NOT ALL OLD KNOWLEDGE IS NEEDED: BUDGETED RELEASE FOR CONTINUAL MULTI-AGENT COORDINATION IN COMORBID TREATMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Critically ill patients often present with multiple interacting conditions, yet reinforcement learning (RL) approaches to ICU treatment have largely been developed for individual diseases. Extending such systems to comorbid treatment introduces a continual-learning challenge: as new disease agents and combinations are introduced, previously learned representations must retain knowledge useful for earlier cohorts while adapting to newly emerging cross-disease dependencies. Existing continual-learning methods can protect past knowledge by constraining changes to historically important representation directions, but protecting all such directions can suppress the adaptation required for new coordination. We propose **COSTAR-MARL**, a continual offline multi-agent RL framework that selectively releases historically protected representation directions under a global adaptation budget. For each direction, COSTAR-MARL considers its historical importance, the benefit of adapting it to the current cohort, and whether its information is functionally supported by another disease agent. Cross-agent support is identified by comparing agents' functional responses on matched patient observations, avoiding direct comparison between heterogeneous latent representations. The resulting budgeted allocation favors directions with high adaptation benefit and low historical cost; under our formulation, its optimal solution has a fractional-knapsack form. We evaluate COSTAR-MARL on continual comorbid treatment for sepsis, acute kidney injury, and respiratory failure across seven disease cohorts in MIMIC-IV and eICU. On both datasets, COSTAR-MARL obtains the highest final off-policy value estimate among all compared methods, the highest across-stage estimate among sequentially trained ones, and the largest backward transfer as new disease combinations are introduced.

1 INTRODUCTION

Critically ill patients often require treatment for multiple interacting conditions, yet reinforcement learning (RL) approaches to retrospective ICU treatment have largely been developed for individual diseases, including sepsis, acute kidney injury (AKI), and respiratory failure (RF) (Komorowski et al., 2018; Zhang et al., 2022; Peine et al., 2021; Wu et al., 2023). These conditions frequently co-occur (Aronsson Dannewitz et al., 2022; Rudd et al., 2020; Bellani et al., 2016), and their treatments are coupled: an intervention beneficial for one condition can alter the treatment requirements of another. Fluid administration, for example, may support perfusion in sepsis while aggravating fluid overload in AKI or pulmonary edema in RF (Meyhoff et al., 2022). Comorbid treatment therefore requires not only disease-specific decision making, but also coordination across interacting treatment objectives. Modeling each disease as an agent provides a natural decomposition: each agent can retain disease-specific observations, actions, and value functions while coordinating its decisions with other agents for the same patient.

Such a system is unlikely to be trained for every disease combination at once: disease-specific models and cohorts become available incrementally, so a system trained for a single disease may later need to treat pairwise and then more complex combinations. Each expansion introduces new coordination requirements. For example, when an AKI agent is added to a previously trained sepsis agent, the sepsis policy must retain knowledge that remains useful for sepsis treatment while adapting to

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.

renal information and treatment interactions that were absent during its earlier training. This creates a continual-learning problem specific to expanding comorbid treatment: how should previously learned disease-specific knowledge be preserved while allowing enough plasticity to learn newly introduced cross-disease coordination?

Existing approaches capture only parts of this problem. Sequential fine-tuning allows disease agents to adapt to newly introduced comorbidities, but can overwrite knowledge acquired from earlier cohorts. Standard value-decomposition methods such as VDN and QMIX (Sunehag et al., 2018; Rashid et al., 2018) learn coordination among a fixed set of agents and do not address the continual expansion considered here. Continual-learning methods based on parameter regularization (Kirkpatrick et al., 2017) or gradient constraints (Lopez-Paz & Ranzato, 2017) protect past knowledge, while subspace-based approaches (Farajtabar et al., 2020; Saha et al., 2021) restrict updates along representation directions used by previous tasks. Such protection reduces forgetting, but creates a difficulty in our setting: directions important for treating an earlier disease cohort may also need to change when a new comorbidity introduces previously unseen treatment dependencies. Protecting all such directions can therefore suppress the very adaptation required for cross-disease coordination.

Our key observation is that historical directions need not all receive the same degree of protection once the disease team expands. Some directions encode knowledge that remains unique to an agent and should stay protected. Others may become *functionally corroborated* by another disease agent, in the sense that a teammate's representation responds to the same patients in the same way, so relaxing their protection can provide capacity for learning new coordination at a lower risk of losing a response that no other agent reproduces. The central question is therefore not simply *how much* adaptation to permit, but *where* that adaptation should be spent. This requires answering three questions for each historical representation direction: how important was it to past decisions, does the current task benefit from changing it, and is the response it produces reproduced elsewhere in the current disease team?

We address this problem with **COSTAR-MARL** (**Co**llaborative **S**elective **T**ransfer **A**nd **R**elease for Multi-Agent Reinforcement Learning), a continual offline MARL framework for expanding comorbid treatment. Each disease agent maintains a compact *private spectral memory* of representation directions that were important to its historical decisions. Because independently trained disease agents need not organize their latent representations in corresponding coordinates, COSTAR-MARL does not compare their directions directly. Instead, it evaluates their *functional responses on matched patient observations* to determine whether a historical direction is functionally corroborated by another agent, that is, whether its response trace is redundant with one a teammate produces. It then allocates a global release budget across historical directions according to their historical importance, current adaptation benefit, and cross-agent support, selectively relaxing directions that can support new coordination at lower historical cost. Under our budgeted objective, this allocation reduces to a fractional-knapsack problem with an efficient optimal solution (Theorem 7); Appendix A discusses related work.

Our contributions are threefold. **1) Continual comorbid treatment formulation.** We formulate comorbid treatment as a continual offline MARL problem in which heterogeneous disease agents and disease combinations are introduced incrementally, exposing a stability–plasticity challenge caused by newly emerging cross-disease dependencies. **2) Selective spectral release with cross-agent support.** We introduce COSTAR-MARL, which combines private spectral memory, functional cross-agent support on matched patient observations, and budgeted release to selectively adapt historically important representation directions. **3) Evaluation across cohorts and datasets.** Across seven disease cohorts on MIMIC-IV and eICU, COSTAR-MARL improves across-stage and final performance over the continual-learning baselines evaluated, achieves positive backward transfer, and our ablations show that its gains arise from selectively allocating adaptation rather than simply permitting more parameter change.

2 Problem Formulation

We study continual offline reinforcement learning over expanding disease combinations. Let $\mathcal{I} = \{1, \ldots, N\}$ be a fixed set of disease agents, here $N = 3$ for sepsis, acute kidney injury (AKI), and respiratory failure (RF), and let $\mathcal{T} \subseteq \mathcal{I}$ be an active team. Agent i observes $o_i \in \mathcal{O}_i$ and acts in $\mathcal{A}_i$, and these spaces differ across agents because diseases require different clinical information and treatment decisions. Agents learn from disease-specific views of the same patient trajectories,

aligned by patient identity and time block so that every agent sees the same clinical moment while keeping its own state, action, reward, and termination; Appendix C gives the aligned data format and the resulting asynchronous terminations.

Training cohorts arrive in three stages $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ of single-disease, two-disease, and three-disease cohorts, so the active teams grow from singletons to all pairs and finally to $\mathcal{I}$. This sequence introduces training cohorts rather than requiring individual patients to progress from one disease to two and then three. Because agents are reused across teams instead of retrained per team, the same agents can cover a comorbid patient's full clinical course as new diseases emerge, which is the open-team setting considered here. Historical cohorts remain accessible only at stage boundaries, for spectral-memory reconstruction and support estimation; optimization within a stage uses the current cohorts only and does not replay earlier ones, so the protocol sits between strict data-free continual learning and replay. This raises *two challenges* we aim to address, namely a) agents must **coordinate with newly introduced teammates**, and b) the resulting parameter updates may **overwrite knowledge from earlier settings**.

3 METHOD: COSTAR-MARL

COSTAR-MARL uses singular value decomposition (SVD) to organize historical representations into orthogonal directions ranked by importance, which lets us identify which past directions to protect or adapt and also provides a common basis for cross-disease communication. A private SVD memory summarizes each agent's dominant historical directions, a shared spectral interface carries current patient information between agents, functional support compares private directions across agents on aligned observations, and a budgeted allocation decides how much gradient reaches each historical direction. Appendix D gives the complete pipeline (Figure 3) and, in Figure 4 of Appendix D, an illustration of how the budget is filled; Appendix B gives the proofs, and Appendix E.12 reports the computational cost of the memory and support computation.

3.1 DISEASE-SPECIFIC REPRESENTATIONS AND PRIVATE SVD MEMORY

Each disease agent uses an independent encoder to accommodate its observation space and extract a patient representation $h_i = E_i(o_i)$. This representation is used directly during single-disease training, but during comorbidity training it is combined with messages from teammates, so the same encoder and downstream layers are updated across multiple team compositions. To characterize the representation structure each agent has acquired from earlier stages before such updates, we maintain a private spectral memory for selected linear layers. Private refers to the coordinate system, not the knowledge: the same pattern can appear in several agents' bases, which is what Section 3.3 compares.

To preserve past knowledge we first need to know which representation directions an agent relied on in earlier stages. Storing every past activation would grow with the number of patients seen, so we summarize them with SVD, which sorts directions by how much of the past activation energy they carry and yields a memory whose size is fixed by the layer width.

For layer ℓ of agent i, let $X_{i\ell} \in \mathbb{R}^{N_{i\ell} \times d_\ell}$ collect its historical input activations, one row per stored input, with d_ℓ the layer input dimension. Decomposing $X_{i\ell} = L_{i\ell}\Sigma_{i\ell}V_{i\ell}^\top$, the columns of $V_{i\ell}$ are orthogonal directions in that input space and the singular values $\sigma_{i\ell 1} \geq \sigma_{i\ell 2} \geq \cdots$ say how strongly each was used. We keep the top $r_{i\ell}$ as the private spectral memory,

$$U_{i\ell}^{\mathrm{pr}} = V_{i\ell}[:, 1 : r_{i\ell}], \qquad \omega_{i\ell k} = \frac{\sigma_{i\ell k}^2}{\sum_{q=1}^{r_{i\ell}} \sigma_{i\ell q}^2}, \qquad p_{i\ell k} = W_{i\ell}^{\mathrm{ref}} u_{i\ell k}, \tag{1}$$

where $u_{i\ell k}$ is the k-th retained direction, $\omega_{i\ell k}$ its share of the retained energy and hence its historical importance, and $p_{i\ell k}$ the output it produced under the parameters $W_{i\ell}^{\mathrm{ref}}$ held when the memory was built. The rank $r_{i\ell}$ is set by the truncation rule in Appendix D and is capped. As in GPM (Saha et al., 2021), a subspace-based continual-learning method, we do not store $X_{i\ell}$ itself. Adding up the outer product of each past input with itself gives $C_{i\ell} = X_{i\ell}^\top X_{i\ell} = V_{i\ell}\Sigma_{i\ell}^2 V_{i\ell}^\top$, which has the same directions $V_{i\ell}$ and the squared singular values, so it is all the memory needs. Because $C_{i\ell}$ is $d_\ell \times d_\ell$, its size is set by the layer width alone and never grows with the number of patients seen.

Given a current layer gradient $G_{i\ell}$, the response $G_{i\ell}U_{i\ell}^{\mathrm{pr}}$ tells us how much this update would push each historical direction, and comparing $G_{i\ell}u_{i\ell k}$ against the stored response $p_{i\ell k}$ tells us whether it reinforces or contradicts what that direction previously represented; Section 3.4 uses both. Private

memory protects one agent's own knowledge, but a comorbid patient needs its diseases to inform each other, so we next let agents exchange information.

3.2 Cross-Disease Communication through Shared SVD Memory

Independent encoders need not assign the same meaning to corresponding private coordinates. Each agent therefore has an adapter that maps its private representation into a latent space of common dimension, $z_i = A_i(h_i) \in \mathbb{R}^{d_z}$, with agent-specific parameters and a common output dimension. A shared spectral basis in this latent space provides the same projection interface for all message senders.

To let different disease agents communicate, we need a common set of directions in the shared latent space, so that a message can be written as coordinates along those directions. These directions should reflect what varies when agents jointly treat the same patients, so we build them only from comorbidity examples, where at least two agents are active. Let the rows of Z_i be the adapter outputs z_i of agent i on those examples, with n_i rows and mean μ_i.

Stacking the Z_i directly would let an agent dominate either because its vectors are larger or because it contributes more rows. We therefore subtract each agent's own mean μ_i and divide by $\sqrt{Mn_i}$ before stacking the M agents with valid comorbidity samples,

$$\begin{aligned} \widetilde{Z} &= \big[\widetilde{Z}_1^\top \ \cdots \ \widetilde{Z}_M^\top\big]^\top, \qquad \widetilde{Z}_i = \frac{Z_i - \mathbf{1}\mu_i^\top}{\sqrt{Mn_i}}, \\ \widetilde{Z} &= L^{\mathrm{sh}}\Sigma^{\mathrm{sh}}(V^{\mathrm{sh}})^\top, \qquad U^{\mathrm{sh}} = V^{\mathrm{sh}}[:, 1 : r_{\mathrm{sh}}]. \end{aligned} \tag{2}$$

The scaling is exactly right rather than approximately so, because $\widetilde{Z}^\top\widetilde{Z} = \frac{1}{M}\sum_i \widehat{\Sigma}_i$ is the plain average of the agents' own within-agent spreads, so every agent enters with equal weight whatever its scale or sample count; Appendix D gives the derivation and Proposition 10 the optimality statement. The columns of U^{sh} are the dominant directions of that average; the rank r_{sh} is capped.

Projecting agent i's current representation onto U^{sh} gives its message, and each agent's policy input concatenates its own representation with the messages it receives,

$$\begin{aligned} m_i &= \tanh\big(\kappa\,(U^{\mathrm{sh}})^\top(z_i - \mu)\big), \qquad \mu = \textstyle\sum_i n_i\mu_i \big/ \sum_i n_i, \\ \xi_i &= \mathrm{concat}\big(h_i,\ \{\mathbf{1}[j \in \mathcal{T},\, j \neq i]\, m_j\}_{j\in\mathcal{I}},\ \mathrm{mask}(\mathcal{T})\big), \end{aligned} \tag{3}$$

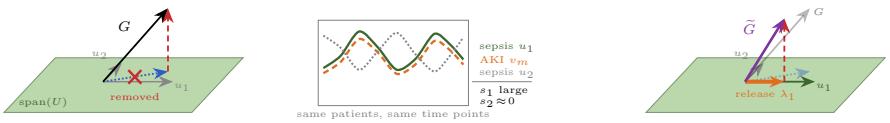
where μ is the pooled mean subtracted so that a message describes how this patient departs from the average comorbid patient, $\kappa > 0$ controls the scale, $\tanh$ acts entrywise and keeps messages bounded, $\mathbf{1}[\cdot]$ is the indicator, and $\mathrm{mask}(\mathcal{T}) \in \{0,1\}^{|\mathcal{I}|}$ records which agents are active. Every agent keeps a fixed slot in ξ_i whether or not it is in the current team, with absent agents sending zero messages, so the same policy architecture serves every disease combination. Before any comorbidity cohort is observed, U^{sh} is undefined; during $\mathcal{D}_2$, messages therefore use the first r_{sh} adapter coordinates, $m_i = \tanh(\kappa[z_i]_{1:r_{\mathrm{sh}}})$, and U^{sh} is built at the end of $\mathcal{D}_2$. The same fallback serves evaluation on comorbidity cohorts after $\mathcal{D}_1$. Unlike the support scores of Section 3.3, U^{sh} is not built from the incoming $\mathcal{D}_2$ cohorts at the boundary: support only compares directions that are already fixed, whereas the shared basis must summarize adapter latents on comorbid patients, on which the adapters have not yet been trained.

Adapting to new combinations may disturb previously learned representations, so we next identify which historical directions are functionally related across agents and can therefore be adapted at lower cost.

3.3 Identifying Cross-Agent Support through Functional Responses

Because each agent learns its own representation, the same physiological pattern may appear along very different directions in different agents. We therefore compare directions by how they respond to the same patient states, rather than by their coordinates.

Consider a comorbidity cohort g in which agents i and j are both active. Let $X_{i\ell}^g$ and $X_{j\ell}^g$ contain their inputs to layer ℓ for the same n_g patient-time observations, so matching rows correspond to the same clinical moment and ℓ refers to corresponding depths in the two agents. Projecting these inputs onto a direction gives a response trace, one value per observation. Because the traces of $u_{i\ell k}$ and $u_{j\ell q}$ cover the same patients at the same times, we can measure directly whether they rise and



(a) Hard projection all historical directions frozen, $\lambda_k = 0$, no room for new coordination

(b) Cross-agent support u_1 and v_m rise and fall together, so the AKI agent also holds that physiology

(c) Budgeted release supported u_1 partly released, unsupported u_2 still frozen, total release capped by B

Figure 1: Direction-wise budgeted release. The green plane holds the directions an agent's past decisions relied on; the gradient splits into an in-plane part (blue) and an orthogonal part (red). (a) Existing subspace methods delete the in-plane part, leaving nothing for new coordination. (b) Projecting the same patients onto two agents' directions gives response curves: sepsis u_1 and AKI v_m rise and fall together, so that physiology survives if one agent changes it, while u_2 has no counterpart. (c) We therefore release u_1 by a fraction λ_1 while u_2 stays protected, giving $\widetilde{G}$; releases follow from past importance, current need, teammate support and a per-direction cap under a budget B, and $B = 0$ recovers (a).

fall together, using the absolute Pearson correlation $\mathsf{R}^g_{ij\ell,kq} = \left|\operatorname{corr}\left(X^g_{i\ell}u_{i\ell k},\, X^g_{j\ell}u_{j\ell q}\right)\right|$. We take the absolute value because a spectral direction is defined only up to sign.

