SIGN LOCK-IN

Randomly Initialized Weight Signs Persist and Bottleneck Sub-Bit Model Compression

Akira Sakai[1,2] Yuma Ichikawa[1,3]

[1] Fujitsu Limited [2] Tokai University [3] RIKEN Center for AIP

1. PROBLEM

As magnitudes are compressed below one bit per weight, the sign bit becomes the dominant bottleneck — the one-bit wall.

2. EMPIRICAL FINDINGS

- Sign matrices are spectrally indistinguishable from i.i.d. Rademacher noise.
- Most sign bits remain inherited from initialization throughout training.

3. SIGN LOCK-IN THEORY

Under bounded updates and a rare re-entry condition, effective sign flips K obey a geometric tail:

$$\Pr\left[K_T^{\mathrm{eff}}(\rho) \geq k\right] \leq h\, g^{b-1} + \delta_{\mathrm{upd}}$$

h: initial-hit factor g: re-entry ratio δ_{upd}: bounded-update slack

4. LOCK-IN ENHANCEMENT (OUR METHOD)

We actively stabilize signs with two lightweight mechanisms that suppress boundary visits and re-entries.

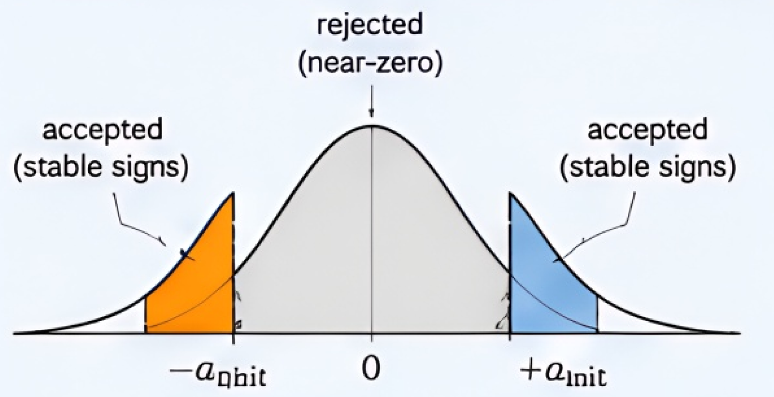
4.1 Gap Initialization

Avoid placing weights near the origin where signs are unstable. Define the gap threshold:

$$a_{\mathrm{init}} := c_{\mathrm{gap}}\, \sigma_{\mathrm{init}}$$

Sample $z \sim \mathcal{N}(0, \sigma_{\mathrm{init}}^2)$ and reject if $|z| < a_{\mathrm{init}}$, repeat until $|z| \geq a_{\mathrm{init}}$.

Equivalently: a two-sided truncated Gaussian on $\mathbb{R} \setminus (-a_{\mathrm{init}}, a_{\mathrm{init}})$.



4.2 Outer-Drift Regularization

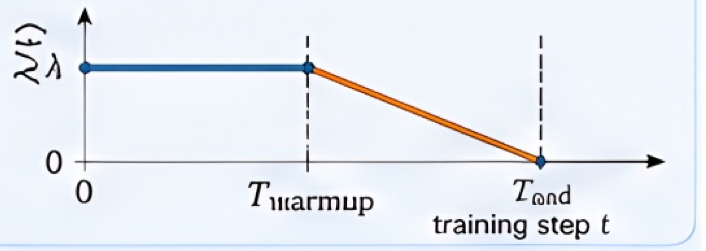
Discourage later drift back toward the boundary with a lightweight log-barrier penalty:

$$R_{\mathrm{LB}}(W; a_{\mathrm{init}}, \epsilon_{\mathrm{lb}}) = \frac{1}{mn} \sum_{i,j} \log\left(\max\left(1, \frac{a_{\mathrm{init}}}{|W_{ij}| + \epsilon_{\mathrm{lb}}}\right)\right)$$

Training objective with selected layers $\mathcal{M}$:

