LOCAL M-COVERS FOR REPLICATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Replicated learning temporarily introduces multiple copies of a model, modifying the training dynamics in ways that can improve generalization while allowing the replicas to contract back to a single deployable model. We ask whether the *internal structure* of this temporary lift can be tuned so that these replica fluctuations induce task-relevant corrections. We study this through an M-cover, which replicates and rewires a model's factor graph, and introduce a local M-cover in which replica fluctuations interact through structured rather than uniform coupling. For a general model, we derive the dynamics of replica-difference modes near synchronization and show that generalization improves when their induced predictor displacement aligns with the source-posterior correction. In an analytically tractable random-feature model with two trainable linear layers, this yields a concrete link between source (teacher-prior)–representation mismatch and the preferred cover geometry, and predicts the transition between improved and degraded generalization. Across broader benchmarks, structured M-covers outperform established replicated-learning methods across tasks, architectures, learning dynamics, and overparameterization regimes. These results identify the internal geometry of temporary replication as a new design axis for generalization: structured replica interactions can improve the final single-model predictor without increasing deployment-time model size.

1 INTRODUCTION

Learning with interacting copies of a model can improve optimization and generalization. Replica-based methods favor dense solution regions by coupling multiple copies (Baldassi et al., 2016; Angelini & Ricci-Tersenghi, 2019; Catania et al., 2024); related approaches include Elastic SGD, replicated SGD, and Entropy-SGD (Zhang et al., 2015; Pittorino et al., 2021; Chaudhari et al., 2019). Together, they reflect the broader principle that generalization depends not only on the solution reached, but also on the surrounding geometry and training dynamics (Hochreiter & Schmidhuber, 1997; Keskar et al., 2016; Baldassi et al., 2022; Zhu et al., 2022; Chaudhari et al., 2017; Koloskova et al., 2020; Wang & Joshi, 2021).

Temporary overparameterization through replicated learning makes this question particularly interesting. The replicated parameter space splits into a shared mode, where all replicas move together, and transverse modes, where they differ. Coupled replication allows differences among replicas to influence their shared learning trajectory before contraction yields a single deployable model. These transverse fluctuations need not all be equally useful for a particular learning problem. This motivates the broader question studied here: *what internal structure of a temporary replicated lift is useful for learning, and can it be designed to steer the shared predictor in task-relevant directions?* Here, that structure is the coupling pattern among replicas.

One natural way to impose such internal structure is through the connectivity of a replicated graphical model. A learning objective can be represented as a factor graph whose factors encode local interactions among trainable variables (Jordan, 2004; Bishop & Nasrabadi, 2006). Within this representation, an M-cover (also called an M-layer) is a transformation that creates M copies of every variable and factor and rewires their incidences across copies. It thereby changes the global connectivity of the lifted graph while preserving the functional form of each factor (Vontobel, 2013; Lucibello et al., 2014; Altieri et al., 2017; Leleu et al., 2026). In the local approximation studied here, this rewiring induces structured interactions among transverse replica fluctuations.

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.

For a general model, we derive the predictor displacement induced by replica fluctuations and show that generalization improves when it aligns with the posterior correction implied by a source prior. In an analytically tractable random-feature model with two trainable linear layers, we evaluate this criterion explicitly under a standard Gaussian source prior and identify a finite range of local, nonuniform mixing for which improvement occurs. Simulations confirm this prediction and show that nonuniform mixing can outperform uniform replicated SGD. We further test the method across diverse tasks, architectures, learning dynamics, and overparameterization regimes, where structured M-covers improve over established replicated-learning approaches.

2 METHOD

2.1 AN M-COVER OF A GENERAL MODEL

Consider a general loss function $\mathcal{L}(\theta)$ expressed as a factor graph (see Fig. 1a),

$$\mathcal{L}(\theta) = \sum_{a \in \mathcal{F}} \ell_a(\{\theta_v\}_{v \in \partial a}). \tag{1}$$

Here θ_v denotes trainable parameter associated with variable v; $\mathcal{F}$, the set of factors; and ∂a denotes the variables incident to a factor a. We define an M-*cover* lift of this factor graph as follows. (1) Make M copies of every variable block and every factor. (2) For each factor–variable incidence (a, v), a permutation π_{av} determines which copy of θ_v is read by factor copy r. A joint permutation law ν then defines the loss of the lifted graph,

$$\mathcal{L}_M^{\text{full}} = \mathbb{E}_{\pi \sim \nu} \frac{1}{M} \sum_{r=1}^{M} \sum_{a \in \mathcal{F}} \ell_a \left(\{\theta_v^{(\pi_{av}(r))}\}_{v \in \partial a} \right), \tag{2}$$

where $\mathbb{E}_{\pi \sim \nu}$ denotes expectation over the collection of permutations $\pi = \{\pi_{av}\}_{a \in \mathcal{F},\, v \in \partial a}$ drawn from the joint distribution ν.

2.2 A LOCAL SIMPLIFICATION FOR NEURAL NETWORKS

Consider an L-layer neural network with parameter blocks $\theta = (\theta_1, \ldots, \theta_L)$ and loss $\mathcal{L}(\theta)$, where θ_ℓ collects the parameters of layer ℓ. Its layer activations z^ℓ, input x, and predictor $f_\theta(x)$ are defined by

$$z^0 = x, \qquad z^\ell = F_\ell(\theta_\ell, z^{\ell-1}), \quad \ell = 1, \ldots, L, \qquad f_\theta(x) = z^L. \tag{3}$$

To construct the local M-cover, we first make M copies of the network, indexed by $r = 1, \ldots, M$, with parameters $\theta^{(r)} = (\theta_1^{(r)}, \ldots, \theta_L^{(r)})$. Rather than permuting the full replica computations, the local construction acts only on parameter fluctuations around their common center. We therefore decompose each replica as

$$\theta_\ell^{(r)} = c_\ell + \delta_\ell^{(r)}, \qquad c_\ell = \frac{1}{M} \sum_r \theta_\ell^{(r)}, \qquad \sum_r \delta_\ell^{(r)} = 0. \tag{4}$$

Here c_ℓ denotes the mean parameter at layer ℓ, and $\delta_\ell^{(r)}$ is the fluctuation of cover r around this mean.

The basic operation of the cover is a local substitution between replicas. Suppose that, at layer ℓ, the fluctuation of cover r is replaced by that of cover s. Holding the input to the layer fixed at the common center representation $\bar{z}^{\ell-1}(c; x)$, the exact change in the local layer output is

$$\Delta z_\ell^{rs}(c; x) = F_\ell\Big(c_\ell + \delta_\ell^{(r)}, \bar{z}^{\ell-1}(c; x)\Big) - F_\ell\Big(c_\ell + \delta_\ell^{(s)}, \bar{z}^{\ell-1}(c; x)\Big), \tag{5}$$

where $\bar{z}^{\ell-1}$ is given as: $\bar{z}^0(c; x) = x$, $\bar{z}^\ell(c; x) = F_\ell(c_\ell, \bar{z}^{\ell-1}(c; x))$, $f_c(x) = \bar{z}^L(c; x)$.

Thus, Δz_ℓ^{rs} measures the local incompatibility between covers r and s: it is the change in the output of the same layer, in the same computational context, when one replica fluctuation is substituted for the other.

Close to the synchronized manifold, the fluctuations are small and can be linearized as follows:

$$\Delta z_\ell^{rs}(c; x) = S_\ell(c; x) \left(\delta_\ell^{(r)} - \delta_\ell^{(s)} \right) + O(\|\delta\|^2), \qquad S_\ell(c; x) = D_{\theta_\ell} F_\ell \left(c_\ell, \bar{z}^{\ell-1}(c; x) \right), \tag{6}$$

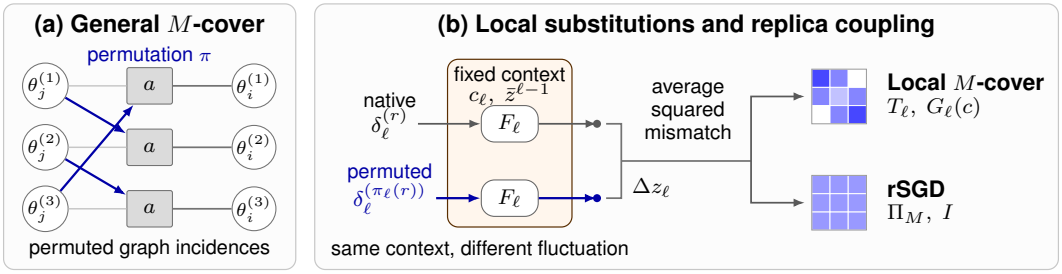


Figure 1: **From covers to local compatibility.** (a) Cover permutations reroute graph incidences. (b) Compare native and substituted layer outputs at fixed center context. The local and tangent approximation to the averaged squared mismatch gives coupling with mixing T_ℓ and metric $G_\ell(c)$. Uniform mixing Π_M and the Euclidean metric I recover rSGD.

Here $S_\ell(c;x)$ denotes the local parameter-to-output response and $D_{\theta_\ell}F_\ell$ is the derivative of the layer output with respect to its trainable parameters. The incoming representation is held fixed when taking this derivative.

We define the local M-cover loss as

$$\mathcal{L}_M(c,\delta) = \frac{1}{M}\sum_r \mathcal{L}\Big(c+\delta^{(r)}\Big) + R(c,\delta), \tag{7}$$

where $R(c,\delta)$ measures the squared local mismatch, summed over replica pairs and averaged over the empirical distribution $\hat{\mu}_D = \frac{1}{D}\sum_{i=1}^{D}\delta_{x_i}$ and cover permutations:

$$R(c,\delta) = \sum_\ell \frac{\gamma_\ell}{4M}\sum_{r,s}(T_\ell)_{rs}\,\mathbb{E}_{x\sim\hat{\mu}_D}\left[\left\|S_\ell(c;x)\left(\delta_\ell^{(r)}-\delta_\ell^{(s)}\right)\right\|^2\right]. \tag{8}$$

Note that we write $c=(c_1,\dots,c_L)$ and $\delta^{(r)}=(\delta_1^{(r)},\dots,\delta_L^{(r)})$. We call this simplification a *local M-cover*, as it evaluates replica mismatch layer by layer (see Fig. 1b).

The average effect of the cover permutations is described by

$$(T_\ell)_{rs} = \mathbb{P}[\pi_\ell(r)=s]\,, \tag{9}$$

where $(T_\ell)_{rs}$ is the probability that cover r is paired with s at layer ℓ. As an average of permutation matrices, T_ℓ is doubly stochastic. The normalization $1/(4M)$ recovers the usual rSGD convention under uniform coupling, and $\gamma_\ell \geq 0$ sets the coupling strength.

The compatibility term can equivalently be written directly in parameter space as follows

$$R(c,\delta) = \frac{1}{2M}\sum_\ell \gamma_\ell\,\delta_\ell^\top\Big[\underbrace{L_\ell}_{\text{replica interaction}} \otimes \underbrace{G_\ell(c)}_{\text{local functional metric}}\Big]\delta_\ell. \tag{10}$$

Here $\delta_\ell = \{\delta_\ell^{(r)}\}_{r=1}^M$ is the stacked fluctuation vector and $\otimes$ denotes the Kronecker product. Thus T_ℓ sets the replica-space mixing, γ_ℓ its strength, and $G_\ell(c)$ the local functional metric:

$$G_\ell(c) = \mathbb{E}_{x\sim\hat{\mu}_D}\left[S_\ell(c;x)^* S_\ell(c;x)\right]. \tag{11}$$

where $*$ denotes the adjoint and $L_\ell = I - \mathrm{Sym}(T_\ell), \qquad \mathrm{Sym}(T) = \frac{T+T^\top}{2}$.

The full M-cover in Eq. (2) generally propagates the effect of substitutions simultaneously through the rest of the computation graph. In the local model, we deliberately replace this global routed computation by the collection of local defects.

2.3 Relation to replicated SGD

The local M-cover contains standard replicated SGD with Euclidean replica attraction as a special case. Let $\Pi_M = \frac{\mathbf{1}\mathbf{1}^\top}{M}$ denote the uniform mixing matrix. If, at every layer, $T_\ell = \Pi_M$, $G_\ell = I$, and

$\gamma_\ell = \gamma$, then the compatibility term becomes

$$R(c, \delta) = R_{\mathrm{RSGD}}(\delta) = \frac{\gamma}{2M} \sum_{\ell,r} \left\| \delta_\ell^{(r)} \right\|^2 . \tag{12}$$

This is simply a quadratic attraction of every replica toward the replica average, while each replica is still trained on its own native task loss.

The local M-cover generalizes this construction in two independent ways. A nonuniform mixing matrix T_ℓ changes which replica differences are penalized and with what strength, while a nonidentity metric $G_\ell(c)$ changes which physical parameter differences are regarded as important, contracting functionally sensitive directions more strongly and *vice versa*. Thus structured mixing generalizes the geometry in replica space, whereas the local functional metric generalizes the geometry in parameter space. In the theory of Secs. 3 and 4, we retain the general functional metric $G_\ell(c)$, whereas in the empirical benchmarks of Sec. 5 we set $G_\ell = I$ to simplify the implementation and isolate the effect of structured replica-space mixing.

3 General theory

The main result of this section shows that the local M-cover improves generalization over the base network when the predictor displacement it induces aligns with the correction preferred by the source posterior.

3.1 Center and fluctuation dynamics

We study gradient flow with learning rate η. For the original model, the parameters evolve according to $\dot{\theta} = -\eta \nabla \mathcal{L}(\theta)$. For the M-cover objective $\mathcal{L}_M$, the replica dynamics can equivalently be decomposed into the evolution of the common center c and of the zero-mean fluctuations $\delta^{(r)}$ around it.

Close to the synchronized manifold, $\delta^{(r)} = 0$, the zero-mean property implies that the fluctuations do not modify the center dynamics at first order. Their leading effect on the center is therefore quadratic. Conversely, the fluctuations themselves evolve linearly to leading order. Introducing the mixing-mode decomposition below makes these two contributions explicit.

3.2 Decomposition into mixing modes

We project the cover-to-cover fluctuations onto the modes of the replica-mixing matrices, separating transverse motion from the common center.