Direction k of agent i is considered *supported*, by which we mean functionally corroborated, when its correlation with a direction from another agent exceeds a threshold $\tau_f \in (0, 1)$. The support score $s_{i\ell k} \in [0, 1]$ summarizes the strongest available match and is zero when no teammate provides a valid match; Appendix D gives the full definition, Appendix B.6 the null model behind the threshold, and Appendix E.11 the support prevalence, release and permutation diagnostics. A high support score means that a teammate produces a response trace that co-varies with this one on the same patients, so the response is redundant within the team. Support is a statement about response redundancy on matched observations, not evidence that the teammate stores the same knowledge or that releasing the direction loses no information; it lowers the estimated cost of adapting the direction rather than removing it, and Appendix B.7 states this limitation.

3.4 Budget-Constrained Adaptation of Historical Directions

We balance coordination against forgetting by controlling how much each historical direction can change: important and uncorroborated directions stay protected, while directions the current task needs and whose responses other agents reproduce can adapt more. Figure 1 illustrates the rule; support alone does not imply release, since the budgeted allocation determines the final fractions.

For clarity we now consider one agent and one of its filtered layers, meaning a linear layer for which a spectral memory is maintained, and omit the indices $i\ell$. We write G for the current gradient, $U = [u_1, \ldots, u_r]$ for the private spectral memory, ω_k for the historical importance of direction u_k, p_k for its stored response, and s_k for its cross-agent support. The same rule is applied independently to every filtered layer. We assign each direction a release fraction $\lambda_k \in [0, 1]$ and collect them in $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_r)$:

$$G_{\text{hard}} = G - GUU^\top, \quad \widetilde{G} = G - GU(I - \Lambda)U^\top, \quad \widetilde{G}U = GU\Lambda, \quad e_k = \|Gu_k\|_2^2,$$
$$\rho_k = \frac{\langle Gu_k, p_k\rangle}{\|Gu_k\|_2\|p_k\|_2}, \quad b_k = [\rho_k]_+ e_k, \quad c_k = \frac{\omega_k e_k}{1+s_k}, \quad \bar{\lambda}_k = \lambda_{\max}\zeta_{\text{sh}} \text{ or } \lambda_{\max}\zeta_{\text{pr}} \text{ as } s_k \gtrless \epsilon. \tag{4}$$

Here $G = \nabla_W \mathcal{L}_\mathcal{T}$, and setting every $\lambda_k = 0$ recovers G_{hard}, so hard projection is a special case of our update. Each quantity then serves a simple role. The term e_k measures **how strongly the current task wants to change direction** k. Since the update is $-\eta\widetilde{G}$, $\rho_k > 0$ means the current objective **pushes against the stored response** p_k, and hard projection blocks exactly this correction, so $b_k = [\rho_k]_+ e_k$ is the **benefit of releasing** the direction while directions with $\rho_k \le 0$ earn none: an update that merely reinforces a stored response is not what the new team composition needs the protected subspace for, and withholding it keeps the budget for the conflicts that hard projection blocks (Appendix D discusses this one-sided rule and its side effect).

The cost c_k works the other way: historically important directions cost more to change, while directions whose responses other agents reproduce cost less, since a redundant response is less likely to be the team's only record of that pattern. The cap $\bar{\lambda}_k$ adds a safeguard, giving supported directions more room than unsupported ones ($0 < \zeta_{\text{pr}} < \zeta_{\text{sh}} \le 1$), with ϵ only a numerical tolerance for detecting support. Together these terms turn the stability–plasticity trade-off into a **direction-wise allocation problem**, with λ_k the fraction of gradient allowed through direction k.

Budgeted allocation. Allowing every useful direction to change freely could still cause excessive forgetting, so we impose a common budget B and allocate it across eligible directions, favoring more benefit per unit cost:

$$\max_{\lambda} \sum_{k\in\mathcal{V}} b_k\lambda_k \;\text{ s.t. }\; \sum_{k\in\mathcal{V}} c_k\lambda_k \le B,\;\; 0 \le \lambda_k \le \bar{\lambda}_k,\;\; B = \beta\sum_{k\in\mathcal{V}} c_k\bar{\lambda}_k. \tag{5}$$

The eligible set $\mathcal{V} = \{k : b_k > 0,\, c_k > \epsilon_0,\, e_k > \epsilon_0\}$ keeps directions with a meaningful adaptation benefit, where $\epsilon_0 > 0$ is a small numerical floor distinct from the support tolerance ϵ of Eq. (4), and all others keep $\lambda_k = 0$. The budget is written as a fraction $\beta \in [0, 1]$ of what it would cost to release every eligible direction to its cap, so β controls the overall amount of release on a scale that does not depend on the layer or the stage: $\beta = 0$ recovers full protection, while larger β allows more adaptation.

Because the objective and the constraint are both linear, the best use of the budget is to spend it first on directions with the largest benefit-to-cost ratio b_k/c_k. We therefore sort eligible directions by this ratio and fill their caps $\bar{\lambda}_k$ in that order until the budget runs out, which takes $O(r \log r)$ time. Appendix B proves that this greedy allocation is exactly **optimal** for Eq. (5), not an approximation (Theorem 7), and that the budget bounds the resulting drift in historical representations (Theorem 2).

Training. Agents are trained from offline trajectories with an actor–critic objective, zero intermediate rewards, and terminal rewards $\pm R$ for survival and death; Appendix D gives the target, the advantage, and the team objective $\mathcal{L}_{\mathcal{T}}$ of Eq. (28). All agents in a team are updated together through the communication paths, so what one agent learns about a patient reaches the others, and every gradient passes through the filter of Eq. (4) before the optimizer sees it. At each stage boundary we rebuild the private memories and the shared basis from the current parameters and compute the support scores on the incoming cohorts, so the next stage is protected according to what the agents know now. These three components stay fixed for the whole stage; only the release fractions λ_k are recomputed at every update (schedule and cost in Appendix E.12).

4 Experiments

We evaluate COSTAR-MARL along two axes. **1) Policy quality** (Section 4.2): whether it learns effective policies as disease combinations expand, at every stage and at the end, against continual-learning, offline RL, and clinical baselines. **2) Whether earlier capabilities survive** (Section 4.3): we follow agents through the sequence and ask about ***retention*** (are single-disease patients still treated well after the system moves on to comorbid cohorts?), ***backward transfer*** (does learning a larger combination leave earlier ones better or worse?) and ***adaptation*** (how much does a new combination gain in the stage that introduces it?). Section 4.4 then ablates the spectral memory, cross-agent support and budgeted release.

4.1 Experimental Setup

Cohorts. We use retrospective ICU trajectories from MIMIC-IV (Johnson et al., 2023) and eICU (Pollard et al., 2018) in seven cohorts, namely sepsis-only, acute kidney injury (AKI)-only and respiratory failure (RF)-only in $\mathcal{D}_1$, the three pairwise combinations in $\mathcal{D}_2$, and sepsis–AKI–RF in $\mathcal{D}_3$. The same three agents are reused throughout, with observation dimensions 47, 101 and 71 and 25, 25 and 48 discrete actions. For eICU, the final analysis includes 70,166 ICU stays, with 42,183 used for training and 27,983 for testing. Cohort sizes are unequal (Tables 3 and 4), but the averages defined below weight cohort–agent pairs equally. Appendix E.1 gives the action definitions and the full training protocol.

Baselines. Each group of baselines tests one thing: **1)** *Whether selective release beats the two extremes of adapting freely and protecting everything:* Sequential fine-tuning, Cumulative Replay, elastic weight consolidation (EWC) (Kirkpatrick et al., 2017) and learning without forgetting (LwF) (Li & Hoiem, 2018). **2)** *Whether releasing directions one by one beats protecting them all equally:* gradient episodic memory (GEM) (Lopez-Paz & Ranzato, 2017), gradient projection memory (GPM) (Saha et al., 2021), scaled gradient projection (SGP) (Saha & Roy, 2023) and Adam-NSCL (Wang et al., 2021). **3)** *Whether the gain comes from continual coordination rather than from offline learning alone:* COMAD (Xiao et al., 2026), counterfactual conservative Q-learning (CFCQL) (Shao et al., 2023), B3C (Kim & Sycara, 2026) and conservative Q-learning (CQL) (Kumar et al., 2020), all independently implemented for the present observation and action spaces. **4)** *Whether our agents beat systems built for a single disease:* AI Clinician (Komorowski et al., 2018)

and WD3QNE (Wu et al., 2023) for sepsis, an adaptation of HRRT (Xu et al., 2026), a hierarchical RL system for renal replacement therapy, for acute kidney injury, and Ventilation FQI (Prasad et al., 2017) for respiratory failure. **5)** *Whether separate agents and stage-wise training are needed at all:* Factored Single Policy, which merges the agents into one network with one head per disease, Balanced Joint, which sees all seven cohorts at once, and Scratch-D2 and Scratch-D3, which train fresh models on $\mathcal{D}_2$ and $\mathcal{D}_3$ alone. Appendix E.1 describes every baseline.

Evaluations. Every entry in Table 1 is a mean over three training runs whose seeds are listed in Table 6 of Appendix E.2. We report agent-level truncated weighted importance sampling (WISt), defined in Eq. (30) of Appendix E.3, with a fixed behaviour reference fitted to the logged clinician actions. WISt estimates the discounted terminal return ($\pm R$ scale) a policy would obtain on the same patients, so higher is better; whether it beats recorded care is a separate comparison against the clinician reference (Above ref. in Table 1; Table 14). Because the same agents must serve every disease combination, we evaluate them on all seven cohorts at every stage. Each agent is scored only on cohorts containing its own disease, so the sepsis–AKI cohort yields one value for the sepsis agent and one for the AKI agent. Across the seven cohorts this gives $J = 12$ values. Let $t \in \{1, 2, 3\}$ index the continual training stage, and let $Y_{t,j}$ denote the WISt of the j-th cohort–agent evaluation measured after stage t. We summarize them in two ways. Final Avg. $= \frac{1}{J}\sum_j Y_{3,j}$ measures whether the **final system is a strong multi-disease policy after the full disease expansion**, so a method that reaches the three-disease cohort by sacrificing earlier capabilities scores poorly. Across-stage Avg. $= \frac{1}{3J}\sum_t \sum_j Y_{t,j}$ measures whether the **system remains effective throughout the sequence**, including on cohorts it has not yet trained on. Table 11 in Appendix E.9 reports the average effective sample size and 95% bootstrap interval of every method's MIMIC-IV final score; ours is [55.809, 63.424] at an average ESS of 762, and several intervals overlap, so we emphasize the larger contrasts. Appendix E.9 also shows that the ranking of COSTAR-MARL is unchanged at every weight-truncation level from 10 to 1000 on both datasets, and that its final average lies above the 95% interval of the clinician reference on both. Run-to-run standard deviations for both datasets are in Appendix E.10.

4.2 Main Performance Comparison

On MIMIC-IV, COSTAR-MARL is highest on both summaries in Table 1, at 59.788 final average and 55.683 across-stage; the strongest baselines are Balanced Joint (57.098 final), EWC (56.590) and SGP (55.113 final, 55.279 across-stage). Leading on both means the agents are strong at the end and usable at every point along the way. Among the methods that protect a stored representation subspace, the ordering follows how strictly that subspace is blocked: Adam-NSCL, which confines updates to directions the old data barely used, reaches 47.839; GPM, which removes the in-subspace component outright, reaches 52.403; and SGP, which scales that component rather than removing it, reaches 55.113. Removing protection altogether is not better either, since sequential fine-tuning reaches 51.448. The best operating point lies between the two extremes, which is what the release budget controls (Section 4.4).

Three further baselines isolate other sources of the advantage: Cumulative Replay (every past cohort available), Balanced Joint (all seven cohorts at once, no continual problem) and Scratch-D3 (the three-disease cohort alone). All three end below us on MIMIC-IV, at 49.247, 57.098 and 41.237 against 59.788, so neither access to past data nor joint access to every cohort is sufficient, although the interval of Balanced Joint overlaps ours (Table 11). The same pattern holds on eICU, where COSTAR-MARL again leads on both summaries, at 60.532 final and 54.446 across-stage against 55.463 (Scratch-D3) and 53.814 (SGP) for the strongest baselines, and attains the largest backward transfer, $+8.413$. COSTAR-MARL is not first only on Above ref., a margin-blind count of scores clearing the clinician reference (SGP $12/12$ on MIMIC-IV, Sequential FT 11.667 on eICU); its final average nonetheless lies above the clinician interval on both datasets (Table 14).

Splitting treatment across agents does not make each agent worse at its own disease: every COSTAR-MARL agent exceeds the clinical method built specifically for its disease, by 50.5 points on sepsis and 58.7 on respiratory failure, and by 0.8 on AKI, where the dedicated baseline HRRT is strongest and the margin is within run-to-run variation (Table 9, Appendix E.5; training seeds in Table 7). Coordinating with teammates therefore does not cost the agents their disease-specific performance.

Table 1: Overall performance on MIMIC-IV and eICU across seven disease cohorts. Mean over three runs per method and dataset. Across-stage Avg. averages all 36 cohort–agent scores over three stages; Final Avg. the final 12; BWT is backward transfer on the nine previously learned pairs; Above ref. is the mean number of final scores (of 12) exceeding the clinician reference. Outcomes: 90-day survival (MIMIC-IV), hospital survival (eICU). Higher is better; bold marks the best mean per metric within each dataset. Dashes indicate metrics that are not applicable to the training protocol.

	MIMIC-IV				eICU			
Method	Across-stage Avg.	Final Avg.	BWT	Above ref. (/12)	Across-stage Avg.	Final Avg.	BWT	Above ref. (/12)
COSTAR-MARL	**55.683**	**59.788**	**+5.271**	11.667	**54.446**	**60.532**	**+8.413**	10.667
Sequential FT	51.590	51.448	−2.017	9.000	51.829	55.404	+2.970	**11.667**
Factored Single Policy	47.707	51.472	+1.879	9.333	47.369	44.649	−7.517	6.000
Cumulative Replay	49.482	49.247	+0.321	8.000	51.218	52.342	+2.832	9.667
EWC	55.079	56.590	+1.669	11.667	52.891	53.693	−0.215	10.000
LwF	50.825	51.768	+0.559	9.667	47.533	45.646	−0.720	8.333
GEM	49.219	49.059	−1.260	7.667	46.092	45.352	−1.876	8.000
GPM	53.399	52.403	−1.941	10.667	52.236	53.255	+0.843	11.333
SGP	55.279	55.113	−1.048	**12.000**	53.814	55.325	+1.831	10.000
Adam-NSCL	48.209	47.839	−1.131	7.667	51.432	49.518	+0.200	10.333
COMAD	40.190	39.499	−2.813	3.667	42.898	42.195	−1.266	5.000
CFCQL	41.882	24.413	−26.742	4.333	32.421	30.795	−0.593	5.667
B3C	40.125	39.137	−3.054	3.667	40.808	37.997	−2.434	2.667
CQL	46.798	42.576	−5.769	8.000	38.179	36.869	+2.491	6.333
Balanced Joint	–	57.098	–	11.000	–	51.006	–	9.667
Scratch-D2	–	49.933	–	8.333	–	51.810	–	9.667
Scratch-D3	–	41.237	–	6.667	–	55.463	–	11.000

4.3 Continual Learning and Adaptation to New Disease Combinations

The comparisons so far summarize each method by where it ends up, not when knowledge is lost or gained; a system whose average holds up while single-disease patients quietly get worse is not safe to deploy. We therefore evaluate every policy after each stage, comparing COSTAR-MARL with the nine baselines that share its three-stage sequence, actor–critic architecture and stage protocol (Sequential FT, Cumulative Replay, EWC, LwF, GEM, GPM, SGP, Adam-NSCL and COMAD).

We report backward transfer (BWT), the average change of each of the nine earlier cohort–agent pairs between the stage that first trained it and the end of the sequence, with positive values meaning that earlier combinations end better than when they were first learned (Eq. (31), Appendix E.6). COSTAR-MARL has the largest mean BWT on both datasets in Table 1, at $+5.271$ on MIMIC-IV, where the next largest is $+1.879$; the run-to-run standard deviation of this mean is large (4.864 over three runs, Table 15), so we read the sign of the average rather than its magnitude. On MIMIC-IV every subspace-protection method ends negative, GPM at -1.941 and SGP at -1.048: blocking updates along the stored directions keeps those directions fixed, but the earlier cohorts are still scored worse at the end than when they were first learned. Several other methods have small positive BWT, but only COSTAR-MARL combines a BWT that is positive on average with the highest final average, so on average learning the comorbid cohorts improves rather than degrades performance on earlier ones. This substantially improves the average stability–plasticity trade-off, although forgetting remains heterogeneous across cohorts and runs: not every earlier cohort–agent pair improves (Figure 2(a); per-agent scores in Table 8, Appendix E.4), and the run-to-run variation means the gain holds on average rather than uniformly.

Resolving the result to the seven disease combinations shows that the gains are unevenly distributed (Figure 2 and Table 8, a single run, seed 42, so values differ slightly from Table 1). The sepsis-only score rises from 49.486 after $\mathcal{D}_1$ to 87.194 after $\mathcal{D}_3$, and sepsis–RF improves from 43.451 after $\mathcal{D}_2$ to 50.493. Two cohorts move the other way: AKI-only declines from 69.117 to 64.275, and RF-only reaches 50.628 after $\mathcal{D}_2$ before falling to 45.406, just below its initial 46.476. AKI–RF also declines, from 61.604 to 58.974. The positive backward transfer therefore reflects large gains on some cohorts outweighing smaller losses on others, not uniform improvement. Figure 5 (Appendix E.7) tracks all ten methods on the three cohort groups, one run each: most competitors peak when a group is first trained and then decline, whereas COSTAR-MARL's single-disease average rises at every stage and it ends highest on all three groups.

