi-bias neurons express a rich one-dimensional shape family with real coefficients, but the fixed-sign deployment corner freezes those coefficients and can symmetry-lock its knots under AdamW. Zero-coefficient nesting brings the deployed parameterization back in line with the theory: it preserves the pretrained function exactly while exposing branch-specific gradients. It is significantly better than matched-site $K = 1$ in all four ResNet-50 cells and never significantly worse in any tested transfer cell; ViT and long-budget Pets are ties. The matched fixed-sign CIFAR-100 arm instead falls significantly below $K = 1$, showing that parameterization, not branch count, determines the sign of the effect. The resulting claim is bounded: identity-exact parameterization makes one-dimensional shape capacity reachable; it does not remove the ceiling imposed by frozen features. Coefficient-space regularization is a concrete route to test next, not a benefit established here.

AI use statement

Generative AI tools were used for research ideation, manuscript drafting and editing, literature retrieval, experiment and code development, and mathematical derivation and proof development. The authors independently checked every proof and take responsibility for every claim. For $K^*(5) \geq 8$, SAT/CEGAR supplied only a candidate truth table; the reported certificate comes from an independent exhaustive exact-integer enumeration that does not rely on the SAT search. The identity and optimizer-symmetry claims have deterministic tests, bibliographic records were checked against publisher metadata, and every reported number is traced to the provenance manifest and experiment outputs. No data were fabricated, and the authors executed and reviewed the final verification and experimental pipelines, text, code, and artifacts.

Ethics statement

This work studies a general activation parameterization using public benchmarks and pretrained models. It introduces no human-subject data, user study, or new surveillance or generative capability.

Its direct risks are therefore limited, although the method can inherit biases and failure modes from the pretrained models and datasets to which it is applied. We report negative and neutral results alongside improvements and avoid making deployment-efficiency claims not supported by measurements.

Reproducibility statement

The supplementary material contains complete proofs, experiment hyperparameters, seed-level results, and commands. The released code includes the activation and optimizer implementations, deterministic tests for the symmetry mechanism and identity guarantee, and scripts for all main experiments. The Boolean lower bound is accompanied by a hard-function witness, an exhaustive ordering oracle generator, and a verifier; every quantitative claim is mapped in the provenance manifest to a raw output, a shipped artifact, or an explicitly labelled legacy aggregate. The manifest preserves these evidence levels rather than pooling results from different configurations.

References

Forest Agostinelli, Matthew Hoffman, Peter Sadowski, and Pierre Baldi. Learning activation functions to improve deep neural networks. In *ICLR Workshop*, 2015.

Marta Andronic and George A. Constantinides. PolyLUT: Learning piecewise polynomials for ultra-low latency FPGA LUT-based inference. In *International Conference on Field Programmable Technology*, pp. 60–68, 2023. doi: 10.1109/ICFPT59805.2023.00012.

Marta Andronic and George A. Constantinides. NeuraLUT: Hiding neural network density in boolean synthesizable functions. In *34th International Conference on Field-Programmable Logic and Applications*, pp. 140–148, 2024. doi: 10.1109/FPL64840.2024.00028.

Alan Tendler Leibel Bacellar, Zachary Susskind, Mauricio Breternitz Jr., Eugene John, Lizy Kurian John, Priscila Machado Vieira Lima, and Felipe M. G. França. Differentiable weightless neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 2277–2295, 2024.

Thomas Bachlechner, Bodhisattwa Prasad Majumder, Huanru Henry Mao, Garrison W. Cottrell, and Julian McAuley. ReZero is all you need: fast convergence at large depth. In *UAI*, 2021.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: simple parameter-efficient fine-tuning for transformer-based masked language-models. In *ACL*, 2022.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *ECCV*, 2014.

Tianqi Chen and Carlos Guestrin. XGBoost: a scalable tree boosting system. In *KDD*, pp. 785–794, 2016.

Edmund Clarke, Orna Grumberg, Somesh Jha, Yuan Lu, and Helmut Veith. Counterexample-guided abstraction refinement for symbolic model checking. *Journal of the ACM*, 50(5):752–794, 2003.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16×16 words: transformers for image recognition at scale. In *ICLR*, 2021.

Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio. Maxout networks. In *ICML*, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In *ICCV*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *ICML*, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: low-rank adaptation of large language models. In *ICLR*, 2022.

Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. In *NeurIPS*, 2016.

Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: training neural networks with low precision weights and activations. *Journal of Machine Learning Research*, 18(187):1–30, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, 2009.

Daniel Kunin, Javier Sagastuy-Brena, Surya Ganguli, Daniel L. K. Yamins, and Hidenori Tanaka. Neural mechanics: symmetry and broken conservation laws in deep learning dynamics. In *ICLR*, 2021.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *EMNLP*, 2021.

Hongyang Li, Wanli Ouyang, and Xiaogang Wang. Multi-bias non-linear activation in deep neural networks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pp. 221–229, 2016.

Xiang Lisa Li and Percy Liang. Prefix-tuning: optimizing continuous prompts for generation. In *ACL*, 2021.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y. Hou, and Max Tegmark. KAN: Kolmogorov-Arnold networks, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Alejandro Molina, Patrick Schramowski, and Kristian Kersting. Padé activation units: end-to-end learning of flexible activation functions in deep networks. In *ICLR*, 2020.

Saburo Muroga. *Threshold logic and its applications*. Wiley-Interscience, 1971.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

Piyush C. Ojha. Enumeration of linear threshold functions from the lattice of hyperplane intersections. *IEEE Transactions on Neural Networks*, 11(4):839–850, 2000. doi: 10.1109/72.857765.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *CVPR*, 2012.

Felix Petersen, Christian Borgelt, Hilde Kuehne, and Oliver Deussen. Deep differentiable logic gate networks. In *NeurIPS*, 2022.

Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. XNOR-Net: ImageNet classification using binary convolutional neural networks. In *ECCV*, 2016.

Vincent Sitzmann, Julien N.P̃. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020.

Richard P. Stanley. An introduction to hyperplane arrangements. In *Geometric Combinatorics*, volume 13 of *IAS/Park City Mathematics Series*, pp. 389–496. American Mathematical Society, 2007.

Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020.

Yaman Umuroglu, Nicholas J. Fraser, Giulio Gambardella, Michaela Blott, Philip H. W. Leong, Magnus Jahre, and Kees Vissers. FINN: A framework for fast, scalable binarized neural network inference. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp. 65–74, 2017. doi: 10.1145/3020078.3021744.

Yaman Umuroglu, Yash Akhauri, Nicholas J. Fraser, and Michaela Blott. LogicNets: Co-designed neural networks and circuits for extreme-throughput applications. In *30th International Conference on Field-Programmable Logic and Applications*, pp. 291–297, 2020. doi: 10.1109/FPL50879.2020.00055.

Erwei Wang, James J. Davis, Peter Y. K. Cheung, and George A. Constantinides. LUTNet: Rethinking inference in FPGA soft logic. In *IEEE 27th Annual International Symposium on Field-Programmable Custom Computing Machines*, pp. 26–34, 2019. doi: 10.1109/FCCM.2019.00014.

R. O. Winder. Enumeration of seven-argument threshold functions. *IEEE Transactions on Electronic Computers*, EC-14(3):315–325, 1965.

Thomas Zaslavsky. Facing up to arrangements: face-count formulas for partitions of space by hyperplanes. *Memoirs of the American Mathematical Society*, 1(154), 1975.

Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Fixup initialization: residual learning without normalization. In *ICLR*, 2019.

A Proofs and verification details

A.1 Identity nesting and progressive growth

For Lemma 1, substitute $a_{c,k_0} = 1$ and $a_{c,k} = 0$ for $k \neq k_0$ into Equation (1):

$$f_c(z) = \sigma(z + b_{c,k_0}) + \sum_{k \neq k_0} 0\, \sigma(z + b_{c,k}) = \sigma(z + b_c).$$

The equality is pointwise and independent of the base activation, input distribution, or locations of the unused knots. Appending a zero-coefficient branch proves Corollary 1. Its coefficient derivative is not forced to zero because $\partial f_c / \partial a_{c,k} = \sigma(z + b_{c,k})$; its bias derivative is $a_{c,k}\sigma'(z + b_{c,k}) = 0$ until the coefficient turns on.

