BELLMAN DEMAND AND CRITIC RESPONSE: A SPECTRAL VIEW OF CONTINUAL RL

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual reinforcement learning requires agents to retain the ability to learn as tasks change over time, yet plasticity loss in neural networks can impede learning on subsequent tasks. In SAC, policy improvement relies on value estimates provided by the critic; therefore, whether the critic can learn the value function of a new task within a finite update budget is critical to continual adaptation. We study this problem through the discrepancy between the Bellman target and the current prediction, examining how the value corrections required by a new task align with the critic's learning responses across different directions. When the dominant correction demand of a new task is concentrated in directions to which the critic responds slowly, this mismatch can hinder value correction. To quantify this difficulty, we derive a demand-weighted spectral burden that combines the fraction of demand carried by each direction with its relative response speed, characterizing the relative directional difficulty of correcting the current residual. Motivated by this analysis, we adopt Periodic Spectral Projection (PSP) in continual SAC, constraining the singular values of each linear layer in both critics to a prescribed interval to modulate their learning responses at the weight level. Experiments on Meta-World and CompoSuite show improved subsequent-task acquisition with PSP. The spectral analyses characterize how the current Bellman correction demand is distributed across the critic's response directions during adaptation.

1 INTRODUCTION

Continual reinforcement learning requires agents to retain the ability to learn as tasks arrive sequentially and change over time Khetarpal et al. (2022); Wołczyk et al. (2021), yet neural networks may gradually lose their ability to learn new objectives after prior training, resulting in plasticity loss Abbas et al. (2023); Dohare et al. (2024). Prior studies have analyzed plasticity loss through network properties such as neuronal activity Sokar et al. (2023), the effective rank of learned features Kumar et al. (2020), and loss curvature Lyle et al. (2023), and have sought to improve subsequent learning through network resetting Nikishin et al. (2022), neuron recycling Sokar et al. (2023); Dohare et al. (2024), and spectral constraints Lewandowski et al. (2025); Chung et al. (2024). Related work has further examined the relationship between target distributions and network responses through task-model alignment analyses in kernel regression Canatar et al. (2021) and residual spectral analyses in continual learning Prakash et al. (2025). In soft actor-critic (SAC), the critic guides policy updates through value estimates Haarnoja et al. (2018a;b) and must adjust its value predictions as tasks change. After a task transition, the discrepancy between the new Bellman target and the current prediction defines the correction that the critic must make, which we refer to as the Bellman correction demand. When the critic continues from parameters learned on previous tasks, its local response may differ across correction directions Jacot et al. (2018). As shown in Figures 1 and 2, continual fine-tuning (FT) struggles to learn after a new task arrives, with a larger fraction of its Bellman correction demand energy concentrated in weak-response directions. This observation prompts us to ask:

> ***If the Bellman correction demand of a new task is concentrated in the critic's slow-response directions, does this hinder its adaptation to the new task?***

To answer this question, we analyze critic adaptation to new tasks from the perspective of the alignment between Bellman correction demand and directional learning response. We decompose

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.

the correction demand during training along the eigendirections of the critic's parameter-learning kernel and examine both the fraction of demand carried by each direction and its response strength. When the dominant correction demand of a new task is concentrated in directions to which the critic responds slowly, this mismatch can impede value correction. To quantify this correction difficulty, we derive a demand-weighted spectral burden that combines the fraction of demand carried by each direction with its relative response speed. For the same fraction of demand, concentration in a slower-response direction produces a larger spectral burden.

Motivated by the above analysis, we adopt Periodic Spectral Projection (PSP) in continual SAC to modulate the critic's learning response by constraining the spectra of its layer weights. Layer weights affect both forward feature propagation and backward gradients, and can therefore influence the critic's local response through the parameter Jacobian. PSP uses two-sided singular-value clipping: while preserving the singular vectors, it periodically constrains the singular values of all linear layers in both critics to a prescribed interval. Experiments on task sequences from Meta-World Yu et al. (2020) and CompoSuite Mendez et al. (2022) jointly examine correction-demand distributions, spectral burden, and task-learning performance. PSP improves subsequent-task acquisition, while the spectral diagnostics characterize the demand–response alignment encountered during training. Our main contributions are as follows:

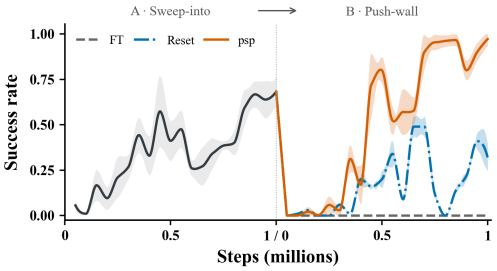


Figure 1: Task acquisition after the Sweep-into → Push-wall transition. All methods share the same first-task training; curves are averaged over 3 seeds.

- **A new perspective on continual critic adaptation.** We analyze new-task learning through the alignment between Bellman correction demand and directional learning response, focusing on how the directional distribution of the correction demand and the critic's response jointly shape adaptation.

- **A demand-weighted spectral burden for quantifying local correction difficulty.** We combine the fraction of correction demand carried by each direction with its relative response speed in a demand-weighted spectral burden that characterizes the relative directional mismatch between the current Bellman correction demand and the critic's learning response.

- **Empirical validation of PSP in continual reinforcement learning.** Experiments on task sequences from Meta-World and CompoSuite demonstrate that PSP effectively improves new-task learning performance in continual SAC.

2 Related Work

2.1 Continual Reinforcement Learning

Continual reinforcement learning studies how agents can continually learn in non-stationary tasks and environments Khetarpal et al. (2022); Abel et al. (2023). Existing methods can be broadly categorized into parameter isolation, knowledge consolidation, and experience replay. Parameter-isolation methods reduce interference across tasks by freezing previously learned parameters or introducing task-specific modules, while reusing prior knowledge through shared representations or modular composition Rusu et al. (2016); Kessler et al. (2022); Malagon et al. (2025); Zhou et al.. Knowledge-consolidation methods preserve and integrate previously acquired knowledge through internal states operating at multiple timescales or through distillation Schwarz et al. (2018); Kaplanis et al. (2018; 2019). Experience-replay methods reuse historical data when training on new tasks and coordinate learning across old and new tasks through value-target adjustment or policy constraintsRolnick et al. (2019); Zhang et al. (2023).

2.2 Plasticity in Continual Reinforcement Learning

Plasticity loss manifests as a decline in a network's ability to continue learning new objectives Klein et al. (2024); Dohare et al. (2024). In deep reinforcement learning, both the training data and bootstrapped targets continually change, causing this problem to arise not only at task transitions but also during within-task training Lyle et al. (2022a); Nikishin et al. (2022). Prior studies have linked plasticity loss to activation sparsification Abbas et al. (2023), the accumulation of low-activity neurons Sokar et al. (2023), reductions in the effective rank of learned features Kumar et al. (2020), and changes in loss curvature Lyle et al. (2023). Corresponding methods seek to preserve subsequent learning ability by constraining selected features from deviating from their initial outputs Lyle et al. (2022a), resetting network components Nikishin et al. (2022), or recycling low-activity and low-utility neurons Sokar et al. (2023); Dohare et al. (2024). For actor-critic architectures, module-level analyses in visual reinforcement learning have further identified critic plasticity as a key bottleneck in the evaluated settings Ma et al. (2024).

2.3 Spectral Analysis of Bellman Updates in RL

Spectral analysis has long been used to characterize value representations and Bellman updates in reinforcement learning. Prior work has exploited spectral properties of transition operators or Bellman structure to construct representations Mahadevan & Maggioni (2007); Wu et al. (2018); Ren et al. (2022); Nabati et al. (2026), and has further analyzed how different spectral components affect the stability and learning speed of temporal-difference (TD) learning Ghosh & Bellemare (2020); Lyle et al. (2022b). Relatedly, neural tangent kernel (NTK) theory describes the local learning dynamics of neural networks as gradient dynamics along kernel eigendirections Jacot et al. (2018), while studies of task-model alignment show that learning behavior depends not only on the kernel spectrum but also on how the target is distributed across these directions Canatar et al. (2021); Spectral Collapse further investigates this issue in continual learning through a spectral decomposition of residuals under a finite update budget Prakash et al. (2025).

3 Preliminaries

3.1 Continual Reinforcement Learning

Continual reinforcement learning (CRL) studies how agents continually learn and adapt in non-stationary environments Khetarpal et al. (2022). We consider an online task-sequence setting with environment interaction Wołczyk et al. (2021); Malagon et al. (2025), in which the agent learns a sequence of tasks ${\mathcal{M}^{(k)}}_{k=1}^{T}$, each described by a Markov decision process (MDP) Sutton et al. (1998):

$$\mathcal{M}^{(k)} = \big(\mathcal{S}, \mathcal{A}, P^{(k)}, r^{(k)}, \rho_0^{(k)}, \gamma\big), \qquad k = 1, \dots, T. \tag{1}$$

Here, $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces shared across tasks, respectively; $P^{(k)}$ is the state transition kernel, $r^{(k)}$ is the expected reward function, $\rho_0^{(k)}$ is the initial-state distribution, and $\gamma \in [0, 1)$ is the discount factor. The MDP remains fixed within each task stage, while the transition dynamics, reward function, or initial-state distribution may change across tasks.