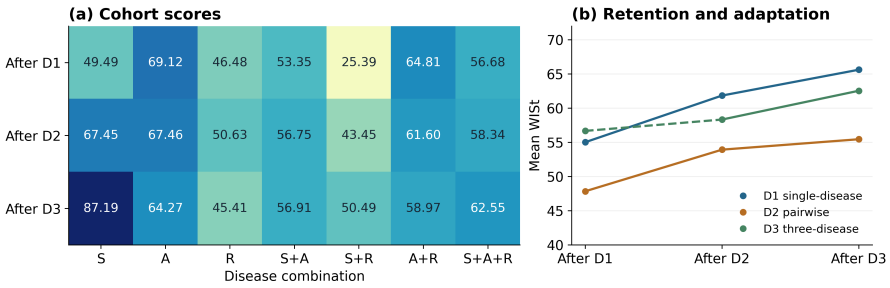


Figure 2: **COSTAR-MARL across the training sequence** (MIMIC-IV, one of the three runs of Table 1, seed 42). (a) WISt on each of the seven cohorts after each stage, averaged over the agents evaluated there; S, A, R denote sepsis, AKI, RF. (b) Group means for the single-, two- and three-disease cohorts, dashed before a group is trained.

4.4 Component and Architecture Ablations

Table 2 ablates the release budget, functional support, private memory and budget allocation, reporting the final average over all 12 cohort–agent pairs and the final score on the hardest combination, the sepsis–AKI–RF cohort.

The release budget matters at both extremes: with $\beta = 0$, which blocks every stored direction as in hard projection, the final average is 52.887, 6.9 points below the full model; with $\beta = 1$, which releases every eligible direction to its cap, the three-disease score falls to 44.432, below even the variant without private memory (Appendix D discusses why). Cross-agent support determines where the budget is spent: setting $s_k = 0$ costs 3.2 points overall and 4.0 on the three-disease cohort. Allocating by benefit per unit cost also matters: spending the same budget uniformly keeps the final average at 58.837 but lowers the three-disease score from 59.034 to 55.851. Removing the private memory, which also removes support estimation and gradient filtering so that $\widetilde{G} = G$ (Appendix D), lowers the final average to 54.880. The communication mechanism has a smaller effect: learned alternatives change the final average by at most 2.7 points, and removing messages costs 1.8 points overall and 3.6 on the three-disease cohort (Table 10, Appendix E.8). Finally, replacing the three agents with a single shared network of matched size reaches 51.472, 8.3 points below ours, so the gain is not explained by network size.

Table 2: Component, architecture, and hyperparameter comparisons on MIMIC-IV (Appendix D). Results are means over three runs per configuration. Final averages the 12 cohort–agent scores after the final stage; 3-dis. averages the three agents on the sepsis–AKI–RF cohort. The full model uses $\beta = 0.15$. Bold indicates the highest displayed mean.

Configuration	Final	3-dis.
COSTAR-MARL	**59.788**	**59.034**
Release budget		
$\beta = 0$ (hard)	52.887	54.047
$\beta = 1$ (caps kept)	51.403	44.432
Functional support		
No support ($s_k = 0$)	56.571	55.053
Memory, alloc., arch.		
No private memory	54.880	56.314
Uniform allocation	58.837	55.851
Factored single policy	51.472	51.461

5 Conclusion

We study comorbidity treatment planning as an open-team problem, where the set of diseases a deployed system handles grows over time. COSTAR-MARL addresses it with budgeted release of historical representation directions, informed by whether teammates functionally corroborate their responses. Across seven ICU cohorts on two independent datasets it attains the highest final and across-stage scores among the continual-learning baselines and the largest mean backward transfer, so earlier disease combinations end better on average than when they were first learned, with substantial run-to-run variation in that gain. Ablations indicate that where adaptation is spent matters as much as how much is permitted, and that response redundancy across teammates supplies useful evidence for that decision, even though it does not certify that the redundant response encodes the same knowledge. A new comorbidity can therefore be added as one more agent rather than a new model, although we verify this on three diseases only, and agents whose action spaces share a treatment variable (fluids for sepsis and AKI) are evaluated independently, so reconciling conflicting recommendations at the bedside remains open, as does extending the release rule to larger teams and testing it prospectively.

AI USE STATEMENT

Generative AI tools provided limited assistance with manuscript writing and editing, development of portions of the code, and figure preparation. The authors take full responsibility for the final manuscript, code, figures, analyses, and scientific claims, including all AI-assisted content.

ETHICS STATEMENT

This study uses retrospective ICU records from MIMIC-IV and the eICU Collaborative Research Database, which provide de-identified data under credentialed-access and data-use requirements. Our evaluation is offline: the reported WISt scores do not establish causal treatment effects, prospective clinical benefit, or safety. Historical treatment decisions, incomplete observations, and unequal cohort representation may introduce confounding and bias; aggregate performance does not establish equitable performance across patient groups. Disease agents may also recommend interventions involving the same treatment variables, so safe reconciliation of their actions requires further study. Any clinical use would require prospective validation, subgroup assessment, and clinician oversight. Reproduction must respect the source databases' access conditions and protect patient confidentiality.

REPRODUCIBILITY STATEMENT

Our code, including the data-processing pipeline, all baseline implementations, and the configuration files behind every table, is provided in the supplementary material. Section 3 and Appendix D describe the spectral memories, communication mechanism, support scores, budgeted allocation, and training objective. The theoretical analysis is provided in Appendix B. Appendix E.1 documents cohort sizes, action spaces, training budgets, baseline implementation choices and, in Table 5, every hyperparameter of COSTAR-MARL, while Appendix E.2 records the training seeds, which run underlies each table and figure, sources of randomness, and fixed data and evaluation settings. Appendices E.3 and E.6 define the off-policy evaluation and retention metrics, and Appendix E.9 describes effective sample sizes and bootstrap uncertainty estimates for the MIMIC-IV main comparison. Appendix E.10 gives run-to-run standard deviations, and Appendix E.11 reports functional-support and release diagnostics, and Appendix E.12 reports computational cost.

REFERENCES

Anna Aronsson Dannewitz, Bodil Svennblad, Karl Michaëlsson, Miklos Lipcsey, and Rolf Gedeborg. Optimized diagnosis-based comorbidity measures for all-cause mortality prediction in a national population-based icu population. *Critical Care*, 26(1):306, 2022. doi: 10.1186/s13054-022-04172-0.

Giacomo Bellani, John G. Laffey, Tài Pham, Eddy Fan, Laurent Brochard, Andres Esteban, Luciano Gattinoni, Frank van Haren, Anders Larsson, Daniel F. McAuley, Marco Ranieri, Gordon Rubenfeld, B. Taylor Thompson, Hermann Wrigge, Arthur S. Slutsky, Antonio Pesenti, LUNG SAFE Investigators, and ESICM Trials Group. Epidemiology, patterns of care, and mortality for patients with acute respiratory distress syndrome in intensive care units in 50 countries. *JAMA*, 315(8): 788–800, 2016. doi: 10.1001/jama.2016.0291.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019.

Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. TarMAC: Targeted multi-agent communication. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1538–1546. PMLR, 2019.

Danruo Deng, Guangyong Chen, Jianye Hao, Qiong Wang, and Pheng-Ann Heng. Flattening sharpness for dynamic gradient projection memory benefits continual learning. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2110.04593.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3762–3773. PMLR, 2020.

Ronald A. Fisher. Frequency distribution of the values of the correlation coefficient in samples from an indefinitely large population. *Biometrika*, 10(4):507–521, 1915.

Jakob N. Foerster, Yannis M. Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 29, pp. 2137–2145. Curran Associates, Inc., 2016.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019. doi: 10.1038/s41591-018-0310-5.

Jiechuan Jiang and Zongqing Lu. Learning attentional communication for multi-agent cooperation. In *Advances in Neural Information Processing Systems*, volume 31, pp. 7254–7264. Curran Associates, Inc., 2018.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, 2023. doi: 10.1038/s41597-022-01899-x.

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. doi: 10.1613/jair.1.13673.

Dongjun Kim, Seohyeon Cha, Huancheng Chen, Chaining Wang, and Haris Vikalo. Quantized gradient projection for memory-efficient continual learning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/24e8b46430df965674221665816a4964-Abstract-Conference.html.

Woojun Kim and Katia P. Sycara. B3C: A minimalist approach to offline multi-agent reinforcement learning. In *Proceedings of the 25th International Conference on Autonomous Agents and Multiagent Systems*, pp. 1755–1763. International Foundation for Autonomous Agents and Multiagent Systems, 2026. doi: 10.65109/UOYL1486.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Matthieu Komorowski, Leo A. Celi, Omar Badawi, Anthony C. Gordon, and Aldo A. Faisal. The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care. *Nature Medicine*, 24(11):1716–1720, 2018. doi: 10.1038/s41591-018-0213-5.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191. Curran Associates, Inc., 2020.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081.

Yan-Shuo Liang and Wu-Jun Li. Adaptive plasticity improvement for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7816–7825, 2023. doi: 10.1109/CVPR52729.2023.00755.

Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. TRGP: Trust region gradient projection for continual learning. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2202.02931.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, pp. 6467–6476. Curran Associates, Inc., 2017.

Tine S. Meyhoff, Peter B. Hjortrup, Jørn Wetterslev, et al. Restriction of intravenous fluid in ICU patients with septic shock. *New England Journal of Medicine*, 386(26):2459–2470, 2022. doi: 10.1056/NEJMoa2202707.

Arne Peine, Ahmed Hallawa, Johannes Bickenbach, Guido Dartmann, Lejla Begic Fazlic, Anke Schmeink, Gerd Ascheid, Christoph Thiemermann, Andreas Schuppert, Ryan Kindle, Leo Celi, Gernot Marx, and Lukas Martin. Development and validation of a reinforcement learning algorithm to dynamically optimize mechanical ventilation in critical care. *npj Digital Medicine*, 4(1): 32, 2021. doi: 10.1038/s41746-021-00388-6.

Tom J. Pollard, Alistair E. W. Johnson, Jesse D. Raffa, Leo A. Celi, Roger G. Mark, and Omar Badawi. The eICU collaborative research database, a freely available multi-center database for critical care research. *Scientific Data*, 5(1):180178, 2018. doi: 10.1038/sdata.2018.178.

Niranjani Prasad, Li-Fang Cheng, Corey Chivers, Michael Draugelis, and Barbara E. Engelhardt. A reinforcement learning approach to weaning of mechanical ventilation in intensive care units. In *Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence*, 2017.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4295–4304. PMLR, 2018.

Kristina E Rudd, Sarah Charlotte Johnson, Kareha M Agesa, Katya Anne Shackelford, Derrick Tsoi, Daniel Rhodes Kievlan, Danny V Colombara, Kevin S Ikuta, Niranjan Kissoon, Simon Finfer, et al. Global, regional, and national sepsis incidence and mortality, 1990–2017: analysis for the global burden of disease study. *The Lancet*, 395(10219):200–211, 2020. doi: 10.1016/S0140-6736(19)32989-7.

Gobinda Saha and Kaushik Roy. Continual learning with scaled gradient projection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9677–9685, 2023. doi: 10.1609/aaai.v37i8.26157.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2103.09762.

Jianzhun Shao, Yun Qu, Chen Chen, Hongchang Zhang, and Xiangyang Ji. Counterfactual conservative Q learning for offline multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77290–77312. Curran Associates, Inc., 2023. doi: 10.52202/075280-3380.

Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. In *Advances in Neural Information Processing Systems*, volume 29, pp. 2244–2252. Curran Associates, Inc., 2016.

Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2085–2087. International Foundation for Autonomous Agents and Multiagent Systems, 2018. doi: 10.5555/3237383.3238080.

Tristan Tomilin, Luka van den Boogaard, Samuel Garcin, Constantin Ruhdorfer, Bram Grooten, Fabrice Kusters, Yali Du, Andreas Bulling, Mykola Pechenizkiy, and Meng Fang. MEAL: A benchmark for continual multi-agent reinforcement learning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. arXiv:2506.14990.

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3):261–272, 2020.

Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 184–193, 2021. doi: 10.1109/CVPR46437.2021.00025.

Peter H. Westfall and S. Stanley Young. *Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment*. Wiley, 1993.

Anderson M Winkler, Matthew A Webster, Diego Vidaurre, Thomas E Nichols, and Stephen M Smith. Multi-level block permutation. *NeuroImage*, 123:253–268, 2015.

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

XiaoDan Wu, RuiChang Li, Zhen He, TianZhi Yu, and ChangQing Cheng. A value-based deep reinforcement learning model with human expertise in optimal treatment of sepsis. *npj Digital Medicine*, 6(1):15, 2023. doi: 10.1038/s41746-023-00755-5.

Yuchen Xiao, Lei Yuan, Ruiqi Xue, Tieyue Yin, and Yang Yu. Offline multi-agent continual cooperation via skill partition and reuse. *arXiv preprint arXiv:2606.25389*, 2026.

Qianyi Xu, Feng Wu, Zi Yi Christopher Thong, Mark Sen Liang Goh, Pengpeng Chen, Jie Yang, Chen Huang, Zhongheng Zhang, Yucai Hong, Kay Choong See, and Mengling Feng. HRRT: hierarchical reinforcement learning for renal replacement therapy decision support. *npj Digital Medicine*, 2026. doi: 10.1038/s41746-026-02900-2.

Yiqin Yang, Xiaoteng Ma, Chenghao Li, Zewu Zheng, Qiyuan Zhang, Gao Huang, Jun Yang, and Qianchuan Zhao. Believe what you see: Implicit constraint approach for offline multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 10299–10312. Curran Associates, Inc., 2021.

Chang Yao, Youfang Lin, Shoucheng Song, Hao Wu, Yuqing Ma, Sheng Han, and Kai Lv. From general relation patterns to task-specific decision-making in continual multi-agent coordination. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 6821–6829, 2025. doi: 10.24963/ijcai.2025/759.

Lei Yuan, Lihe Li, Ziqian Zhang, Fuxiang Zhang, Cong Guan, and Yang Yu. Multi-agent continual coordination via progressive task contextualization. *IEEE Transactions on Neural Networks and Learning Systems*, 2024. arXiv:2305.13937.

Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continual learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1(8):364–372, 2019. doi: 10.1038/s42256-019-0080-x.

Kristine Zhang, Henry Wang, Jianzhun Du, Brian Chu, Aldo Robles Arévalo, Ryan Kindle, Leo Anthony Celi, and Finale Doshi-Velez. An interpretable RL framework for pre-deployment modeling in ICU hypotension management. *npj Digital Medicine*, 5(1):173, 2022. doi: 10.1038/s41746-022-00708-4.

A Related Work

Clinical Reinforcement Learning for Comorbid Treatment. RL has been applied to sepsis resuscitation (Komorowski et al., 2018; Wu et al., 2023), mechanical ventilation (Prasad et al., 2017; Peine et al., 2021), hemodynamic management (Zhang et al., 2022), and renal replacement therapy (Xu et al., 2026), almost always under a single-disease reward, and safety concerns for offline clinical RL are well documented (Gottesman et al., 2019). We instead assign one agent per disease, so that distinct clinical views and treatment preferences remain explicit and can be coordinated, and take the learning machinery from offline MARL (Yang et al., 2021; Shao et al., 2023; Kim &

Sycara, 2026). Standard MARL communication methods learn message representations for coordination within a fixed team (Sukhbaatar et al., 2016; Foerster et al., 2016; Jiang & Lu, 2018); our shared spectral interface serves the same purpose but is not learned by gradient descent: it is recomputed only at stage boundaries, which keeps it fixed within a stage while the agents adapt. All of these methods learn treatment policies for a predefined disease setting. We instead study how disease-specific agents continually acquire new coordination behavior as the set of diseases they must handle together expands.

Continual Reinforcement Learning and Continual Multi-Agent Coordination. Continual RL seeks stability, plasticity, and transfer at once (Khetarpal et al., 2022; Wołczyk et al., 2021), through regularization (Kirkpatrick et al., 2017), replay and gradient constraints (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2019), or architectural growth. Its multi-agent counterpart is comparatively recent. MACPro obtains a factorized policy with progressively expanded task heads selected by learned task contextualization (Yuan et al., 2024); RPG argues that the transferable object in continual coordination is the relation pattern between entities and maps task-agnostic patterns into task-specific decision makers via a hypernetwork (Yao et al., 2025); MEAL provides the first dedicated CMARL benchmark and shows that naively combining popular CL and MARL methods fails on settings that require sustained coordination (Tomilin et al., 2026); and COMAD addresses offline multi-agent continual cooperation by partitioning coordination skills across expandable heads and reusing them via a density-based reusability estimator (Xiao et al., 2026). These methods preserve or reuse knowledge at the level of tasks, policy heads, relation patterns, replayed data, or skills. COSTAR-MARL asks a finer question: which individual historical representation directions inside heterogeneous agents should be protected, and which should be adapted, when new coordination relationships appear.

Subspace-Based Continual Learning. The methods closest to ours protect old knowledge in representation or gradient space. OGD projects new gradients away from directions that would alter previous outputs (Farajtabar et al., 2020); OWM constrains updates to the orthogonal complement of the input space of prior tasks (Zeng et al., 2019); GPM builds the protected subspace explicitly by applying SVD to historical layer inputs and projecting new gradients onto its orthogonal complement (Saha et al., 2021); Adam-NSCL steps within the null space of the feature covariance (Wang et al., 2021); and later work relaxes or reshapes that subspace, by scaling updates along stored directions according to their importance (Saha & Roy, 2023), by making the stored memory itself dynamic (Deng et al., 2021), by adjusting the retained subspace when plasticity is exhausted (Liang & Li, 2023), by reusing a correlated old subspace through a trust-region scaling matrix (Lin et al., 2022), or by quantizing it for memory efficiency (Kim et al., 2026).