We assume first that the replica interaction L_ℓ share a common eigenbasis for all layer ℓ. Let $e_q \in \mathbb{R}^M$, $q = 1, \ldots, M-1$, denote these eigenvectors with $L_\ell e_q = \kappa_{\ell q} e_q$ and normalized as $e_q^\top e_p = M \delta_{qp}$ and $\sum_r e_q(r) = 0$, where δ_{qp} is the Kronecker delta and $\kappa_{\ell q}$ is the eigenvalue of the interaction matrix at layer ℓ in cover mode q.

We expand the replica fluctuations in this basis as

$$\delta^{(r)} = \sum_q e_q(r) h_q, \qquad \frac{1}{M} \sum_r \left\| \delta^{(r)} \right\|^2 = \sum_q \|h_q\|^2 . \tag{13}$$

Each h_q collects all layer fluctuations in one cover mode; since e_q is orthogonal to the uniform replica direction, these are *transverse modes*.

Close to the uniform replica manifold, $h_q = 0$, we expand the local M-cover loss of Eq. (7) to second order in these transverse fluctuations. The zero-mean property eliminates the term linear in h_q, giving

$$\mathcal{L}_M(c, h) = \mathcal{L}(c) + \frac{1}{2} \sum_q h_q^\top \mathbb{H}_q(c) h_q + O(\|h\|^3). \tag{14}$$

The matrix $\mathbb{H}_q(c)$ is the effective curvature experienced by cover mode q. It contains the curvature of the original task together with the restoring force generated by local compatibility,

$$\mathbb{H}_q(c) = H(c) + \mathrm{diag}_\ell \left[\gamma_\ell \kappa_{\ell q} G_\ell(c) \right], \qquad H(c) = \nabla^2 \mathcal{L}(c). \tag{15}$$

Here, diag_ℓ denotes the block-diagonal matrix formed by assembling the blocks for all layers. Thus, the native loss and the M-cover compatibility act together on each transverse mode.

Gradient flow with learning rate η then gives, close to the synchronized manifold (see App. Sec. A),

$$\dot{h}_q = -\eta \left[\mathbb{H}_q(c) h_q + O(\|h\|^2)\right], \tag{16}$$

$$\dot{c} = -\eta \left[\nabla \mathcal{L}(c) + \sum_q \mathbb{N}_q(c)[h_q, h_q] + O(\|h\|^3)\right]. \tag{17}$$

The corresponding quadratic center force is defined by $\mathbb{N}_q(c)[u,v] = \frac{1}{2}\nabla_c \left[u^\top \mathbb{H}_q(c) v\right]$, where u and v are held fixed when taking the derivative with respect to the center. Hence the same center-dependent transverse operator $\mathbb{H}_q(c)$ determines both the lifetime of a fluctuation and the force that this fluctuation exerts on the center.

3.3 Contraction

We now ask when the transverse modes decay to zero, so that the replicated dynamics collapse onto the center trajectory. At a fixed center $c_\star$, contraction occurs whenever the transverse operator is positive definite on the directions considered,

$$\mathbb{H}_q(c_\star) \succ 0. \tag{18}$$

Since the compatibility term adds positive curvature, sufficiently large γ_ℓ can overcome any finite negative native task curvature, provided that the corresponding local metrics $G_\ell(c_\star)$ are strictly positive in those directions. Directions lying in the nullspace of the local metrics, i.e., directions v satisfying $G_\ell(c_\star)v = 0$, cannot be stabilized in this way.

3.4 Center-force displacement for a general model

Before contracting, the transverse fluctuations exert a transient quadratic center force that can leave a finite displacement of the deployed predictor.

Let $c_0(t)$ denote the center trajectory obtained when the replicas follow the base dynamics, and let $c_M(t)$ denote the center trajectory under the M-cover dynamics. The corresponding predictors are $f_{c_0(t)}$ and $f_{c_M(t)}$, respectively. Their difference at time T is

$$\Delta f(T) = f_{c_M(T)} - f_{c_0(T)} = \epsilon^2 D_c f_c\big|_{c=c_0(T)} [\xi(T)] + O(\epsilon^3). \tag{19}$$

Here $\xi(T)$ is the leading displacement of the M-cover center relative to the base center trajectory, while $D_c f_c|_{c=c_0(T)}$ is the derivative of the predictor with respect to the center parameters, evaluated along the base trajectory.

To compute $\xi(T)$, define the transverse and center propagators by

$$\partial_t U_q(t,u) = -\eta\, \mathbb{H}_q(c_0(t)) U_q(t,u), \qquad \partial_t \Phi(t,u) = -\eta\, H(c_0(t))\Phi(t,u), \tag{20}$$

with $U_q(u,u) = \Phi(u,u) = I$. We take the M-cover and base trajectories to have the same initial center, $c_M(0) = c_0(0)$, and initialize each transverse mode as $h_q(0) = \epsilon v_q^0$, where $\epsilon \ll 1$. Assuming the perturbations remain small, the accumulated center displacement is then (see App. Sec. B)

$$\xi(T) = -\eta \sum_q \int_0^T \Phi(T,t)\, \mathbb{N}_q(c_0(t)) \left[U_q(t,0)v_q^0, U_q(t,0)v_q^0\right] \mathrm{d}t. \tag{21}$$

Thus, the final displacement is obtained by integrating the quadratic center force generated by each transverse mode over its lifetime. The propagator U_q determines how much of the initial fluctuation survives at time t, while Φ determines how much of the resulting center displacement remains at time T. This result is general: no particular model architecture or source prior has been assumed.

3.5 A general source prior

To characterize generalization, we adopt a standard teacher–student formulation (Sollich & Halees, 2002; Bordelon et al., 2020; Canatar et al., 2021; Bordelon et al., 2024). Inputs follow $x \sim \mu$, the

target $f_\star$ is drawn from a source prior P_0 with finite second moment under μ, and the dataset $\mathcal{D}$ is generated according to an observation model. The prior encodes which target functions are likely but need not be known by the training algorithm.

Under squared population error, the Bayes-optimal predictor is the posterior mean

$$f_B(x) = \mathbb{E}[f_\star(x) \mid \mathcal{D}], \qquad \mathcal{E}(f; f_\star) = \frac{1}{2}\,\mathbb{E}_{x\sim\mu}\,|f(x) - f_\star(x)|^2 . \tag{22}$$

Here the conditional expectation is taken under the posterior induced by the source prior P_0 and the observation model, while the expectation in $\mathcal{E}$ is over a fresh input $x \sim \mu$. We ask whether the temporary M-cover fluctuations displace f_c toward f_B.

3.6 Posterior alignment as the generalization criterion

We can now combine the dynamical effect of the M-cover with the source prior introduced in Sec. 3.5. Section 3.4 gives the predictor displacement $\Delta f(T)$ produced by the transient cover fluctuations. The posterior mean f_B determines whether this displacement is favorable for generalization.

We define the posterior-averaged population error as $\overline{\mathcal{E}}(f) = \mathbb{E}[\mathcal{E}(f) \mid \mathcal{D}]$, where the expectation is over the posterior distribution of the unknown target function $f_\star$, conditioned on the observed dataset $\mathcal{D}$. For squared population error, f_B minimizes $\overline{\mathcal{E}}(f)$.

The difference in posterior-averaged population error is

$$\overline{\mathcal{E}}\big(f_{c_M(T)}\big) - \overline{\mathcal{E}}\big(f_{c_0(T)}\big) = - \underbrace{\big\langle f_B - f_{c_0(T)}, \Delta f(T)\big\rangle_\mu}_{\text{alignment with the posterior-preferred direction}} + \underbrace{\frac{1}{2}\|\Delta f(T)\|_\mu^2}_{\text{overshoot}} . \tag{23}$$

Here $\langle f, g\rangle_\mu = \mathbb{E}_{x\sim\mu}\left[f(x)^\top g(x)\right]$.

The first term measures whether the M-cover displacement points from the base predictor toward the posterior mean predictor, while the second penalizes a displacement that is too large.

Posterior-alignment criterion: For sufficiently small initial cover fluctuations, $\Delta f(T) = O(\epsilon^2)$. If its leading-order component satisfies

$$\big\langle f_B - f_{c_0(T)}, \Delta f(T)\big\rangle_\mu > 0 \tag{24}$$

at order $O(\epsilon^2)$, then for sufficiently small nonzero ϵ the M-cover has lower posterior-averaged population error than the corresponding base predictor. Here $\Delta f(T)$ is the M-cover-induced predictor displacement defined in Eqs. (19)–(21).

Appendix C derives an endpoint-local closed form: when transverse fluctuations are introduced near a stationary point of the base dynamics, integrating their covariance dynamics yields Lyapunov (symmetric Sylvester) equations for their lifetime-integrated covariances, yielding an explicit predictor displacement and posterior-alignment condition. In the next section, Sec. 4, we introduce an analytically tractable random-feature model, evaluate this criterion explicitly, and identify a finite region of M-cover parameters for which the induced displacement improves generalization. Appendix D specializes the posterior-alignment theory to the factorized random-feature model.

4 An analytical simplified model

4.1 Random features followed by two trainable linear layers

We specialize the general theory to a network that maps each input x through a frozen random-feature dictionary of size P, shared across replicas, followed by two trainable linear layers, with predictor $f_{A,B}(x) = BAz_P(x)$ and loss $\mathcal{L}(A, B) = \frac{1}{2D}\,\|BAZ - Y\|_F^2$. This retains a nontrivial representation spectrum while making the cover-induced response analytically tractable:

$$z_P(x) = \frac{1}{\sqrt{P}}\big(\psi_{\omega_1}(x), \ldots, \psi_{\omega_P}(x)\big)^\top, \qquad \Lambda_P = \mathbb{E}_x[z_P(x) z_P(x)^\top]. \tag{25}$$

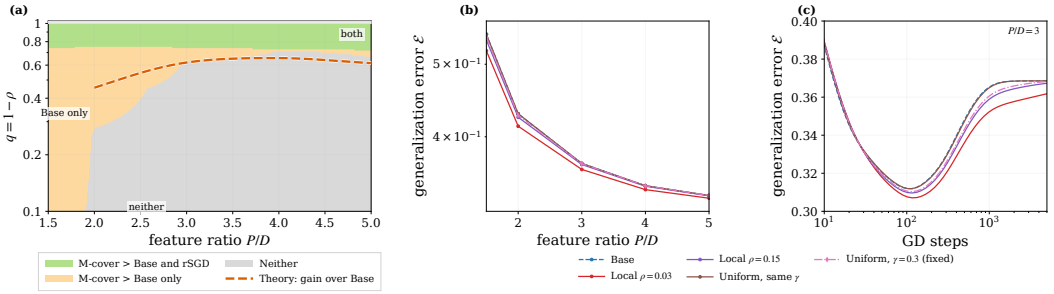


Figure 2: **Phase structure and generalization of the local M-cover in the random-feature model.** **(a)** Monte Carlo phase portrait in $\phi = P/D$ and $q = 1 - \rho$ for replicas initialized as zero-mean perturbations of a common random network, with fixed $\gamma = 2$. Colors indicate whether the M-cover outperforms the single-model Base, the fixed uniform reference ($\rho = 1, \gamma = 0.3$), both, or neither. The orange dashed curve is the leading-order finite-time posterior-alignment prediction for the Base-relative zero-gain boundary, computed from Eqs. (21) and (73). **(b)** Final population generalization error $\mathcal{E}$ versus P/D for the Base, representative structured M-covers, the uniform endpoint at $\gamma = 2$, and the fixed uniform reference. **(c)** Corresponding learning curves at $P/D = 3$, shown from GD step 10 and zoomed near the minimum to resolve differences between methods.

Here $A \in \mathbb{R}^{m\times P}$ and $B \in \mathbb{R}^{d_y\times m}$ are trainable, while Z and Y collect the features and targets of the D training examples. Although the predictor is linear in z_P, its factorized dynamics are nonlinear (Fukumizu, 1998; Saxe et al., 2014; Varre et al., 2023). The width P determines the represented function space, whereas m controls its factorization; thus, $\phi = P/D$ is the feature-to-sample ratio governing interpolation and overparameterization.

4.2 Gaussian source prior and spectral mismatch

We expand the target in the population representation modes:

$$f_{\star,o}(x) = \sum_j v_{oj}\varphi_j(x), \qquad v_{oj} \sim \mathcal{N}(0, p_j), \qquad p_j = \mathbb{E}[v_{oj}^2]. \tag{26}$$

Here, λ_j is the eigenvalue of the population feature covariance $\Lambda_P = \mathbb{E}_{x\sim\mu}[z_P(x)z_P(x)^\top]$ corresponding to the representation mode φ_j[1], while p_j is the expected target power carried by that mode. We choose

$$\lambda_j = \lambda_0 j^{-b}, \qquad p_j = C_a j^{-a}, \qquad s = \frac{a-1}{b}, \qquad \sum_{j:\lambda_j \lesssim u} p_j \asymp u^s, \tag{27}$$

with $a, b > 1$ and C_a chosen to keep the total target power fixed. Smaller s therefore places more target power in weakly represented modes.

4.3 Synthetic verification

We test the predicted generalization transition of the analytical model against Monte Carlo simulations. We arrange the M replicas on a ring and mix their fluctuations using $T = \exp[-t_\rho \bar{Q}]$, where $\bar{Q}$ is the normalized ring Laplacian and $t_\rho = -\log(1-\rho)$, so that ρ interpolates from no mixing at $\rho = 0$ to uniform averaging as $\rho \to 1$. Replicas are initialized as zero-mean perturbations of a common random network and trained at fixed structured-coupling strength $\gamma = 2$. We vary the feature-to-sample ratio $\phi = P/D$ and measure generalization by the population half-MSE $\mathcal{E}_\rho = \frac{1}{2d_y}\mathbb{E}_{x\sim\mu}\left[\|f_\rho(x) - f_\star(x)\|_2^2\right]$. In Fig. 2a, colors indicate whether Monte Carlo simulations show the M-cover improving over the single-model Base, the fixed uniform reference, both, or neither. The finite-time response in Eq. (21), combined with the posterior-risk identity in Eq. (23) and

[1]More precisely, if $\Lambda_P u_j = \lambda_j u_j$, then $\varphi_j(x) = \lambda_j^{-1/2} u_j^\top z_P(x)$; equivalently, φ_j is an eigenfunction with eigenvalue λ_j of the population kernel operator associated with $K_P(x, x') = z_P(x)^\top z_P(x')$.