3.2 Soft Actor-Critic and the Bellman Learning Spectrum

We use soft actor-critic (SAC) as the base algorithm Haarnoja et al. (2018a;b), maintaining a stochastic policy π_ϕ, two critics $Q_{\theta_1}, Q_{\theta_2}$, and their corresponding target networks. Given a replay sample $(s, a, r, s', z) \sim \mathcal{D}$, the two critics fit a shared Bellman target by minimizing the squared error Haarnoja et al. (2018b), with the target defined in Equation 2.

$$y = c_r r + \gamma(1 - z)\left[\min_{j\in\{1,2\}} Q_{\bar{\theta}_j}(s', a') - \alpha \log \pi_\phi(a' \mid s')\right], \qquad a' \sim \pi_\phi(\cdot \mid s'). \tag{2}$$

Here, $\bar{\theta}_j$ denotes the target-network parameters, γ is the discount factor, α is the entropy temperature, c_r is the reward scaling coefficient, and $z \in \{0, 1\}$ indicates environment termination. During critic updates, gradients are stopped through the Bellman target y, and the target-network parameters are

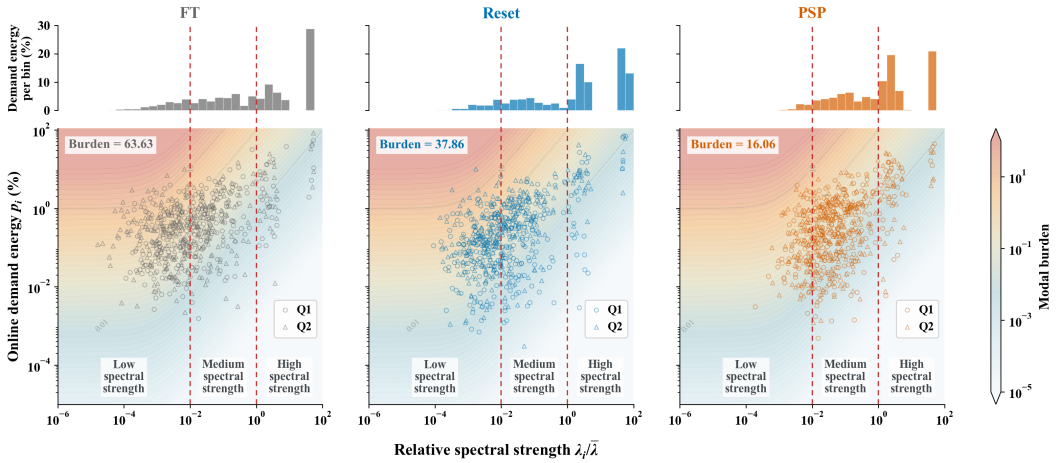


Figure 2: Distribution of Bellman correction demand over the relative learning spectrum. The bottom row shows how demand energy is distributed across directions of different relative spectral strengths: the horizontal axis gives the relative spectral strength $\tilde{\lambda}_i$, the vertical axis gives the percentage of demand energy, and circles and triangles denote the two critics, respectively. The background contours indicate the per-direction contribution $p_i/(\tilde{\lambda}_i + \varepsilon)$ in Equation 11. The top row further aggregates the demand energy carried by different spectral ranges, with the values for each method summing to 100%.

updated via Polyak averaging of the corresponding online critic parameters; the actor is updated using the minimum value estimate from the two critics together with the entropy term.

To characterize how the critic learns to match Bellman targets, we analyze the two critics separately on a fixed experience set :

$$\mathcal{B} = \{(s_\ell, a_\ell, r_\ell, s'_\ell, z_\ell)\}_{\ell=1}^{n}. \tag{3}$$

At update step t, let $q_t = [Q_{\theta_t}(s_\ell, a_\ell)]_{\ell=1}^{n}$, and let y_t denote the target vector computed according to Equation 2. The empirical residual and corresponding loss are

$$d_t = y_t - q_t, \qquad L_Q(\theta_t) = \frac{1}{2n}\|q_t - y_t\|_2^2. \tag{4}$$

The residual d_t represents the output change that the current target requires the critic to make, which we refer to as the Bellman correction demand. The critic's local response to this demand is characterized by the parameter Jacobian and the corresponding empirical parameter-learning kernel Jacot et al. (2018):

$$J_t = \frac{\partial q_t}{\partial \theta_t} \in \mathbb{R}^{n\times p}, \qquad K_t = \frac{1}{n} J_t J_t^\top = U_t \,\mathrm{diag}\,(\lambda_{1,t}, \ldots, \lambda_{n,t})\, U_t^\top. \tag{5}$$

Here, p denotes the number of trainable parameters in the critic, including biases. K_t is an uncentered positive semidefinite parameter-learning kernel, whose spectrum we refer to as the Bellman learning spectrum. The orthonormal eigenvectors $u_{i,t}$ describe directions of output variation over the experience set, the eigenvalues $\lambda_{i,t} \geq 0$ specify the response scale of each direction in the local gradient model, and $u_{i,t}^\top d_t$ gives the corresponding demand component. Section 4 uses these shared directions to analyze how a given demand is corrected under a finite update budget.

4 Spectral Analysis of Critic Adaptation

This section investigates whether Bellman correction demand concentrated in weak-response spectral directions limits critic adaptation to new tasks. We first analyze directional value correction under a finite update budget, then introduce a demand-weighted spectral burden to characterize the relative mismatch between correction demand and learning response.

4.1 Learning with a Finite Update Budget

On a new task, we construct the Bellman target y according to Equation 2 and denote the inherited critic output by q_0. Their difference, $d_0 = y - q_0$, defines the value correction that the current model needs to make. To analyze how these corrections are learned along different directions, we fix the experience inputs and sampled targets and adopt a local affine model at θ_0 Jacot et al. (2018):$q_\theta = q_0 + J(\theta - \theta_0)$, where J is the fixed parameter Jacobian. Under this model, parameter updates are mapped to output changes through the learning kernel $K = JJ^\top/n$, where n is the number of experience inputs. Let the current demand be $d = y - q_\theta$. Performing one full-batch gradient descent step with step size $\eta > 0$ on the loss $L_Q = \|q_\theta - y\|_2^2/(2n)$ yields:

$$\nabla_\theta L_Q = -\frac{1}{n} J^\top d, \qquad \Delta q = \eta K d, \qquad u_i^\top \Delta q = \eta \lambda_i \, u_i^\top d. \tag{6}$$

Here, u_i is an orthonormal eigenvector of the learning kernel, and λ_i is its corresponding eigenvalue.

In Equation 5, $\eta\lambda_i$ is the response coefficient of the critic to the correction demand along direction u_i. To analyze how weak-response directions affect value learning on a new task, we further examine the distribution of the initial demand across the eigendirections of the learning kernel. For a nonzero initial demand d_0, we define the fraction of its energy along direction u_i as:

$$p_i = \frac{|u_i^\top d_0|^2}{\|d_0\|_2^2}, \qquad \sum_i p_i = 1. \tag{7}$$

Proposition 4.1 (Demand Distribution and Value Correction under a Finite Update Budget)**.** *Under the fixed-input, fixed-target affine model described above, let $d_0 \neq 0$ and perform $H \in \mathbb{N}_0$ steps of full-batch gradient descent with step size $\eta > 0$, starting from θ_0. Let the residual after H updates be $d_H = y - q_{\theta_H}$. Then the normalized residual energy satisfies:*

$$\mathcal{R}_H := \frac{\|d_H\|_2^2}{\|d_0\|_2^2} = \sum_i p_i (1 - \eta\lambda_i)^{2H}. \tag{8}$$

This result follows by expanding the residual recursion $d_{h+1} = (I - \eta K) d_h$ along the eigendirections of the learning kernel; the proof is provided in Appendix B.1

Under this model, when the step size satisfies $0 \leq \eta\lambda_i \leq 1$, prediction errors along weak-response spectral directions, i.e., directions associated with smaller eigenvalues of the learning kernel, decay more slowly. Equation 8 shows that the contribution of each such direction to the normalized fitting error is $p_i(1 - \eta\lambda_i)^{2H}$. Therefore, if the Bellman correction demand of a new task is concentrated primarily in weak-response spectral directions, the dominant errors that the critic must eliminate are precisely those that decay slowly, thereby limiting its ability to fit the new target within a finite update budget. Conversely, if the fraction of demand energy in these directions is small, their contribution to the normalized fitting error remains small even when the corresponding errors decay slowly.

4.2 Demand-Weighted Spectral Burden

The section 4.1 showed that prediction errors along weak-response directions decay more slowly, while their impact on fitting a new target depends on how much of the initial demand is concentrated in those directions. To account for both the fraction of error and its decay rate, we further examine the cumulative residual energy contributed by each direction during the local learning process. For the same initial error magnitude, a more slowly corrected error makes a larger cumulative contribution.

To compare how the demand is distributed across relatively fast and slow response directions, we normalize the residual energy by the initial demand energy $\|d_0\|_2^2$ and scale the response by the mean eigenvalue of the learning kernel, $\bar{\lambda} = \mathrm{tr}(K)/n > 0$. Under this normalization, we rescale gradient-flow time from t to $s = \bar{\lambda}t$, and define $\tilde{K} = K/\bar{\lambda}$, whose eigenvalues are $\tilde{\lambda}_i = \lambda_i/\bar{\lambda}$. This allows us to compare the fraction of error carried by each direction and its decay rate relative to the average response, rather than differences in initial error magnitude or the scale of the learning kernel.

Under the same fixed-target affine model, the normalized gradient flow satisfies $\frac{\mathrm{d}}{\mathrm{d}s}d(s) = -\widetilde{K}d(s)$ with $d(0) = d_0$. Solving along the eigendirections of the learning kernel gives:

$$\frac{|u_i^\top d(s)|^2}{\|d_0\|_2^2} = p_i e^{-2\tilde{\lambda}_i s}. \tag{9}$$

Here, p_i gives the fraction of the initial error carried by direction u_i, while the exponential term describes how this error decays with learning time. We integrate this process with temporal discounting, assigning less weight to residuals at later times, to obtain the demand-weighted spectral burden.

Proposition 4.2 (Demand-Weighted Spectral Burden)**.** *Under the fixed-target affine model of proposition 4.1, let $d_0 \neq 0$, $\bar{\lambda} > 0$, and $\varepsilon > 0$. For the normalized gradient flow defined above, define the discounted demand-weighted spectral burden as:*

$$\widetilde{P}(d_0, K) := 2\int_0^\infty e^{-2\varepsilon s}\frac{\|d(s)\|_2^2}{\|d_0\|_2^2}\,\mathrm{d}s. \tag{10}$$

Then $\widetilde{P}(d_0, K)$ admits the closed-form spectral decomposition:

$$\widetilde{P}(d_0, K) = \frac{d_0^\top(\widetilde{K} + \varepsilon I)^{-1}d_0}{\|d_0\|_2^2} = \sum_{i=1}^{n}\frac{p_i}{\tilde{\lambda}_i + \varepsilon}, \tag{11}$$

where $p_i = (d_0^\top u_i)^2/\|d_0\|_2^2$ is the fraction of initial demand energy aligned with the i-th eigendirection u_i of $\widetilde{K}$.

Each term $p_i/(\tilde{\lambda}_i + \varepsilon)$ in Equation 11 is the discounted cumulative residual contribution of the corresponding direction. For the same fraction of initial error, a relatively weak-response direction decays more slowly and therefore contributes more; among directions with the same response strength, the contribution increases with the fraction of demand energy they carry. Thus, even a weak direction that carries only a small fraction of the initial error can make a large cumulative contribution because of its slow decay. The parameter ε controls the temporal discounting and keeps the contribution of zero-response directions finite. In the subsequent analyses, we set $\varepsilon = 10^{-3}$ and refer to directions with $\tilde{\lambda}_i \leq 0.01$ as weak-response directions. The proof is provided in Appendix B.2.

We apply this analysis to the current state of the online critic. At each recorded checkpoint, we compute the directional energy fractions and relative spectral strengths from the Bellman demand and learning kernel at that time, and then evaluate the spectral burden according to Equation 11.