COSTAR-MARL differs at three levels. First, release is continuous and *direction-wise*, decided per minibatch from the current gradient response, rather than a per-task decision to reuse or exclude a stored subspace. Second, the release decision uses *cross-agent functional support*, evidence from other disease agents' spectral responses on the same patients, which has no analogue in single-agent continual learning. Third, the total amount of release is set by an explicit *budgeted allocation problem* with a drift guarantee, so relaxing protection is a controlled quantity rather than a tuned coefficient; hard projection is recovered exactly at zero budget.

The first difference also explains which of these methods can serve as baselines here. TRGP, API and FS-DGPM select or reshape a stored subspace once per task, using a task identity that is available at training and, for TRGP, at inference. Our setting has no such identity: the active team $\mathcal{T}$ changes within a stage, a cohort seen at one stage reappears at later ones, and the same agent participates in several teams, so there is no discrete task index to condition a trust region or a per-task subspace on. We therefore compare against the subspace methods that operate without task identity, GPM, SGP and Adam-NSCL, together with GEM, EWC and LwF; adapting trust-region selection to open teams is an interesting direction we do not pursue.

B Theoretical Analysis

This section analyses the allocator defined in Section 3.4. We inherit e_k, ρ_k, b_k, c_k, $\bar{\lambda}_k$, ω_k, s_k, $\mathcal{V}$, B and β from Eqs. (4)–(5) and drop agent and layer indices. Two results are central: release enters the first-order learning signal linearly and the retained historical drift quadratically (Corollary 6), and the budget of Eq. (5) controls that drift (Theorem 2).

B.1 Objects Used in the Analysis

Fix one agent, one filtered linear layer, and one update. Let $W, G \in \mathbb{R}^{m\times d}$ with $G = \nabla_W \mathcal{L}_\mathcal{T}$. The memory matrix $X \in \mathbb{R}^{n\times d}$ holds the layer inputs frozen at memory construction, with uncentered second moment

$$C = X^\top X = \sum_{q=1}^{d} \sigma_q^2 v_q v_q^\top, \qquad \sigma_1^2 \geq \cdots \geq \sigma_d^2 \geq 0, \tag{6}$$

so that $U = [v_1, \ldots, v_r]$ is the retained private basis of Eq. (1) and $u_k = v_k$. We write

$$S_\sigma = \sum_{k=1}^{r} \sigma_k^2, \qquad \Lambda = \mathrm{diag}(\lambda), \qquad P_\perp = I - UU^\top, \tag{7}$$

so $\omega_k = \sigma_k^2/S_\sigma$ as in Eq. (1). We assume $r \geq 1$ and $S_\sigma > 0$; when no nonzero memory exists the method uses G unchanged and nothing below applies. Rescaling C and the drift of Eq. (9) by n gives per-example quantities.

The filtered gradient of Eq. (4) and the *virtual SGD step* it defines are

$$\widetilde{G} = G - GU(I-\Lambda)U^\top = GP_\perp + GU\Lambda U^\top, \qquad W^+ = W - \eta\widetilde{G}, \quad \eta > 0. \tag{8}$$

The learning rate is η; the cap coefficients of Eq. (4) are $\zeta_{\mathrm{pr}}, \zeta_{\mathrm{sh}}$. Every single-step statement conditions on X, U, G, s, ρ and the caps, so these are fixed when λ varies. Note that $[\rho_k]_+$ enters b_k as a continuous weight and not only through eligibility: two directions with $\rho_k = 0.1$ and $\rho_k = 0.9$ receive different benefits.

The quantity we control is the squared response drift of the virtual step on the frozen historical inputs,

$$\mathcal{S}(\lambda) = \big\|(W^+ - W)X^\top\big\|_F^2 = \sum_{t=1}^{n} \big\|(W^+ - W)x_t\big\|_2^2. \tag{9}$$

B.2 Release Fractions Act Exactly on Historical Responses

Proposition 1 (Response identity and pointwise bound)**.** *$\widetilde{G}U = GU\Lambda$, so for every $x = Ua \in \mathrm{span}(U)$,*

$$(W^+ - W)x = -\eta GU\Lambda a = -\eta \sum_{k=1}^{r} \lambda_k a_k \, Gu_k, \tag{10}$$

and

$$\big\|(W^+ - W)x\big\|_2 \;\leq\; \eta\|GU\Lambda\|_2\|a\|_2 \;\leq\; \eta\Big(\sum_{k=1}^{r}\lambda_k^2 e_k\Big)^{1/2}\|a\|_2. \tag{11}$$

Proof. Right-multiplying Eq. (8) by U and using $U^\top U = I_r$ gives $\widetilde{G}U = GU - GU(I-\Lambda) = GU\Lambda$; substituting $x = Ua$ yields Eq. (10). The first inequality is the definition of the induced 2-norm and the second uses $\|\cdot\|_2 \leq \|\cdot\|_F$ together with $\|GU\Lambda\|_F^2 = \sum_k \lambda_k^2\|Gu_k\|_2^2$. □

The vectors Gu_k are in general not mutually orthogonal, so the pointwise cross terms in Eq. (10) cannot be discarded and the square-root form holds only as the bound (11). The next result removes cross terms by averaging against C, not by assuming orthogonality of the responses.

Theorem 2 (Exact spectral decomposition and budget guarantee)**.** *For every retained rank r in Eq.* (6)*,*

$$\mathcal{S}(\lambda) = \underbrace{\eta^2 \sum_{k=1}^{r} \sigma_k^2 \lambda_k^2 e_k}_{\mathcal{S}_{\mathrm{ret}}(\lambda)} + \underbrace{\eta^2 \sum_{q=r+1}^{d} \sigma_q^2 \|Gv_q\|_2^2}_{\mathcal{S}_{\mathrm{tail}}}, \qquad \mathcal{S}_{\mathrm{ret}}(\lambda) = \eta^2 S_\sigma \sum_{k=1}^{r} \omega_k e_k \lambda_k^2. \tag{12}$$

This is an identity, not a bound, and requires neither $\mathrm{rank}(C) \leq r$ nor any property of λ; the tail is independent of the current allocation. If $\mathcal{V} \neq \varnothing$, put $\alpha_{\max} = \max_{k\in\mathcal{V}}(1+s_k)\bar{\lambda}_k$ and $s_{\max} = \max_{k\in\mathcal{V}} s_k$. Every feasible λ of Eq. (5) *then satisfies*

$$\mathcal{S}_{\mathrm{ret}}(\lambda) \;\leq\; \eta^2 S_\sigma \, \alpha_{\max} B \;\leq\; \eta^2 S_\sigma \, \lambda_{\max}(1+s_{\max}) B \;\leq\; 2\,\eta^2 S_\sigma \, \lambda_{\max} B. \tag{13}$$

If $\mathcal{V} = \varnothing$ then $\lambda = 0$ and $\mathcal{S}_{\mathrm{ret}} = 0$. Total drift adds $\mathcal{S}_{\mathrm{tail}}$ to each right-hand side.

Proof. By the trace identity and the eigendecomposition of C, $\mathcal{S} = \eta^2 \operatorname{tr}(\widetilde{G} C \widetilde{G}^\top) = \eta^2 \sum_q \sigma_q^2 \|\widetilde{G} v_q\|_2^2$. For $q \le r$, Proposition 1 gives $\widetilde{G} v_q = \lambda_q G v_q$; for $q > r$, $U^\top v_q = 0$ gives $\widetilde{G} v_q = G v_q$. Substituting $\sigma_k^2 = S_\sigma \omega_k$ gives Eq. (12). For the budget bound, $\lambda_k = 0$ off $\mathcal{V}$ and $\omega_k e_k = (1 + s_k) c_k$ by Eq. (4), so with $\lambda_k \le \bar{\lambda}_k$,

$$\sum_k \omega_k e_k \lambda_k^2 = \sum_{k \in \mathcal{V}} (1 + s_k) c_k \lambda_k^2 \le \sum_{k \in \mathcal{V}} (1 + s_k) \bar{\lambda}_k \, c_k \lambda_k \le \alpha_{\max} \sum_{k \in \mathcal{V}} c_k \lambda_k \le \alpha_{\max} B.$$

The remaining inequalities use $\bar{\lambda}_k \le \lambda_{\max}$ and $s_k \le 1$. □

Corollary 3 (Zero budget recovers hard projection)**.** *If $\beta = 0$ then every eligible coordinate has positive cost, so feasibility forces $\lambda = 0$, $\widetilde{G} = G P_\perp$ as in Eq.* (4)*, and $\mathcal{S} = \mathcal{S}_{\mathrm{tail}}$. Hard projection is therefore the zero-budget endpoint of the method rather than a separate algorithm. When $\operatorname{rank}(C) \le r$ this gives $\mathcal{S} = 0$; under truncation, zero retained drift does not imply zero total drift.*

Equation (12) presumes that U holds eigenvectors of the stated C. It is exact at any retained rank of a truncated spectral memory, but is not a claim of exactness for an arbitrary approximate basis; eigensolver residuals or approximate accumulated moments need separate error accounting.

B.3 First-Order Plasticity and Finite-Step Descent

Theorem 4 (Alignment of the filtered gradient)**.** *For $0 \le \lambda_k \le 1$ let $A = \langle G, \widetilde{G} \rangle_F$ and $D = \|\widetilde{G}\|_F^2$. Then*

$$A = \|G P_\perp\|_F^2 + \sum_{k=1}^{r} \lambda_k e_k, \qquad D = \|G P_\perp\|_F^2 + \sum_{k=1}^{r} \lambda_k^2 e_k, \qquad A \ge D \ge 0, \tag{14}$$

and $\partial A / \partial \lambda_k = e_k \ge 0$, strictly positive exactly when $G u_k \ne 0$. If $\mathcal{L}_\mathcal{T}$ is L_{sm}-smooth in W with $L_{\mathrm{sm}} > 0$, then

$$\mathcal{L}_\mathcal{T}(W^+) \le \mathcal{L}_\mathcal{T}(W) - \eta A + \tfrac{1}{2} L_{\mathrm{sm}} \eta^2 D \le \mathcal{L}_\mathcal{T}(W) - \eta\big(1 - \tfrac{1}{2}\eta L_{\mathrm{sm}}\big) D. \tag{15}$$

If $D = 0$ the virtual step is zero; if $D > 0$, any $0 < \eta < 2/L_{\mathrm{sm}}$ gives strict decrease.

Proof. The right-input subspaces of $G P_\perp$ and $G U \Lambda U^\top$ are orthogonal, so their Frobenius inner product vanishes; expanding Eq. (8) gives the two equalities, and $A \ge D$ follows from $\lambda_k \ge \lambda_k^2$ on $[0, 1]$. The first inequality in Eq. (15) is the descent lemma applied with $W^+ - W = -\eta \widetilde{G}$, and the second substitutes $A \ge D$. □

Corollary 5 (Blockwise-filtered virtual step on the full model)**.** *Let θ collect all parameter blocks, $\mathcal{F}$ the filtered linear layers, and $\widetilde{\mathbf{G}}$ the concatenation of $\widetilde{G}_\ell$ for $\ell \in \mathcal{F}$ with the unmodified G_j for $j \notin \mathcal{F}$. Then*

$$\langle \nabla_\theta \mathcal{L}_\mathcal{T}, \widetilde{\mathbf{G}} \rangle = \sum_{\ell \in \mathcal{F}} A_\ell + \sum_{j \notin \mathcal{F}} \|G_j\|_F^2 \;\ge\; \sum_{\ell \in \mathcal{F}} D_\ell + \sum_{j \notin \mathcal{F}} \|G_j\|_F^2 = \|\widetilde{\mathbf{G}}\|^2, \tag{16}$$

so if $\mathcal{L}_\mathcal{T}$ is L_{sm}-smooth in θ, the joint virtual step $\theta^+ = \theta - \eta \widetilde{\mathbf{G}}$ strictly decreases $\mathcal{L}_\mathcal{T}$ for any $0 < \eta < 2/L_{\mathrm{sm}}$ whenever $\widetilde{\mathbf{G}} \ne 0$.

Proof. The inner product and squared norm decompose blockwise; Theorem 4 gives $A_\ell \ge D_\ell$ for each filtered block, and unfiltered blocks contribute $\|G_j\|_F^2$ to both sides. Apply the descent lemma in θ as in Eq. (15). □

Corollary 5 covers the joint blockwise-filtered gradient produced across all active agents and filtered layers before optimizer processing, with the loss backpropagated through the communication paths of Eq. (3). It is a statement about the corresponding virtual SGD step; the realized clipped Adam step is discussed in Appendix B.7.

Corollary 6 (Linear plasticity, quadratic drift)**.** *Combining Theorems 2 and 4, the same release vector enters the two quantities with different order:*

$$\frac{\partial A}{\partial \lambda_k} = e_k, \qquad \frac{\partial \mathcal{S}_{\mathrm{ret}}}{\partial \lambda_k} = 2\eta^2 S_\sigma\, \omega_k e_k \lambda_k. \tag{17}$$

Hence for a direction with $Gu_k \neq 0$ the marginal first-order gain is positive and constant, whereas the marginal retained drift vanishes at $\lambda_k = 0$. Near the origin the two therefore grow at different orders,

$$\Delta A = O(\lambda_k), \qquad \Delta \mathcal{S}_{\mathrm{ret}} = O(\lambda_k^2), \tag{18}$$

and wherever both derivatives are positive the marginal retained drift per unit of marginal first-order alignment gain is

$$\frac{\partial \mathcal{S}_{\mathrm{ret}}/\partial \lambda_k}{\partial A/\partial \lambda_k} = 2\eta^2 S_\sigma\, \omega_k \lambda_k, \tag{19}$$

which increases in the historical energy share ω_k and in the release already granted. The reciprocal is the corresponding alignment-gain-per-unit-drift ratio.

The asymmetry in Eq. (18) provides a local theoretical motivation for soft release rather than uniform hard projection: at $\lambda_k = 0$ the retained-subspace drift is stationary while the first-order alignment gain is not. It does not show that the particular allocator of Eq. (5), with its $[\rho_k]_+$ weighting, support discount and linear budget, is an optimal design. Two further qualifications matter. First, being locally second-order in retained drift is not being free in the allocator: the constraint of Eq. (5) charges $c_k\lambda_k$ linearly, so a direction consumes budget at first order even where it produces drift only at second order. Second, Eq. (19) concerns the current objective and the frozen historical responses, and does not assert that larger release improves finite-step loss or policy return.

B.4 What the Implemented Allocator Optimizes

The cost $c_k = \omega_k e_k/(1+s_k)$ of Eq. (4) is a *support-discounted proxy*: by Eq. (12) the true retained drift carries $\omega_k e_k$ with no support term. Granting corroborated directions more capacity is a design choice, and the factor $1 + s_{\max}$ in Eq. (13) bounds its effect on the guarantee.

Theorem 7 (Greedy solves Eq. (5) exactly)**.** *Order the eligible positive-cap coordinates by non-increasing $\varrho_k = b_k/c_k$. Starting from $B_{\mathrm{rem}} = B$, assign in this order*

$$\lambda_k = \min\{\bar{\lambda}_k,\ B_{\mathrm{rem}}/c_k\}, \qquad B_{\mathrm{rem}} \leftarrow B_{\mathrm{rem}} - c_k\lambda_k, \tag{20}$$

with zero for ineligible and zero-cap coordinates. This is a global optimum of Eq. (5)*, has at most one coordinate strictly between zero and its cap, and costs $O(r \log r)$ after score construction. Ties may admit other optima, so the maximizer need not be unique.*

Proof. Substituting $y_k = c_k\lambda_k$ and $a_k = c_k\bar{\lambda}_k$ turns Eq. (5) into $\max \sum_k \varrho_k y_k$ subject to $0 \le y_k \le a_k$ and $\sum_k y_k \le B$, with $\varrho_k > 0$ and $c_k > 0$ on the eligible set. Let y^* be the greedy profile. If $B = 0$ every feasible y is zero and there is nothing to prove. Otherwise the coordinates with $y_k^* > 0$ form a prefix of the sorted order, and we let ϱ^* be the ratio of the last of them. Take any feasible y. If $y_k^* > y_k \ge 0$, then k lies in that prefix, so $\varrho_k \ge \varrho^*$. If $y_k^* < y_k \le a_k$, then $y_k^* < a_k$, so k is the last filled coordinate or comes after it, and $\varrho_k \le \varrho^*$. In both cases $\varrho_k(y_k^* - y_k) \ge \varrho^*(y_k^* - y_k)$, and summing over k,

$$\sum_k \varrho_k(y_k^* - y_k) \ \ge\ \varrho^*\Big(\sum_k y_k^* - \sum_k y_k\Big).$$

If the budget is exhausted, $\sum_k y_k^* = B \ge \sum_k y_k$ and the right side is nonnegative. Otherwise every positive-cap coordinate is saturated, so $y^* \ge y$ componentwise and the left side is nonnegative because every $\varrho_k > 0$. Hence y^* is optimal. Fill-to-cap leaves at most one coordinate strictly between zero and its cap, and sorting dominates the cost. □

What β means. By Eq. (5), β is the fraction of the proxy cost needed to bring every eligible direction to its cap. It is dimensionless but is not a fraction of true squared drift. Under $G \mapsto aG$ with $a > 0$ the quantities e_k, b_k, c_k and B all scale by a^2 while ρ_k is unchanged, so the allocation is scale invariant *provided* the eligibility floor ϵ_0 of Eq. (5) does not change $\mathcal{V}$. Because ϵ_0 is absolute, shrinking gradients late in a stage can contract $\mathcal{V}$ and implicitly anneal release; the size of $\mathcal{V}$ over training is therefore a quantity to monitor, and Appendix E.11 reports what our records

retain: $|\mathcal{V}|$ over eight-batch continuations and full-run eligibility fractions. Per-step $|\mathcal{V}|$ over the full runs was not logged, so late-stage contraction is neither established nor excluded. Note also that $\varrho_k = [\rho_k]_+(1+s_k)/\omega_k$ does not depend on e_k: gradient energy sets how much budget a direction consumes, not its rank in the fill.