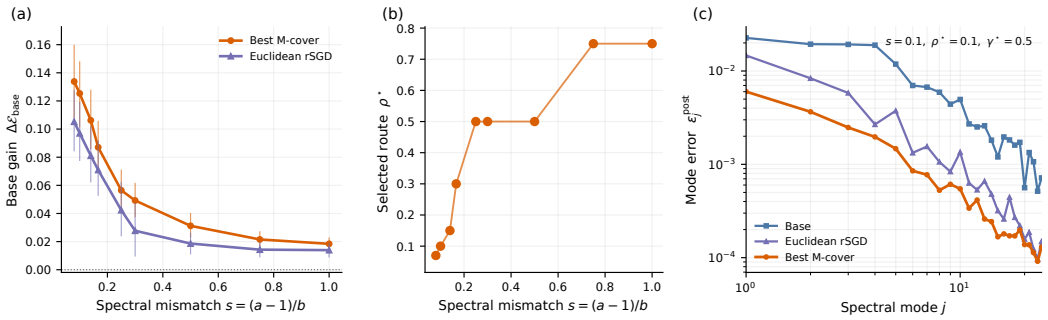


Figure 3: **Spectral mismatch selects the M-cover route and reshapes the learned spectrum. (a)** Fresh-task population gain over the single-network Base, $\Delta\mathcal{E}_{\text{base}} \equiv \mathcal{E}_{\text{Base}} - \mathcal{E}_m$, for the validation-selected M-cover pair $(\rho^\star, \gamma^\star)$ and independently tuned Euclidean replicated SGD; positive values indicate improvement over Base. The M-cover selection uses validation data only and includes the uniform endpoint $\rho = 1$ in the same candidate family; no separate uniform curve is shown. **(b)** Selected mixing $\rho^\star$ versus source mismatch $s = (a-1)/b$; marker color denotes the selected coupling strength $\gamma^\star$. **(c)** Mode-resolved error to the Bayes posterior mean, $\varepsilon_j^{\text{post}} \equiv \|\hat{f}_j - f_{B,j}\|^2/(2d_y)$, at a representative high-mismatch source ($s = 0.1$). Lower values indicate closer agreement with the Bayes posterior spectrum. Errors are measured in the orthonormal function basis and are not normalized by the Bayes mode power.

its coefficient-space form in Eq. (73), gives a leading-order prediction for the Base-relative zero-gain boundary, $\mathcal{E}_{\text{Base}} = \mathcal{E}_M$. Appendix Fig. 5 shows the relation between γ and ρ at the predicted boundary of zero generalization gain over the single-model Base.

We next examine how source–representation mismatch affects optimal M-cover coupling. We fix the feature spectrum $\lambda_j \propto j^{-b}$ and vary the source prior $p_j \propto j^{-a}$ through $s = (a-1)/b$. For each source condition we jointly select the available M-cover pair (ρ, γ) using held-out validation data. Without knowing the prior, an interior ρ can already help; optimizing it requires tuning. Figure 3 measures predictor error relative to the Bayes posterior mean. As the source–representation mismatch increases ($s \to 0$), the optimal mixing parameter ρ^* moves toward smaller, but nonzero, values, while the optimized M-cover achieves increasingly larger gains over both the base model and Euclidean rSGD through closer mode-by-mode agreement with the Bayes posterior predictor.

5 Empirical Benchmarks

Controlled MNIST benchmark. We train $M = 4$ replicas of a one-hidden-layer ReLU classifier, varying the declared load P/D through the hidden width. The replicas retain their native task losses and are coupled by the ring M-cover operator introduced above.

We examine the coupling scale γ in two complementary settings (Fig. 4). First, an annealed-γ sweep at $P/D = 2$ shows that validation performance depends jointly on ρ and γ, with several interior ρ values outperforming the independently tuned rSGD endpoint. Second, an adaptive controller selects the target contraction time τ on validation data and freezes the resulting policy before evaluation on fresh seeds for both M-cover and rSGD. Across $P/D = 2, 3, 4$, the structured M-cover consistently improves over rSGD. Experiments up to $M = 10$ further show increasing gains and a shift of $\rho^\star$ toward more local coupling (Appendix G). Appendix F further examines how changing the mixing topology beyond the ring-diffusion family affects validation performance.

Cross-benchmark comparison. Table 1 summarizes results across image classification, language modeling, physics-informed PDE solving, and time-series forecasting. The M-cover outperforms the single-model Base and replica-based baselines (rSGD and Elastic SGD) in all reported comparisons. SAM shows task-dependent behavior, yielding substantial gains in some settings but degrading performance in others (Singh et al., 2025; Kim et al., 2023). Where SAM is beneficial, combining it with the M-cover yields further improvements; where SAM is detrimental, the M-cover alone still improves on Base. See the appendix for experimental details and additional results.

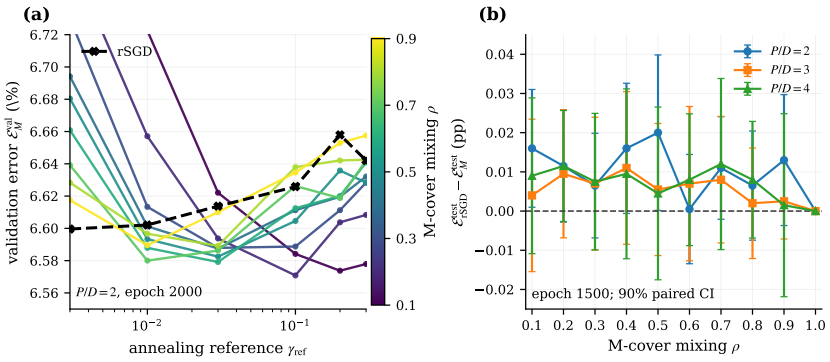


Figure 4: **MNIST: structured transverse mixing relative to replicated SGD. (a)** Classification error for the annealed-γ local M-cover at $P/D = 2$ after 2000 epochs. Curves correspond to interior ring-mixing parameters $0 < \rho < 1$; rSGD is the uniform-mixing endpoint $\rho = 1$. **(b)** Adaptive-γ benchmark at epoch 1500 for $P/D \in \{2, 3, 4\}$. Points show the paired test-error gain $\mathcal{E}^{\text{test}}_{\text{rSGD}} - \mathcal{E}^{\text{test}}_{M}$; positive values favor the M-cover. Error bars are 90% paired Student-t confidence intervals over 20 confirmation seeds, and $\rho = 1$ coincides with rSGD. Both experiments use $M = 4$ replicas. See Appendix E.1 for experimental and hyperparameter-selection details.

Table 1: **Benchmark comparison.** Mean test performance after validation-based selection or evaluation following prior work. Lower is better for all reported metrics. See Appendix E for details.

Benchmark	Setting	Metric	Base	rSGD	Elastic SGD	SAM	M-cover	SAM + M-cover
MNIST	MLP-1H, $P/D = 4$, $M = 4$	Err. (%)	5.3515	5.3350	5.3950	5.0175	5.3275	**4.9895**
Fashion-MNIST	MLP-2H, $P/D = 11.16$, $M = 5$	Err. (%)	9.874	9.612	9.588	9.574	9.441	**9.342**
TinyStories	8-layer Transformer, $M = 6$	PPL	7.6951	7.6301	7.6020	7.8553	**7.4613**	7.5994

Benchmark	Setting	Metric	Base	rSGD	M-cover
CIFAR-10	ResNet-18, $M = 6$	Err. (%)	6.168	5.683	**5.545**
Tiny ImageNet	ResNet-18, $M = 6$	Err. (%)	42.499	42.503	**42.427**
PINNs	MLP-3H, Fourier features, $M = 4$	Rel. L_2	2.17×10^{-6}	2.19×10^{-6}	$\mathbf{1.64 \times 10^{-6}}$
ETTm2	CycleNet/MLP, $M = 4$	MSE	0.163890	0.161841	**0.161600**

6 Discussion

Our results show that transient replica fluctuations can improve generalization when their induced predictor displacement aligns with the correction preferred by the source posterior. Future work should examine whether transverse replica modes provide temporary escape directions and how these relate to saddle avoidance, flatness, and local-entropy approaches (Dauphin et al., 2014; Hochreiter & Schmidhuber, 1997; Baldassi et al., 2016; Chaudhari et al., 2019; Pittorino et al., 2021).

Structured mixing improves the generalization of the final deployed model over matched uniform rSGD while still contracting to a single network. Moreover, the local coupling used in our benchmarks can in principle reduce replica-coupling costs relative to global rSGD interactions. A separate question is whether temporary replication is a more effective use of training-time overparameterization than conventional widening followed by pruning to the same deployment size, at matched training compute. Future work should also derive non-replicated effective dynamics by integrating out transverse modes, extend the local tangent construction to sampled routed permutations, and adaptively learn richer layer- and time-dependent mixing spectra beyond our single-parameter family.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish writing, including improvements to grammar, spelling, clarity, and presentation, and for retrieval and discovery of related literature. We also used generative AI tools as research assistants to check and discuss mathematical derivations, assist in preparing dataset and benchmark setups, support the implementation and verification of experimental pipelines, and help organize and inspect data produced by the benchmarks.

All AI-assisted mathematical work was independently checked by the authors. In particular, mathematical derivations, identities, and theoretical claims appearing in the paper were reviewed and verified by the authors rather than accepted on the basis of AI-generated output. AI-assisted literature searches were followed by examination of the original sources, and references were included only after verification of their relevance and bibliographic information. AI-assisted code, dataset preparation, benchmark configurations, and data-processing procedures were inspected and tested by the authors, and the reported experimental results were obtained from the authors' own computational experiments. Generative AI was not used to fabricate experimental results or citations.

We have reviewed all AI-assisted work and take responsibility for the final content of this work, including all text, mathematical claims, code, experimental procedures, results, and other artifacts produced with the aid of generative AI.

REFERENCES

Ada Altieri, Maria Chiara Angelini, Carlo Lucibello, Giorgio Parisi, Federico Ricci-Tersenghi, and Tommaso Rizzo. Loop expansion around the bethe approximation through the m-layer construction. *Journal of Statistical Mechanics: Theory and Experiment*, 2017(11):113303, 2017.

Maria Chiara Angelini and Federico Ricci-Tersenghi. Monte carlo algorithms are very effective in finding the largest independent set in sparse random graphs. *Physical Review E*, 100(1):013302, 2019.

Carlo Baldassi, Christian Borgs, Jennifer T Chayes, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes. *Proceedings of the National Academy of Sciences*, 113(48):E7655–E7662, 2016.

Carlo Baldassi, Clarissa Lauditi, Enrico M Malatesta, Rosalba Pacelli, Gabriele Perugini, and Riccardo Zecchina. Learning through atypical phase transitions in overparameterized neural networks. *Physical Review E*, 106(1):014116, 2022.

Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Blake Bordelon, Abdulkadir Canatar, and Cengiz Pehlevan. Spectrum dependent learning curves in kernel regression and wide neural networks. In *International Conference on Machine Learning*, pp. 1024–1034. PMLR, 2020.

Blake Bordelon, Alexander Atanasov, and Cengiz Pehlevan. A dynamical model of neural scaling laws. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Abdulkadir Canatar, Blake Bordelon, and Cengiz Pehlevan. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021.

Giovanni Catania, Aurélien Decelle, and Beatriz Seoane. Copycat perceptron: Smashing barriers through collective learning. *Physical Review E*, 109(6):065313, 2024.

Pratik Chaudhari, Carlo Baldassi, Riccardo Zecchina, Stefano Soatto, Ameet Talwalkar, and Adam Oberman. Parle: parallelizing stochastic gradient descent. *arXiv preprint arXiv:1707.00424*, 2017.

Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: biasing gradient descent into wide valleys*. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, dec 2019. doi: 10.1088/1742-5468/ab39d9. URL https://doi.org/10.1088/1742-5468/ab39d9.

Jung Min Choi, Vijaya Krishna Yalavarthi, and Lars Schmidt-Thieme. Hpmixer: Hierarchical patching for multivariate time series forecasting. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 42–53. Springer, 2026.

Yann N. Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pp. 2933–2941, Cambridge, MA, USA, 2014. MIT Press.

Kenji Fukumizu. Effect of batch learning in multilayer neural networks. In *Proceedings of the Fifth International Conference on Neural Information Processing (ICONIP '98)*, pp. 67–70, 1998. URL https://www.ism.ac.jp/~fukumizu/papers/research.html.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

Michael I. Jordan. Graphical models. *Statistical Science*, 19(1):140–155, 2004. ISSN 08834237. URL http://www.jstor.org/stable/4144379.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Hoki Kim, Jinseong Park, Yujin Choi, and Jaewook Lee. Stability analysis of sharpness-aware minimization. *arXiv preprint arXiv:2301.06308*, 2023.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized sgd with changing topology and local updates. In *International conference on machine learning*, pp. 5381–5393. PMLR, 2020.

Timothee Leleu, Sam Reifenstein, Atsushi Yamamura, and Surya Ganguli. Reshaping global loop structure to accelerate local optimization by smoothing rugged landscapes. *arXiv preprint arXiv:2602.01490*, 2026.

Shengsheng Lin, Weiwei Lin, Xinyi Hu, Wentai Wu, Ruichao Mo, and Haocheng Zhong. Cyclenet: Enhancing time series forecasting through modeling periodic patterns. *Advances in Neural Information Processing Systems*, 37:106315–106345, 2024.

Carlo Lucibello, Flaviano Morone, Giorgio Parisi, Federico Ricci-Tersenghi, and Tommaso Rizzo. Finite-size corrections to disordered ising models on random regular graphs. *Physical Review E*, 90(1):012146, 2014.

Fabrizio Pittorino, Carlo Lucibello, Christoph Feinauer, Gabriele Perugini, Carlo Baldassi, Elizaveta Demyanenko, and Riccardo Zecchina. Entropic gradient descent algorithms and wide flat minima. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124015, 2021.

Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Representations*, 2014. URL https://arxiv.org/abs/1312.6120.

Sidak Pal Singh, Hossein Mobahi, Atish Agarwala, and Yann Dauphin. Avoiding spurious sharpness minimization broadens applicability of sam. *arXiv preprint arXiv:2502.02407*, 2025.