The above analysis provides an explanation for why FT in Figure 1 makes little progress in learning the new task after the task transition. As shown in Figure 2, when learning Push-wall after Sweep-into, a larger fraction of FT's Bellman correction demand lies in weak spectral directions, whereas the corresponding fraction is lower for PSP. This observation is consistent with our proposed explanation: when a new task requires the critic to make value corrections along weak spectral directions, insufficient learning responses in those directions may prevent the critic from completing the required value adjustment, potentially leading to failure in learning the new task.

5 Periodic Spectral Projection

Building on the analysis and empirical observations in Section 4, we adopt Periodic Spectral Projection (PSP), which periodically constrains the layer-weight spectra of both critics using existing two-sided singular-value clipping Jia et al. (2017). By controlling the singular-value range of the layer weights, PSP affects forward feature propagation and backward gradients, thereby influencing the parameter Jacobian and modulating critic's local learning response at the weight level.

Specifically, PSP applies a common spectral interval $[\sigma_-, \sigma_+]$ to all linear layers of both critics, where $0 < \sigma_- < \sigma_+$. For any linear layer, let the pre-projection weight matrix be $W \in \mathbb{R}^{r\times c}$, with singular value decomposition $W = U_W\,\mathrm{diag}(\sigma_1, \ldots, \sigma_{k_W})V_W^\top$, where $k_W = \min(r, c)$ and $\sigma_1 \geq \cdots \geq \sigma_{k_W} \geq 0$. We preserve the original left and right singular vectors, clip each singular value to the prescribed interval, and reconstruct the projected weight matrix as:

$$\widehat{W} = \Pi_{[\sigma_-,\sigma_+]}(W) := U_W\,\mathrm{diag}(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_{k_W})\,V_W^\top. \tag{12}$$

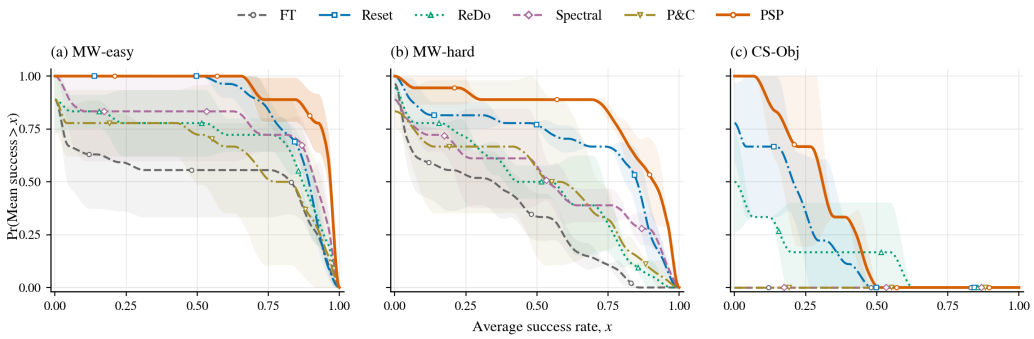


Figure 3: Performance Profiles of Mean Success Rates Across Subsequent Tasks. Each subplot summarizes performance on the subsequent tasks after training on the first task. For each subsequent task and each random seed, we first compute the mean success rate over all valid evaluation records collected during that task's training. For a success-rate threshold x, we then compute, separately for each seed, the fraction of subsequent tasks whose mean success rate is at least x. The plotted curve shows the mean of this fraction across the three seeds, and the shaded region denotes ± 1 standard deviation. Curves that remain higher across thresholds indicate that a larger fraction of subsequent tasks achieve the corresponding levels of average training performance.

Here, $\widehat{\sigma}_i = \min\{\sigma_+, \max\{\sigma_-, \sigma_i\}\}$, for $i = 1, \ldots, k_W$. This operation raises singular values below the lower bound to σ_-, lowers singular values above the upper bound to σ_+, and leaves those within the interval unchanged, thereby constraining the scaling of the linear map along singular-vector directions from both ends.

Among all weight matrices satisfying the spectral interval constraint, the clipped weight $\widehat{W}$ minimizes the Frobenius distance to the original weight matrix W:

$$\widehat{W} \in \underset{W' \in \mathcal{S}_{[\sigma_-, \sigma_+]}}{\arg\min} \frac{1}{2} \|W' - W\|_F^2, \tag{13}$$

where $\mathcal{S}_{[\sigma_-, \sigma_+]}$ denotes the set of $r \times c$ matrices whose k_W singular values all lie in $[\sigma_-, \sigma_+]$. The complete statement and proof of this projection property are provided in Lemma B.1.

We integrate this projection operation into continual SAC. The first task is trained with the original SAC procedure; starting from the second task, both critics are projected once at each task boundary and then projected again every Δ_{env} environment steps within the task. Between successive projections, training follows the original SAC updates; the complete procedure is given in Algorithm 1.

6 Experiments

We evaluate our method on two widely used CRL benchmarks: Meta-World Yu et al. (2020) and CompoSuite Mendez et al. (2022). Specifically, we use the Easy and Hard task sequences from Meta-World, each comprising 10 tasks, and the four-task CS-Obj sequence from CompoSuite. Detailed descriptions are provided in Appendix D.

6.1 Baselines

We compare against five categories of baselines: (1) continual fine-tuning (FT); (2) network resetting, Reset Nikishin et al. (2022); (3) dormant-neuron recycling, ReDo Sokar et al. (2023); (4) spectral regularization, Spectral Regularization (Spectral) Lewandowski et al. (2025); and (5) knowledge compression, Progress & Compress (P&C) Schwarz et al. (2018).

6.2 Task Acquisition across Continual Learning Sequences

Figure 3 shows that PSP attains favorable task-acquisition profiles across the evaluated continual task sequences. On both MW-hard and MW-easy, PSP covers a larger fraction of tasks at high success-

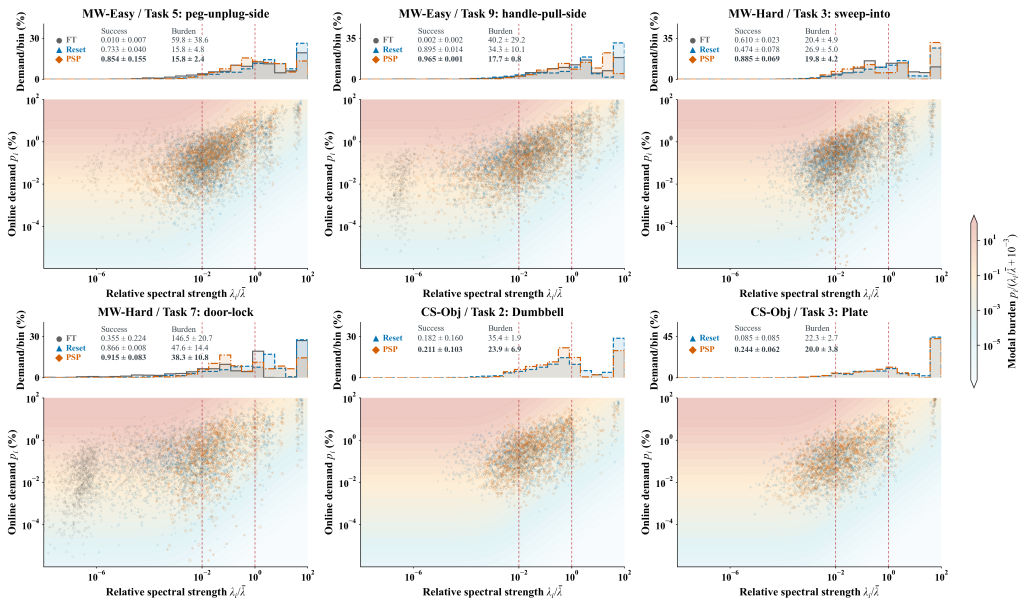


Figure 4: Bellman Demand–Spectrum Alignment in New-Task Adaptation. In Figure 4, the horizontal axis denotes the relative spectral strength $\lambda_i/\bar{\lambda}$, while the vertical axis shows the demand energy carried by each direction. The background color encodes the demand-weighted modal burden; consequently, concentrating more correction demand in weak-response directions results in a larger spectral burden. The histogram above further summarizes the distribution of demand energy across different relative spectral ranges. All results are aggregated over three seeds.

rate thresholds and exhibits better performance throughout training. On MW-hard, the fraction of task–seed mean success rates exceeding 50% reaches 88.9% for PSP, compared with 77.8% for Reset. Moreover, PSP maintains its advantage over a broad range of thresholds, indicating that its gains extend across multiple tasks rather than being concentrated in a small number of high-performing cases. On CS-Obj, the average success rates of FT, Spectral, and P&C on subsequent tasks are all zero, whereas PSP covers more tasks over the low-to-moderate threshold range and achieves an average success rate of 30.1% across tasks, exceeding 18.9% for Reset and 12.9% for ReDo. Although ReDo retains a small number of cases in the high-threshold tail, PSP provides broader gains, reflected in a larger number of tasks with nonzero and moderate performance. These results support the effectiveness of critic spectral projection for subsequent-task acquisition in the evaluated setting.

6.3 Bellman Demand–Spectrum Alignment

To further examine the critic's local correction state under different methods, we select challenging tasks from the three benchmark sequences and representative methods, and decompose the Bellman correction demand observed during training along the eigendirections of the parameter-learning kernel. In Figure 4, in addition to FT, we include the best-performing baseline for each selected task to provide a stronger reference for comparison with PSP. Propositions 4.1 and 4.2 describe how demand aligned with slow kernel directions persists in the local fixed-target model. The online measurements below characterize the directional composition of the remaining residual at each checkpoint. Across the selected tasks in Figure 4, FT or Reset, when adapting poorly, retains substantial Bellman correction demand in weak-response directions, whereas PSP is associated with a lower spectral burden and a higher average success rate over training. Conversely, cases in which PSP exhibits a higher spectral burden are also accompanied by lower success rates.