A.2 Proof of Proposition 1

Let the distinct knots be $t_1 < \cdots < t_K$ and write

$$f(z) = \sum_{k=1}^{K} a_k \,\mathrm{ReLU}(z - t_k).$$

For $z < t_1$, $f(z) = 0$. On an interval $t_j < z < t_{j+1}$,

$$f'(z) = \sum_{k=1}^{j} a_k,$$

so f is continuous piecewise linear, has at most K knots, and its slope change at t_k is a_k.

Conversely, let g be continuous piecewise linear with knots $t_1 < \cdots < t_K$, far-left value zero, and consecutive slopes $m_0 = 0, m_1, \ldots, m_K$. Taking $a_k = m_k - m_{k-1}$ reconstructs g. For an arbitrary target on a compact interval I, add one knot $t_0 < \inf I$. The term $m_0 \,\mathrm{ReLU}(z - t_0)$ supplies the desired initial slope on I, and a downstream scalar bias supplies the additive constant. Continuous piecewise-linear interpolation is uniformly dense in $C(I)$, proving the approximation statement.

A.3 XOR AND BOOLEAN COMPLETENESS

The explicit XOR certificate in Corollary 2 uses coefficients $(1, -2)$, not fixed alternating signs. For Boolean completeness, put $N = 2^n$ and choose $w_i = 2^{i-1}$. The largest nonzero term in any difference of two signed binary sums strictly exceeds the sum of all lower-order terms, so $z(x) = \langle w, x \rangle$ is injective on $\{-1, +1\}^n$.

Order the projected values $z_1 < \cdots < z_N$ and linearly interpolate the target values $T(x_i)$ between adjacent points. The resulting continuous piecewise-linear function has at most $N - 2$ internal knots. One additional hinge to the left of z_1 supplies its initial slope, and the classifier threshold absorbs its additive constant. Proposition 1 therefore realizes the labels with $K \leq N - 1 = 2^n - 1$. This argument requires unconstrained real slope changes and hence real branch coefficients.

A.4 EXACT FIXED-ORDERING COUNT

Fix any strict ordering of the $N = 2^n$ cube vertices. A thresholded multi-bias output with at most K class changes induces a binary sequence of length N with at most K transitions. Conversely, linear interpolation realizes every such sequence. Choose the initial bit in two ways and choose j transition locations among the $N - 1$ adjacent pairs. Thus

$$M(K, n) = 2 \sum_{j=0}^{\min(K, N-1)} \binom{N-1}{j}.$$

The exact values used in the paper are: This statistic concerns one feasible generic ordering. It is

Table 7: Exact fixed-ordering indicator counts.

K	$n = 2$	$n = 3$	$n = 4$	$n = 5$
1	8	16	32	64
2	14	58	242	994
3	16	128	1,152	9,984
4	16	198	3,882	72,914
5	16	240	9,888	412,736

neither an upper bound on all threshold functions nor a claim of completeness at $K = 5$.

A.5 EXACT EXHAUSTIVE WITNESS FOR $K^*(5) \geq 8$

Inputs are in lexicographic order on $\{-1, +1\}^5$. The certified hard truth table is

index	0	1	2	3	4	5	6	7
$T^\star$	0	0	1	0	0	1	1	1
index	8	9	10	11	12	13	14	15
$T^\star$	1	0	1	0	1	0	1	1
index	16	17	18	19	20	21	22	23
$T^\star$	0	0	1	0	1	1	0	1
index	24	25	26	27	28	29	30	31
$T^\star$	1	1	0	1	1	0	0	0