Peter Sollich and Anason Halees. Learning curves for gaussian process regression: Approximations and bounds. *Neural computation*, 14(6):1393–1428, 2002.

Juan Diego Toscano, Daniel T Chen, Vivek Ooomen, Jérôme Darbon, and George Em Karniadakis. A variational framework for residual-based adaptivity in neural pde solvers and operator learning. *NPJ Artificial Intelligence*, 2(1):32, 2026.

Aditya Vardhan Varre, Maria-Luiza Vladarean, Loucas Pillaud-Vivien, and Nicolas Flammarion. On the spectral bias of two-layer linear networks. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/cad2fd66cf88226d868f90a7cbaa4a53-Abstract-Conference.html.

Pascal O Vontobel. Counting in graph covers: A combinatorial characterization of the bethe entropy function. *IEEE Transactions on Information Theory*, 59(9):6018–6048, 2013.

Jianyu Wang and Gauri Joshi. Cooperative sgd: A unified framework for the design and analysis of local-update sgd algorithms. *Journal of Machine Learning Research*, 22(213):1–50, 2021.

Sifan Wang, Bowen Li, Yuhan Chen, and Paris Perdikaris. Piratenets: physics-informed deep learning with residual adaptive networks. *J. Mach. Learn. Res.*, 25(1), January 2024. ISSN 1532-4435.

Sifan Wang, Bowen Li, Paris Perdikaris, et al. Gradient alignment in physics-informed neural networks: A second-order optimization perspective. *Advances in Neural Information Processing Systems*, 38:168482–168532, 2026.

Boya Zhang, Shuaijie Yin, Huiwen Zhu, and Xing He. Freqcycle: A multi-scale time-frequency analysis method for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 28159–28167, 2026.

Sixin Zhang, Anna Choromanska, and Yann LeCun. Deep learning with elastic averaging sgd. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pp. 685–693, Cambridge, MA, USA, 2015. MIT Press.

Tongtian Zhu, Fengxiang He, Lan Zhang, Zhengyang Niu, Mingli Song, and Dacheng Tao. Topology-aware generalization of decentralized sgd. In *International Conference on Machine Learning*, pp. 27479–27503. PMLR, 2022.

A Derivation of the reduced center–fluctuation flow

We derive Eq. (17) from the local M-cover objective. Assume that $\mathcal{L}$ is sufficiently smooth and use the replica gradient flow

$$\dot{\theta}^{(r)} = -M\eta\, \nabla_{\theta^{(r)}} \mathcal{L}_M. \tag{28}$$

The factor M compensates for the replica average in Eq. (7), so that the synchronized manifold follows $\dot{c} = -\eta \nabla \mathcal{L}(c)$.

The normalization of the mixing modes gives the inverse relations

$$c = \frac{1}{M}\sum_r \theta^{(r)}, \qquad h_q = \frac{1}{M}\sum_r e_q(r)\theta^{(r)}. \tag{29}$$

Consequently, the chain rule and Eq. (28) yield

$$\dot{c} = -\eta \nabla_c \mathcal{L}_M, \qquad \dot{h}_q = -\eta \nabla_{h_q} \mathcal{L}_M. \tag{30}$$

We now expand the native part of the objective around the synchronized manifold. Writing $H(c) = \nabla^2 \mathcal{L}(c)$ and using $\sum_r \delta^{(r)} = 0$,

$$\begin{aligned} \frac{1}{M}\sum_r \mathcal{L}(c+\delta^{(r)}) &= \mathcal{L}(c) + \frac{1}{2M}\sum_r \delta^{(r)\top} H(c)\delta^{(r)} + O(\|h\|^3) \\ &= \mathcal{L}(c) + \frac{1}{2}\sum_q h_q^\top H(c) h_q + O(\|h\|^3), \end{aligned} \tag{31}$$

where $\|h\|^2 = \sum_q \|h_q\|^2$. The linear term vanishes by the zero-mean constraint, while the second equality follows from $M^{-1}\sum_r e_q(r)e_p(r) = \delta_{qp}$.

Let $h_{\ell q}$ denote the component of h_q at layer ℓ. Substituting $\delta_\ell = \sum_q e_q \otimes h_{\ell q}$ into Eq. (10) gives

$$\begin{aligned} R(c,h) &= \frac{1}{2M}\sum_{\ell,q,p} \gamma_\ell (e_q^\top L_\ell e_p)\, h_{\ell q}^\top G_\ell(c) h_{\ell p} \\ &= \frac{1}{2}\sum_q h_q^\top \operatorname{diag}_\ell\left[\gamma_\ell \kappa_{\ell q} G_\ell(c)\right] h_q, \end{aligned} \tag{32}$$

because $L_\ell e_p = \kappa_{\ell p} e_p$ and $e_q^\top e_p = M\delta_{qp}$. Combining Eqs. (31) and (32) therefore gives

$$\mathcal{L}_M(c,h) = \mathcal{L}(c) + \frac{1}{2}\sum_q h_q^\top \mathbb{H}_q(c) h_q + O(\|h\|^3), \tag{33}$$

with $\mathbb{H}_q(c)$ defined in Eq. (15).

Differentiating with respect to h_q and c gives

$$\nabla_{h_q}\mathcal{L}_M = \mathbb{H}_q(c)h_q + O(\|h\|^2), \tag{34}$$

$$\nabla_c \mathcal{L}_M = \nabla\mathcal{L}(c) + \frac{1}{2}\sum_q \nabla_c\left[h_q^\top \mathbb{H}_q(c) h_q\right] + O(\|h\|^3). \tag{35}$$

Here the modes are held fixed when taking the center derivative. Defining

$$\mathbb{N}_q(c)[u,v] = \frac{1}{2}\nabla_c\left[u^\top \mathbb{H}_q(c) v\right]$$

and substituting these gradients into Eq. (30) gives Eq. (17).

B Derivation of the accumulated center response

We derive Eq. (21) perturbatively from the reduced flow in Eq. (17). Let the base trajectory satisfy

$$\dot{c}_0(t) = -\eta \nabla \mathcal{L}(c_0(t)), \tag{36}$$

and assume that $c_M(0) = c_0(0)$ and $h_q(0) = \epsilon v_q^0$. Because the transverse force on the center is quadratic, we expand

$$h_q(t) = \epsilon u_q(t) + O(\epsilon^2), \qquad c_M(t) = c_0(t) + \epsilon^2 \xi(t) + O(\epsilon^3), \tag{37}$$

with $u_q(0) = v_q^0$ and $\xi(0) = 0$.

At order $O(\epsilon)$, the transverse equation in Eq. (17) gives

$$\dot{u}_q(t) = -\eta \, \mathbb{H}_q(c_0(t)) u_q(t). \tag{38}$$

The dependence of $\mathbb{H}_q$ on the center may be evaluated along $c_0(t)$ because $c_M(t) - c_0(t) = O(\epsilon^2)$, which affects h_q only at order $O(\epsilon^3)$. By definition of the transverse propagator,

$$u_q(t) = U_q(t, 0) v_q^0. \tag{39}$$

To obtain the center response, expand the two terms in its reduced flow:

$$\nabla \mathcal{L}(c_M) = \nabla \mathcal{L}(c_0) + \epsilon^2 H(c_0) \xi + O(\epsilon^3), \tag{40}$$

$$\mathbb{N}_q(c_M)[h_q, h_q] = \epsilon^2 \mathbb{N}_q(c_0)[u_q, u_q] + O(\epsilon^3). \tag{41}$$

Subtracting the base equation and collecting the terms of order $O(\epsilon^2)$ yields the forced linear equation

$$\dot{\xi}(t) = -\eta H(c_0(t)) \xi(t) - \eta \sum_q \mathbb{N}_q(c_0(t))[u_q(t), u_q(t)], \qquad \xi(0) = 0. \tag{42}$$

Let $\Phi(t, u)$ be the propagator of the homogeneous center equation, $\partial_t \Phi(t, u) = -\eta H(c_0(t)) \Phi(t, u)$ with $\Phi(u, u) = I$. Applying the variation-of-constants formula to Eq. (42) gives

$$\xi(T) = -\eta \sum_q \int_0^T \Phi(T, t) \, \mathbb{N}_q(c_0(t)) \, [u_q(t), u_q(t)] \, \mathrm{d}t. \tag{43}$$

Finally, substituting $u_q(t) = U_q(t, 0) v_q^0$ from Eq. (39) gives

$$\xi(T) = -\eta \sum_q \int_0^T \Phi(T, t) \, \mathbb{N}_q(c_0(t)) \, \left[U_q(t, 0) v_q^0, U_q(t, 0) v_q^0\right] \, \mathrm{d}t, \tag{44}$$

which is Eq. (21).

C AN ENDPOINT-LOCAL SYLVESTER CLOSURE

The general response in Eq. (21) depends on the complete base trajectory. We now derive an endpoint-local closed form by introducing the transverse fluctuations at a stationary point $c_\star$ satisfying $\nabla \mathcal{L}(c_\star) = 0$. The formulas are exact for the resulting frozen linearized dynamics and provide the leading local approximation when the fluctuations are introduced sufficiently close to $c_\star$.

Let

$$H_0 = H(c_\star) \succeq 0, \qquad \mathbb{H}_q^\star = \mathbb{H}_q(c_\star) \succ 0. \tag{45}$$

We assume that $\ker H_0$ is tangent to a smooth local manifold of stationary solutions. We also assume a locally normally attracting stationary manifold, with positive normal and excited transverse spectral gaps, and that sufficiently small perturbed trajectories remain in this neighborhood. The endpoint expansion below further assumes uniform control of the perturbative remainder as $t \to \infty$. Without this uniform control, the formulas describe the endpoint of the frozen leading-order response equation. The base trajectory is then constant, $c_0(t) = c_\star$, and the propagators in Eq. (21) reduce to

$$U_q(t, 0) = e^{-\eta \mathbb{H}_q^\star t}, \qquad \Phi(t, u) = e^{-\eta H_0 (t - u)}. \tag{46}$$

Assume that the initial transverse directions have zero mean and covariance

$$\Sigma_q = \mathbb{E}\left[v_q^0 v_q^{0\top}\right], \tag{47}$$

so that the covariance of $h_q(0) = \epsilon v_q^0$ is $\epsilon^2 \Sigma_q$. The leading transverse solution is

$$u_q(t) = e^{-\eta \mathbb{H}_q^\star t} v_q^0, \tag{48}$$

with covariance

$$C_q(t) = \mathbb{E}\big[u_q(t) u_q(t)^\top\big] = e^{-\eta \mathbb{H}_q^\star t} \Sigma_q e^{-\eta \mathbb{H}_q^\star t}. \tag{49}$$

Its lifetime-integrated covariance is

$$X_q = \eta \int_0^\infty C_q(t)\, \mathrm{d}t. \tag{50}$$

Because $\mathbb{H}_q^\star \succ 0$, the integral converges. Differentiating $C_q(t)$ and integrating over time shows that X_q is the unique solution of

$$\mathbb{H}_q^\star X_q + X_q \mathbb{H}_q^\star = \Sigma_q. \tag{51}$$

This is a Lyapunov equation, or equivalently a symmetric Sylvester equation. If $\mathbb{H}_q^\star = V_q \operatorname{diag}(\mu_{qi}) V_q^\top$, then

$$\big(V_q^\top X_q V_q\big)_{ij} = \frac{\big(V_q^\top \Sigma_q V_q\big)_{ij}}{\mu_{qi} + \mu_{qj}}, \tag{52}$$

which makes the dependence on the transverse lifetimes explicit.

Let $d_c = \dim(c)$ and define the endpoint force vector $g_q^\star \in \mathbb{R}^{d_c}$ componentwise by

$$\big(g_q^\star\big)_a = \frac{1}{2} \operatorname{Tr}\left[\left. \frac{\partial \mathbb{H}_q(c)}{\partial c_a} \right|_{c=c_\star} X_q \right], \qquad a = 1, \dots, d_c. \tag{53}$$

Indeed, using the definition of $\mathbb{N}_q$ gives

$$\eta \int_0^\infty \mathbb{E}[\mathbb{N}_q(c_\star)[u_q(t), u_q(t)]]\, \mathrm{d}t = g_q^\star. \tag{54}$$

The center propagator satisfies $e^{-\eta H_0 t} \to P_0$ as $t \to \infty$, where

$$P_0 = I - H_0^\dagger H_0 \tag{55}$$

is the orthogonal projector onto $\ker H_0$. Thus, stable center directions relax after the transverse force vanishes, whereas displacements tangent to the stationary manifold persist. Taking the infinite-time limit of Eq. (21) therefore gives

$$\mathbb{E}[\xi_\infty] = -P_0 \sum_q g_q^\star. \tag{56}$$

Consequently, the persistent mean predictor displacement is

$$\mathbb{E}[\Delta f_\infty] = -\epsilon^2\, D_c f_c|_{c=c_\star} \left[P_0 \sum_q g_q^\star \right] + O(\epsilon^3). \tag{57}$$

A persistent predictor change at this order requires $D_c f_{c_\star} P_0 \neq 0$. In particular, the response vanishes if $H_0 \succ 0$ or if all center-null directions are pure reparameterizations. This endpoint statement is distinct from the finite-time response in Eq. (21).

Substitution into the posterior-alignment criterion gives the endpoint-local condition

$$-\left\langle f_B - f_{c_\star},\, D_c f_c|_{c=c_\star} \left[P_0 \sum_q g_q^\star \right] \right\rangle_\mu > 0. \tag{58}$$

Hence the Sylvester solution X_q encodes the lifetime of each transverse mode, while the center derivatives of $\mathbb{H}_q$ determine the direction of its persistent force.