6.4 Effects of Spectral Constraints and Critic Layer Scope

Constraint Type. As shown in Figure 5a, on the MW-Hard sequence, the two-sided constraint achieves the highest average success rate over training, followed by the upper-bound-only constraint,

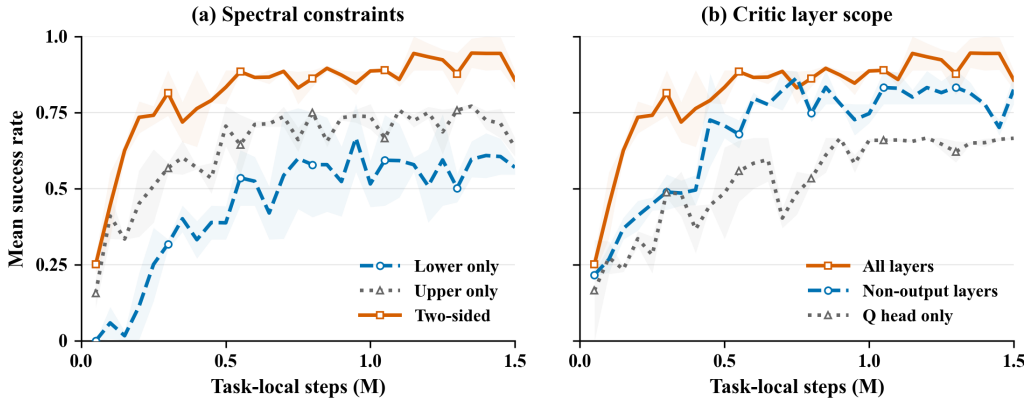


Figure 5: Effects of spectral constraints and critic layer scope on MW-Hard. (a) Comparison of lower-only, upper-only, and two-sided spectral constraints. (b) Projection applied to all critic linear layers, all non-output layers, or only the Q head. Curves show mean success across tasks 2–10 over 3 random seeds at each task-local step; shaded regions indicate the SEM across seeds.

while the lower-bound-only constraint performs substantially worse. The two-sided variant separates from both one-sided variants early and maintains its lead across most task-local steps.Constraining only the large singular values already yields markedly better task acquisition, while additionally constraining the lower end of the spectrum further improves performance. Thus, in this setting, regulating the two ends of the spectrum has a complementary effect.

Critic Layer Scope. As shown in Figure 5b, applying the two-sided projection to all linear layers of the critic yields the best task-acquisition performance, followed by applying it to the non-output layers, while restricting the projection to the Q-output head results in the slowest acquisition. This result indicates that modifying only the final value-output mapping is insufficient to reproduce the full benefit of PSP, and that spectral intervention in the critic backbone plays an important role in modulating its learning response. The average online spectral burdens under the three settings are 31.82, 38.45, and 100.73, respectively, consistent with their ordering in task-acquisition performance. In particular, intervention restricted to the Q head exhibits both slower task acquisition and a substantially higher demand-weighted burden, suggesting that extending spectral projection to the critic backbone is more effective at improving the alignment between the current Bellman correction demand and directional learning response. Additional experimental results are provided in the Appendix H.

7 Conclusion

This work provides an analytical perspective on new-task adaptation in continual reinforcement learning by examining the alignment between Bellman correction demand and the critic's directional learning response, and derives a demand-weighted spectral burden to quantify the local difficulty of correcting the current target. Analyses of demand distributions and spectral burden on Meta-World and CompoSuite support our central hypothesis: when a larger fraction of the required corrections is concentrated in weak-response directions, value correction by the critic can become more difficult. Sequence experiments further validate the effectiveness of PSP, showing that periodically regulating the critic's weight spectrum can mitigate this mismatch and improve learning performance on new tasks. These findings offer a new perspective for understanding critic plasticity from the standpoint of task-specific correction demand and for improving continual adaptation through spectral intervention.

Limitations. Future work could further investigate the joint evolution and interactions of the experience distribution, Bellman targets, and policy updates during training, providing a more complete characterization of how the critic adapts to new tasks. In addition, PSP currently relies on a predefined spectral interval and projection frequency; adaptively adjusting spectral interventions in response to changes in demand and learning response is an important direction for future investigation.

AI USE STATEMENT

In this work, we used generative AI tools solely to assist with the stylistic presentation of tables and figures, such as formatting, layout, and visual consistency, without generating or modifying their underlying data or substantive content.

REPRODUCIBILITY STATEMENT

Algorithm 1 describes the PSP update procedure, and Appendix B provides the theoretical proofs. Appendices C to E specify the baseline implementations, task sequences, interaction budgets, architectures, optimization settings, and projection schedules. Additional task-level observations and ablations appear in Appendices G and H.

REFERENCES

Zaheer Abbas, Rosie Zhao, Joseph Modayil, Adam White, and Marlos C Machado. Loss of plasticity in continual deep reinforcement learning. In *Conference on lifelong learning agents*, pp. 620–636. PMLR, 2023.

David Abel, André Barreto, Benjamin Van Roy, Doina Precup, Hado Van Hasselt, and Satinder Singh. A definition of continual reinforcement learning. *Advances in Neural Information Processing Systems*, 36:50377–50407, 2023.

Hongjoon Ahn, Jinu Hyeon, Youngmin Oh, Bosun Hwang, and Taesup Moon. Prevalence of negative transfer in continual reinforcement learning: Analyses and a simple baseline. In *International Conference on Learning Representations*, volume 2025, pp. 75036–75060, 2025.

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.

Wesley Chung, Lynn Cherif, David Meger, and Doina Precup. Parseval regularization for continual reinforcement learning. *Advances in Neural Information Processing Systems*, 37:127937–127967, 2024.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026): 768–774, 2024.

Dibya Ghosh and Marc G Bellemare. Representations for stable off-policy reinforcement learning. In *International Conference on Machine Learning*, pp. 3556–3565. PMLR, 2020.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. Pmlr, 2018a.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018b.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Kui Jia, Dacheng Tao, Shenghua Gao, and Xiangmin Xu. Improving training of deep neural networks via singular value bounding. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3994–4002. IEEE, 2017.

Christos Kaplanis, Murray Shanahan, and Claudia Clopath. Continual reinforcement learning with complex synapses. In *International Conference on Machine Learning*, pp. 2497–2506. PMLR, 2018.

Christos Kaplanis, Murray Shanahan, and Claudia Clopath. Policy consolidation for continual reinforcement learning. *arXiv preprint arXiv:1902.00255*, 2019.

Tyler Kastner, Nimrod De La Vega, and Amir-massoud Farahmand. Singularclip: Preventing spectral collapse to maintain plasticity in continual and reinforcement learning. *arXiv preprint arXiv:2608.18319*, 2026.

Samuel Kessler, Jack Parker-Holder, Philip Ball, Stefan Zohren, and Stephen J Roberts. Same state, different task: Continual reinforcement learning without interference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7143–7151, 2022.

Khimya Khetarpal, Matthew Riemer, Irina Rish, and Doina Precup. Towards continual reinforcement learning: A review and perspectives. *Journal of Artificial Intelligence Research*, 75:1401–1476, 2022.

Timo Klein, Christoph Luther, Manus McAuliffe, Lukas Miklautz, Claudia Plant, and Sebastian Tschiatschek. Plasticity loss in deep reinforcement learning: A survey. *arXiv preprint arXiv:2411.04832*, 2024.

Aviral Kumar, Rishabh Agarwal, Dibya Ghosh, and Sergey Levine. Implicit under-parameterization inhibits data-efficient deep reinforcement learning. *arXiv preprint arXiv:2010.14498*, 2020.

Alex Lewandowski, Michał Bortkiewicz, Saurabh Kumar, Andras Gyorgy, Dale Schuurmans, Mateusz Ostaszewski, and Marlos C Machado. Learning continually by spectral regularization. In *International Conference on Learning Representations*, volume 2025, pp. 34504–34530, 2025.

Clare Lyle, Mark Rowland, and Will Dabney. Understanding and preventing capacity loss in reinforcement learning. *arXiv preprint arXiv:2204.09560*, 2022a.

Clare Lyle, Mark Rowland, Will Dabney, Marta Kwiatkowska, and Yarin Gal. Learning dynamics and generalization in deep reinforcement learning. In *International conference on machine learning*, pp. 14560–14581. PMLR, 2022b.

Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In *International conference on machine learning*, pp. 23190–23211. PMLR, 2023.

Guozheng Ma, Lu Li, Sen Zhang, Zixuan Liu, Zhen Wang, Yixin Chen, Li Shen, Xueqian Wang, and Dacheng Tao. Revisiting plasticity in visual reinforcement learning: Data, modules and training stages. In *International Conference on Learning Representations*, volume 2024, pp. 19798–19822, 2024.

Sridhar Mahadevan and Mauro Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *Journal of Machine Learning Research*, 8(10), 2007.

Mikel Malagon, Josu Ceberio, and Jose A Lozano. Self-composing policies for scalable continual reinforcement learning. *arXiv preprint arXiv:2506.14811*, 2025.

Jorge A Mendez, Marcel Hussing, Meghna Gummadi, and Eric Eaton. Composuite: A compositional reinforcement learning benchmark. *arXiv preprint arXiv:2207.04136*, 2022.

Ofir Nabati, Bo Dai, Shie Mannor, and Guy Tennenholtz. Spectral bellman method: Unifying representation and exploration in rl. In *International Conference on Learning Representations*, volume 2026, pp. 53939–53958, 2026.

Evgenii Nikishin, Max Schwarzer, Pierluca D'Oro, Pierre-Luc Bacon, and Aaron Courville. The primacy bias in deep reinforcement learning. In *International conference on machine learning*, pp. 16828–16847. PMLR, 2022.

Arjun Prakash, Naicheng He, Kaicheng Guo, Saket Tiwari, Ruo Yu Tao, Tyrone Serapio, Amy Greenwald, and George Konidaris. Spectral collapse drives loss of plasticity in deep continual learning. *arXiv preprint arXiv:2509.22335*, 2025.

Tongzheng Ren, Tianjun Zhang, Lisa Lee, Joseph E Gonzalez, Dale Schuurmans, and Bo Dai. Spectral decomposition representation for reinforcement learning. *arXiv preprint arXiv:2208.09515*, 2022.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International conference on machine learning*, pp. 4528–4537. PMLR, 2018.

Ghada Sokar, Rishabh Agarwal, Pablo Samuel Castro, and Utku Evci. The dormant neuron phenomenon in deep reinforcement learning. In *International Conference on Machine Learning*, pp. 32145–32168. PMLR, 2023.

Richard S Sutton, Andrew G Barto, and Andrew Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Maciej Wołczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. Continual world: A robotic benchmark for continual reinforcement learning. *Advances in Neural Information Processing Systems*, 34:28496–28510, 2021.

Maciej Wolczyk, Michał Zając, Razvan Pascanu, Łukasz Kuciński, and Piotr Miłoś. Disentangling transfer in continual reinforcement learning. *Advances in Neural Information Processing Systems*, 35:6304–6317, 2022.

Yifan Wu, George Tucker, and Ofir Nachum. The laplacian in rl: Learning representations with efficient approximations. *arXiv preprint arXiv:1810.04586*, 2018.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020.

Tiantian Zhang, Kevin Zehua Shen, Zichuan Lin, Bo Yuan, Xueqian Wang, Xiu Li, and Deheng Ye. Replay-enhanced continual reinforcement learning. *arXiv preprint arXiv:2311.11557*, 2023.