SAT-guided CEGAR (Clarke et al., 2003) was used only to search for this candidate truth table. The claimed lower bound does not rely on the SAT solver or its search trace: the independent verifier evaluates the displayed table against an exhaustive exact-arithmetic enumeration. Specifically, the hyperplanes $w^\top(u - v) = 0$ partition weight space into chambers, each inducing one strict

cube ordering. Signed coordinate permutations act freely, and every chamber orbit has a unique representative with $0 < w_1 < \cdots < w_5$. After scaling its strict integer inequalities to $Aw \geq \mathbf{1}$, a vertex and Cramer's rule give an integer chamber point. Hadamard's bound gives $\max_i w_i \leq \lfloor 5^{5/2} \rfloor = 55$. The verifier therefore enumerates all $\binom{55}{5} = 3{,}478{,}761$ increasing positive integer vectors, rejects tied projections using exact arithmetic, and obtains exactly 516 orbit representatives. Expanding their 3,840 signed coordinate permutations yields $R(5) = 1{,}981{,}440$ strict orderings. For every ordering π, the verifier computes

$$\mathrm{bc}(T^\star, \pi) = \#\{i : T^\star(\pi_i) \neq T^\star(\pi_{i+1})\}.$$

The minimum is eight. A ReLU multi-bias classifier with K knots has at most K threshold crossings along a strict projection ordering, so $K_{\min}(T^\star) \geq 8$ and therefore $K^*(5) \geq 8$.

For completeness, this argument also covers nongeneric realizing directions and zero or flat threshold segments. On the finite cube, raise the classifier threshold by less than the positive-class margin and perturb the direction away from all ordering hyperplanes by a sufficiently small amount. Labels are unchanged and the resulting ordering is strict. The far-left ReLU segment is constant; after it, each of the K affine segments crosses the perturbed threshold at most once.

The certificate is stored in `artifacts/sat_witness_M5_geq_8.json`. Reproduce the oracle and verification with:

```
python experiments/chamber_enum.py --n 5 \
  --out-npy ~/scratch/multi_bias/chamber/R5_orderings.npy
python experiments/sat_max_bc.py --n 5 \
  --rn-npy ~/scratch/multi_bias/chamber/R5_orderings.npy \
  --verify-witness-json artifacts/sat_witness_M5_geq_8.json \
  --verification-out \
    artifacts/sat_witness_M5_geq_8.verification.json
```

A.6 Proof of the symmetry-lock proposition

At tied biases, all branches evaluate the same derivative $\sigma'(z + b)$, so $g_{t,k} = a_k q_t$. Adam's moment recurrences are linear and quadratic in the gradient:

$$m_{t,k} = \beta_1 m_{t-1,k} + (1 - \beta_1) a_k q_t, \qquad v_{t,k} = \beta_2 v_{t-1,k} + (1 - \beta_2) a_k^2 q_t^2.$$

With equal zero initial states, induction gives $m_{t,k} = a_k \bar{m}_t$ and $v_{t,k} = a_k^2 \bar{v}_t$. Bias correction is shared across coordinates and does not alter the factorization. For $\epsilon = 0$, $m_{t,k}/\sqrt{v_{t,k}}$ depends on a_k only through its sign. Same-sign branches therefore receive equal data updates. If their biases are equal, decoupled weight decay also gives equal updates, preserving the tie. A positive ϵ introduces only the small scale-dependent difference measured in the TinyNet experiment. Sharing v across branches instead preserves the magnitude factor a_k, which explains the GroupAdamW diagnostic.

B A corrected Shannon–Whittaker special case

This supporting result is not used by any experiment or headline claim. We state the form whose assumptions rule out out-of-band aliasing.

Theorem 2 (Lattice-bias reconstruction)**.** *Let $\sigma \in L^2(\mathbb{R})$ be band-limited to $[-W, W]$ and satisfy $0 < \eta \leq |\widehat{\sigma}(\omega)| \leq C < \infty$ almost everywhere on that band. For every $f \in L^2(\mathbb{R})$ band-limited to $[-W, W]$, there are real coefficients a_k and lattice biases $b_k = k\pi/W$ such that*

$$f_K(z) = \sum_{|k| \leq K} a_k \sigma(z + b_k) \longrightarrow f(z) \quad \text{in } L^2.$$