D COEFFICIENT-SPACE POSTERIOR ALIGNMENT

We now specialize the endpoint-local response to the factorized random-feature model. For the two-factor model, define

$$\widehat{C} = \frac{ZZ^\top}{D}, \qquad \widehat{R} = (BAZ - Y)\frac{Z^\top}{D}, \qquad \alpha_q = \gamma_A \kappa_{Aq}, \quad \beta_q = \gamma_B \kappa_{Bq}.$$

For fluctuations (U, V) of (A, B), the local metrics act as $G_A[U] = U\widehat{C}$ and $G_B[V] = VA\widehat{C}A^\top$. The transverse operator is

$$\big(\mathbb{H}_q[U,V]\big)_A = (B^\top B + \alpha_q I_m)U\widehat{C} + B^\top VA\widehat{C} + V^\top \widehat{R}, \tag{59}$$

$$\big(\mathbb{H}_q[U,V]\big)_B = BU\widehat{C}A^\top + \widehat{R}U^\top + (1+\beta_q)VA\widehat{C}A^\top. \tag{60}$$

The center force follows by differentiating this operator as in Eq. (17); the dependence of G_B on A is differentiated as well.

Under the Gaussian source prior and observation model

$$y = f_\star(x) + \xi, \qquad \xi \sim \mathcal{N}(0, \sigma_\xi^2 I), \tag{61}$$

the posterior mean predictor is

$$f_B(x) = Y\left[K_\star(X,X) + \sigma_\xi^2 I_D\right]^{-1} k_\star(X,x), \qquad k_\star(x,x') = \sum_j p_j \varphi_j(x)\varphi_j(x'). \tag{62}$$

Because the finite model represents only predictors of the form $\Theta z_P(x)$, only the population-orthogonal projection of f_B onto this space enters the alignment condition. Its coefficient matrix is

$$\Theta_B = \mathbb{E}_x\left[f_B(x) z_P(x)^\top\right]\Lambda_P^\dagger, \qquad f_{B,P}(x) = \Theta_B z_P(x). \tag{63}$$

Let $c_\star = (A_\star, B_\star)$ be a stationary synchronized endpoint and define the corresponding coefficient predictor

$$\Theta_0 = \mathcal{C}_\Theta(c_\star) = B_\star A_\star, \qquad \mathcal{C}_\Theta(A,B) = BA. \tag{64}$$

We use Θ_0, rather than Θ_1, to avoid confusion with the uniform-mixing endpoint $\rho = 1$. The derivative of the coefficient map is

$$D\mathcal{C}_\Theta(A_\star, B_\star)[U,V] = B_\star U + VA_\star. \tag{65}$$

For a given mixing parameter ρ, let $\mathbb{H}_{q,\rho}(c)$ denote the transverse operator of cover mode q. Assume that the initial covariance is isotropic on a fixed subspace $\mathcal{S}_q$ with orthonormal basis matrix W_q. For each value of ρ considered, assume

$$\Sigma_q = \sigma_q^2 W_q W_q^\top, \qquad (I - W_q W_q^\top)\mathbb{H}_{q,\rho}(c_\star)W_q = 0, \qquad W_q^\top \mathbb{H}_{q,\rho}(c_\star)W_q \succ 0.$$

Thus $\mathcal{S}_q$ is invariant under the frozen transverse dynamics. The Lyapunov equation gives

$$X_{q,\rho} = \frac{\sigma_q^2}{2} W_q \left[W_q^\top \mathbb{H}_{q,\rho}(c_\star)W_q\right]^{-1} W_q^\top. \tag{66}$$

This restriction is important in the overparameterized random-feature model because the empirical first-layer metric has null directions. Writing $\widehat{C} = ZZ^\top/D$, any first-layer direction U satisfying $UZ = 0$ is unpenalized by the empirical first-layer metric.

Using $\partial_c \log\det A = \mathrm{Tr}(A^{-1}\partial_c A)$ in the endpoint response therefore yields

$$\mathbb{E}[\Theta_\rho - \Theta_0] = \epsilon^2 G_\rho + O(\epsilon^3), \tag{67}$$

where

$$G_\rho = -\frac{1}{4} D\mathcal{C}_\Theta(c_\star)\left[P_0\, \nabla_c \left(\sum_q \sigma_q^2 \log\det_{\mathcal{S}_q} \mathbb{H}_{q,\rho}(c)\right)\Big|_{c=c_\star}\right]. \tag{68}$$

Here

$$H_0 = \nabla^2 \mathcal{L}(c_\star), \qquad P_0 = I - H_0^\dagger H_0 \tag{69}$$

projects onto center directions that retain endpoint memory. Here $\log\det_{\mathcal{S}_q} \mathbb{H}_{q,\rho}(c)$ denotes $\log\det[W_q^\top \mathbb{H}_{q,\rho}(c) W_q]$, with W_q held fixed when differentiating. It equals the log product of all positive eigenvalues only when $\mathcal{S}_q$ is the entire positive spectral subspace. Without the invariance and isotropy assumptions, we use the Lyapunov solution in Eq. (51) directly. The center derivative is taken before evaluating at $c_\star$.

Define the coefficient-space alignment score

$$\mathcal{A}_\rho = \mathrm{Tr}\big[(\Theta_B - \Theta_0)\Lambda_P G_\rho^\top\big]. \tag{70}$$

Since

$$\langle (\Theta_B - \Theta_0) z_P, G_\rho z_P \rangle_\mu = \mathcal{A}_\rho, \tag{71}$$

the general posterior-alignment criterion reduces to

$$\mathcal{A}_\rho > 0 \tag{72}$$

when the M-cover is compared with the synchronized base predictor.

For fluctuations introduced along a nonstationary base trajectory, define

$$\Theta_0(T) = \mathcal{C}_\Theta(c_0(T)), \qquad G_\rho(T) = D\mathcal{C}_\Theta(c_0(T))\big[\mathbb{E}_{\mathrm{init}}\xi_\rho(T)\big],$$

where $\xi_\rho(T)$ is given by Eq. (21). Conditional on the base initialization and dataset, the finite-time alignment score is

$$\mathcal{A}_\rho(T) = \mathrm{Tr}\big[(\Theta_B - \Theta_0(T))\Lambda_P G_\rho(T)^\top\big]. \tag{73}$$

For the summed-output risk of Eq. (22),

$$\mathbb{E}_{\mathrm{init}}\left[\overline{\mathcal{E}}(\Theta_\rho(T)) - \overline{\mathcal{E}}(\Theta_0(T))\right] = -\epsilon^2 \mathcal{A}_\rho(T) + O(\epsilon^3).$$

Figure 5 extends the theory calculation in Fig. 2 by varying the coupling strength as well as the mixing parameter. We evaluate Eq. (21) along the same randomly initialized Base trajectories, retaining the original hidden widths and initial replica fluctuations. The leading posterior gain is

$$\mathcal{T}_{\gamma,\rho}(T) = \frac{1}{d_y}\mathrm{Tr}\Big[(\Theta_B - \Theta_0(T))\Lambda_P\big(\Delta\Theta^{(2)}_{\gamma,\rho}(T)\big)^\top\Big], \tag{74}$$

where $\Delta\Theta^{(2)}$ includes the factor ϵ^2 and $\Theta_0(T)$ is the Base predictor at the same horizon. The zero of the task-averaged score defines the predicted Base-relative boundary. This finite-time calculation includes forces accumulated during training; it does not replace the trajectory by a stationary endpoint.

All settings match Fig. 2: $D = 8$, $d_y = 4$, $K = 128$, $M = 5$, $a = 1.25$, $b = 1.5$, $\sigma_\xi = 0.2$, $m = \max(4, \lceil 0.6P \rceil)$, and $\epsilon = 0.3$. We use the same 16 task seeds, random common centers, centered replica perturbations, and 5000 gradient steps of size 0.02, giving $T = 100$. Only the scan over (γ, ρ) is extended. Uniform and structured mixing use the same local functional metrics.

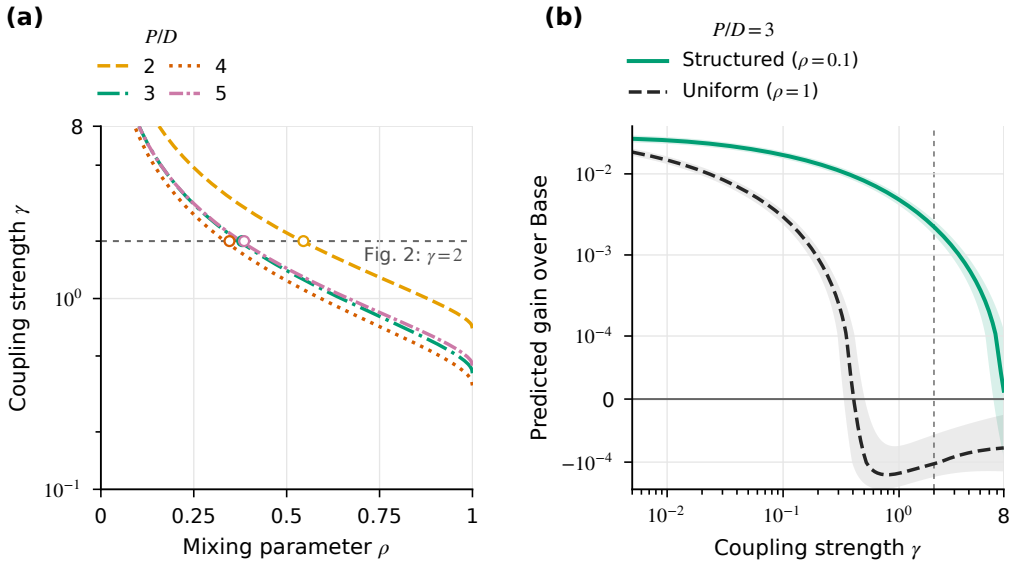


Figure 5: **Finite-time theory. (a)** Zero contours of the task-averaged leading posterior gain over Base for different P/D. The dashed horizontal line marks the original $\gamma = 2$ slice. Open circles mark the crossings obtained from the original Fig. 2 mixing grid; small offsets from the curves reflect the denser interpolation used here. For $P/D = 1.5$, the mean predicted gain remains positive throughout the scanned domain, so no zero contour is shown. **(b)** Predicted gain over Base at $P/D = 3$, comparing structured ($\rho = 0.1$) and uniform ($\rho = 1$) functional coupling at the same γ. Shading shows one standard error across the same 16 paired tasks.

E Experimental setup

E.1 MNIST MLP benchmark

We evaluate the local M-cover on MNIST using a one-hidden-layer 784–m–10 network with ReLU activation and biases. Each run is fitted on $4{,}000$ examples sampled from the training set and evaluated on the official $10{,}000$-example test set. The hidden widths $m = 101$, 151, and 201 give $80{,}305$, $120{,}055$, and $159{,}805$ trainable parameters, respectively. For this multi-output benchmark, the reported load P/D divides the parameter count by the $4{,}000 \times 10$ scalar one-hot targets, giving the nominal ratios $P/D \in \{2, 3, 4\}$.

All methods minimize mean-squared error on one-hot labels for $2{,}000$ epochs using mini-batch SGD with batch size 128, learning rate 0.003, and momentum 0.95. Replicated methods use $M = 4$ copies initialized as zero-mean orthogonal perturbations of a common Xavier-initialized network, with relative perturbation scale $\nu = 0.1$; deployment uses the network obtained by averaging the replica parameters. The M-cover uses the ring-diffusion mixing family with $\rho \in \{0, 0.1, \ldots, 1\}$, where $\rho = 1$ is the uniform rSGD endpoint. Rather than tuning a fixed coupling strength, an adaptive controller updates γ every ten epochs to target a prescribed disagreement e-folding time $\tau \in \{125, 250, 500, 1000, 2000, 4000\}$ epochs. Thus, rSGD selects τ at $\rho = 1$, whereas the structured M-cover jointly selects (ρ, τ).

Hyperparameters are selected using a fixed stratified holdout of $20{,}000$ training images that is excluded from every fitting subset. The full grid is evaluated over 24 discovery seeds, shortlisted candidates receive 24 additional refinement runs, and the final policy is selected from the pooled validation results. The selected policy is then frozen and restarted from scratch on 20 fresh paired confirmation seeds, with matched fitting subsets, initializations, and mini-batch orders across methods. Table 2 reports the resulting mean test classification error at epoch 1500.

Table 2: MNIST confirmation error for rSGD, M-cover, and the single-network Base reference.

P/D	$\mathcal{E}^{\text{test}}_{\text{rSGD}}$	$\mathcal{E}^{\text{test}}_{M}$	$\mathcal{E}^{\text{test}}_{\text{Base}}$
2	6.2245	**6.2130**	6.2735
3	5.5925	**5.5815**	5.6480
4	5.3350	**5.3275**	5.3515

E.2 FASHION-MNIST MLP BENCHMARK

Data and architecture. We use a fully connected 784–512–512–10 network with ReLU hidden activations and biases, containing $P = 669{,}706$ trainable parameters. Images are scaled to $[0, 1]$ and normalized using mean 0.2860 and standard deviation 0.3530, without augmentation. Hyperparameter selection uses a fixed stratified $50{,}000/10{,}000$ training/validation split of the official training set. Final runs use all $D = 60{,}000$ training examples and the official $10{,}000$ test examples, giving $P/D = 11.16$.

Training. All methods use cross-entropy, batch size 128, and 300 epochs of SGD with Nesterov momentum 0.9 and cosine learning-rate decay to zero. The initial learning rate is 0.02 for vanilla SGD and SAM, and 0.05 for replicated SGD (rSGD), Elastic SGD, M-cover, and M-cover + SAM. We use no weight decay, dropout, or batch normalization. Replicated methods use $M = 5$ independently initialized networks with standard PyTorch linear-layer initialization. Each replica processes the full training set through an independently shuffled minibatch stream. Corresponding runs share initializations and data-order seeds across methods.

Replicated SGD and M-cover. Let Θ contain the replica parameters as rows, including all weights and biases. Every tenth minibatch, starting with the first minibatch of each epoch, we apply the coupling of Eq. (10), adding $\gamma(t)K_\rho\Theta$ to the gradients of the summed replica losses, where $K_\rho = I - T_\rho$. For rSGD, we use uniform mixing ($\rho = 1$) as in Sec. 2.3. For M-cover, we use the ring-diffusion mixing family of Sec. 4.3 on the five replicas, with the ring Laplacian divided by its largest eigenvalue. The coupling uses the Euclidean parameter metric.