A quadratic alternative, for analysis only. Replacing the linear cost by a quadratic one, with the same benefits, caps and eligible set, gives a different problem that the method does *not* solve:

$$\max_{0\le\lambda\le\bar\lambda} \sum_{k\in\mathcal{V}} b_k\lambda_k \quad \text{s.t.} \quad \mathcal{Q}(\lambda) := \sum_{k\in\mathcal{V}} c_k\lambda_k^2 \le B_{\mathrm{quad}}, \qquad \lambda_k = 0\ (k\notin\mathcal{V}). \tag{21}$$

Write $\mathcal{Q}_{\mathrm{cap}} = \sum_{k\in\mathcal{V}} c_k\bar\lambda_k^2$. The solution is $\lambda = 0$ when $B_{\mathrm{quad}} = 0$ and $\lambda = \bar\lambda$ when $B_{\mathrm{quad}} \ge \mathcal{Q}_{\mathrm{cap}}$; for $0 < B_{\mathrm{quad}} < \mathcal{Q}_{\mathrm{cap}}$ it is the water-filling profile

$$\lambda_k^\star = \min\Big\{\bar\lambda_k,\ \frac{b_k}{2\psi c_k}\Big\} = \min\Big\{\bar\lambda_k,\ \frac{[\rho_k]_+(1+s_k)}{2\psi\,\omega_k}\Big\}, \tag{22}$$

with the unique $\psi > 0$ making $\mathcal{Q}(\lambda^\star) = B_{\mathrm{quad}}$. The gradient energy e_k cancels: it sets how much budget a direction consumes, not how much release it is granted. Holding the remaining factors fixed, a larger historical energy share ω_k yields a smaller uncapped level, consistent with Eq. (19); the alignment weight $[\rho_k]_+$, the support discount $1+s_k$ and the cap $\bar\lambda_k$ enter as well, so the profile is not determined by ω_k alone.

Derivation. The objective is linear and the feasible set convex; in the stated nondegenerate regime a sufficiently small strictly positive vector lies below the caps and satisfies the constraint strictly, so Slater's condition holds. For fixed $\psi > 0$, maximizing $\sum_k(b_k\lambda_k - \psi c_k\lambda_k^2)$ over the box gives Eq. (22) by scalar stationarity and clipping, uniquely since $b_k, c_k > 0$. Its cost is continuous and non-increasing in ψ, equals $\mathcal{Q}_{\mathrm{cap}}$ for small ψ and tends to zero as $\psi\to\infty$, so the multiplier exists and is unique for budgets strictly between the two extremes. KKT gives optimality; the boundary budgets are handled directly. □

Remark 8 (A shared preference score is not a shared allocation). The uncapped levels in Eq. (22) are proportional to the same $\varrho_k = b_k/c_k = [\rho_k]_+(1+s_k)/\omega_k$ that orders the greedy fill, but the two allocations differ in form: greedy saturates caps in order, whereas with positive quadratic budget every eligible coordinate is positive. Saturation order there is governed by $\varrho_k/\bar\lambda_k$, so unequal caps can reorder it. Moreover, for every feasible λ of Eq. (5), $\mathcal{Q}(\lambda) \le \lambda_{\max}\sum_k c_k\lambda_k \le \lambda_{\max}B$, which establishes feasibility, not equality of optima. At equal right-hand sides and $\lambda_k \le 1$ the linear constraint is the more restrictive of the two, so it is a linear-cost surrogate rather than a relaxation. With a single positive-benefit direction the two rules release $\beta\bar\lambda$ and $\sqrt{\beta_{\mathrm{quad}}}\bar\lambda$; at $\beta = \beta_{\mathrm{quad}} = 0.15$ that is 15% against roughly 38.7% of the cap. Switching solvers at the same numerical budget fraction is an algorithm change, not a correction.

B.5 Shared Basis: Optimal Linear Compression of Agent-Centered Variation

Let $Z_i \in \mathbb{R}^{n_i\times d_z}$ hold agent i's comorbidity latents with $n_i > 0$ and mean μ_i, and write $\mathring{Z}_i = Z_i - \mathbf{1}\mu_i^\top$ for the centered blocks used in Eq. (2), so that

$$\widehat{\Sigma}_i = \tfrac{1}{n_i}\mathring{Z}_i^\top\mathring{Z}_i, \qquad \bar\Sigma = \tfrac{1}{M}\sum_{i=1}^{M}\widehat{\Sigma}_i. \tag{23}$$

Lemma 9 (Agent-balanced empirical covariance). $\widetilde{Z}^\top\widetilde{Z} = \bar\Sigma$*: each agent's empirical covariance enters with weight* $1/M$ *regardless of* n_i*, and replicating one agent's rows leaves the result unchanged.*

Proof. Block i of Eq. (2) contributes $\mathring{Z}_i^\top\mathring{Z}_i/(Mn_i) = \widehat{\Sigma}_i/M$; summing gives the identity. Row replication scales numerator and count equally and preserves μ_i. □

Proposition 10 (Optimality for agent-centered variation). *Draw an agent* I *uniformly from* $\{1,\dots,M\}$*, then a row* z_I *uniformly from its stored samples, and set* $\varsigma = z_I - \mu_I$*. Among* $Q\in\mathbb{R}^{d_z\times r_{\mathrm{sh}}}$ *with* $Q^\top Q = I$*, the leading* r_{sh} *eigenvectors* U^{sh} *of* $\bar\Sigma$ *minimize*

$$J_{\mathrm{within}}(Q) = \tfrac{1}{M}\sum_{i=1}^{M}\mathbb{E}_i\big\|(I-QQ^\top)(z_i-\mu_i)\big\|_2^2 = \mathbb{E}\big\|(I-QQ^\top)\varsigma\big\|_2^2, \tag{24}$$

with minimum $\sum_{q>r_{\mathrm{sh}}}(\sigma_q^{\mathrm{sh}})^2$*, where* σ_q^{sh} *are the singular values in Eq.* (2)*. If* $\mathrm{tr}(\bar{\Sigma})>0$ *the retained fraction of balanced within-agent variance is* $\sum_{q\le r_{\mathrm{sh}}}(\sigma_q^{\mathrm{sh}})^2/\sum_q(\sigma_q^{\mathrm{sh}})^2$*. Eigenvalue ties can make the optimal subspace non-unique.*

Proof. The sampling rule gives $\mathbb{E}[\varsigma\varsigma^\top]=\bar{\Sigma}$ by Lemma 9. Since $QQ^\top$ is an orthogonal projector, $J_{\mathrm{within}}(Q)=\mathrm{tr}(\bar{\Sigma})-\mathrm{tr}(Q^\top\bar{\Sigma}Q)$. Writing $\bar{\Sigma}=\sum_q \nu_q v_q v_q^\top$ with non-increasing ν_q, we get $\mathrm{tr}(Q^\top\bar{\Sigma}Q)=\sum_q \nu_q\|Q^\top v_q\|_2^2$ with weights in $[0,1]$ summing to r_{sh}; this is at most $\sum_{q\le r_{\mathrm{sh}}}\nu_q$, attained by the leading eigenvectors. □

The transmitted message is a different object. The message of Eq. (3) uses the pooled center μ and a $\tanh$ nonlinearity. For any common center, with $B_\mu=\frac{1}{M}\sum_i(\mu_i-\mu)(\mu_i-\mu)^\top$,

$$\mathbb{E}\big[(z_I-\mu)(z_I-\mu)^\top\big]=\bar{\Sigma}+B_\mu,\qquad J_\mu(Q)=J_{\mathrm{within}}(Q)+\tfrac{1}{M}\sum_i\big\|(I-QQ^\top)(\mu_i-\mu)\big\|_2^2, \tag{25}$$

the cross terms vanishing because $\mathbb{E}_i(z_i-\mu_i)=0$. Proposition 10 optimizes the first term only. Using the balanced center $\bar{\mu}=M^{-1}\sum_i\mu_i$ does not remove the second, since $B_\mu=B_{\bar{\mu}}+(\bar{\mu}-\mu)(\bar{\mu}-\mu)^\top$ and $B_{\bar{\mu}}$ generally survives. For instance, with $z_1=(\xi,a)$ and $z_2=(\xi,-a)$ where $\mathbb{E}\xi=0$, $\mathrm{Var}\,\xi=1$ and $a>1$, we get $\bar{\Sigma}=\mathrm{diag}(1,0)$, whose leading direction is the first coordinate, whereas the common-centered second moment is $\mathrm{diag}(1,a^2)$, whose leading direction is the second. Re-centering alone therefore cannot extend the proposition.

B.6 Absolute Correlation and Multiple Support Comparisons

For a comparison $h=(i,j,\ell,g,k,q)$ let $\widehat{r}_h$ be the *signed* Pearson correlation, so that the support score $\mathsf{R}^g_{ij\ell,kq}$ of Section 3.3 is $\mathsf{R}_h=|\widehat{r}_h|$; constant or unavailable responses are set to zero and cannot pass a positive threshold. Let $\mathcal{M}$ denote the declared family of comparisons behind a support claim, including every component match over which a maximum is taken and, as applicable, agent pairs, layers, groups and stages.

Proposition 11 (Idealized null and a family-wise bound)**.** *Fix* h *and suppose its* $n_h>2$ *response pairs are i.i.d. draws from a nondegenerate bivariate Gaussian with zero correlation, the component functions and the family being fixed independently of these draws or the statement holding conditionally on their construction. Then* $T_h=\widehat{r}_h\sqrt{(n_h-2)/(1-\widehat{r}_h^2)}\sim t_{n_h-2}$*, so for* $0<\tau_f<1$*,*

$$p_h(\tau_f)=2\Big[1-F_{t_{n_h-2}}\Big(\tau_f\sqrt{\tfrac{n_h-2}{1-\tau_f^2}}\Big)\Big],\qquad \Pr\Big(\max_{h\in\mathcal{M}}\mathsf{R}_h>\tau_f\Big)\le\min\Big\{1,\sum_{h\in\mathcal{M}}p_h(\tau_f)\Big\}. \tag{26}$$

The union bound requires no independence across comparisons; for a mixture of null and non-null pairs it applies to the true nulls.

Proof. Under the stated null the signed correlation has density $(1-r^2)^{(n_h-4)/2}/\mathrm{B}(\frac{1}{2},\frac{n_h-2}{2})$ on $(-1,1)$ (Fisher, 1915), equivalently the exact beta null implemented in Virtanen et al. (2020); the change of variable gives t_{n_h-2}, and symmetry with monotonicity in $|r|$ gives the two-sided tail. The union of events has probability at most the sum of theirs. □

Remark 12 (The transform must precede the absolute value). For $n_h>3$ the familiar approximation applies to the signed correlation, $Z_h=\mathrm{arctanh}\,\widehat{r}_h\mathrel{\dot\sim}\mathcal{N}(0,(n_h-3)^{-1})$, whereas $\mathrm{arctanh}\,\mathsf{R}_h=|Z_h|$ is approximately folded normal rather than centered normal; its tail is $2[1-\Phi(\sqrt{n_h-3}\,\mathrm{arctanh}\,\tau_f)]$. For extreme tails Eq. (26) should be evaluated directly rather than extrapolated from the normal approximation. A single 32×32 match matrix already contributes 1024 comparisons before layers, groups and agent pairs are counted.

The idealized null is not the operative one here. Rows from one patient need not be independent, so a row count is not a count of independent observations; agents also share patient state, time trends and overlapping inputs. A calibrated diagnostic therefore requires a patient-level exchangeability scheme that preserves within-patient trajectories and the intended background time structure (Winkler et al., 2015), repeating the full matching and selection pipeline under each permutation b and recording $T^{(b)}=\max_{h\in\mathcal{M}}|\widehat{r}_h^{(b)}|$, with

$$\widehat{p}_{\max}=\frac{1+\sum_{b=1}^{N_{\mathrm{perm}}}\mathbf{1}[T^{(b)}\ge T^{\mathrm{obs}}]}{N_{\mathrm{perm}}+1},\qquad T^{\mathrm{obs}}=\max_{h\in\mathcal{M}}\mathsf{R}_h, \tag{27}$$

the max-statistic construction of Westfall & Young (1993). Shuffling individual time rows does not preserve those dependencies, and a percentile of the pooled distribution of individual correlations does not calibrate a maximum over the family. A basis may be held fixed only when the design justifies conditioning on it. This is an omnibus test of the declared global null; family-wise control under arbitrary mixtures needs a further procedure. A support claim should accordingly state the declared family and its size, row and unique-patient counts, strata and alignment rules, permutation count, and the null maxima against τ_f; Appendix E.11 reports these quantities, except unique-person counts, which are unavailable to that audit, for the experiments of Section 4.

B.7 Scope of the Guarantees

Frozen inputs versus current inputs. Equation (12) is exact for the stored X at any layer, but for a deeper layer the current upstream parameters produce different historical inputs X_t, and $\Delta W X_t^\top = \Delta W X^\top + \Delta W (X_t - X)^\top$ with $\|\Delta W (X_t - X)^\top\|_F \le \|\Delta W\|_2 \|X_t - X\|_F$. Only the first term is controlled. Biases, nonlinearities, unfiltered parameters and changing team context lie outside the weight-only identity; for a first layer with fixed raw inputs the second term is absent.

Accumulation over a stage. For a fixed basis across a stage, $\|(W_T - W_0)U\|_F \le \sum_t \eta_t \|G_t U \Lambda_t\|_F$ and $\|(W_T - W_0)X^\top\|_F \le \sum_t \sqrt{\mathcal{S}_t}$. Turning either into an $O(T)$ statement requires a uniform bound on the summands, so a per-step budget does not by itself give a horizon-independent forgetting guarantee; memory is also rebuilt between stages, which prevents reuse of a single fixed-basis statement across stages.

Virtual step versus realized step. Filtering is applied to layer-weight gradients before clipping and Adam, so the realized update is generally not $-\eta \widetilde{G}$ and momentum or coordinate-wise preconditioning can reintroduce components along protected directions. Proposition 1 still describes the filtered gradient exactly, but Theorems 2 and 4 and Corollary 5 are statements about the virtual SGD step. Empirical evaluation should therefore distinguish the virtual-step drift from the realized parameter-step drift, and agreement between them is not verification of a theorem about Adam.

Representation drift versus policy value. No result here bounds the change in offline policy value. Stability of layer responses is not stability of returns, and offline value estimates carry their own error; we treat the two as separate empirical questions.

Compression versus communication. Proposition 10 concerns an orthogonal linear projection of agent-centered residuals before $\tanh$. It does not establish optimality of the transmitted message, generalization to future latents, or any advantage over a learned message head; that comparison requires an ablation matching message width, receiver architecture, data access and training budget (Appendix E.8).

Association versus function. A thresholded correlation is evidence of association between component responses on aligned observations, that is, of response redundancy within the team. It does not establish that the teammate stores the same knowledge, that the two directions serve the same causal function, or that releasing a corroborated direction loses no information; the support term in c_k is a design choice rather than a consequence of Eq. (12).

Protocol scope. These results assume a given frozen memory and do not establish a replay-free protocol; historical cohorts are accessed at stage boundaries to rebuild the memories and support scores, while within-stage optimization does not replay them, as stated in Section 2 and Appendix C.

C Problem Formulation Details

Aligned training examples. For patients with comorbidities, the disease-specific views are aligned by patient identity and time block, so an aligned training example is $(\mathcal{T}, \{(o_i, a_i, r_i, o_i', \delta_i)\}_{i \in \mathcal{T}})$, where a_i is the recorded treatment action, r_i the transition reward, o_i' the next observation, and δ_i the termination indicator of agent i's disease-specific trajectory. Because disease-specific trajectories may terminate at different times, rewards and termination indicators remain agent-specific. Action spaces may also share treatment variables, so we do not assume that agents control disjoint interventions.

Stage-wise team collections. The team collection at stage s is $\mathfrak{T}_1 = \{\{i\} : i \in \mathcal{I}\}$, $\mathfrak{T}_2 = \{\{i, j\} : i, j \in \mathcal{I},\ i < j\}$ and $\mathfrak{T}_3 = \{\mathcal{I}\}$. At each stage, agents are optimized on the current cohorts only; earlier data are accessed at stage boundaries solely to rebuild the spectral memories, and aligned comorbidity observations are used to estimate cross-agent associations; no earlier cohort is replayed during within-stage optimization, with the goal of learning new disease combinations without losing performance on earlier ones.