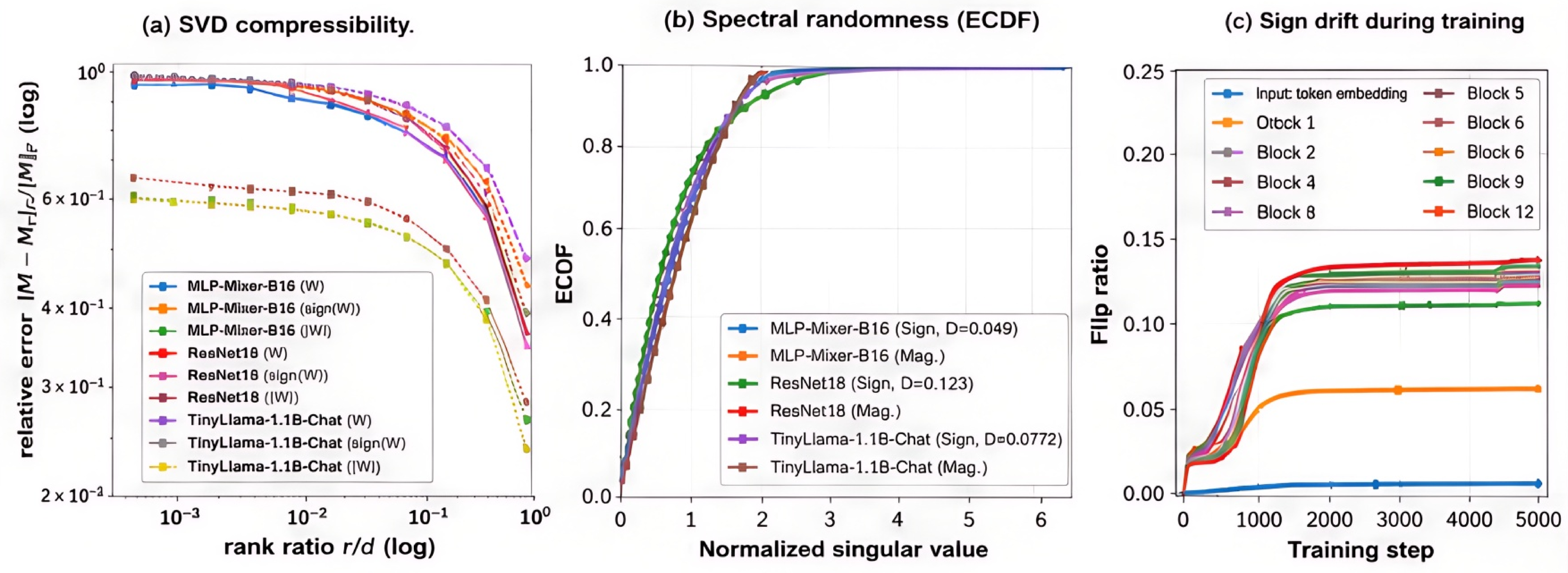
$$\mathcal{L}_{\mathrm{total}}(\theta; t) = \mathcal{L}_{\mathrm{task}}(\theta) + \lambda(t) \sum_{\ell \in \mathcal{M}} R_{\mathrm{LB}}\left(W^{(\ell)}; a_{\mathrm{init}}, \epsilon_{\mathrm{lb}}\right)$$

Schedule: $\lambda(t)$ is held constant during an initial warmup phase, then linearly annealed to 0.



Eapinit → suppresses **INITIAL** boundary hits (reduces h).
Outer-drift reg → suppresses **LATER** re-entries (reduces g).

A. EMPIRICAL VALIDATION OF THE ONE-BIT WALL



Sign matrices are far less compressible than magnitudes (a), are spectrally Rademacher-like (b), and most signs are inherited from initialization (c).

B. RE-GENERABLE SIGN TEMPLATES

(a) Random sign vs. rank-1 template

Random sign S_{rand}

$$\begin{bmatrix} 1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 \end{bmatrix}$$