The rSGD coupling is controlled by a target contraction time τ. For layer ℓ, define

$$E_\ell(t) = \sum_{a=1}^{M} \left\| \theta_\ell^{(a)}(t) - \bar{\theta}_\ell(t) \right\|_2^2, \qquad d(t) = \left[\frac{1}{3} \sum_{\ell=1}^{3} \frac{E_\ell(t)}{E_\ell(0)} \right]^{1/2}.$$

Every ten epochs, we estimate the decay rate of $d(t)$ from its log difference over the preceding interval and smooth this rate with an exponential moving average of weight 0.5. Writing the smoothed rate as $\bar{q}$, the update is

$$\gamma_{\text{next}} = \text{clip}_{[10^{-8},\,1]} \left[\gamma \, 1.1^{\text{clip}_{[-1,\,1]}(1-\tau\bar{q})} \right].$$

We initialize $\gamma = 0.003$ and initialize the moving average with the first observed rate. Disagreement is floored at 10^{-6} when taking logarithms; below this threshold, the controller decreases γ by a factor of 1.1. Feedback uses parameter disagreement only.

We first select τ for rSGD, then tune M-cover using

$$\gamma_M(t) = \frac{c}{\rho} \gamma_{\text{rSGD}}(t).$$

Each M-cover run replays the rSGD schedule from the same seed and data split, without feedback from its own disagreement. The selected values are $\tau = 173.20508$ epochs, $\rho = 0.5$, and $c = 3$, giving $\gamma_M(t) = 6\gamma_{\text{rSGD}}(t)$. Final runs generate fresh rSGD schedules on the full training set.

Elastic SGD. Elastic SGD uses five workers and a persistent center z, initialized to their parameter mean. At the same ten-minibatch exchange cadence, let $d_a = \theta^{(a)} - z$ before the update. The exchange is

$$\theta_{\text{next}}^{(a)} = \theta^{(a)} + \Delta_{\text{SGD}}^{(a)} - \alpha d_a, \qquad z_{\text{next}} = z + \alpha \sum_{a=1}^{M} d_a,$$

Table 3: Fashion-MNIST hyperparameter searches. The M-cover search uses the selected rSGD timescale; the combination search holds all M-cover settings fixed.

Method	Parameter	Coarse grid	Added points
rSGD	τ	$\{30, 100, 300\}$	$\{173.20508, 519.61524\}$
M-cover	(ρ, c)	$\rho \in \{0.25, 0.5, 0.75\}$ $c \in \{1.5, 2, 3\}$	—
Elastic SGD	α	$\{0.02, 0.1, 0.18\}$	$\{0.14, 0.19\}$
SAM	r	$\{0.01, 0.03, 0.05, 0.1, 0.2, 0.5\}$	$\{0.14142136, 0.31622777\}$
M-cover + SAM	r	$\{0, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$	$\{0.14142136, 0.31622777\}$

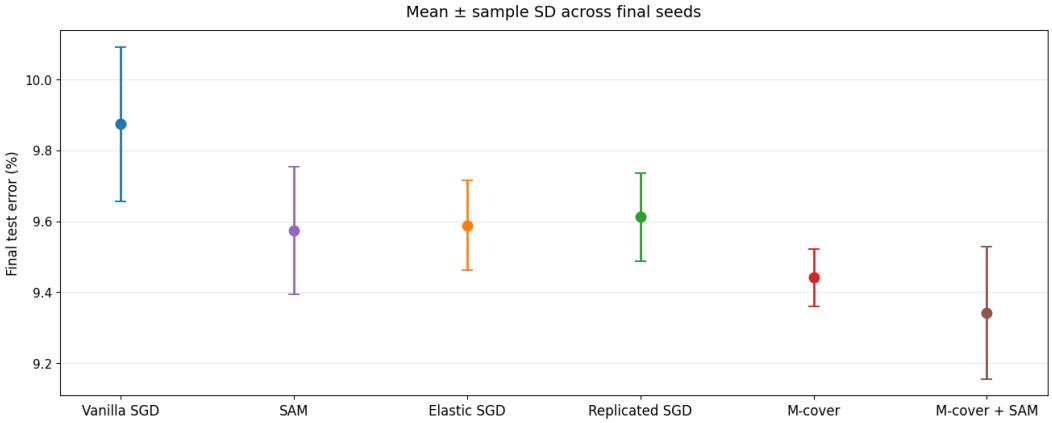


Figure 6: Fashion-MNIST test classification error after 300 epochs. Points show means over ten final seeds; error bars indicate ± 1 sample standard deviation. Hyperparameters are selected using validation data.

where $\Delta_{\mathrm{SGD}}^{(a)}$ is the Nesterov update from the worker's data loss. The exchange is applied separately from momentum. The selected coefficient $\alpha = 0.18$ remains fixed throughout training.

SAM and M-cover + SAM. We use standard Euclidean SAM with perturbation radius r:

$$\epsilon = r \frac{g}{\|g\|_2 + 10^{-12}}, \qquad g = \nabla_\theta \ell_{\mathcal{B}}(\theta).$$

The second gradient is evaluated at $\theta + \epsilon$ on the same minibatch, and the parameters are restored before the SGD update. The norm includes all weights and biases. For M-cover + SAM, SAM is applied separately to each replica's data loss; the coupling gradient is evaluated at the original parameters and added afterward. We freeze the selected M-cover settings and tune only r. Both standalone SAM and M-cover + SAM select $r = 0.2$. SAM uses two data-gradient evaluations per update; the comparison fixes training epochs rather than computational cost.

Hyperparameter selection. The search grids and additional refinement points are listed below. Coarse and refinement runs use seed 2. The best two candidates for each method are additionally evaluated with seeds 7 and 11, and selection minimizes the mean classification error at epoch 300 over all three seeds. Cross-entropy and intermediate checkpoints are not used for selection. For M-cover + SAM, the existing zero-radius M-cover results are also retained as a control.

Evaluation and results. After selection, each method is trained from scratch using ten seeds, 101–110. Test performance is evaluated only at epoch 300. For rSGD and both M-cover variants, inference uses the network obtained by averaging replica parameters; Elastic SGD uses its persistent center. Figure 6 reports mean test classification error and sample standard deviation.

M-cover achieves the lowest mean test error among the individual methods, at $9.441 \pm 0.081\%$. Combining it with SAM further reduces the mean error to $9.342 \pm 0.186\%$.

E.3 CIFAR-10 ResNet benchmark

Data and architecture. We evaluate the local M-cover on CIFAR-10 using a CIFAR-adapted ResNet-18 with $11{,}173{,}962$ trainable parameters. Hyperparameter selection uses a fixed stratified split of the official training set into $45{,}000$ fitting and $5{,}000$ validation images. After selection, the frozen configurations are retrained from scratch on all $50{,}000$ training images and evaluated on the official $10{,}000$-image test set. Training images use random cropping and horizontal flipping; validation and test evaluation are deterministic.

Training and replicated models. All methods minimize cross-entropy for 100 epochs with batch size 128, Nesterov momentum 0.9, weight decay 5×10^{-4}, a three-epoch linear warmup, and cosine learning-rate decay to zero. The learning rate is selected independently for the single-network Base and is fixed at 0.1 for the replicated methods. rSGD and M-cover use $M = 6$ replicas. The replicas are initialized as zero-mean perturbations of a common ResNet initialization, with perturbation scale equal to 1% of each layer's initial weight RMS. Corresponding rSGD and M-cover runs share the common initialization and the per-replica data and augmentation streams. Inference uses a single network obtained by averaging replica parameters, rather than an ensemble of predictions. Batch-normalization statistics are recalibrated using training images after parameter averaging.

Adaptive coupling and M-cover replay. rSGD uses uniform replica mixing ($\rho = 1$). Its coupling strength is updated every three epochs by an adaptive controller that targets a prescribed disagreement contraction time τ; the initial coupling rate is 0.003, and each controller observation changes it by at most a factor of 1.1. The M-cover uses the ring-diffusion mixing family and replays the adaptive rSGD schedule from the matched seed according to

$$\gamma_M(t) = \frac{c}{\rho}\,\gamma_{\mathrm{rSGD}}(t).$$

Thus the M-cover search changes the mixing geometry and the replay multiplier without using its own disagreement to modify the schedule. The implementation uses the Euclidean parameter metric for coupling.

Hyperparameter selection. Base learning rates $\{0.05, 0.1, 0.2\}$ and rSGD contraction times $\tau \in \{10, 20, 30, 50, 90\}$ epochs are first evaluated on two discovery seeds. The two leading Base and rSGD candidates are then evaluated on six fresh confirmation seeds. After selecting $\tau = 10$ for rSGD, the M-cover search evaluates $\rho \in \{0.25, 0.5, 0.75\}$ and $c \in \{1.5, 2.0, 2.5\}$ on two discovery seeds, followed by six-seed confirmation of the two leading configurations. Selection minimizes validation classification error at epoch 100. This procedure selects Base learning rate 0.05, rSGD $\tau = 10$, and M-cover $(\rho, c) = (0.5, 2.5)$.

Final evaluation. The selected policies are frozen and retrained with ten fresh paired seeds, 101–110. Mean test errors (sample standard deviations) are $6.168 \pm 0.153\%$ for Base, $5.683 \pm 0.262\%$ for rSGD, and $5.545 \pm 0.208\%$ for M-cover. The paired gain of M-cover over rSGD, defined as $\mathcal{E}_{\mathrm{rSGD}} - \mathcal{E}_M$, is 0.138 percentage points, with a paired-t 95% confidence interval of $[-0.146, 0.422]$ percentage points.

E.4 Tiny ImageNet ResNet benchmark

Data and architecture. We evaluate the structured Euclidean M-cover on Tiny ImageNet-200, which contains $100{,}000$ training images and $10{,}000$ labeled validation images from 200 classes. Hyperparameter selection uses a fixed class-stratified split of the official training set into $90{,}000$ fitting and $10{,}000$ internal-validation images. The official validation set is not accessed until all configurations have been selected and frozen. Final runs are retrained from scratch on all $100{,}000$ training images and evaluated once on the official validation set, which serves as the held-out endpoint because test labels are not publicly available.

The model is a BasicBlock ResNet-18 with $11{,}271{,}432$ trainable parameters. Its 64×64 input stem consists of a 3×3 stride-two convolution followed by batch normalization and ReLU, without max pooling. The four residual stages therefore operate at spatial resolutions 32, 16, 8, and 4, followed by global average pooling and a 200-class linear classifier.

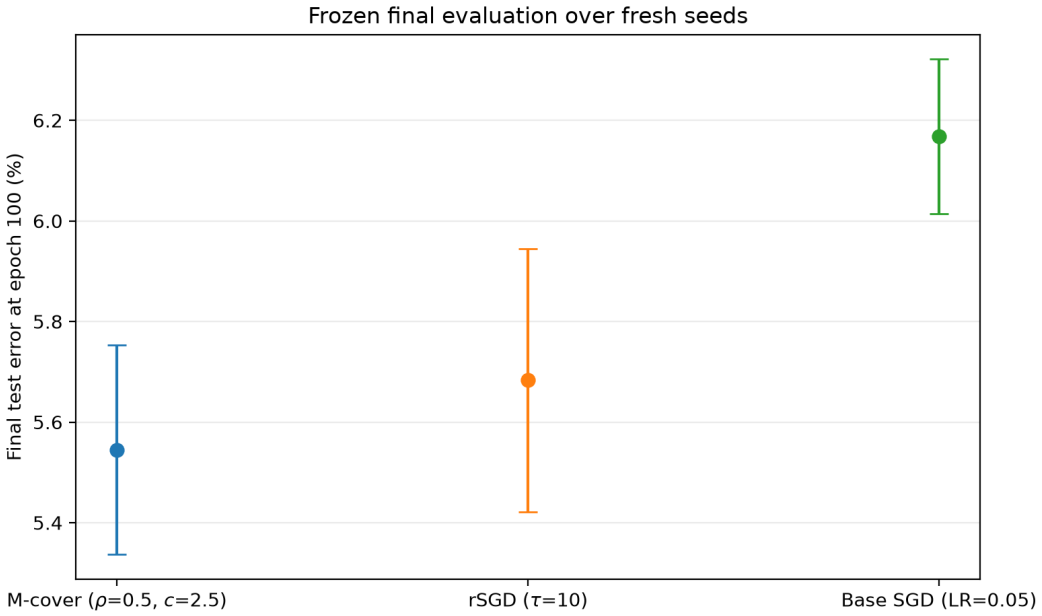


Figure 7: CIFAR-10 test classification error after 100 epochs. Points show means over ten fresh final seeds and error bars show ± 1 sample standard deviation. Hyperparameters are selected using only the held-out validation split. The deployed predictor for each replicated method is the single parameter-averaged network.

Training and replicated models. All methods minimize cross-entropy for 80 epochs using batches of 128 images per replica, Nesterov SGD with momentum 0.9, weight decay 10^{-4}, a five-epoch linear warmup, and cosine learning-rate decay to zero. Training images are padded by eight pixels, randomly cropped to 64×64, and horizontally flipped with probability 0.5. Base tunes its peak learning rate over $\{0.05, 0.1, 0.2\}$, whereas the replicated methods use peak learning rate 0.1.

rSGD and the M-cover use $M = 6$ replicas initialized as centered perturbations of a common ResNet initialization. The perturbation RMS is 1% of the initial RMS of each eligible parameter tensor, and the perturbations sum exactly to zero across replicas. Corresponding rSGD and M-cover runs use matched initializations and matched independently keyed per-replica data and augmentation streams. Deployment uses a single network obtained by averaging the replica parameters. Batch-normalization running statistics are reset and recalibrated on deterministic training images after parameter averaging.

Adaptive rSGD and structured M-cover. For rSGD, the supervised gradient of replica a is augmented by the barycentric attraction

$$g_a \leftarrow g_a + \gamma_t \left(\theta_a - \bar{\theta}\right), \qquad \bar{\theta} = \frac{1}{M} \sum_{a=1}^{M} \theta_a.$$

An adaptive controller observes normalized replica disagreement every two epochs and updates γ_t to target a prescribed contraction time $\tau \in \{8, 24, 72\}$ epochs.

For each architectural site b, the M-cover uses a fixed replica permutation P_b and mixing kernel

$$Q_b(\rho) = \rho J + (1 - \rho) P_b, \qquad J = \frac{1}{M} \mathbf{1}\mathbf{1}^\top.$$

Its parameter-space coupling is

$$g_b \leftarrow g_b + \gamma_t^M \left[\theta_b - Q_b(\rho)\theta_b\right], \qquad \gamma_t^M = \frac{c}{\rho} \gamma_t^{\mathrm{rSGD}}.$$

Each M-cover run replays the complete adaptive rSGD schedule from the matched seed. We fix $c = 2$ and select $\rho \in \{0.25, 0.5, 0.75\}$. Convolutional, linear, bias, and batch-normalization affine parameters participate in the coupling, whereas batch-normalization running buffers remain replica-local.