D METHOD DETAILS

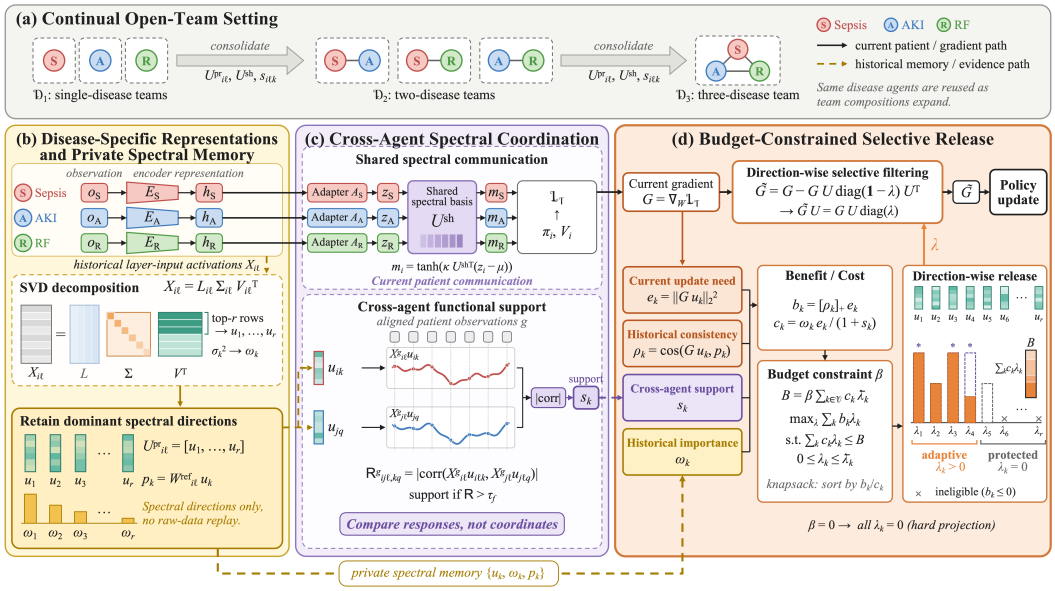


Figure 3: Overview of COSTAR-MARL. (a) The same disease-specific agents are reused across single-disease, two-disease, and three-disease training stages. (b) Private spectral memories summarize historical representation directions and their importance. (c) Shared spectral communication exchanges current patient information, while functional support compares agents' responses on aligned observations. (d) Current gradients, historical importance, and functional support jointly determine direction-wise release under individual caps and a cost-weighted budget. Dashed outlines mark the per-direction caps $\bar{\lambda}_k$, asterisks mark supported directions ($s_k > \epsilon$), and crosses mark ineligible ones.

Figure 1 in the main text abstracts modules (b)–(d).

Rank of the private spectral memory. In Eq. (1), we retain the leading eigenvectors of the accumulated second-moment matrix, discarding eigenvalues at or below 10^{-8} and retaining at most $r_{\max} = 32$ directions per protected layer. No cumulative spectral-energy threshold is used. The memory is stored as the uncentered second moment $C_{i\ell} \in \mathbb{R}^{d_\ell \times d_\ell}$, whose size depends only on the layer width, so it does not grow with the number of patients seen and no historical activations need to be replayed.

Why the agent-balanced stacking is exact. Writing the blocks of Eq. (2) out gives $\widetilde{Z}^\top \widetilde{Z} = \frac{1}{M}\sum_{i=1}^{M} \widehat{\Sigma}_i$, where $\widehat{\Sigma}_i = (Z_i - \mathbf{1}\mu_i^\top)^\top (Z_i - \mathbf{1}\mu_i^\top)/n_i$ is agent i's own average within-agent covariance. Centering removes the raw-magnitude problem, since $\widehat{\Sigma}_i$ depends only on variation around each agent's own mean. Dividing by n_i removes the sample-count problem, turning agent i's contribution into a per-sample average rather than a sum that grows with n_i, and the outer $\frac{1}{M}$ then gives every agent an equal share regardless of how many comorbidity samples it has. Lemma 9 and Proposition 10 state this formally.

Cross-agent support scores. A direction q of agent j matches direction k of agent i when $\mathsf{R}^g_{ij\ell,kq} > \tau_f$ for a threshold $\tau_f \in (0, 1)$, with R the absolute Pearson correlation defined in Section 3.3. We collect the agents providing at least one match, $\mathcal{J}_{i\ell k} = \{j \neq i : \max_{g,q} \mathsf{R}^g_{ij\ell,kq} > \tau_f\}$,

and record the strongest matched correlation $f_{i\ell k} = \max_{j\in\mathcal{J}_{i\ell k}} \max_{g,q} \mathsf{R}^{g}_{ij\ell,kq}$. Because a single component pair can match by chance, we also keep the coarser agent-level statistic $\varphi_{ij} \in [0,1]$, the fraction of agent i's retained directions in layer ℓ that find any match in agent j, which is stable when individual correlations are noisy. The effective support takes whichever evidence is stronger, $s_{i\ell k} = \max(f_{i\ell k}, \max_{j\in\mathcal{J}_{i\ell k}} \varphi_{ij})$ when $\mathcal{J}_{i\ell k} \neq \emptyset$ and $s_{i\ell k} = 0$ otherwise. Proposition 11 bounds the chance that an unrelated pair passes the threshold.

Offline policy learning objective. Agents are trained with zero intermediate rewards and terminal rewards $+R$ and $-R$ for survival and death. The clipped target and the advantage are $y_i = \mathrm{clip}(r_i + \gamma(1-\delta_i)V_{\bar\theta_i}(\xi_i'), -R, R)$ and $\widehat{A}_i = y_i - V_{\theta_i}(\xi_i)$, where $\gamma \in (0,1)$ is the discount, θ_i and $\bar\theta_i$ are agent i's critic parameters and their slowly updated target copy, and ξ_i' is the policy input of Eq. (3) built from the next observation. For an active team $\mathcal{T}$ the objective is

$$\mathcal{L}_{\mathcal{T}} = \frac{1}{|\mathcal{T}|}\sum_{i\in\mathcal{T}} \mathbb{E}_{\mathcal{D}_{s,\mathcal{T}}}\Big[-\log \pi_i(a_i|\xi_i)\,\mathrm{sg}(\widehat{A}_i) + c_V\,\ell_{\mathrm{Huber}}\big(V_{\theta_i}(\xi_i), \mathrm{sg}(y_i)\big) - c_H\mathcal{H}(\pi_i) - c_{\mathrm{BC}}\log\pi_i(a_i|\xi_i)\Big], \tag{28}$$

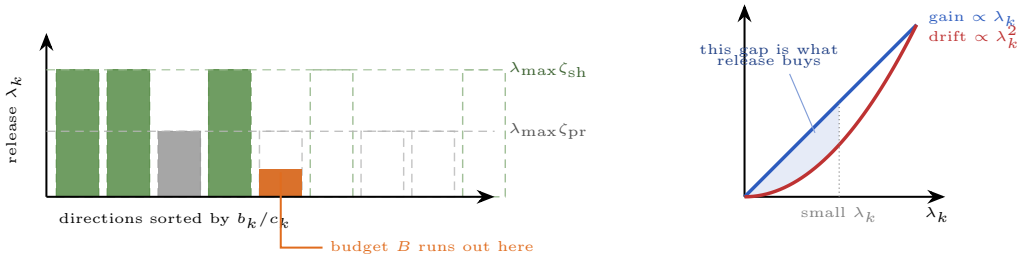
where $\mathcal{D}_{s,\mathcal{T}}$ denotes the stage-s training examples whose active team is $\mathcal{T}$, ℓ_{Huber} is the Huber loss, $\mathcal{H}(\pi_i)$ is the policy entropy, and $c_V, c_H, c_{\mathrm{BC}} > 0$ weight the critic, entropy, and behaviour-cloning terms. Filtered gradients are applied before clipping and the Adam update.

No private memory. This ablation variant in Table 2 disables the construction and updating of agent-specific SVD memories and skips the associated functional-support estimation and budgeted gradient filtering. Thus, $\widetilde{G} = G$ before gradient clipping and Adam updates. The disease-specific networks, shared spectral communication, and shared-basis updates are retained.

Why $\beta = 1$ underperforms unfiltered training. In Table 2, releasing every eligible direction to its cap ($\beta = 1$) is worse than removing the private memory altogether, even though it permits less change inside the retained subspace. We attribute this to the eligibility rule. Only directions with $\rho_k > 0$ are ever released, and at $\beta = 1$ all of them are, so within the retained subspace the filter acts as a one-sided gate on the minibatch gradient: components that would move a historical response away from p_k pass, and components that would move it back are blocked. For a direction the current task is indifferent to, the minibatch component along u_k is zero-mean noise, but the gated component is not; its expectation points away from p_k, so historical responses drift in directions the task never asked for. The unfiltered gradient carries the same noise with zero mean, and hard projection freezes such directions, which is why both extremes outperform $\beta = 1$. A small budget avoids this because the greedy ratio $b_k/c_k = [\rho_k]_+(1+s_k)/\omega_k$ does not depend on e_k: it ranks directions by alignment alone, and a budget of $\beta = 0.15$ is typically exhausted by the few directions with large ρ_k before reaching those whose alignment is of the order $m^{-1/2}$ expected from noise for a layer of output width m (the row dimension of G and p_k). We have not isolated this effect experimentally; it is the explanation most consistent with the ordering of the three rows.

Why only conflicting directions earn benefit. The benefit $b_k = [\rho_k]_+ e_k$ is zero when $\rho_k \leq 0$, that is, when the current update would move the response along u_k toward its stored value p_k rather than away from it. A reviewer may ask why an update that agrees with old knowledge is withheld. The reason is what the budget is for. Hard projection already blocks every in-subspace component, and the release budget exists to buy back the specific components that new coordination needs and that hard projection prevents: those that contradict a historical response. An update that reinforces p_k still changes the retained response (Theorem 2 carries no sign term), so it consumes drift, but it resolves no conflict between the old team and the new one, and the current task can still express the same preference through the unprotected complement and through the messages of Eq. (3). Using $|\rho_k|$ instead would also make the fill order $|\rho_k|(1+s_k)/\omega_k$ blind to the direction of the conflict, so the budget would be spent on directions the old and new objectives already agree on. The cost of this choice is the one-sided gate described in the preceding paragraph, which is benign at small budgets and is the reason a small β is preferable to $\beta = 1$. A symmetric variant with $b_k = |\rho_k| e_k$ is a natural ablation that we have not run.

Illustration of Budgeted Release. Figure 4 illustrates how the release budget is allocated and why partial release provides a useful local trade-off. It is a schematic of the allocation rule and its response-level analysis.



(a) How the budget is spent

caps filled in order of benefit per unit cost until $\sum_k c_k\lambda_k = B$; the rest stay frozen

(b) Why release is cheap

adaptation grows with the release, disturbance only with its square

Figure 4: Schematic of budget allocation and local release scaling. (a) For one agent and one filtered layer, eligible directions are ordered by decreasing benefit-to-cost ratio b_k/c_k. Dashed outlines indicate directional caps; filled bars indicate allocated release. Green and grey directions have the supported and unsupported caps, respectively, and orange marks a partially filled direction. The greedy allocation has at most one partial fill per independent budget problem. (b) With the gradient and historical basis fixed, the allocation benefit is linear in release, while squared retained-response drift under a virtual SGD step is quadratic. The curves are normalized illustrations; their vertical separation is not a measured net benefit.

How the budget is allocated. For each agent–layer problem, the allocator maximizes $\sum_{k\in\mathcal{V}} b_k\lambda_k$ subject to $\sum_{k\in\mathcal{V}} c_k\lambda_k \le B$ and $0 \le \lambda_k \le \bar{\lambda}_k$, where $\mathcal{V}$ contains the eligible directions. It fills caps in decreasing order of b_k/c_k, using a fixed order for ties. If the remaining budget cannot fill the next cap, that direction receives the remainder and later directions receive zero. Supported directions have cap $\lambda_{\max}\zeta_{\text{sh}}$, whereas unsupported directions have the smaller positive cap $\lambda_{\max}\zeta_{\text{pr}}$. Support therefore changes the permitted release but does not guarantee allocation. With $B = \beta\sum_{k\in\mathcal{V}} c_k\bar{\lambda}_k$, $\beta = 0$ recovers hard protection and $\beta = 1$ fills all eligible caps. Zero release blocks the gradient along the corresponding retained direction; learning in the orthogonal complement remains possible.

Interpreting the linear and quadratic curves. For a fixed gradient G and retained basis, the allocation benefit contributed by direction k is $b_k\lambda_k$. Under the ideal filtered update $\Delta W = -\eta\widetilde{G}$, its squared historical response change is $\eta^2\nu_k\|Gu_k\|_2^2\lambda_k^2$, where ν_k is the corresponding eigenvalue of the historical input second moment. Thus, for an eligible direction and fixed coefficients, benefit changes at first order in release and squared response drift at second order near zero. Panel (b) depicts this local scaling; the two curves do not share a common utility scale. A small total budget may still fill a few individual caps, so it does not force every release fraction to be small. These statements concern retained linear responses under virtual SGD, rather than a direct guarantee for Adam updates, WISt improvement, or clinical outcomes.

E Additional Experimental Details and Results

E.1 Cohorts, Actions, and Training Budgets

The eICU trajectories contain 1,770,824 four-hour observations, comprising 1,065,332 training observations and 705,492 test observations. The data split is held fixed across methods and training seeds.

The sepsis action space discretizes intravenous fluids and vasopressors into a 5×5 grid. The AKI action space uses a 5×5 grid of fluids and ultrafiltration output. The RF action space comprises $4 \times 4 \times 3$ combinations of inspired oxygen fraction, set respiratory rate, and positive end-expiratory pressure. Disease-specific observations and action spaces are preserved across cohorts. All methods use standardized inputs and action boundaries fitted from the corresponding $\mathcal{D}_1$ training data.

Table 3: Number of MIMIC-IV ICU stays in the training and test sets. Counts refer to ICU stays rather than unique patients. Each stay is counted once within a cohort, regardless of the number of disease agents evaluated. Agents is the number of disease agents evaluated on the cohort.

Cohort	Stage	Train stays	Test stays	Total	Agents
Sepsis only	D1	6,190	4,128	10,318	1
AKI only	D1	13,939	9,294	23,233	1
RF only	D1	1,860	1,240	3,100	1
Sepsis + AKI	D2	8,481	5,654	14,135	2
Sepsis + RF	D2	678	452	1,130	2
AKI + RF	D2	771	514	1,285	2
Sepsis + AKI + RF	D3	1,106	738	1,844	3
Total	–	33,025	22,020	55,045	–

Table 4: Number of ICU stays in the eICU cohorts after preprocessing. The seven cohorts are mutually exclusive. Each stay is counted once, regardless of the number of disease agents evaluated. Counts refer to ICU stays rather than unique patients. Agents is the number of disease agents evaluated on the cohort.

Cohort	Stage	Train stays	Test stays	Total	Agents
Sepsis only	D1	2,156	1,434	3,590	1
AKI only	D1	31,154	20,624	51,778	1
RF only	D1	2,425	1,617	4,042	1
Sepsis + AKI	D2	2,005	1,349	3,354	2
Sepsis + RF	D2	699	432	1,131	2
AKI + RF	D2	2,937	1,966	4,903	2
Sepsis + AKI + RF	D3	807	561	1,368	3
Total	–	42,183	27,983	70,166	–

Training protocol. COSTAR-MARL receives gradient updates from the current stage and accesses previously seen training observations when consolidating memory at stage boundaries. We train for 4, 2, and 5 epochs in $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$, respectively, with batch size 64. On MIMIC-IV this gives 19,616, 3,478, and 1,090 training iterations (eICU counts are in Appendix E.2). The encoder and hidden widths are 128; the common latent dimension is 64; private and shared spectral ranks are capped at 32 and 16. We use Adam with learning rate 3×10^{-4} and release-budget fraction $\beta = 0.15$. Rewards are zero before termination and $+100$ or -100 at termination according to survival or death (90-day survival on MIMIC-IV, hospital survival on eICU), with discount factor $\gamma = 0.99$. Each sequential method spanning all three diseases is evaluated on all seven test cohorts at the end of each stage, following the fixed training schedule.

COSTAR-MARL and the continual-learning baselines using the same actor–critic architecture align observations by ICU-stay identifier and recorded block index while retaining agent-specific transitions. CFCQL, B3C, COMAD, CQL, Factored Single Policy, and the clinical baselines additionally require consistency of outcome, reward, termination, and successor block across participating disease views, with an observed common successor for nonterminal transitions. Each method uses the eligible transitions satisfying its alignment requirements. Where training iteration counts are matched, eligible transitions are reused as needed.

EWC, LwF, GEM, GPM, SGP, and Adam-NSCL use the common actor–critic architecture with their respective retention mechanisms. LwF distills the previous model's policy outputs on current-stage observations, while GEM constrains updates using gradients computed from stored examples. GPM and SGP use SGD; Adam-NSCL projects the candidate Adam parameter update; EWC and Sequential FT use Adam. LwF and GEM use incremental shared-memory updates, whereas COSTAR-MARL, GPM, and SGP rebuild shared memory using previously seen training data. Historical-data access therefore differs across these implementations. Cumulative Replay uses earlier observations for gradient training, while COSTAR-MARL uses them for memory consolidation. Balanced Joint trains on all cohorts from the outset.

Auxiliary training includes behavior-model fitting for CFCQL, encoder training for HRRT, and the fitting procedures for AI Clinician and Ventilation FQI. We use an implemented AKI adaptation of HRRT (Xu et al., 2026) with 25 discrete actions. Factored Single Policy and COSTAR-MARL

have online parameter counts of 254,387 and 253,989, respectively, excluding target networks and memory buffers.

Hyperparameters. Table 5 collects every hyperparameter of COSTAR-MARL in one place; the values are those of the configuration files in the supplementary material, which also document the baseline-specific settings. Gradients are processed in the order filtering (Eq. (4)), global L_2-norm clipping over each agent's parameters, then the Adam update; the target critic is initialized as a copy of the online critic and updated after every optimization step by $\bar{\theta}_i \leftarrow 0.99\,\bar{\theta}_i + 0.01\,\theta_i$.

Table 5: Hyperparameters of COSTAR-MARL. The same values are used on MIMIC-IV and eICU unless stated. Symbols refer to Sections 3.1–3.4 and Appendix D.