Qiyang Zhou, Xu Ruihang, Peng Wang, WenJie Lu, Xiaochun Cao, Naiqiang Tan, and Li Shen. Htac: Hierarchical task-aware composition for continual offline reinforcement learning. In *Forty-third International Conference on Machine Learning*.

A ALGORITHMIC DETAILS OF PSP IN CONTINUAL SAC

Algorithm 1 provides the complete execution procedure of PSP in continual SAC. The first task is trained using standard SAC; starting from the second task, spectral projection is applied to both critics at each task boundary, after which the projected online critics are synchronized with their corresponding target critics. During within-task training, PSP projects the critic weights again according to a predefined environment-step interval, while all other updates to the actor, critics, temperature parameter, and target networks follow the original SAC procedure unchanged. The projection is performed layer by layer by computing an SVD, clipping the singular values to the interval $[\sigma_-, \sigma_+]$, and reconstructing the corresponding weight matrix; biases, actor parameters, and optimizer states are left unchanged.

Algorithm 1 Periodic Spectral Projection (PSP) in Continual SAC

Require: Task sequence $\{\mathcal{M}^{(k)}\}_{k=1}^{T}$, interaction budgets $\{N_k\}_{k=1}^{T}$, spectral interval $[\sigma_-, \sigma_+]$, projection period Δ_{env}
Ensure: Policy π_ϕ and twin critics $Q_{\theta_1}, Q_{\theta_2}$

Initialize SAC networks, temperature parameters, replay buffer, and optimizer states
for $k = 1, \ldots, T$ **do**
  Begin task $\mathcal{M}^{(k)}$ following the base SAC configuration
  $t_{\text{env}} \leftarrow 0$, $t_{\text{next}} \leftarrow \Delta_{\text{env}}$
  if $k > 1$ **then**
    $\theta_j \leftarrow \mathcal{P}(\theta_j), \quad j \in \{1, 2\}$
    $\bar{\theta}_j \leftarrow \theta_j, \quad j \in \{1, 2\}$
  end if
  while $t_{\text{env}} < N_k$ **do**
    Collect experience for Δt environment steps and store it in replay buffer $\mathcal{D}$
    $t_{\text{env}} \leftarrow t_{\text{env}} + \Delta t$
    $G \leftarrow$ number of critic updates scheduled by base SAC for this iteration
    for $g = 1, \ldots, G$ **do**
      $\mathcal{B} \sim \mathcal{D}$
      $\theta_j \leftarrow \mathrm{Adam}_{Q_j}\Big(\theta_j, \nabla_{\theta_j} \widehat{L}_{Q_j}(\theta_j; \mathcal{B})\Big), \quad j \in \{1, 2\}$
      if $k > 1 \wedge t_{\text{env}} \geq t_{\text{next}} \wedge g = G$ **then**
        $\theta_j \leftarrow \mathcal{P}(\theta_j), \quad j \in \{1, 2\}$
        $t_{\text{next}} \leftarrow \left(\left\lfloor \frac{t_{\text{env}}}{\Delta_{\text{env}}} \right\rfloor + 1\right) \Delta_{\text{env}}$
      end if
      if an actor update is due under the base SAC schedule **then**
        $\phi \leftarrow \mathrm{Adam}_{\pi}\Big(\phi, \nabla_{\phi} \widehat{L}_{\pi}(\phi; \mathcal{B})\Big)$
      end if
      if a temperature update is due under the base SAC schedule **then**
        $\beta \leftarrow \mathrm{Adam}_{\alpha}\Big(\beta, \nabla_{\beta} \widehat{L}_{\alpha}(\beta; \mathcal{B})\Big)$
      end if
      if a target critic update is due under the base SAC schedule **then**
        $\bar{\theta}_j \leftarrow (1 - \tau)\bar{\theta}_j + \tau\theta_j, \quad j \in \{1, 2\}$
      end if
    end for
  end while
end for
return $\pi_\phi, Q_{\theta_1}, Q_{\theta_2}$

B Proofs of Main Theoretical Results

B.1 Proof of Proposition 4.1

Proposition 4.1. Consider an affine critic evaluated on a fixed input set,

$$q_\theta = q_0 + J(\theta - \theta_0), \tag{14}$$

where $q_0 \in \mathbb{R}^n$ and $J \in \mathbb{R}^{n\times p}$ are fixed. Let the target $y \in \mathbb{R}^n$ be fixed, and define the nonzero initial demand as

$$d_0 = y - q_0 \neq 0. \tag{15}$$

Starting from θ_0, perform full-batch gradient descent with step size $\eta > 0$ on the loss

$$L_Q(\theta) = \frac{1}{2n} \|q_\theta - y\|_2^2 . \tag{16}$$

Let $d_h = y - q_{\theta_h}$ and define

$$K = \frac{1}{n} JJ^\top = U \operatorname{diag}(\lambda_1, \ldots, \lambda_n) U^\top, \tag{17}$$

where $U = [u_1, \ldots, u_n]$ is orthogonal. Then, for any $H \in \mathbb{N}_0$,

$$\frac{\|d_H\|_2^2}{\|d_0\|_2^2} = \sum_{i=1}^{n} p_i (1 - \eta\lambda_i)^{2H}, \qquad p_i = \frac{|u_i^\top d_0|^2}{\|d_0\|_2^2}. \tag{18}$$

Proof. Since both the target y and the Jacobian J are fixed, the gradient of the critic loss at iteration h is

$$\nabla_\theta L_Q(\theta_h) = \frac{1}{n} J^\top (q_{\theta_h} - y) \tag{19}$$

$$= -\frac{1}{n} J^\top d_h. \tag{20}$$

The corresponding gradient-descent update is therefore

$$\theta_{h+1} - \theta_h = -\eta \nabla_\theta L_Q(\theta_h) = \frac{\eta}{n} J^\top d_h. \tag{21}$$

By the affine form of the critic, the corresponding change in its output is

$$q_{\theta_{h+1}} - q_{\theta_h} = J(\theta_{h+1} - \theta_h) \tag{22}$$

$$= \frac{\eta}{n} JJ^\top d_h \tag{23}$$

$$= \eta K d_h. \tag{24}$$

Consequently, the demand evolves as

$$d_{h+1} = y - q_{\theta_{h+1}} \tag{25}$$

$$= y - q_{\theta_h} - \left(q_{\theta_{h+1}} - q_{\theta_h}\right) \tag{26}$$

$$= d_h - \eta K d_h \tag{27}$$

$$= (I - \eta K) d_h. \tag{28}$$

Recursively applying equation 28 from d_0 gives

$$d_H = (I - \eta K)^H d_0. \tag{29}$$

Because K is a real symmetric positive semidefinite matrix, its eigenvectors form an orthonormal basis. Expanding d_0 in this basis gives

$$d_0 = \sum_{i=1}^{n} (u_i^\top d_0) u_i. \tag{30}$$

Using $Ku_i = \lambda_i u_i$ in equation 29, we obtain

$$d_H = \sum_{i=1}^{n} (1 - \eta\lambda_i)^H (u_i^\top d_0) u_i. \tag{31}$$

By orthonormality,

$$\|d_H\|_2^2 = \sum_{i=1}^{n} (1 - \eta\lambda_i)^{2H} |u_i^\top d_0|^2. \tag{32}$$

Dividing both sides of equation 32 by $\|d_0\|_2^2 > 0$ yields equation 18, corresponding to equation 8 in the main text.

When $H = 0$, we have $(I - \eta K)^0 = I$, so every directional retention factor is equal to one. Moreover, when $0 \le \eta\lambda_i \le 1$, the retention factor $(1 - \eta\lambda_i)^{2H}$ is monotonically non-increasing in λ_i for any fixed $H \ge 1$. Thus, directions associated with smaller eigenvalues retain a larger fraction of their initial demand. In particular, if $\lambda_i = 0$, then

$$u_i^\top d_H = u_i^\top d_0, \tag{33}$$

so the demand component along u_i is not reduced by gradient descent. □

B.2 Proof of Proposition 4.2

Proposition 4.2. Under the fixed-target affine model of Proposition 4.1, let $d_0 \neq 0$, $\bar{\lambda} = \mathrm{tr}(K)/n > 0$, and $\varepsilon > 0$. Define

$$\widetilde{K} = \frac{K}{\bar{\lambda}}, \qquad \widetilde{\lambda}_i = \frac{\lambda_i}{\bar{\lambda}}. \tag{34}$$

Let $d(s)$ satisfy

$$\frac{\mathrm{d}}{\mathrm{d}s} d(s) = -\widetilde{K} d(s), \qquad d(0) = d_0. \tag{35}$$

Then the discounted cumulative residual defined in equation 10 of the main text satisfies

$$2 \int_0^\infty e^{-2\varepsilon s} \frac{\|d(s)\|_2^2}{\|d_0\|_2^2} \,\mathrm{d}s = \sum_{i=1}^{n} \frac{p_i}{\widetilde{\lambda}_i + \varepsilon} = \frac{d_0^\top \left(\widetilde{K} + \varepsilon I\right)^{-1} d_0}{\|d_0\|_2^2}, \tag{36}$$

where

$$p_i = \frac{|u_i^\top d_0|^2}{\|d_0\|_2^2}. \tag{37}$$

Proof. We first establish the correspondence between the time normalization and the loss dynamics. Applying parameter gradient flow to the loss in Proposition 4.1 gives

$$\frac{\mathrm{d}}{\mathrm{d}t} \theta(t) = -\nabla_\theta L_Q\big(\theta(t)\big). \tag{38}$$

Because the target y and the Jacobian J are fixed, the gradient of the loss is

$$\nabla_\theta L_Q\big(\theta(t)\big) = -\frac{1}{n} J^\top d(t). \tag{39}$$

It follows that

$$\frac{\mathrm{d}}{\mathrm{d}t} \theta(t) = \frac{1}{n} J^\top d(t). \tag{40}$$

Therefore, using $d(t) = y - q_{\theta(t)}$ and the affine form of the critic, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t} d(t) = -J \frac{\mathrm{d}}{\mathrm{d}t} \theta(t) \tag{41}$$

$$= -\frac{1}{n} J J^\top d(t) \tag{42}$$

$$= -K d(t). \tag{43}$$

Let $s = \bar{\lambda}t$. By the chain rule, the reparameterized residual trajectory satisfies

$$\frac{\mathrm{d}}{\mathrm{d}s}d(s) = \frac{1}{\bar{\lambda}}\frac{\mathrm{d}}{\mathrm{d}t}d(t) \tag{44}$$

$$= -\frac{K}{\bar{\lambda}}d(s) \tag{45}$$

$$= -\widetilde{K}d(s), \tag{46}$$

which recovers equation 35.