Rank-1 template S_{temp}

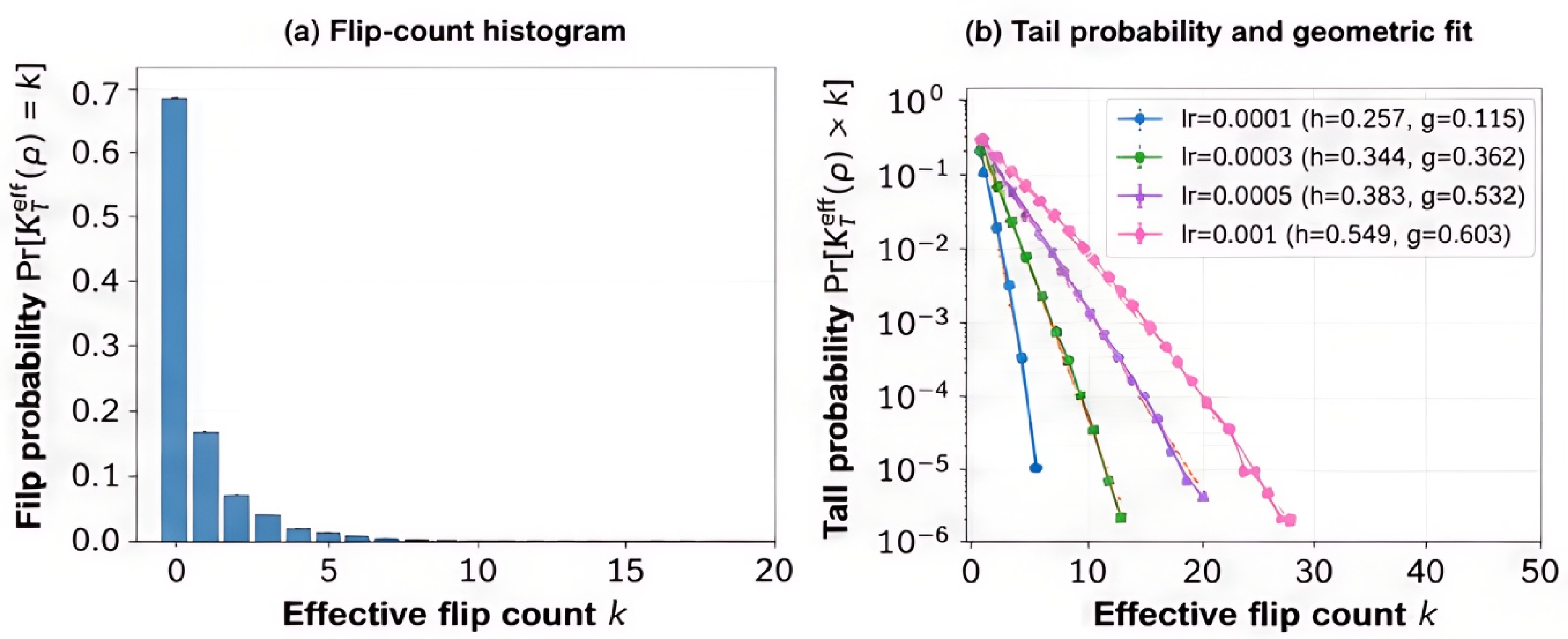
$$\begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & -1 & -1 \end{bmatrix}$$

(b) Takeaway

- Random patterns lack low-rank structure and cost $\approx$ 1 bit/weight.
- Re-generable templates can be rank-one (or low-rank), driving effective sign cost to zero.

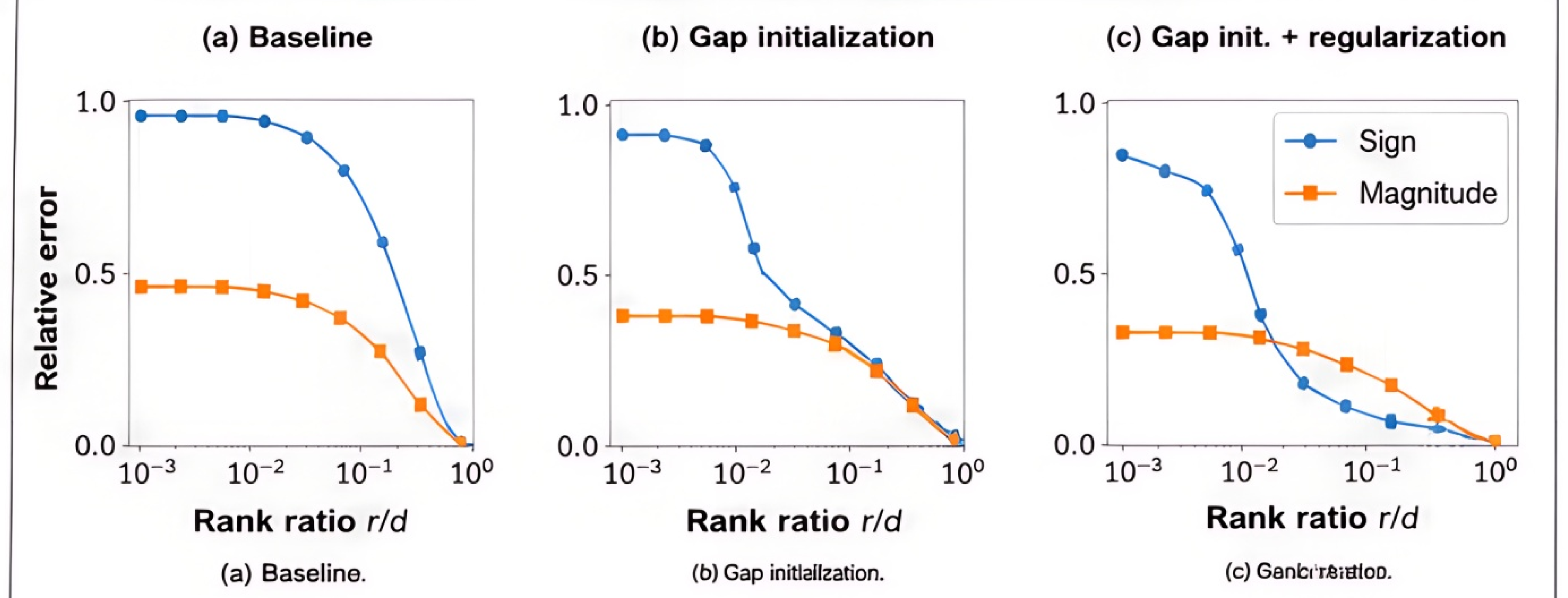
Template structure collapses the effective sign cost.