Group	Hyperparameter	Value
Architecture	Encoder and hidden width	128
	Common latent dimension d_z	64
	Private spectral rank cap $r_{\max}$	32
	Shared spectral rank cap r_{sh}	16
	Private eigenvalue floor	10^{-8} (no energy threshold)
	Message scale κ	0.25
Optimization	Optimizer / learning rate	Adam / 3×10^{-4}
	Batch size	64
	Epochs per stage ($\mathcal{D}_1/\mathcal{D}_2/\mathcal{D}_3$)	4 / 2 / 5
	Global gradient-norm clipping	10.0
	Target-critic update	Polyak, 0.01 per step
	Discount γ	0.99
	Terminal reward $\pm R$	± 100
	Loss weights c_V, c_H, c_{BC}	0.5, 0.01, 0.05
	Huber threshold	1.0
Release	Budget fraction β	0.15
	Supported cap $\lambda_{\max}\zeta_{\mathrm{sh}}$	0.15
	Unsupported cap $\lambda_{\max}\zeta_{\mathrm{pr}}$	0.0495
	Eligibility floor ϵ_0	10^{-8}
	Support tolerance ϵ	10^{-8}
Support	Matching threshold τ_f	0.35
	Matching layers	adapter, first actor, first critic
	Aligned rows per cohort pair	$\leq$ 8,192 (min. 256)
	Subsampling seed	42 (cohort offsets)
Evaluation	WISt weight truncation	100
	Behavior-reference seed	42
	Bootstrap resamples	2,000

E.2 Random Seeds and Reproducibility

Which runs underlie each result. Tables 1, 9 and 11 report means over three training runs per method and dataset, with the seeds listed in Tables 6 and 7. The same three runs are used for all reported metrics within a method–dataset pair, including Across-stage Avg., Final Avg., BWT, and Above ref. All three COSTAR-MARL disease agents in Table 9 come from the same three trained systems as the MIMIC-IV main comparison. Tables 2 and 10 report means over three runs per configuration; their full-model row reuses the three COSTAR-MARL runs of Table 1.

Table 8, Figure 2 and Figure 5 instead show a single training run per method, selected for display because per-cohort trajectories do not average meaningfully across runs. For COSTAR-MARL this is the first of its three MIMIC-IV runs, seed 42. Values in these displays therefore differ from the three-run means: the seed-42 run has a 12-pair final average of 59.772 and a three-disease score of 62.548, against the three-run means of 59.788 and 59.034 in Tables 1 and 2; its per-disease four-cohort averages are 66.107 (sepsis), 56.367 (AKI) and 56.842 (RF), against 64.226, 56.588 and 58.551 in Table 9. All quantitative claims in the main text other than Section 4.3's per-cohort trajectories use the three-run means.

Sources of training randomness. Each run initializes the Python, NumPy, and PyTorch random-number generators with its training seed. This seed controls neural-network initialization and train-

Table 6: Training seeds used for the three-run means in Table 1. The same three runs are used for all reported metrics within each method–dataset pair.

Method	MIMIC-IV	eICU
COSTAR-MARL	42, 43, 45	42, 49, 50
Sequential FT	44, 45, 46	42, 43, 44
Factored Single Policy	42, 44, 45	42, 43, 45
Cumulative Replay	43, 44, 46	42, 45, 46
EWC	43, 44, 46	42, 44, 46
LwF	42, 43, 46	42, 45, 46
GEM	42, 43, 46	43, 45, 46
GPM	42, 45, 46	42, 43, 44
SGP	42, 43, 45	43, 44, 46
Adam-NSCL	43, 44, 45	42, 44, 46
COMAD	42, 43, 44	43, 44, 46
CFCQL	43, 44, 46	42, 44, 46
B3C	42, 43, 46	44, 45, 46
CQL	43, 45, 46	42, 43, 45
Balanced Joint	44, 50, 51	42, 43, 45
Scratch-D2	43, 45, 46	42, 43, 45
Scratch-D3	42, 43, 45	44, 45, 46

Table 7: Training seeds used for the MIMIC-IV clinical comparison in Table 9. Across all three diseases, COSTAR-MARL uses the same three trained systems as in the MIMIC-IV main comparison.

Method	Training seeds
COSTAR-MARL (all three diseases)	42, 43, 45
AI Clinician	42, 45, 46
WD3QNE	42, 44, 46
HRRT	42, 43, 45
Ventilation FQI	42, 44, 46

ing minibatch shuffling or sampling. It also controls method-specific stochastic operations, including action sampling for EWC's Fisher estimate and B3C's training objective, and observation perturbations in COMAD. For the clinical methods, it controls clustering initialization in AI Clinician, training-patient resampling and randomized feature selection in WD3QNE, encoder training and minibatch sampling in HRRT, and regression-sample selection and randomized tree fitting in Ventilation FQI. Where stage-specific random streams are used, their offsets from the training seed are fixed.

Fixed data and configurations. Within each dataset, cohort membership, stored training/test partitions, dataset-level preprocessing, action discretization, feature normalization, stage order, and evaluation cohorts are held fixed. Changing a training seed does not repartition the data or refit dataset-level preprocessing. Within each method, the architecture specification, optimizer, learning rates, regularization settings, memory limits, and training-data access rules are fixed across runs.

For methods using the common neural training schedule, the stage budgets follow $4/2/5$ epochs with batch size 64; the resulting reference update counts are $(19{,}616, 3{,}478, 1{,}090)$ on MIMIC-IV and $(56{,}468, 4{,}704, 900)$ on eICU. Balanced Joint and Scratch variants retain their respective training budgets, while clinical fitting procedures retain their method-specific budgets. The discount factor is $\gamma = 0.99$, with zero intermediate rewards and terminal rewards ± 100. The terminal endpoint is 90-day survival on MIMIC-IV and hospital survival on eICU. These settings are fixed within a dataset; method-specific hyperparameters need not be identical across methods.

Auxiliary sampling and evaluation. Auxiliary sampling is separated from the training random stream. The actor–critic baselines use a fixed auxiliary data seed of 42 for retention-statistic sampling and GEM memory selection, with fixed stage-specific offsets. COMAD calibration uses the same fixed auxiliary seed convention. COSTAR-MARL's functional-support subsampling uses a fixed seed of 42 with cohort-specific offsets. Private and shared spectral bases are obtained from the collected representation statistics without separately sampling a random basis; their values can nevertheless change across runs because the learned representations change.

For each dataset, the fitted behavior references used for off-policy evaluation are constructed with seed 42 and reused across all methods and training seeds. These fixed evaluation references are distinct from any trainable behavior model used inside a baseline. The WISt truncation threshold is fixed at 100, and aggregation weights and the unit-level zero-completion rule are fixed across runs. WISt point estimates use the fixed test trajectories and do not depend on bootstrap resampling. The bootstrap intervals in Appendix E.9 quantify test-stay sampling uncertainty conditional on fitted policies and behavior references.

E.3 Agent-level Off-policy Evaluation

For cohort–agent pair j, let trajectory n have discounted return $\mathcal{G}_{n,j}$ and recorded actions $a_{n,t}$. Using the fixed fitted behavior reference $\widehat{\mu}_j$ and the target agent's policy π_j, we compute

$$\varpi_{n,j} = \prod_t \max\left\{ \frac{\pi_j(a_{n,t} \mid \xi_{n,t})}{\max\{\widehat{\mu}_j(a_{n,t} \mid o_{n,t}), 10^{-10}\}}, 10^{-300} \right\}, \tag{29}$$

$$w_{n,j} = \min\{\varpi_{n,j}, 100\}, \qquad \widehat{V}_j = \frac{\sum_n w_{n,j}\mathcal{G}_{n,j}}{\sum_n w_{n,j}}. \tag{30}$$

The product is computed in log space, converted to a trajectory weight, and truncated at 100. Probability floors provide numerical stabilization but do not establish that the behavior data support the target policy. A reported WISt value does not by itself establish adequate effective sample size or statistical significance.

The clinician reference is evaluated with the same fixed protocol. Reported differences subtract its WISt for the same cohort and disease agent before aggregation.

E.4 Disease-agent Results within Each Cohort

Table 8: Final WISt for each disease agent across the seven MIMIC-IV cohorts, for the seed-42 run of COSTAR-MARL shown in Figure 2 (single run; three-run means are in Table 1). Differences from the clinician reference use unrounded scores and are expressed in WISt points, not mortality percentages.

Cohort	Agent	Clinician	**COSTAR-MARL**	Δ
Sepsis only	Sepsis	47.680	87.194	+39.514
AKI only	AKI	62.935	64.275	+1.340
RF only	RF	23.075	45.406	+22.332
Sepsis + AKI	Sepsis	52.350	59.127	+6.778
Sepsis + AKI	AKI	52.350	54.686	+2.336
Sepsis + RF	Sepsis	6.222	53.327	+47.106
Sepsis + RF	RF	6.294	47.658	+41.364
AKI + RF	AKI	53.890	55.446	+1.556
AKI + RF	RF	58.979	62.502	+3.523
Sepsis + AKI + RF	Sepsis	45.087	64.780	+19.694
Sepsis + AKI + RF	AKI	45.087	51.062	+5.976
Sepsis + AKI + RF	RF	45.078	71.802	+26.725

Table 8 reports, for the seed-42 run, the final WISt for all 12 cohort–agent pairs across the seven patient cohorts, together with the corresponding clinician references and score differences. In this run COSTAR-MARL exceeds the clinician reference in all 12 pairs; across its three runs it does so in 11.667 of 12 on average (Table 1). Averaged across the four cohorts containing each disease, the improvements in this run are 28.273 WISt points for sepsis, 2.802 for AKI, and 23.486 for RF.

E.5 Comparison with Disease-Specific Clinical Methods

Table 9 compares each disease-specific clinical baseline with the corresponding COSTAR-MARL agent on the four cohorts containing that disease, using three-run means. COSTAR-MARL achieves the highest mean scores for both Across-stage Avg. and Final WISt across all three diseases. For sepsis, its final average WISt is 64.226, compared with −7.632 for AI Clinician and 13.686 for WD3QNE. For AKI, COSTAR-MARL obtains 56.588 versus 55.768 for HRRT, a gain of 0.820 points that lies within one standard deviation of either method. For RF, it reaches 58.551, compared with −0.152 for Ventilation FQI.

Table 9: Comparison with disease-specific clinical methods on MIMIC-IV. Results are reported as mean ± standard deviation over three runs. For each disease, Across-stage Avg. averages the 12 scores across four cohorts and three training stages, while Final WISt averages the four scores after the final stage. Unestimable scores are assigned zero before aggregation. Higher values are better, and bold indicates the highest mean for each metric within each disease.

Disease	Method	Across-stage Avg.	Final WISt
Sepsis	**COSTAR-MARL**	$\mathbf{51.822 \pm 3.819}$	$\mathbf{64.226 \pm 1.691}$
	AI Clinician	22.425 ± 9.219	-7.632 ± 10.527
	WD3QNE	10.429 ± 2.897	13.686 ± 23.704
AKI	**COSTAR-MARL**	$\mathbf{58.180 \pm 0.300}$	$\mathbf{56.588 \pm 1.032}$
	HRRT	53.525 ± 0.356	55.768 ± 1.470
RF	**COSTAR-MARL**	$\mathbf{57.048 \pm 1.503}$	$\mathbf{58.551 \pm 1.585}$
	Ventilation FQI	-5.562 ± 15.969	-0.152 ± 9.247

E.6 Retention and Backward-Transfer Definitions

Following the evaluation-matrix formulation of continual learning (Lopez-Paz & Ranzato, 2017; Saha et al., 2021), let $a(j) \in \{1, 2, 3\}$ denote the stage when the cohort associated with pair j is first used for training. The set $\mathcal{J}_{\text{old}} = \{j : a(j) < 3\}$ contains the nine cohort–agent pairs from the first two stages. We compute backward transfer as

$$\text{BWT} = \frac{1}{|\mathcal{J}_{\text{old}}|} \sum_{j \in \mathcal{J}_{\text{old}}} \left(Y_{3,j} - Y_{a(j),j} \right). \tag{31}$$

Positive BWT indicates that performance on earlier disease combinations improves on average relative to the scores immediately after they were first learned. BWT is reported alongside the final average, since a stable score can remain low and a high final average can hide declines in individual cohorts; Figure 2 shows the per-cohort trajectories behind it for one run.

E.7 Group Trajectories across the Training Sequence

Figure 5 compares the ten sequentially trained methods of Section 4.3 on MIMIC-IV. Unlike Table 1, each curve is a single training run per method; COSTAR-MARL is shown with seed 42, the same run as in Figure 2 and Table 8 (Appendix E.2). The single-disease, two-disease, and three-disease panels average three, six, and three disease-agent scores, respectively, and the comparisons below are between these single runs.

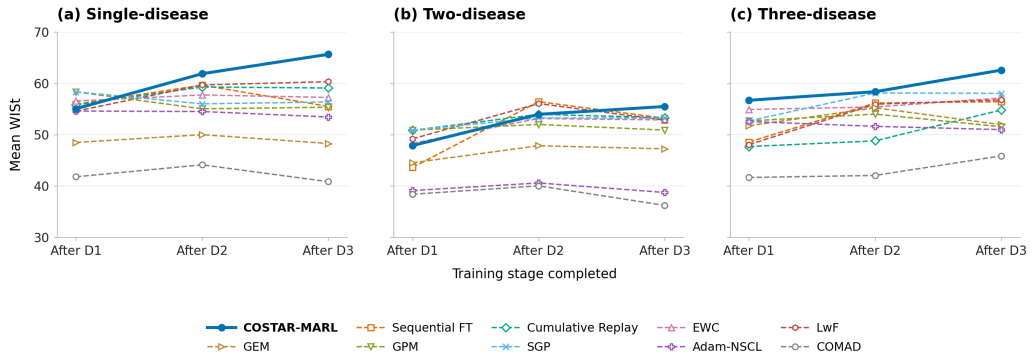


Figure 5: **Comparison of retention and adaptation across disease combinations.** Mean WISt for ten methods on MIMIC-IV after each training stage on (a) single-disease cohorts, (b) two-disease cohorts, and (c) the three-disease cohort, one training run per method (COSTAR-MARL: seed 42) rather than the three-run means of Table 1. Each point averages three, six, and three disease-agent scores, respectively. These groups are first used for training in $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$. Colors, markers, and line styles identify methods consistently across all panels, which share one vertical scale and legend.

In Figure 5(a), COSTAR-MARL improves from 55.026 to 61.848 and then 65.625 over the three stages. Sequential FT rises from 55.895 to 59.623 before falling to 55.474. LwF rises from 54.554 to 59.656 and then 60.298, the highest final single-disease score among the displayed comparison methods. Cumulative Replay and EWC end at 59.059 and 57.221, respectively. Adam-NSCL ends at 53.418. COSTAR-MARL therefore increases its single-disease average while learning more complex disease combinations, although individual cohorts can still decline, as shown in Figure 2(a).

Figure 5(b) separates adaptation during $\mathcal{D}_2$ from retention during subsequent training. COSTAR-MARL improves from 47.852 before $\mathcal{D}_2$ training to 53.935 afterward, a gain of 6.083 points. Sequential FT and LwF achieve higher immediate scores of 56.400 and 56.027, respectively. During $\mathcal{D}_3$, COSTAR-MARL increases further to 55.458, while Sequential FT and LwF fall to 53.172 and 52.805, respectively, and Cumulative Replay ends at 53.277. The comparison thus distinguishes immediate adaptation from the performance retained after further learning.

In Figure 5(c), COSTAR-MARL improves from 58.337 before $\mathcal{D}_3$ training to 62.548 afterward. This final score is the highest among the ten displayed methods, followed by SGP at 58.021, EWC at 57.126, and LwF at 56.748. Cumulative Replay has a larger training-stage gain, 6.017 points compared with COSTAR-MARL's 4.211, but starts lower and ends at 54.782. Comparing starting and final scores therefore complements the adaptation gain. The three-disease score averages only the three agents in that cohort; the final average of this run over all 12 cohort–agent pairs is 59.772, against the three-run mean of 59.788 in Table 1.

E.8 Communication Ablation

Table 10 compares the spectral message in Eq. (3) with a No-message variant and three learned alternatives: MLP, CommNet-style (Sukhbaatar et al., 2016), and TarMAC-style (Das et al., 2019) communication. The MLP uses hidden width 64 and message scale 0.5. Apart from the communication module, the remaining model components and training settings are held fixed.

COSTAR-MARL obtains a final average WISt of 59.788 and a three-disease score of 59.034. The No-message variant obtains 57.947 and 55.393, respectively, corresponding to gaps of 1.841 and 3.641 points in the reported summaries. The larger gap on the three-disease cohort indicates that the overall average alone does not fully reflect the observed differences between communication settings.

Table 10: Communication ablation on MIMIC-IV. Results are means over three runs per configuration. Final Avg. averages the 12 cohort–agent WISt scores after the final training stage; 3-dis. averages the three agents on the sepsis–AKI–RF cohort. COSTAR-MARL uses spectral communication and repeats Table 2.

Configuration	Final Avg.	3-dis.
COSTAR-MARL	**59.788**	**59.034**
No-message	57.947	55.393
MLP	58.079	56.334
CommNet-style	58.490	55.484
TarMAC-style	57.127	53.921

Among the learned alternatives, CommNet-style communication achieves the highest final average at 58.490, whereas MLP achieves the highest three-disease score at 56.334. COSTAR-MARL exceeds these respective values by 1.298 and 2.700 points. TarMAC-style communication obtains a final average of 57.127 and a three-disease score of 53.921. Spectral communication therefore has the highest mean on both metrics within the displayed three-run summaries.

E.9 Effective Sample Size and Uncertainty

We evaluate all 17 general methods on the same 12 MIMIC-IV cohort–agent pairs, using the same three runs per method as in Table 1. The training seeds and run protocol are given in Appendix E.2. Sequential methods are evaluated after the third stage; Balanced Joint and Scratch-D2/D3 are evaluated when their respective training procedures finish.