Because $\widetilde{K} = K/\bar{\lambda}$, the matrices $\widetilde{K}$ and K share the same orthonormal eigenvectors. In particular,

$$\widetilde{K}u_i = \widetilde{\lambda}_i u_i. \tag{47}$$

Projecting equation 35 onto u_i gives

$$\frac{\mathrm{d}}{\mathrm{d}s}\big(u_i^\top d(s)\big) = -\widetilde{\lambda}_i u_i^\top d(s). \tag{48}$$

Applying the initial condition $d(0) = d_0$ yields

$$u_i^\top d(s) = e^{-\widetilde{\lambda}_i s}u_i^\top d_0. \tag{49}$$

By orthonormality of the eigenvectors,

$$\frac{\|d(s)\|_2^2}{\|d_0\|_2^2} = \sum_{i=1}^{n} e^{-2\widetilde{\lambda}_i s}\frac{|u_i^\top d_0|^2}{\|d_0\|_2^2} \tag{50}$$

$$= \sum_{i=1}^{n} p_i e^{-2\widetilde{\lambda}_i s}. \tag{51}$$

Since $\widetilde{\lambda}_i \geq 0$ and $\varepsilon > 0$, each discounted integral converges. Because the sum contains only finitely many terms, we may integrate term by term:

$$2\int_0^\infty e^{-2\varepsilon s}\frac{\|d(s)\|_2^2}{\|d_0\|_2^2}\,\mathrm{d}s = 2\sum_{i=1}^{n} p_i \int_0^\infty e^{-2(\widetilde{\lambda}_i+\varepsilon)s}\,\mathrm{d}s \tag{52}$$

$$= \sum_{i=1}^{n}\frac{p_i}{\widetilde{\lambda}_i + \varepsilon}. \tag{53}$$

Finally, all eigenvalues of $\widetilde{K} + \varepsilon I$ are strictly positive because $\widetilde{\lambda}_i \geq 0$ and $\varepsilon > 0$. Hence, $\widetilde{K} + \varepsilon I$ is invertible, with

$$\big(\widetilde{K} + \varepsilon I\big)^{-1} = U\,\mathrm{diag}\left(\frac{1}{\widetilde{\lambda}_1 + \varepsilon}, \ldots, \frac{1}{\widetilde{\lambda}_n + \varepsilon}\right)U^\top. \tag{54}$$

Substituting equation 54 into the associated quadratic form gives

$$\frac{d_0^\top\big(\widetilde{K} + \varepsilon I\big)^{-1}d_0}{\|d_0\|_2^2} = \sum_{i=1}^{n}\frac{|u_i^\top d_0|^2}{\|d_0\|_2^2\big(\widetilde{\lambda}_i + \varepsilon\big)} \tag{55}$$

$$= \sum_{i=1}^{n}\frac{p_i}{\widetilde{\lambda}_i + \varepsilon}. \tag{56}$$

Together with equation 53, this establishes equation 36, corresponding to equation 11 in the main text. □

B.3 Proof of the Spectral Projection Property

Lemma B.1 (Spectral-Interval Projection Kastner et al. (2026)). *Let $W \in \mathbb{R}^{r\times c}$, $k_W = \min\{r, c\}$, and $0 < \sigma_- < \sigma_+ < \infty$. Consider a compact singular value decomposition containing all k_W singular values:*

$$W = U_W\,\mathrm{diag}\big(\sigma_1, \ldots, \sigma_{k_W}\big)V_W^\top, \tag{57}$$

where

$$U_W^\top U_W = V_W^\top V_W = I_{k_W}, \tag{58}$$

and

$$\sigma_1 \geq \cdots \geq \sigma_{k_W} \geq 0. \tag{59}$$

For each $i \in \{1, \ldots, k_W\}$, define

$$\widehat{\sigma}_i = \min\{\sigma_+, \max\{\sigma_-, \sigma_i\}\}, \tag{60}$$

and let

$$\widehat{W} = U_W \operatorname{diag}\big(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_{k_W}\big) V_W^\top. \tag{61}$$

Let $\mathcal{S}_{[\sigma_-,\sigma_+]}$ denote the set of $r \times c$ matrices whose k_W singular values all lie in $[\sigma_-, \sigma_+]$. Then

$$\widehat{W} \in \operatorname*{arg\,min}_{W' \in \mathcal{S}_{[\sigma_-,\sigma_+]}} \frac{1}{2} \|W' - W\|_F^2. \tag{62}$$

Proof. Take any $W' \in \mathcal{S}_{[\sigma_-,\sigma_+]}$, and denote its singular values in non-increasing order by $s_1, \ldots, s_{k_W}$. By von Neumann's trace inequality,

$$\operatorname{tr}\big((W')^\top W\big) \leq \sum_{i=1}^{k_W} s_i \sigma_i. \tag{63}$$

Moreover, the Frobenius norm is invariant under orthogonal transformations, so

$$\|W'\|_F^2 = \sum_{i=1}^{k_W} s_i^2, \qquad \|W\|_F^2 = \sum_{i=1}^{k_W} \sigma_i^2. \tag{64}$$

Using equation 63 and equation 64, we obtain

$$\begin{aligned}
\frac{1}{2}\|W' - W\|_F^2 &= \frac{1}{2}\|W'\|_F^2 + \frac{1}{2}\|W\|_F^2 - \operatorname{tr}\big((W')^\top W\big) \\
&\geq \frac{1}{2} \sum_{i=1}^{k_W} \big(s_i^2 + \sigma_i^2 - 2 s_i \sigma_i\big) \\
&= \frac{1}{2} \sum_{i=1}^{k_W} (s_i - \sigma_i)^2 \\
&\geq \frac{1}{2} \sum_{i=1}^{k_W} (\widehat{\sigma}_i - \sigma_i)^2.
\end{aligned} \tag{65}$$

The final inequality follows because $\widehat{\sigma}_i$ is the Euclidean projection of σ_i onto $[\sigma_-, \sigma_+]$, namely,

$$\widehat{\sigma}_i \in \operatorname*{arg\,min}_{s \in [\sigma_-,\sigma_+]} \frac{1}{2}(s - \sigma_i)^2. \tag{66}$$

The clipping map

$$s \longmapsto \min\{\sigma_+, \max\{\sigma_-, s\}\} \tag{67}$$

is monotonically non-decreasing. Therefore,

$$\widehat{\sigma}_1 \geq \cdots \geq \widehat{\sigma}_{k_W}, \tag{68}$$

and every $\widehat{\sigma}_i$ lies in $[\sigma_-, \sigma_+]$. Hence,

$$\widehat{W} \in \mathcal{S}_{[\sigma_-,\sigma_+]}. \tag{69}$$

Finally, because U_W and V_W have orthonormal columns,

$$\begin{aligned}
\frac{1}{2}\left\|\widehat{W} - W\right\|_F^2 &= \frac{1}{2}\left\|U_W \operatorname{diag}(\widehat{\sigma}_1 - \sigma_1, \ldots, \widehat{\sigma}_{k_W} - \sigma_{k_W})\, V_W^\top\right\|_F^2 \\
&= \frac{1}{2} \sum_{i=1}^{k_W} (\widehat{\sigma}_i - \sigma_i)^2.
\end{aligned} \tag{70}$$

Thus, $\widehat{W}$ attains the lower bound in equation 65 over all matrices in $\mathcal{S}_{[\sigma_-,\sigma_+]}$. This proves equation 62, corresponding to equation 13 in the main text. □

If W has zero singular values, the corresponding left and right singular vectors need not be unique. Nevertheless, choosing any compatible orthonormal singular-vector bases and applying the construction in equation 61 still yields an optimizer. Therefore, the conclusion in equation 62 is stated using set membership rather than equality with a unique minimizer.

For the multiple weight matrices projected in Algorithm 1, the layerwise constraints are independent, and the total squared parameter change is the sum of the layerwise squared changes. Applying Lemma B.1 independently to each projected layer therefore minimizes the total squared change over all feasible projected weight matrices.

C Baseline Implementations

FT. Continual fine-tuning introduces no additional continual-learning mechanism. After each task transition, both the actor and critics inherit the parameters obtained at the end of the previous task and continue training on the new task using the standard SAC objectives. This baseline measures the model's ability to learn subsequent tasks under uninterrupted continual optimization.

Reset. Reset Nikishin et al. (2022) restores subsequent learning capacity by reinitializing network parameters. At each task transition, we restore both online critics to their respective initial parameters from the beginning of training, clear the corresponding Adam optimizer states, and synchronize the reset parameters to the target critics. The actor parameters are left unchanged, restricting the intervention to the critics' ability to relearn.

ReDo. ReDo Sokar et al. (2023) maintains network plasticity by periodically recycling low-activity hidden neurons. Every 1,000 environment steps, we evaluate hidden neurons in the actor and both critics using normalized mean absolute activation as the activity measure. For neurons with activity no greater than 0.1, we reinitialize their incoming connections, set the corresponding outgoing connections to zero, and clear the associated optimizer states, allowing these neurons to participate in subsequent learning again.

Spectral. Spectral Regularization Lewandowski et al. (2025) constrains network parameters by adding spectral regularization terms to the training objectives. We regularize every linear layer of the actor and both critics by driving the largest singular value toward 1 and the bias toward 0. A regularization coefficient of 10^{-4} is used for all three networks, and the largest singular value is estimated with one power iteration at each update. Unlike the periodic hard projection used by PSP, this method continuously constrains the weight spectrum throughout training via soft regularization.

P&C. Progress & Compress (P&C) Schwarz et al. (2018) performs continual learning by alternating between knowledge acquisition and knowledge compression. The active network first learns the current task while leveraging information stored in the knowledge base. At each task transition, the knowledge acquired by the active network is distilled into the knowledge base, while a consolidation regularizer limits changes to previously acquired knowledge. The active column and adapters are then reinitialized for learning the next task. We set the consolidation regularization coefficient of the knowledge base to 1.

D Benchmark Details

Meta-World. Meta-World Yu et al. (2020) is a continuous-control benchmark for robotic manipulation, comprising a diverse set of tasks with different objectives and interaction patterns. Inspired by Ahn et al. (2025),we construct two continual-learning sequences, **MW-Easy** and **MW-Hard**, using tasks from Meta-World v2, with 10 tasks in each sequence. Tasks are trained sequentially in a predefined order, with 1.5×10^6 environment interaction steps allocated to each task. Although the tasks share the same robotic control interface, they differ in manipulated objects, target locations, and manipulation skills, requiring the inherited actor and critics to continually adapt to new control objectives.