D. SIGN LOCK-IN — GEOMETRIC TAIL



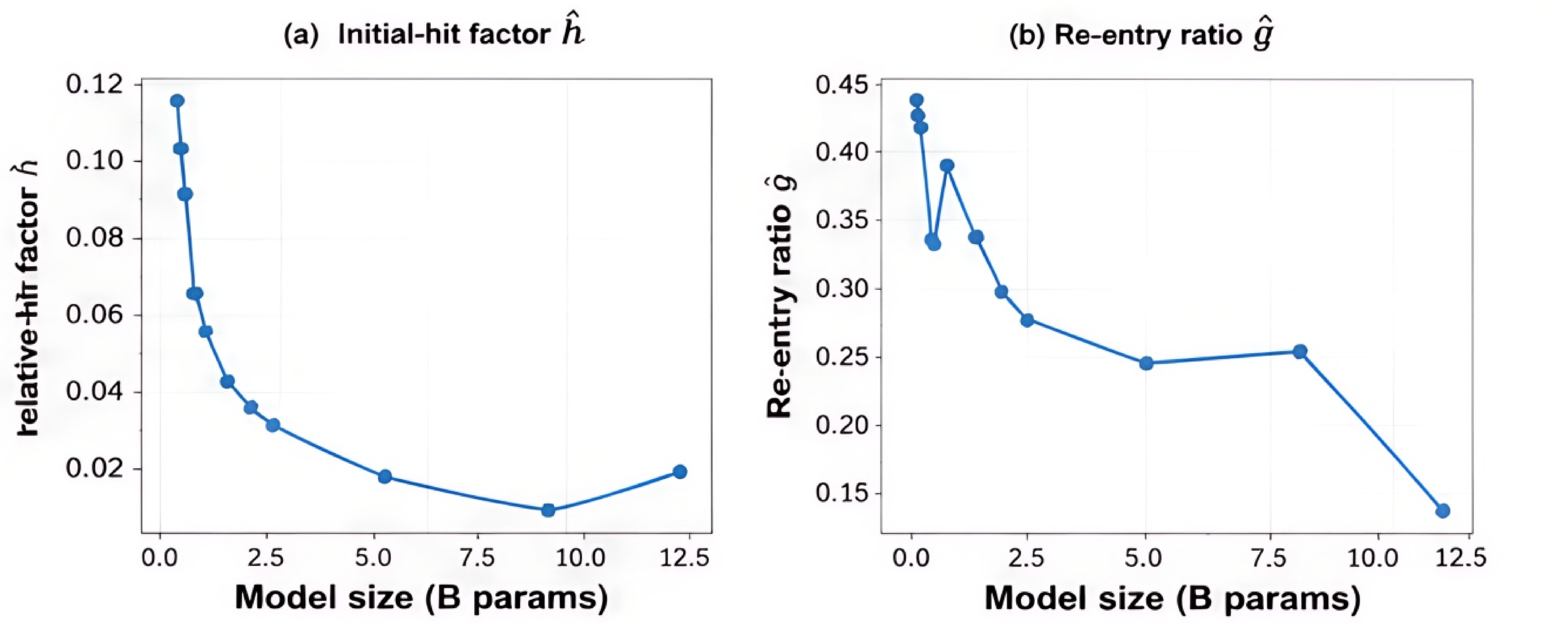
Effective sign flips are exponentially rare; the tail follows a geometric law with parameters (h, g).

C. SIGN vs. MAGNITUDE COMPRESSIBILITY



Gap init. and outer-drift regularization dramatically improve sign-matrix compressibility.