Proof. On $[-W, W]$, define $h(\omega) = \widehat{f}(\omega)/\widehat{\sigma}(\omega)$. The lower bound gives $h \in L^2[-W, W]$. Expand h in the Fourier basis $\{e^{i\omega k\pi/W}\}_{k \in \mathbb{Z}}$ and let h_K be its symmetric partial sums. Then $h_K \to h$ in $L^2[-W, W]$. Translation gives $\widehat{f}_K = \widehat{\sigma} h_K$ on the band, while both sides vanish outside it because σ is band-limited. Hence

$$\|\widehat{f}_K - \widehat{f}\|_2 \leq C \|h_K - h\|_2 \to 0.$$

Plancherel proves the result. Conjugate-symmetric coefficients give real f_K when f and σ are real. □

The band-limiting assumption on the base is essential here. Without it, shifted copies can introduce uncontrolled out-of-band terms; we make no such claim for ReLU, GELU, sigmoid, or Gaussian bases.

C Experimental details and complete results

C.1 Primary PEFT protocol

For ResNet-50, all convolutional weights are frozen; the classifier and activation parameters at every ReLU are trained with AdamW for ten epochs. The classifier learning rate is 10^{-3}, coefficients use the same rate, and knot biases use 10^{-2}. Batch size is 64. ViT-B/16 uses batch size 32 and trains for 15 epochs. Standard train augmentations are random resized crop and horizontal flip; evaluation uses deterministic resize/center crop. Dataset-defined train and test splits are used. ResNet seeds are 0, 1, and 2; ViT also uses seed 3.

Table 8: Parameter counts and per-seed primary PEFT accuracies (%).

Dataset	Arm	Trainable	Seed 0	Seed 1	Seed 2	Seed 3	Mean
CIFAR-100	$K = 1$	229,668	81.68	81.62	81.28	–	81.53
	fixed-sign tied $K = 3$	279,204	79.94	80.45	80.25	–	80.21
	nested $K = 3$	353,508	82.25	82.30	82.39	–	82.31
Food-101	$K = 1$	231,717	81.37	81.41	81.44	–	81.41
	nested $K = 3$	355,557	81.87	82.06	81.96	–	81.96
Pets-37	$K = 1$	100,581	91.63	91.82	91.80	–	91.75
	nested $K = 3$	224,421	92.53	92.45	92.29	–	92.42
Flowers-102	$K = 1$	233,766	84.19	84.37	83.85	–	84.14
	nested $K = 3$	357,606	85.72	85.97	85.62	–	85.77
ViT CIFAR-100	$K = 1$	113,764	87.00	86.98	86.88	86.75	86.90
	nested $K = 3$	298,084	86.87	87.19	86.97	87.17	87.05

C.2 Mechanism and robustness checks

The identity-deviation grid uses 257 evenly spaced scalar inputs and reports $\|f - \mathrm{ReLU}\|_2 / \|\mathrm{ReLU}\|_2$. The TinyNet lock diagnostic uses 80 AdamW steps, learning rate 0.01, batch size 256, 16 input features, and seed 0. Exact configurations and outputs are in `docs/identity_deviation_vs_gap.json` and `artifacts/symmetry_lock_tinynet.json`.

For the noisy-label check, 20% of training labels are replaced symmetrically while validation and test labels remain clean. Last-epoch train/test gaps are computed from the same histories as Table 2; no best-checkpoint selection is used. Re-monotonization replaces learned negative slope increments after training and compares the accuracy drop with a control that perturbs the same number of already monotone channels.

Table 9: Flowers-102 budget durability (three-seed test accuracy). Nested retains a 0.80-point edge at 30/60 epochs; separated stays below $K = 1$. This is an independent rerun (85.73 versus 85.77 in Table 2 at 10 epochs).

Arm	10 epochs	30 epochs	60 epochs
$K = 1$	84.14±0.22	86.75±0.18	87.24±0.25
Nested $K = 3$	**85.73±0.15**	**87.55±0.19**	**88.04±0.16**
Separated $K = 3$	79.30±0.52	85.13±0.32	86.13±0.40

Table 9 shows that longer training improves all arms without erasing the initialization distinction: nested stays ahead, while separated repays most of its damage only after six times the budget.