MW-Easy: `button-press` → `plate-slide-back-side` → `window-close` → `plate-slide-side` → `peg-unplug-side` → `plate-slide-back` → `coffee-button` → `window-open` → `handle-pull-side` → `door-close`,

MW-Hard: `plate-slide-back-side` → `soccer` → `sweep-into` → `handle-pull-side` → `plate-slide-side` → `peg-unplug-side` → `door-lock` → `reach` → `plate-slide-back` → `coffee-button`.

CompoSuite. CompoSuite Mendez et al. (2022) constructs robotic manipulation tasks by composing robots, manipulated objects, obstacles, and task objectives. We construct the **CS-Obj** sequence by varying only the manipulated object while keeping all other factors fixed: we use the IIWA robot with a joint controller, fix the task objective to PickPlace, and adopt an obstacle-free setting. The four tasks use **Box, Dumbbell, Plate, and Hollowbox** in sequence, so this setting primarily evaluates continual adaptation to changes in the manipulated object while the task objective remains unchanged. Each object is trained for 1.0×10^6 environment steps. CS-Obj uses state observations and dense rewards. The observation includes information about the object, target, robot state, and environment-provided component information. The action space is 8-dimensional, and each episode is limited to at most 500 environment steps.

Continual-learning protocol. Continual-learning protocol. All sequences are trained task by task in a fixed order; once the interaction budget for the current task is exhausted, training immediately proceeds to the next task, with task boundaries known to the agent. The two Meta-World sequences induce continual adaptation across manipulation skills by changing the robotic manipulation task, whereas CS-Obj varies only the manipulated object while keeping the manipulation objective fixed, thereby providing a more controlled form of task variation.

E Hyperparameters and Experimental Details

Our implementation uses soft actor-critic (SAC) with two online critics and their corresponding target networks. PSP is applied to the online critic weights at task boundaries and, where specified, periodically within each task. Table 1 summarizes the training settings, and Table 2 lists the sequence-specific PSP configurations.

E.1 Architecture and Optimization

Both critics use multilayer perceptrons with two hidden layers of 256 units each. In Meta-World, the actor uses a shared two-layer ReLU backbone of the same width and task-specific Gaussian policy output heads selected by the task position. The networks are optimized using Adam.

For Meta-World, the replay buffer is cleared at each task transition, and critic updates begin after collecting 10^4 transitions from the new task. The entropy temperature and its optimizer are reinitialized at task boundaries. The target entropy is the negative action dimensionality. Target critics follow the online critics through Polyak updates,

$$\bar{\theta} \leftarrow (1-\tau)\bar{\theta} + \tau\theta.$$

Bootstrapping is disabled for terminal transitions and retained for time-limit truncations.

E.2 PSP Configuration

PSP acts on all linear weight matrices in both online critics, including the input layer, hidden layer, and scalar Q-value output layer. Within a run, all affected layers use the same absolute singular-value bounds. The first task is trained without projection; interventions begin at the second task.

At task entry, projection is followed by a hard synchronization of the target critics with the online critics. Within a task, periodic projection is performed after the final critic update of the current data-collection round and before the actor update. The target critics subsequently follow the standard

Table 1: SAC training and evaluation settings.

Hyperparameter	Meta-World	CompoSuite
Architecture and Optimization		
Number of online critics	2	2
Number of target critics	2	2
Critic hidden dimensions	$256, 256$	$256, 256$
Optimizer	Adam	Adam
Learning rate	3×10^{-4}	3×10^{-4}
Batch size	64	256
Update-to-data ratio	1	1
Entropy temperature	Automatic	Automatic
Discount factor γ	0.99	0.99
Reward scale	1	1
Target update coefficient τ	0.005	0.005
Replay buffer capacity	10^6	10^6
Replay warm-up steps per task	10^4	10^4
Training Budget and Evaluation		
Tasks per sequence	10	4
Environment steps per task	1.5×10^6	10^6
Evaluation episodes	50	50
Evaluation policy	Deterministic	Deterministic

Table 2: PSP configurations used in the reported experiments.

Sequence	Spectral interval	Schedule
MW-easy	$[0.25, 4]$	Entry only
MW-hard	$[0.0625, 16]$	Entry + 200k
CS-Obj	$[0.25, 4]$	Entry + 400k

Polyak update. The projection clock restarts at each task boundary and includes replay warm-up interactions.

Projection preserves the singular vectors of each weight matrix and clips its singular values to the specified interval. Critic biases, actor parameters, and Adam optimizer states are unchanged by the projection operation. No additional regularization loss or distillation stage is introduced.

F Relation to Continual SAC and Spectral Plasticity Studies

Continual SAC and the critic. Component-wise analyses of continual SAC have examined how the actor, critic, exploration, and data contribute to transfer Wolczyk et al. (2022). A module-level study in visual reinforcement learning has also identified critic plasticity as a bottleneck in its evaluated setting Ma et al. (2024). We examine the correction required of the critic at a new task: how the current Bellman target differs from its prediction, where that difference lies among the critic's learning directions, and how strongly the critic responds along those directions.

Spectral analyses of learning. Spectral properties of transition operators and Bellman structure have been used to construct value representations Mahadevan & Maggioni (2007); Wu et al. (2018); Ren et al. (2022); Nabati et al. (2026), while analyses of temporal-difference learning relate spectral components to stability and learning speed Ghosh & Bellemare (2020); Lyle et al. (2022b). Kernel task–model alignment shows that learning depends on the distribution of target energy across kernel eigendirections as well as their eigenvalues Canatar et al. (2021). Spectral Collapse likewise studies residual energy in fast kernel eigenspaces under a finite update budget in continual learning Prakash et al. (2025). Our analysis applies this residual-conditioned view to the current soft Bellman target in continual SAC. Our demand-weighted spectral burden in Equation 11 characterizes the relative directional mismatch between the current Bellman correction demand and the critic's learning response, separately from residual magnitude and the overall kernel scale.

Weight-spectrum interventions. Chung et al. (2024) regularization encourages weight-matrix orthogonality to preserve optimization properties in continual reinforcement learning . Kastner et al. (2026) studies plasticity through weight-spectrum anisotropy and gradient propagation. PSP applies two-sided projection to the twin critics of continual SAC at task boundaries and, when enabled, at scheduled points within each task, leaving the actor unchanged. Our focus is the task-specific value correction: we analyze how Bellman correction demand aligns with the eigendirections of the critic's parameter-learning kernel, and use the demand-weighted spectral burden to quantify the relative directional difficulty of that correction.

G Online Spectral Burden and Task-Level Learning Progres

New-task adaptation depends on whether the critic can complete the required Bellman corrections in a timely manner. When the correction demand is concentrated in weak-response directions of the learning kernel, value updates become slower and the new task becomes more difficult to learn. To examine this relationship in greater detail, we report, on a task-by-task basis, the average success rate over training and the demand-weighted spectral burden for all six methods on both the easy and hard sequences, directly comparing the critic's local correction state with its actual task performance.

MW-Easy Sequence As shown in the Figure 6, on the easy sequence, PSP achieves both higher average success rates over training and lower spectral burdens on most subsequent tasks. On window-close, PSP achieves a success rate of 0.972 with a spectral burden of 15.37, compared with 0.722 and 45.16 for FT. The contrast becomes more pronounced on harder tasks: on peg-unplug-side, PSP achieves a success rate of 0.678 with a spectral burden of 13.41, whereas FT, ReDo, and P&C attain success rates close to zero, with corresponding spectral burdens of 59.83, 138.25, and 68.88. On handle-pull-side, PSP reaches a success rate of 0.970 with a spectral burden of 18.52, whereas FT and P&C achieve success rates of only 0.002 and 0.433, respectively. These task-level results indicate that the gains from PSP are not confined to a small number of easy tasks; across multiple task transitions, a lighter current Bellman correction burden is accompanied by better task acquisition.

ReDo exhibits a different adaptation pattern. On plate-slide-back and coffee-button, ReDo achieves success rates of 0.886 and 0.933 despite relatively high spectral burdens of 101.59 and 134.74, respectively. ReDo periodically recycles low-activity neurons in both the actor and critics, thereby changing the network parameters used in subsequent training as well as the critic's learning response. The spectral burden characterizes the current critic's local response to the Bellman correction demand, whereas ReDo repeatedly modifies neuronal connections, network representations, and the parameter Jacobian during subsequent training while simultaneously updating the actor. Consequently, a high spectral burden at a given checkpoint does not directly characterize the model's adaptation capability over the remainder of training on that task. ReDo can therefore exhibit a high instantaneous spectral burden while still gradually adapting to the new task through subsequent neuron recycling and policy updates.

MW-Hard Sequence The hard sequence further widens the performance gap across methods on individual tasks. On door-lock, for example, PSP achieves an average success rate of 0.918 over training with a spectral burden of 38.30, whereas FT, ReDo, Spectral, and P&C all achieve success rates no higher than 0.392, with spectral burdens of 146.44, 209.90, 141.67, and 264.04, respectively. A similar pattern appears on plate-slide-back: PSP reaches a success rate of 0.936 with a spectral burden of 18.09, while FT, ReDo, and P&C achieve success rates of 0.628, 0.565, and 0.624, respectively, all with higher burdens. Under the more challenging task ordering, PSP maintains a lighter current value-correction burden on multiple tasks, accompanied by higher average success rates throughout training.

H Additional Ablations

We conduct a more extensive ablation study on MW-Hard, as detailed below.

Spectral interval across projection periods. Figure 8 compares the task-acquisition trajectories under different spectral intervals and projection periods. Across all three projection periods, $[1/16, 16]$ achieves the highest average success rate over training. Its advantage is most pronounced with a 200k projection period, where it reaches 81.78%, compared with 69.26% for $[1/4, 4]$ and 58.77%

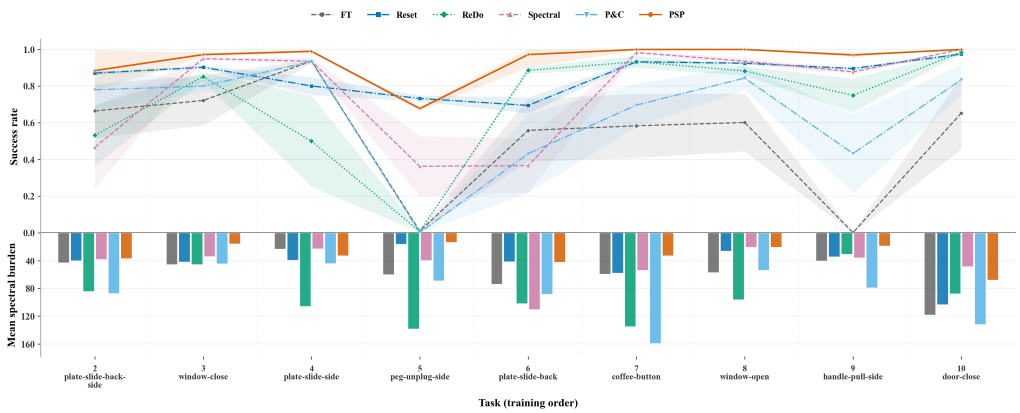


Figure 6: Task-wise success rate and spectral burden in the MW-Easy Sequence. We consider only tasks 2–10 in training order, with all results aggregated over three random seeds. Lines show mean success rates, shaded regions show the mean ± standard deviation, and downward bars show mean spectral burden.