Selection and final evaluation. Selection uses endpoint internal-validation top-1 error at epoch 80. Base and rSGD candidates are screened using two seeds, after which the two leading candidates receive three and four fresh confirmation seeds, respectively. The three M-cover values of ρ are evaluated using four development seeds and three fresh replay-confirmation seeds. A predeclared one-standard-error rule favors the central grid value when it is statistically competitive. The selected configurations are Base learning rate 0.2, rSGD $\tau = 72$, and M-cover $(\rho, c) = (0.75, 2)$.

The frozen policies are retrained using eight fresh paired seeds, 101–108. Mean official-validation top-1 errors are 42.499% for Base, 42.503% for rSGD, and 42.427% for the M-cover. The paired gain of the M-cover over rSGD, defined as $\mathcal{E}_{\mathrm{rSGD}} - \mathcal{E}_M$, is 0.075 percentage points, with a paired 95% confidence interval of $[-0.352, 0.502]$ percentage points and an exact two-sided sign-flip value of $p = 0.680$. Each method wins four of the eight paired comparisons.

E.5 TinyStories Transformer benchmark

Data and architecture. A deterministic story-level hash split reserves 2% of the official training corpus for hyperparameter selection; this subset is excluded from all fitting runs. A byte-level BPE tokenizer with vocabulary size $10{,}000$ is trained only on the remaining 98% fitting pool and is then frozen. The official TinyStories validation split is not accessed until all method configurations have been selected and sealed. The model is an approximately 8.94M-parameter pre-LayerNorm causal Transformer with eight layers, width 256, eight attention heads, MLP width 1024, learned positional embeddings, tied input and output token embeddings, context length 256, and no dropout.

Training. All methods use AdamW with $\beta = (0.9, 0.95)$, weight decay 0.1, per-replica gradient clipping at 1, a 2% linear warmup, and cosine decay to 10% of the peak learning rate. Parameters and optimizer states remain in FP32, while forward and backward computation uses BF16 autocasting. Training lasts $1{,}500$ optimizer updates with $65{,}536$ tokens per update and replica, corresponding to $98{,}304{,}000$ token presentations per replica. Replicated methods use $M = 6$ identically initialized copies with independently keyed token streams. Deployment and evaluation use the parameter barycenter, not a logits ensemble.

rSGD and structured M-cover coupling. Coupling is applied every four AdamW updates as a separate exact parameter-space operator; gradients and Adam moments are never mixed. For a trainable parameter block, let $X \in \mathbb{R}^{M \times d}$ stack the replicas and let $J = \mathbf{1}\mathbf{1}^\top / M$. With normalized training progress $u \in [0, 1]$, the focusing clock is

$$H_\tau(u) = \left(\frac{u}{\tau}\right)^2.$$

For rSGD, an increment ΔH gives the exact global contraction

$$X^+ = JX + e^{-\Delta H}(I - J)X.$$

For the M-cover, each embedding, attention-input, attention-output, MLP-input, MLP-output, and normalization block receives a fixed replica-cycle Laplacian $L_{\mathrm{cyc},b}$, normalized by its spectral gap. Different blocks use different immutable cycles. Defining

$$L_b(\rho) = \rho(I - J) + (1 - \rho)L_{\mathrm{cyc},b},$$

the structured update is

$$X_b^+ = \exp[-c\,\Delta H\, L_b(\rho)]X_b.$$

Thus ρ controls the interpolation between global and block-routed coupling, while c controls its overall magnitude. All trainable Transformer parameters, including embeddings, attention and MLP weights and biases, and LayerNorm parameters, are coupled along the replica axis.

Selection and final evaluation. Selection is staged to avoid leakage. Base first selects the peak learning rate from $\{2, 4, 8\} \times 10^{-4}$. Holding that value fixed, rSGD selects $\tau \in \{0.5, 0.75, 1, 1.5\}$. The M-cover then inherits the selected learning rate and τ and selects $\rho \in \{0.25, 0.5, 0.75\}$ and $c \in \{1.5, 2, 2.5\}$. Discovery uses two seeds per candidate; the two best candidates at each stage are evaluated on three fresh confirmation seeds. Selection minimizes endpoint token-weighted NLL on the reserved selection split. The locked configurations are learning rate 8×10^{-4} for all methods,

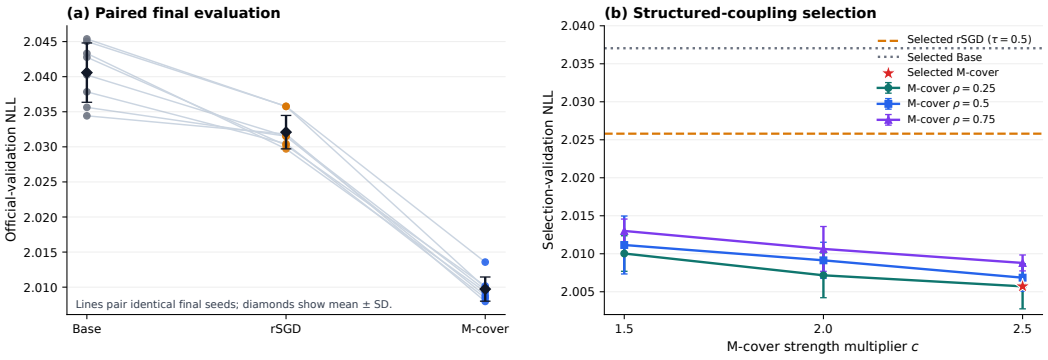


Figure 8: **TinyStories final performance and hyperparameter selection. (a)** Official-validation negative log-likelihood (NLL) for eight fresh paired final seeds. Thin lines join runs with the same seed; diamonds and error bars show the mean and one sample standard deviation. **(b)** Selection-validation NLL as a function of the M-cover strength multiplier c, with separate curves for the route parameter ρ. Error bars show one sample standard deviation over two discovery seeds. Horizontal lines show the selected Base and rSGD references, and the star marks the M-cover candidate advanced to confirmation. Lower is better in both panels.

$\tau = 0.5$ for rSGD, and $(\tau, \rho, c) = (0.5, 0.25, 2.5)$ for the M-cover. Final evaluation restarts all methods from scratch on eight fresh paired seeds and uses only the official validation split. Mean perplexities are 7.6951 for Base, 7.6301 for rSGD, and 7.4613 for the M-cover. The primary paired difference is $\mathrm{NLL}_M - \mathrm{NLL}_{\mathrm{rSGD}} = -0.02237$ with 95% confidence interval $[-0.02392, -0.02082]$ and exact two-sided sign-flip $p = 0.0078125$; all eight paired differences favor the M-cover.

The selected M-cover has effective contraction time $\tau/\sqrt{c} \simeq 0.316$, compared with $\tau = 0.5$ for rSGD. Accordingly, this experiment compares independently tuned methods and does not by itself isolate coupling topology from coupling magnitude.

E.6 KdV physics-informed neural network benchmark

Problem and reference method. We consider the Korteweg–de Vries (KdV) equation, which describes nonlinear dispersive waves,

$$u_t + 6uu_x + u_{xxx} = 0, \qquad (t, x) \in [0, 5] \times [0, 20].$$

Initial values, Dirichlet data at both spatial boundaries, and the derivative at the right boundary are taken from the two-soliton solution in Toscano et al. (2026), with speeds 6 and 2 and shifts -2 and 2. Our reference is their state-of-the-art KdV configuration: exponential variational residual-based attention (vRBA), which emphasizes large PDE residuals, combined with Adam pretraining and the self-scaled Broyden (SSBroyden) quasi-Newton optimizer. Their Table 4 reports a relative L^2 error of 2.17×10^{-6}, below PirateNet by Wang et al. (2024) with Adam or SOAP by Wang et al. (2026) and SSBroyden with residual-based adaptive distribution (RAD).

Architecture and data. Our PINN normalizes inputs to $[0, 1]^2$ and uses a trainable Fourier embedding with five frequencies, initialized from a zero-mean Gaussian with standard deviation 3, followed by three width-30 tanh hidden layers and a scalar linear output (2,231 parameters). An output ansatz enforces the initial and Dirichlet conditions exactly; the right-boundary derivative is enforced through a Neumann loss. We use a fixed pool of 200,000 Latin-hypercube collocation points, 512 uniformly sampled Neumann points, and batches of 20,000 collocation points.

Training. Training uses JAX in double precision, with 5,000 Adam updates followed by 800 SSBroyden blocks of at most 100 iterations per replica. Adam starts at learning rate 10^{-3}, with exponential decay by a factor of 0.9 over its 5,000 updates. Exponential vRBA controls residual weighting and adaptive sampling during Adam, and sampling before each SSBroyden block; the latter minimizes the unweighted sampled PDE loss plus the Neumann loss. The vRBA schedule uses

logarithmic annealing, initial memory 0.999, and update scale 0.01. The Neumann loss weight is balanced using gradient norms during Adam and frozen afterward.

Replicated learning. We use $M = 4$ replicas with identical initial Fourier embeddings and zero-mean Gaussian MLP perturbations around a shared initialization, with relative RMS norm 0.02. All trainable parameters are coupled using the ring-diffusion operator K_ρ, now on four replicas; $\rho = 1$ gives uniform rSGD coupling. During Adam, the coupling gradient is $\gamma(t)K_\rho\Theta$, with a cosine ramp in log space from 10^3 to 2×10^4. After each task-only SSBroyden block, parameter deviations from their mean are contracted by $(I - cK_\rho)^n$, where n is the largest number of iterations executed by any replica in that block. A cosine schedule increases c from $0.003s$ to $0.015s$. Optimizer states remain replica-specific, and coupling is excluded from the SSBroyden objective and curvature updates.

Selection and evaluation. We first sweep $s \in \{0.3, 1, 3, 5, 10, 15\}$ for rSGD, selecting $s = 3$. We then evaluate $\rho \in \{0.8, 0.5, 0.2\}$ with $s = (3/\rho) \times \{1, 2, 4\}$, keeping the Adam coupling schedule fixed. All candidates use seed 10003 with matched initializations, point pools, and sampling seeds. We retain the SSBroyden checkpoint minimizing the center model's mean absolute PDE residual on the full collocation pool. Inference uses the network at the mean replica parameters. Hyperparameters are selected by its relative L^2 error against the exact solution on a uniform 100×256 time–space grid.

Results. The selected M-cover setting $(\rho, s) = (0.5, 12)$ achieves 1.6419×10^{-6} relative L^2 error, compared with 2.1988×10^{-6} for the selected rSGD setting, a 25.3% reduction. These are exploratory single-seed results selected on the evaluation grid, without independent validation or confirmation runs. The 2.17×10^{-6} Base entry is taken from the cited paper and constitutes a literature comparison.

E.7 ETTm2 CycleNet forecasting benchmark

Data and architecture. We evaluate CycleNet/MLP (Lin et al., 2024) on ETTm2, predicting the next 96 observations of all seven variables from the preceding 96 observations. At the 15-minute sampling interval, this corresponds to forecasting 24 hours from 24 hours of history. We retain the official chronological split: 34,560 observations for training, followed by 11,520 for validation and 11,520 for testing. Validation and test windows may use the preceding 96 observations as input context, while their forecast targets remain within the respective split. Each variable is standardized using training statistics only.

The backbone is a 96–512–96 MLP with one ReLU hidden layer and biases, with weights shared across the seven variables. CycleNet additionally learns a periodic tensor $Q \in \mathbb{R}^{96\times7}$: it subtracts the phase-aligned cycle from the input, predicts the residual with the MLP, and adds the corresponding future cycle. We retain the model's instance normalization and output de-normalization. The single-model Base uses the official initialization, including $Q = 0$.

Training. All methods minimize mean-squared prediction error using Adam with $(\beta_1, \beta_2) = (0.95, 0.999)$, $\epsilon = 10^{-8}$, and no weight decay. The batch size is 256. Every run completes exactly seven epochs, corresponding to 134 updates per epoch and $S = 938$ updates in total; evaluation uses the final checkpoint. We retain the effective learning rates of the reproduced CycleNet implementation:

$$(\eta_1, \ldots, \eta_7) = 10^{-3}(0.2, 5, 5, 5, 4, 3.2, 2.56).$$

The official loader shuffles training and validation windows and drops incomplete batches; test evaluation includes all windows. The epoch budget is matched across methods, while replicated training uses four model copies.

Replicated models and coupling. rSGD and M-cover use $M = 4$ replicas. Their MLP parameters are initialized as centered regular-simplex perturbations of a common initialization, with $\sqrt{M^{-1}\sum_{r=1}^{M} \|\delta_{\mathrm{MLP}}^{(r)}\|_2^2} = 0.1\|\theta_{\mathrm{MLP}}^{(0)}\|_2$. The cycle tensors receive centered perturbations around zero, with per-coordinate RMS matched to the MLP perturbations. All replicas process the same minibatch at each update; corresponding runs share initialization and data-order seeds across methods.

Let Θ contain the complete replica parameters as rows, including the MLP weights, biases, and Q. At every minibatch, the objective is

$$\sum_{r=1}^{M} \ell_B(\theta^{(r)}) + \frac{\gamma_k}{2} \operatorname{tr}(\Theta^\top K_\rho \Theta),$$

where ℓ_B averages squared error over the batch, forecast horizon, and seven variables. Thus the coupling gradient is $\gamma_k K_\rho \Theta$, with no additional normalization by replica or parameter count. The task and coupling gradients are combined before the Adam update, with separate optimizer moments for each replica's parameters.

We use the Euclidean parameter metric and the same undirected four-node ring for every parameter block. Writing $L = L_{\text{ring}}/\lambda_{\max}(L_{\text{ring}})$, the mixing family is

$$K_\rho = I - \exp[\log(1-\rho)L], \qquad 0 < \rho < 1, \qquad K_1 = I - \frac{\mathbf{1}\mathbf{1}^\top}{M}.$$

The endpoint $\rho = 1$ is uniform rSGD coupling.