C.3 PETS LOCALIZATION: OLD SEPARATED-BIAS RECIPE

This experiment predates NESTEDMULTIBIAS and uses the old fixed-sign, minimum-gap initialization. It is retained only as localization evidence; it does not validate the identity-exact repair.

Table 10: Pets-37 localization, mean over three seeds. LoRA is strongest. The $K = 3$ row uses the old minimum-gap recipe.

Task	Strategy	Trainable	mIoU/IoU	Acc@0.5	Acc@0.7
Segmentation	Head only	4k	82.64	–	–
	BitFit	57k	93.91	–	–
	Old multi-bias $K = 3$	72k	94.99	–	–
Bbox	Head only	8k	45.03	43.64	13.62
	BitFit	61k	66.04	84.20	50.32
	Adapter	211k	69.09	89.15	56.40
	Same-site $K = 1$	33k	75.95	95.82	75.61
	Old multi-bias $K = 3$	82k	77.91	96.96	80.47
	LoRA layer 4	78k	**81.11**	**97.77**	**88.74**

C.4 OOD DETECTION: OLD SEPARATED-BIAS RECIPE

These runs also predate NESTEDMULTIBIAS and use the old fixed-sign, minimum-gap recipe. Four seeds train a VGG-style CIFAR-10 classifier for 30 epochs. The result is detector specific: Mahalanobis improves on both OOD sets, while maximum-softmax probability and firing-rate scores do not. We use the feature-space Mahalanobis detector of Lee et al. (2018). Table 11 reports the original four-seed aggregates. A later three-seed rerun changed the absolute SVHN Mahalanobis values ($0.564/0.912$ for ReLU/old multi-bias) but reproduced the direction; we therefore retain this table only as legacy, detector-specific evidence.

Table 11: CIFAR-10 OOD AUROC. Positive conclusions are restricted to the Mahalanobis detector and the old recipe.

OOD set	Variant	In-acc.	Mahalanobis	MSP	Firing
SVHN	ReLU	90.34	0.6272	**0.8980**	**0.9651**
	Old multi-bias $K = 3$	89.15	**0.9029**	0.8557	0.8502
	Δ	-1.19 pp	$+27.56$ pp	-4.23 pp	-11.50 pp
Textures	ReLU	90.38	0.7327	**0.8351**	0.7773
	Old multi-bias $K = 3$	89.27	**0.8793**	0.8132	**0.8121**
	Δ	-1.11 pp	$+14.66$ pp	-2.20 pp	$+3.48$ pp

C.5 FINN-CNV COEFFICIENT-ONLY ABLATION

This is external corroboration of the coefficient mechanism, not a compression, area, latency, or hardware-efficiency claim. All cells use the same FINN-CNV topology, optimizer, schedule, and three seeds; only the multi-bias coefficients change from fixed alternating signs to learned real values. The summed multi-bias output is not re-binarized, so these cells must not be compared as W1A1 hardware operating points.

Table 12: FINN-CNV CIFAR-10 coefficient ablation (test accuracy, three seeds). At each K, learning the coefficients is the only changed arm.

K	Fixed alternating	Learned real	Δ
2	70.15 ± 2.16	$\mathbf{78.04 \pm 0.63}$	$+7.89$
3	80.74 ± 0.66	$\mathbf{83.79 \pm 0.16}$	$+3.05$

C.6 MATCHED LEARNED-ACTIVATION BASELINES

All methods replace the same ResNet-50 activation sites on CIFAR-100 and use the same frozen-backbone protocol. The stabilized KAN run uses an fp32 cubic B-spline basis, coefficient learning-rate multiplier 2, gradient clipping 1.0, and $K = 5$ control points (six trainable scalars per channel, matching nested $K = 3$).

Table 13: Matched-site learned activations under the same frozen-backbone protocol, three seeds. The fixed-sign row is the matched replication reported in this paper; the other rows predate the nested repair.