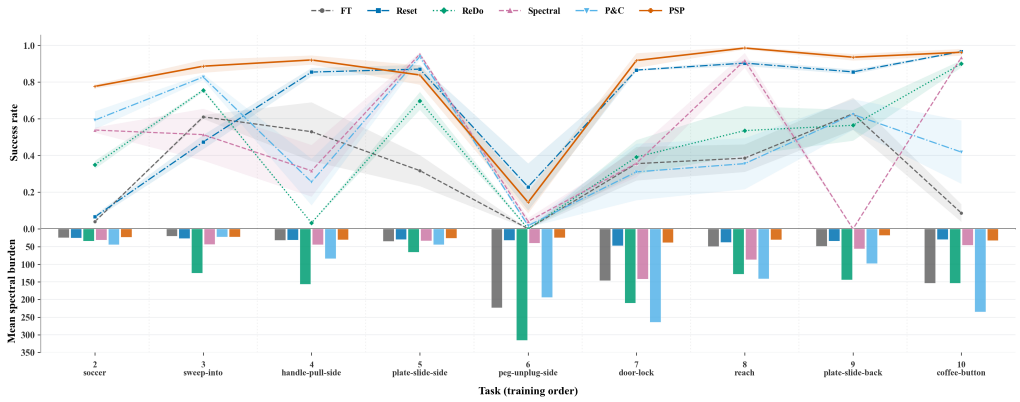


Figure 7: Task-wise success rate and spectral burden in the MW-Hard Sequence. We consider only tasks 2–10 in training order, with all results aggregated over three random seeds. Lines show mean success rates, shaded regions show the mean ± standard deviation, and downward bars show mean spectral burden.

for $[1/8, 8]$. With the spectral interval fixed to $[1/16, 16]$, the 200k period yields the highest average success rate over training, whereas the 400k period achieves the highest final-task success rate of 92.56%. The ordering between the intermediate interval $[1/8, 8]$ and the narrower interval $[1/4, 4]$ changes with the projection period, indicating that the spectral interval and projection timing jointly shape the learning trajectories on subsequent tasks.

Spectral interval across projection periods. Table 3 compares three spectral intervals under different projection schedules. The $[1/16, 16]$ interval achieves the highest mean process success at every periodic schedule. Its advantage is most pronounced at 200k: process success reaches 0.82 ± 0.04, compared with 0.69 ± 0.11 for $[1/4, 4]$ and 0.59 ± 0.08 for $[1/8, 8]$. The ordering of the two narrower intervals changes with the projection period, indicating that interval width and projection frequency jointly shape the acquisition trajectory. For $[1/16, 16]$, the 200k schedule gives the highest process success, exceeding both 100k (0.71 ± 0.07) and 400k (0.77 ± 0.01). The 400k schedule attains a higher task-end mean (0.93 ± 0.05 versus 0.89 ± 0.04 at 200k), but learns less

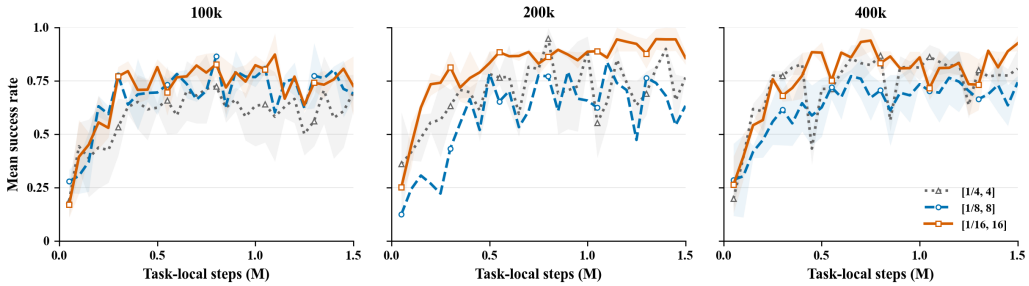


Figure 8: Spectral-interval sensitivity across projection periods on MW-Hard. The left, middle, and right panels use task-local projection periods of 100k, 200k, and 400k environment steps, respectively. All variants also project at task entry. Curves show the mean success rate across tasks 2–10 at each task-local step without smoothing. All results are averaged over three seeds, and shaded regions indicate SEM.

Table 3: Task-acquisition and task-end success across spectral intervals and projection schedules on MW-Hard. Values are mean ± standard deviation across 3 independent seeds.

Spectral interval	Projection schedule	Seeds	Process success	Task-end success
$[1/4, 4]$	Entry only	3	0.79 ± 0.10	0.93 ± 0.14
$[1/4, 4]$	Entry + 100k	3	0.59 ± 0.17	0.69 ± 0.12
$[1/4, 4]$	Entry + 200k	3	0.69 ± 0.11	0.77 ± 0.00
$[1/4, 4]$	Entry + 400k	3	0.73 ± 0.02	0.81 ± 0.15
$[1/8, 8]$	Entry only	–	–	–
$[1/8, 8]$	Entry + 100k	3	0.68 ± 0.02	0.69 ± 0.03
$[1/8, 8]$	Entry + 200k	3	0.59 ± 0.08	0.63 ± 0.12
$[1/8, 8]$	Entry + 400k	3	0.64 ± 0.13	0.74 ± 0.04
$[1/16, 16]$	Entry only	3	0.74 ± 0.08	0.89 ± 0.10
$[1/16, 16]$	Entry + 100k	3	0.71 ± 0.07	0.73 ± 0.19
$[1/16, 16]$	Entry + 200k	3	0.82 ± 0.04	0.89 ± 0.04
$[1/16, 16]$	Entry + 400k	3	0.77 ± 0.01	0.93 ± 0.05

Note. Process success averages 30 evaluations from 50k to 1.5M task-local steps over tasks 2–10. Task-end success averages the nine evaluations at 1.5M steps. Standard deviations are calculated across independent seed-level summaries using the $n - 1$ denominator.

successfully over the full task budget. At the same interval, entry-only projection attains a process mean of 0.74 ± 0.08 and a task-end mean of 0.89 ± 0.10. The 200k periodic configuration therefore has a higher observed process mean, while both schedules have the same task-end mean of 0.89. The reported MW-Hard configuration uses $[1/16, 16]$ with entry and 200k periodic projection.

Constraint type across projection periods. As shown in the Figure 9, we compare the task-acquisition trajectories under lower-bound-only, upper-bound-only, and two-sided singular-value constraints. The average success rates over training under the lower-bound-only constraint are 45.06%, 45.01%, and 41.98% for projection periods of 100k, 200k, and 400k, respectively, all lower than those of the other two settings. Imposing only an upper bound already yields a substantial improvement in success rate; at 200k and 400k, the two-sided constraint further increases the average success rate over training from 63.70% to 81.78% and from 68.90% to 77.20%, respectively. At the 100k period, the two-sided and upper-bound-only constraints exhibit similar training performance. These results indicate that limiting excessively large singular values is an important component of effective intervention in the current setting, while the additional benefit of raising small singular values depends jointly on the projection period.

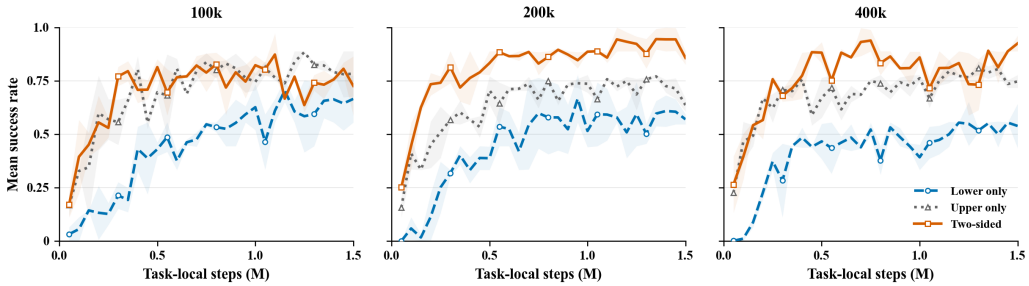


Figure 9: Constraint types across projection periods on MW-Hard. Lower-only projection uses the $1/16$ bound, upper-only projection uses the 16 bound, and two-sided projection uses both. All variants act on every linear layer of both critics. The left, middle, and right panels show projection periods of 100k, 200k, and 400k environment steps, respectively, with projection also applied at each task entry. Curves show the mean success rate across tasks 2–10 at each task-local step. All results are averaged over three seeds, and shaded regions indicate SEM.

I Thinking and Discussion: Alignment Between Task Demands and Learning Directions

PSP provides a perspective on plasticity through the alignment between task demands and the network's learning directions: the difficulty of adapting to a new task may depend on whether the required functional changes lie in directions to which the network can respond rapidly. Our SAC experiments provide support for this interpretation and offer a starting point for further studying the relationship among demand distributions, optimizer-induced directional responses, and finite update budgets.

The present work focuses primarily on SAC, partly because SAC and PPO differ in how learning demands arise and how they influence policy improvement. SAC learns an action-value function $Q(s, a)$ from replay data, and the actor is updated directly using its action gradients; consequently, slow critic adaptation can more directly impede policy improvement. PPO, by contrast, collects trajectories using the current policy and uses a state-value function $V(s)$ to support advantage estimation. When the policy continues to visit failure trajectories, the critic may accurately predict low returns while the data still lack reliable signals for improving the action choice; in such cases, the evaluation demand reflected by the current value residual may deviate from the improvement demand required to escape failure.

In our preliminary exploration with PPO, we also observed phenomena consistent with this interpretation, but systematically validating the underlying mechanism is beyond the scope of this work and is therefore not included in the main experiments. Future work could extend this perspective to PPO by constructing learning demands that are more directly tied to policy improvement and provide effective state coverage and action discrimination, and then examining their relationship with the critic's directional response and behavioral improvement. This also raises a broader question for future study: how can we distinguish between adaptation bottlenecks caused by the absence of effective learning signals and those caused by existing signals that cannot be learned efficiently within a finite update budget?