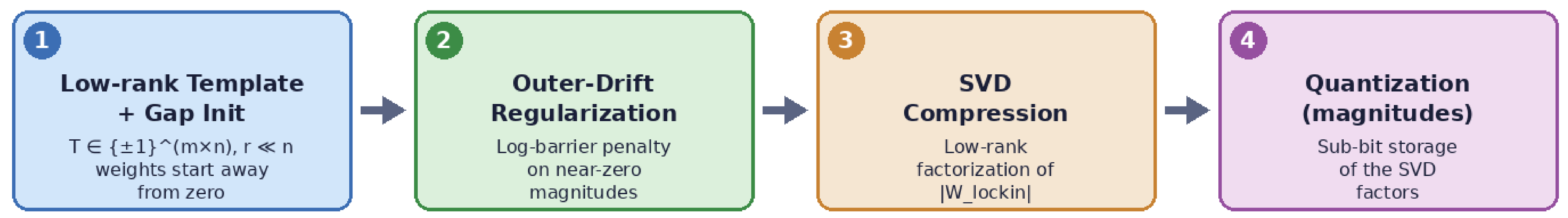
E. LOCK-IN PARAMETERS SWEEP

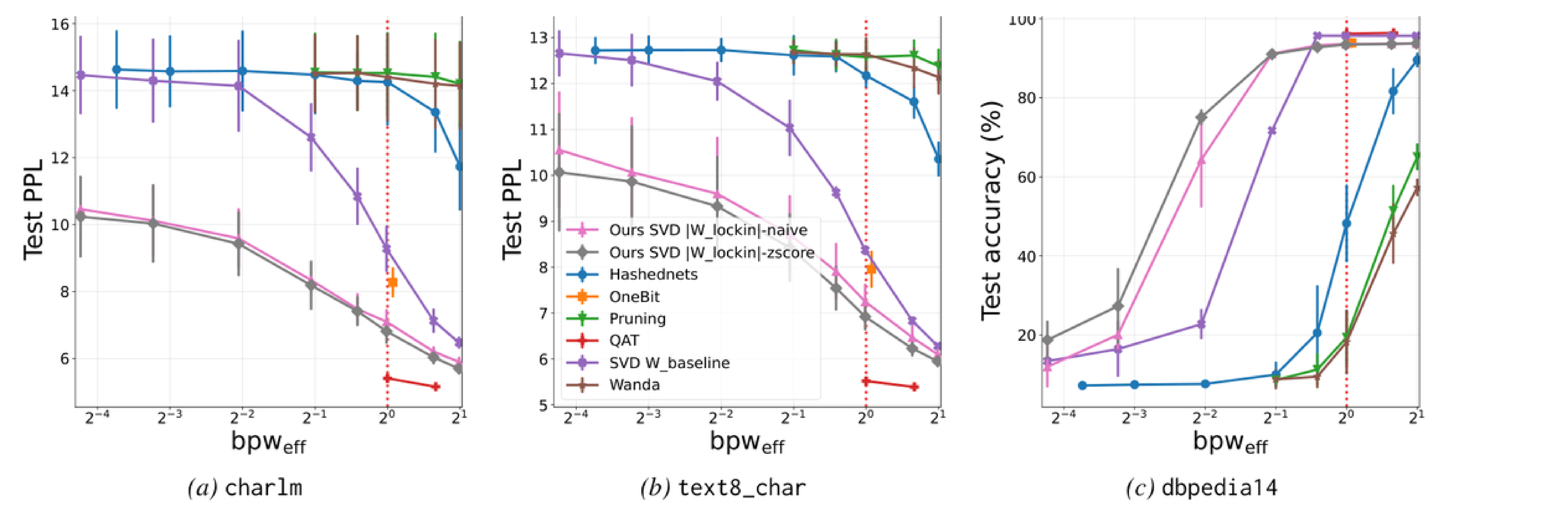


Larger models exhibit lower initial-hit factors $\hat{h}$ and re-entry ratios $\hat{g}$, making lock-in even stronger at scale.

F. DOWNSTREAM PERFORMANCE vs. EFFECTIVE BITS PER WEIGHT

OUR METHOD





Across CharLM, Text8-Char, and DBPedia14, our method dominates baselines in the sub-bit regime (bpw_eff < 1).

KEY TAKEAWAY

Random signs lock in.
Most sign bits never flip due to rare boundary visits and re-entries.

Lock-in can be made stronger.
Gap initialization lowers initial hits (h); outer-drift regularization lowers re-entries (g).

Break the one-bit wall.
Stronger lock-in enables lower effective bpw without sacrificing downstream performance.

Sign lock-in explains the fundamental bottleneck at sub-bit precision — and provides a principled, practical path to overcome it.