Activation	Trainable	Accuracy
Maxout $K = 3$	354k	81.66 ± 0.06
APL $K = 3$	428k	$\mathbf{82.48 \pm 0.21}$
KAN-style spline $K = 5$	approximately matched	82.40 ± 0.24
Fixed-sign multi-bias $K = 3$	279k	80.21 ± 0.21

The fixed-sign estimate supersedes a legacy 82.31 ± 0.10 aggregate used in an earlier draft, which inadvertently duplicated the nested result. The matched replication is 1.32 points below the same-site $K = 1$ control (81.53 ± 0.18), whereas nested $K = 3$ is 0.78 points above it.

C.7 RELATION TO LI ET AL. (2016)

Li et al. (2016) emit K shifted activations as KC separate channels. We sum them back into C channels. At an activation between a C-channel convolution and a C'-output $k \times k$ convolution, our additional parameters are $2KC$. The separate-channel construction adds

$$KC + (K - 1)CC'k^2,$$

because the following convolution widens from C to KC inputs. For $C = C' = 256$, $k = 3$, and $K = 3$, these totals are 1,536 versus 1,180,416 parameters per site. A one-site Li-style baseline on CIFAR-100 uses 1.975M trainable parameters and reaches $73.93 \pm 0.14\%$ in a three-seed replication; nested multi-bias at all sites uses 354k and reaches $82.31 \pm 0.06\%$. Because the wiring and parameter counts differ, this is contextual evidence rather than a strict matched-budget comparison.

C.8 DEMOTED CAPABILITY PROBES

These probes map scope and are not evidence for the main optimization claim:

- **One-dimensional and density targets.** A single multi-bias unit is strongest on localized Gaussian and mixture targets; matched tiny networks win all eight tested 1D/2D Gaussian-mixture cells. A β-VAE improves ELBO by 4.93, while a two-dimensional normalizing flow ties its affine baseline.
- **PARITY and INR.** At width four, $K = 3$ reaches 100% on PARITY-3/4; both ReLU and multi-bias saturate by width 16 through PARITY-8. On two small image-INR probes, multi-bias improves over ReLU but remains far below specialized SIREN (Sitzmann et al., 2020) and Fourier-feature models (Tancik et al., 2020).
- **Tabular.** In the matched-small-model study, multi-bias trails ReLU and XGBoost (Chen & Guestrin, 2016) on Adult and Wine, and slightly leads on Covertype (83.30% versus 82.24% ReLU and 82.74% XGBoost). Separate larger-MLP runs reach 86.25% versus 82.80%; these configurations are not pooled.
- **PINNs.** ReLU multi-bias fails Poisson and Helmholtz residual training because its second derivative is zero almost everywhere. Smooth tanh/SiLU bases are required.
- **Tiny GPT.** On a 3.2M-parameter character model, identity-init $K = 3$ ties ReLU in best validation BPC (2.231 ± 0.013 versus 2.230 ± 0.012, $p = 0.93$); other tested K values are worse. This is a neutral small-scale result, not evidence of language-model improvement.

D REPRODUCTION COMMANDS

```
# Deterministic mechanism checks
python tests/test_symmetry_under_adam.py
python -m pytest tests/test_nested_identity.py
python -m pytest tests/test_slope_palette.py

# ResNet-50 / ViT validation and ceiling
python experiments/multibias_theory_audit.py \
  --phase validate --seeds 3 --epochs 10
python experiments/multibias_theory_audit.py \
  --phase ceiling --seeds 3

# FINN-CNV coefficient grid
for K in 2 3; do
  for SIGNS in alt learned; do
    python experiments/exp_qonnx_bnn.py \
      --arch cnv --K "$K" --sign-pattern "$SIGNS" \
      --arms multibias --epochs 100 --seeds 3 \
      --out-dir "results/finn_cnv_K${K}_${SIGNS}"
  done
done
```

The compact, repository-shipped summary `artifacts/iclr_main_results.json` maps main-text numbers to raw outputs or shipped artifacts and labels appendix-only legacy aggregates separately. It never pools distinct configurations. Large checkpoints, datasets, and the 1.98M-row ordering oracle are intentionally excluded from the submission archive.