Coupling schedule and hyperparameter selection. Selection minimizes mean validation MSE at epoch seven over the five paired seeds 2024–2028. We first tune uniform rSGD using exponential schedules starting at 10^{-6}, with final strengths $\{3000, 5000, 7500, 10000, 15000, 20000, 30000\}$. The selected endpoint is 10^4, defining the fixed reference schedule

$$\gamma_{\text{ref}}(k) = 10^{-6} \left(\frac{10^4}{10^{-6}}\right)^{k/(S-1)}, \qquad k = 0, \ldots, S-1.$$

For M-cover, we use $\gamma_{\rho,c}(k) = (c/\rho)\gamma_{\text{ref}}(k)$ and search

$$\rho \in \{0.1, 0.25, 0.5, 0.75, 0.9\},$$
$$c \in \{1, 1.5, 2, 3, 4, 6, 10, 20, 30, 50, 100\}.$$

This multiplier scales the entire schedule, including its initial value. We additionally evaluate uniform controls with $\rho = 1$ and $c \in \{10, 20, 30, 50, 100\}$, and reselect rSGD from the union of these controls and the original endpoint sweep. The reference schedule used by M-cover remains fixed. This procedure selects $(\rho, c) = (1, 100)$ for the reported rSGD baseline and $(\rho, c) = (0.75, 50)$ for M-cover.

Evaluation and results. For replicated methods, inference uses one network at the mean replica parameters, including the mean cycle tensor. Test MSE is averaged over all test windows, forecast steps, and standardized variables. The selected configurations are evaluated using the same five training seeds as the hyperparameter search; these are not independent confirmation runs.

Figure 9 reports mean test MSE and sample standard deviation: 0.163890 ± 0.000735 for Base, 0.161841 ± 0.000236 for rSGD, and 0.161600 ± 0.000193 for M-cover. The improvement over rSGD is small (2.4×10^{-4} MSE, approximately 0.15%), but M-cover achieves lower test MSE on all five paired seeds. Relative to Base, its mean MSE is reduced by approximately 1.40%.

Comparison with reported forecasting results. For the same seven-variable ETTm2 task with input and forecast lengths of 96, reported MSEs are 0.163 for CycleNet/MLP (Lin et al., 2024), 0.162 for FreqCycle (Zhang et al., 2026), and 0.160 for HPMixer (Choi et al., 2026). Our M-cover result, 0.161600, improves on the reported CycleNet/MLP mean and matches FreqCycle at the reported three-decimal precision, while HPMixer reports a lower MSE. These literature values provide context; training budgets, hyperparameter searches, and checkpoint-selection protocols differ across studies.

F Effect of the mixing topology

We examine whether changing the replica mixing topology can improve generalization beyond the ring-diffusion family.

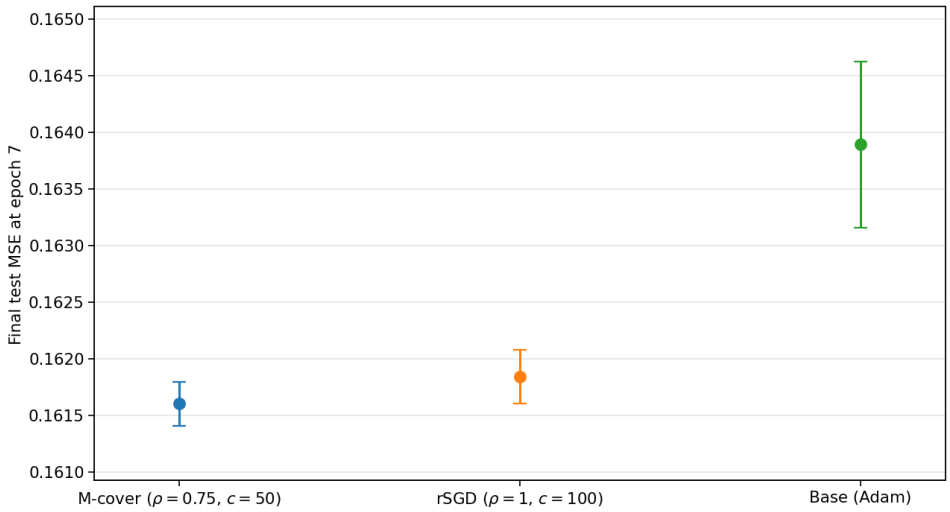


Figure 9: ETTm2 test MSE after seven epochs for CycleNet/MLP with input and forecast lengths of 96 and seven variables. Points show means over five paired seeds; error bars indicate ± 1 sample standard deviation. Hyperparameters are selected by validation MSE using these same training seeds. The selected configurations are rSGD with $(\rho, c) = (1, 100)$ and M-cover with $(\rho, c) = (0.75, 50)$, both using $M = 4$ replicas and a single parameter-averaged network for inference.

Experimental setup. We use a fixed stratified split of the official Fashion-MNIST training set into 50,000 training and 10,000 validation images, and compare mixing topologies using validation classification error. Each of the $M = 5$ replicas is a 784–512–512–10 ReLU network with biases and 669,706 parameters. Images are scaled to $[0, 1]$ and normalized with mean 0.2860 and standard deviation 0.3530, without augmentation. Training minimizes cross-entropy for 300 epochs using batch size 128, SGD with Nesterov momentum 0.9, and cosine learning-rate decay from 0.05 to zero, without weight decay, dropout, or batch normalization. Replicas use independent default PyTorch initializations and independently shuffled streams covering the full training set. Corresponding runs share initial weights and minibatch orders across topologies.

Mixing matrices. For a symmetric weighted adjacency matrix A, define

$$L_0 = \operatorname{diag}(A\mathbf{1}) - A, \qquad L = \frac{L_0}{\lambda_{\max}(L_0)}, \qquad T_\rho = \exp[\log(1-\rho)L], \qquad K_\rho = I - T_\rho.$$

We fix $\rho = 0.5$ for the nonuniform topologies. The uniform control uses $\rho = 1$, with $T_1 = \Pi_M = \mathbf{1}\mathbf{1}^\top/M$ and $K_1 = I - \Pi_M$. The underlying graphs are fixed and undirected, with unit edge weights unless specified otherwise:

- **Ring:** the cycle 1–2–3–4–5–1.
- **Uniform:** all-to-all coupling at the uniform endpoint.
- **Path:** the chain 1–2–3–4–5.
- **Star:** replica 1 connects to replicas $2, 3, 4, 5$.
- $2 + 3$ **clusters:** groups $\{1, 2\}$ and $\{3, 4, 5\}$, with weights 1 within groups and 0.1 between groups.
- **Weighted ring:** the ring edges above have weights $(1, 0.3, 1, 0.3, 0.1)$ in cyclic order.
- **Random:** a complete graph with independently sampled log-uniform edge weights on $[0.1, 1]$. A single matrix generated using a fixed seed is shared across all runs.

The matrix exponential includes indirect connections. The same K_ρ acts on every layer, using the Euclidean parameter metric.

Coupling protocol. Let Θ contain all replica weights and biases as rows. Every tenth minibatch, starting with the first minibatch of each epoch, we add $\gamma(t)\,\mathrm{tr}(\Theta^\top K_\rho \Theta)/2$ to the sum of the replica losses. We use the adaptive rSGD controller specified in the preceding Fashion-MNIST benchmark,

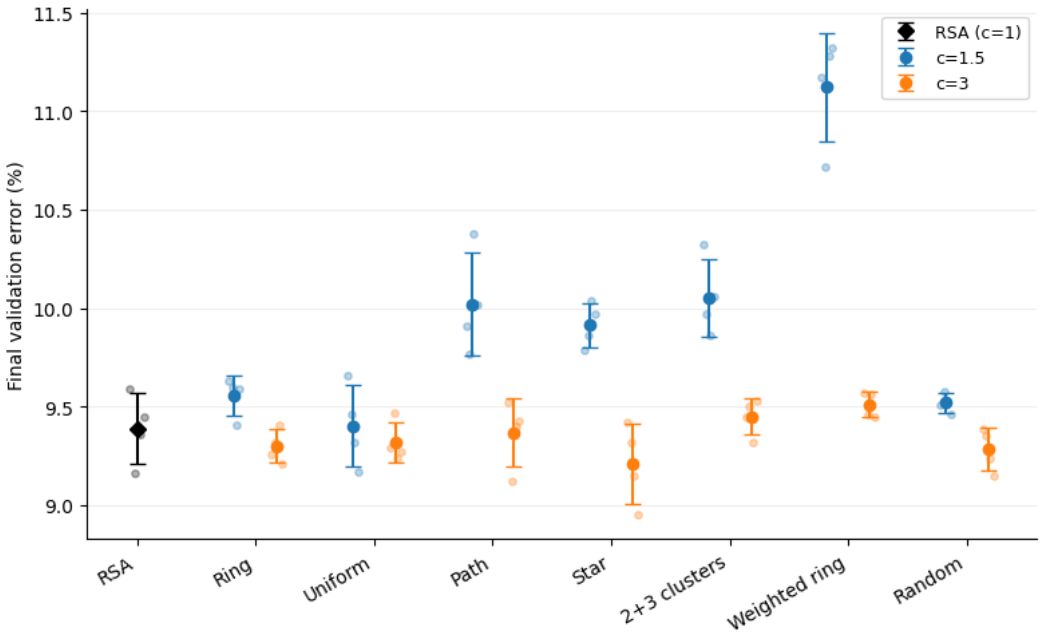


Figure 10: Effect of the mixing topology on Fashion-MNIST validation classification error after 300 epochs. Points show means and error bars indicate ± 1 sample standard deviation across four training seeds, pooled from two separately tuned rSGD seed sets; faint points show individual runs. RSA denotes rSGD at $c = 1$; colors distinguish $c = 1.5$ and $c = 3$.

updated every ten epochs. Each topology replays the rSGD schedule from the same training seed using

$$\gamma(t) = \frac{c}{\rho}\gamma_{\mathrm{rSGD}}(t), \qquad c \in \{1.5, 3\},$$

without feedback from its own disagreement. Since $\lambda_{\max}(K_\rho) = \rho$, this gives $\lambda_{\max}(\gamma(t)K_\rho) = c\gamma_{\mathrm{rSGD}}(t)$: the strongest coupling mode is matched across topologies while the remaining modes can differ. Uniform therefore corresponds to rSGD with its schedule multiplied by c; the original rSGD baseline uses $c = 1$.

Search and evaluation. The rSGD contraction timescale τ is selected independently for seed sets $(2, 7, 11)$ and $(101, 107, 113)$. In each set, the first seed evaluates the coarse grid $\tau \in \{30, 100, 300\}$ epochs and the two refinement points $\{173.20508, 519.61524\}$. The best two candidates are then evaluated on the remaining two seeds; selection minimizes mean validation classification error at epoch 300. The selected timescales are 173.20508 and 300 epochs, respectively. All seven topologies and both multipliers are evaluated on the first two seeds of each set, giving four runs per configuration on seeds $2, 7, 101, 107$. The rSGD reference uses these same four seeds. Evaluation uses the parameter-averaged network at epoch 300. We pool the four raw results to compute means and sample standard deviations; these include variation across both training seeds and the two selected rSGD timescales.

Results. Figure 10 shows that increasing c from 1.5 to 3 reduces mean validation error for every topology. At $c = 3$, star has the lowest mean error, 9.21%, compared with 9.30% for ring, 9.32% for uniform, and 9.39% for the original rSGD reference. These exploratory four-seed results suggest that selecting the mixing topology can further improve generalization beyond the ring family.

G Scaling with the number of replicas.

To assess whether the generalization advantage of the local M-cover changes with the number of replicas, we performed an experiment at the overparameterization ratio $P/D = 4$. The number

of replicas was varied over $M \in \{3, 4, 5, 6, 8, 10\}$, while the remaining training protocol was kept fixed. Hyperparameters were selected on validation data separately for each M, and the resulting policies were restarted from scratch and evaluated over 450 fresh confirmation seeds. Figure 11 shows the confirmation test error at epoch 1500 for the selected M-cover, the independently tuned replicated-SGD reference, and the single-network Base model. At $M = 6$, the selected M-cover achieves a larger improvement over rSGD. From the candidate set of $\rho \in \{0.3, 0.4, 0.85, 0.9, 1\}$, the selected structured mixing parameters are $\rho^\star = 0.9$, 0.4, 0.3, 0.3, and 0.3 for $M = 4$, 5, 6, 8, and 10, respectively, indicating progressively more nonuniform mixing as the number of replicas increases. Although the available data are insufficient to reliably estimate asymptotic scaling, the observed gains are consistent with approximate $M^{0.3}$ and $M^{0.24}$ scaling relative to Base and rSGD, respectively.

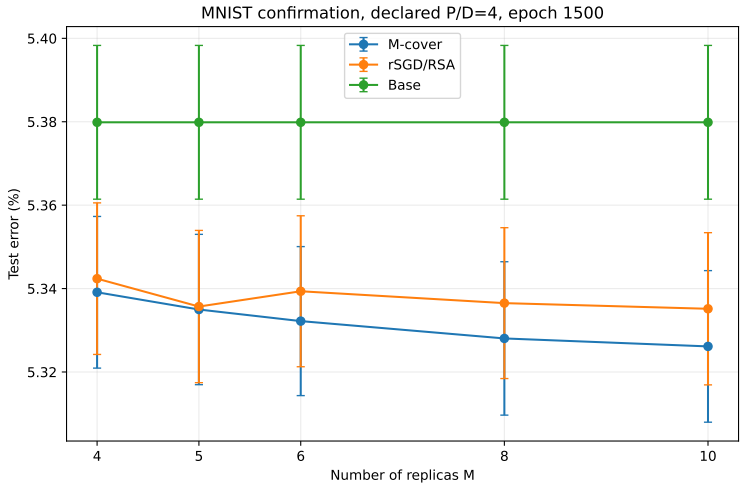


Figure 11: **MNIST confirmation error as a function of the number of replicas at $P/D = 4$.** Test classification error at epoch 1500 for the validation-selected local M-cover, independently tuned replicated SGD (rSGD), and the single-network Base model, evaluated over 450 fresh confirmation seeds. Error bars denote 95% paired Student-t confidence intervals. At larger M, the selected M-cover achieves a larger improvement.