Let $r \in \{1, 2, 3\}$ index the three runs for a method and $j \in \{1, \ldots, J\}$ index the cohort–agent pairs, where $J = 12$. Exact zero target-action probabilities yield zero trajectory weights. Positive weights are computed in log space and capped at 100. Denoting these weights by $\widetilde{w}_{n,r,j}$, we compute

$$\mathrm{ESS}_{r,j} = \frac{\left(\sum_n \widetilde{w}_{n,r,j}\right)^2}{\sum_n \widetilde{w}_{n,r,j}^2}, \qquad \overline{\mathrm{ESS}} = \frac{1}{3J} \sum_{r=1}^{3} \sum_{j=1}^{J} \mathrm{ESS}_{r,j}, \tag{32}$$

setting $\mathrm{ESS}_{r,j} = 0$ when its denominator is zero. Average ESS is the arithmetic mean of the 36 run–pair ESS values; trajectory weights are not pooled across runs. The reported final score uses the

same fixed aggregation denominator:

$$\overline{V}^{(0)} = \frac{1}{3J}\sum_{r=1}^{3}\sum_{j=1}^{J}\widehat{V}_{r,j}^{(0)},$$
$$\widehat{V}_{r,j}^{(0)} = \begin{cases} \dfrac{\sum_n \widetilde{w}_{n,r,j}\mathcal{G}_{n,j}}{\sum_n \widetilde{w}_{n,r,j}}, & \sum_n \widetilde{w}_{n,r,j} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{33}$$

Undefined run–pair scores contribute zero; finite negative scores are retained. These three-run means agree with the MIMIC-IV Final Avg. column of Table 1 at the reported precision. Zero completion is a reporting convention, not evidence that an unsupported policy has value zero.

We draw 2,000 bootstrap resamples of complete ICU stays within each cohort, using common resampling multiplicities across agents, methods, and all three training runs. In each resample, we recompute every run–pair score, apply its zero convention, and then average over the same 36 run–pair estimates as in Equation 33. The 2.5th and 97.5th percentiles of these aggregate scores form the 95% interval. Thus, the interval is computed for the three-run mean, rather than by averaging endpoints of separate run-level intervals. Policies, behavior models, preprocessing, and the retained run sets remain fixed. The intervals quantify test-stay sampling uncertainty and exclude retraining and run-selection variability. Dependence between repeated stays of the same person is not modeled.

Table 11: Final WISt, 95% bootstrap intervals, and average effective sample size (ESS) on MIMIC-IV, using the same three runs per method as in Table 1. Final WISt and ESS are averaged equally over three runs and 12 cohort–agent pairs, with trajectory weights truncated at 100. Undefined run–pair scores contribute zero before averaging. Intervals are computed from 2,000 common within-cohort ICU-stay resamples, shared across runs, agents, and methods, with the same aggregation and zero convention applied in every resample. They quantify test-stay sampling uncertainty conditional on the fitted policies and behavior models. Bold indicates the highest Final WISt.

Method	Final WISt	95% CI	Average ESS
COSTAR-MARL	**59.788**	[55.809, 63.424]	762.416
Sequential FT	51.448	[48.867, 53.894]	1128.180
Factored Single Policy	51.472	[45.030, 57.069]	401.289
Cumulative Replay	49.247	[40.758, 57.129]	1065.760
EWC	56.590	[53.464, 59.583]	957.158
LwF	51.768	[48.288, 55.151]	829.140
GEM	49.059	[46.171, 51.873]	985.337
GPM	52.403	[48.492, 56.323]	541.822
SGP	55.113	[51.322, 58.780]	669.661
Adam-NSCL	47.839	[42.485, 52.810]	690.487
COMAD	39.499	[36.782, 42.164]	1332.993
CFCQL	24.413	[18.071, 36.686]	62.959
B3C	39.137	[36.304, 41.777]	1211.785
CQL	42.576	[32.249, 50.623]	110.163
Balanced Joint	57.098	[51.931, 61.921]	664.204
Scratch-D2	49.933	[47.384, 52.378]	1330.146
Scratch-D3	41.237	[38.823, 43.557]	852.870

Table 11 reports a three-run mean Final WISt of 59.788 for COSTAR-MARL, with a 95% interval of [55.809, 63.424] and an average ESS of 762.416. ESS measures weight concentration rather than policy performance: COMAD has a larger average ESS of 1,332.993 but a lower mean WISt of 39.499. Several intervals overlap; point-estimate rankings alone do not establish statistically significant pairwise differences.

CFCQL and CQL have smaller average ESS values of 62.959 and 110.163, respectively, and wider intervals than COSTAR-MARL. Among the 36 original-sample run–pair estimates, CFCQL and CQL require zero completion in 5 and 3 cases, respectively. Both methods require zero completion in at least one run–pair estimate in each of the 2,000 bootstrap resamples. Their intervals describe the zero-completed three-run summary and do not resolve the underlying lack of support.

Sensitivity to the truncation level. The truncation level of 100 in Eq. (30) is a choice, so Tables 12 and 13 recompute the Final Avg. of every method at truncation levels of 10, 25, 50, 100, 250 and

Table 12: Final Avg. on MIMIC-IV at different trajectory-weight truncation levels. Entries average the same three runs and 12 cohort–agent pairs as Table 1. Whole-trajectory weights are truncated before self-normalization. Undefined pair-level scores contribute zero. Bold marks the highest value at each level.

Method	10	25	50	100	250	1000	Untruncated
COSTAR-MARL	**60.231**	**59.968**	**59.943**	**59.788**	**59.351**	**58.724**	**52.346**
Sequential FT	52.548	52.063	51.790	51.448	50.923	50.077	10.488
Factored Single Policy	53.030	52.402	51.998	51.472	50.684	49.761	11.823
Cumulative Replay	51.700	50.911	50.198	49.247	47.767	45.880	30.152
EWC	57.498	57.171	57.006	56.590	55.891	54.780	41.175
LwF	53.366	52.702	52.351	51.768	50.948	49.631	18.876
GEM	50.851	50.197	49.661	49.059	48.471	47.500	21.956
GPM	54.386	53.782	53.138	52.403	51.439	49.989	24.004
SGP	56.852	56.216	55.738	55.113	54.156	52.692	28.765
Adam-NSCL	49.529	48.876	48.456	47.839	47.049	45.358	29.130
COMAD	40.141	39.912	39.740	39.499	39.174	38.741	38.762
CFCQL	24.428	24.473	24.512	24.413	23.912	23.048	11.403
B3C	39.687	39.380	39.269	39.137	38.895	38.455	26.754
CQL	40.316	40.469	41.237	42.576	44.145	44.214	15.831
Balanced Joint	59.008	58.115	57.575	57.098	56.294	54.720	37.343
Scratch-D2	50.959	50.573	50.335	49.933	49.294	48.262	3.581
Scratch-D3	43.325	42.584	41.891	41.237	40.399	39.097	36.948

Table 13: Final Avg. on eICU at different trajectory-weight truncation levels, with the same conventions as Table 12. Bold marks the highest value at each level.

Method	10	25	50	100	250	1000	Untruncated
COSTAR-MARL	**60.979**	**60.764**	**60.486**	**60.532**	**60.571**	**60.557**	70.610
Sequential FT	56.306	55.907	55.626	55.404	55.263	54.888	66.842
Factored Single Policy	44.760	44.333	44.432	44.649	44.650	44.043	52.696
Cumulative Replay	52.980	52.804	52.606	52.342	51.914	51.050	68.853
EWC	54.579	54.090	53.945	53.693	53.356	52.777	50.040
LwF	46.572	46.040	45.795	45.646	45.415	44.725	55.602
GEM	46.594	45.899	45.512	45.352	45.089	44.646	56.038
GPM	54.233	53.780	53.430	53.255	52.994	52.346	70.788
SGP	56.406	55.871	55.636	55.325	54.954	54.400	66.249
Adam-NSCL	50.817	50.135	49.825	49.518	49.307	48.911	51.432
COMAD	43.024	42.596	42.426	42.195	41.924	41.655	53.085
CFCQL	32.141	31.231	30.987	30.795	30.505	29.610	50.897
B3C	38.585	38.270	38.160	37.997	37.815	37.591	34.717
CQL	37.727	36.972	36.875	36.869	36.803	36.386	46.930
Balanced Joint	52.139	51.630	51.244	51.006	50.420	49.122	62.097
Scratch-D2	52.872	52.392	52.023	51.810	51.604	50.945	**71.153**
Scratch-D3	56.553	56.094	55.747	55.463	55.190	54.827	67.675

1000, and without truncation, using the same three runs, the same 12 cohort–agent pairs and the same zero convention as Table 1. COSTAR-MARL has the highest Final Avg. at every level from 10 to 1000 on both datasets, and the gap to the next method does not close as the level changes: on MIMIC-IV it ranges from 1.2 (against Balanced Joint at 10) to 3.9 points (at 1000), and on eICU from 4.4 to 5.7 points. Untruncated WIS is dominated by a few trajectories with extreme weights and its rankings are not stable across levels, so we do not interpret that column.

Uncertainty of the clinician reference. The Above ref. column of Table 1 compares final scores with a point estimate of the clinician reference. Table 14 gives that reference with a 95% bootstrap interval for every cohort–agent pair and for the overall average, using the same 2,000 common ICU-stay resamples as Table 11. The overall clinician reference is 41.585 [39.271, 43.847] on MIMIC-IV and 40.560 [38.237, 42.969] on eICU, so the COSTAR-MARL Final Avg. of 59.788 and 60.532 lies above the upper end of the clinician interval on both datasets, and its own MIMIC-IV interval [55.809, 63.424] does not overlap the clinician one. As with the other intervals in this section, these quantify test-stay sampling uncertainty with the fitted models held fixed.

E.10 Performance Variation across Runs

Tables 15 and 16 provide the sample standard deviations accompanying the means in Table 1, using the same three runs for each method. Standard deviations describe variation within these reported run sets.

Table 14: Clinician reference values with 95% bootstrap confidence intervals. Each pair-level value is the mean observed discounted terminal return. Intervals use 2,000 common ICU-stay resamples within each cohort; the same multiplicities are shared across agents and target policies. The overall average gives equal weight to the 12 cohort–agent pairs. Two agents evaluated on the same cohort can have slightly different references (e.g. Sepsis–RF on MIMIC-IV) because each agent's trajectories end at its own disease-specific termination indicator (Appendix C), so the set of scored trajectories is agent-specific. Endpoints are 90-day mortality for MIMIC-IV and hospital mortality for eICU.

Cohort	Agent	MIMIC-IV	eICU
Sepsis only	Sepsis	$47.680\ [45.323, 49.950]$	$63.628\ [60.234, 66.994]$
AKI only	AKI	$62.935\ [61.664, 64.151]$	$62.272\ [61.574, 62.922]$
RF only	RF	$23.075\ [18.261, 28.016]$	$39.180\ [35.925, 42.371]$
Sepsis–AKI	Sepsis	$52.350\ [50.353, 54.311]$	$53.770\ [50.023, 57.397]$
Sepsis–AKI	AKI	$52.350\ [50.353, 54.311]$	$53.770\ [50.023, 57.397]$
Sepsis–RF	Sepsis	$6.222\ [-1.920, 14.006]$	$43.147\ [35.672, 50.671]$
Sepsis–RF	RF	$6.294\ [-1.849, 14.088]$	$43.147\ [35.672, 50.671]$
AKI–RF	AKI	$53.890\ [48.791, 58.813]$	$23.197\ [20.139, 26.356]$
AKI–RF	RF	$58.979\ [53.022, 64.406]$	$23.197\ [20.139, 26.356]$
Sepsis–AKI–RF	Sepsis	$45.087\ [39.756, 50.538]$	$27.139\ [20.073, 33.934]$
Sepsis–AKI–RF	AKI	$45.087\ [39.756, 50.538]$	$27.139\ [20.073, 33.934]$
Sepsis–AKI–RF	RF	$45.078\ [39.602, 50.569]$	$27.139\ [20.073, 33.934]$
Overall average	–	$41.585\ [39.271, 43.847]$	$40.560\ [38.237, 42.969]$

Table 15: Overall performance on MIMIC-IV across seven disease cohorts. Results are reported as mean $\pm$ sample standard deviation over the same three runs per method used in Table 1. Across-stage Avg. averages the 36 cohort–agent scores over three stages; Final Avg. averages the final 12 scores. BWT measures backward transfer on the nine previously learned cohort–agent pairs. Above ref. counts final scores exceeding the corresponding clinician reference, out of 12. Higher values are better; bold indicates the highest mean in each column. Dashes indicate metrics that are not applicable to the training protocol.

Method	Across-stage Avg.	Final Avg.	BWT	Above ref. (/12)
COSTAR-MARL	$\mathbf{55.683 \pm 1.615}$	$\mathbf{59.788 \pm 0.300}$	$\mathbf{+5.271 \pm 4.864}$	11.667 ± 0.577
Sequential FT	51.590 ± 1.141	51.448 ± 0.580	-2.017 ± 2.531	9.000 ± 0.000
Factored Single Policy	47.707 ± 2.491	51.472 ± 6.245	$+1.879 \pm 10.186$	9.333 ± 1.155
Cumulative Replay	49.482 ± 2.281	49.247 ± 3.201	$+0.321 \pm 3.058$	8.000 ± 2.646
EWC	55.079 ± 0.989	56.590 ± 1.371	$+1.669 \pm 1.407$	11.667 ± 0.577
LwF	50.825 ± 1.337	51.768 ± 1.264	$+0.559 \pm 2.125$	9.667 ± 1.155
GEM	49.219 ± 0.568	49.059 ± 0.364	-1.260 ± 0.700	7.667 ± 0.577
GPM	53.399 ± 0.480	52.403 ± 0.293	-1.941 ± 0.202	10.667 ± 0.577
SGP	55.279 ± 0.785	55.113 ± 0.082	-1.048 ± 0.537	$\mathbf{12.000 \pm 0.000}$
Adam-NSCL	48.209 ± 2.093	47.839 ± 2.237	-1.131 ± 0.658	7.667 ± 0.577
COMAD	40.190 ± 0.215	39.499 ± 0.423	-2.813 ± 0.335	3.667 ± 0.577
CFCQL	41.882 ± 12.081	24.413 ± 6.484	-26.742 ± 22.140	4.333 ± 1.155
B3C	40.125 ± 0.093	39.137 ± 0.176	-3.054 ± 0.340	3.667 ± 0.577
CQL	46.798 ± 3.535	42.576 ± 4.257	-5.769 ± 3.831	8.000 ± 1.000
Balanced Joint	–	57.098 ± 2.399	–	11.000 ± 0.000
Scratch-D2	–	49.933 ± 0.303	–	8.333 ± 1.155
Scratch-D3	–	41.237 ± 0.251	–	6.667 ± 0.577

Table 16: Overall performance on eICU across seven disease cohorts. Results are reported as mean ± sample standard deviation over the same three runs per method used in Table 1. Across-stage Avg. averages the 36 cohort–agent scores over three stages; Final Avg. averages the final 12 scores. BWT measures backward transfer on the nine previously learned cohort–agent pairs. Above ref. counts final scores exceeding the corresponding clinician reference, out of 12. Higher values are better; bold indicates the highest mean in each column. Dashes indicate metrics that are not applicable to the training protocol.

Method	Across-stage Avg.	Final Avg.	BWT	Above ref. (/12)
COSTAR-MARL	$\mathbf{54.446 \pm 1.504}$	$\mathbf{60.532 \pm 3.441}$	$\mathbf{+8.413 \pm 2.512}$	10.667 ± 0.577
Sequential FT	51.829 ± 1.357	55.404 ± 2.273	$+2.970 \pm 2.051$	$\mathbf{11.667 \pm 0.577}$
Factored Single Policy	47.369 ± 3.390	44.649 ± 3.723	-7.517 ± 4.683	6.000 ± 1.000
Cumulative Replay	51.218 ± 1.280	52.342 ± 2.456	$+2.832 \pm 2.991$	9.667 ± 2.082
EWC	52.891 ± 0.634	53.693 ± 1.986	-0.215 ± 1.533	10.000 ± 1.000
LwF	47.533 ± 9.117	45.646 ± 12.967	-0.720 ± 1.941	8.333 ± 2.082
GEM	46.092 ± 0.608	45.352 ± 0.551	-1.876 ± 1.040	8.000 ± 1.000
GPM	52.236 ± 0.812	53.255 ± 0.501	$+0.843 \pm 0.546$	11.333 ± 0.577
SGP	53.814 ± 0.762	55.325 ± 2.097	$+1.831 \pm 1.520$	10.000 ± 2.000
Adam-NSCL	51.432 ± 1.248	49.518 ± 4.115	$+0.200 \pm 0.122$	10.333 ± 1.528
COMAD	42.898 ± 2.055	42.195 ± 1.246	-1.266 ± 0.293	5.000 ± 1.000
CFCQL	32.421 ± 5.357	30.795 ± 6.306	-0.593 ± 0.256	5.667 ± 0.577
B3C	40.808 ± 0.807	37.997 ± 0.332	-2.434 ± 2.787	2.667 ± 0.577
CQL	38.179 ± 4.859	36.869 ± 0.685	$+2.491 \pm 6.932$	6.333 ± 1.155
Balanced Joint	–	51.006 ± 0.975	–	9.667 ± 0.577
Scratch-D2	–	51.810 ± 1.080	–	9.667 ± 0.577
Scratch-D3	–	55.463 ± 1.438	–	11.000 ± 0.000

Table 17: Native support prevalence and positive-score distribution. Brackets denote the interquartile range; zeros are excluded only from the last column.

Seed	Stage	Supported directions	Positive s_k: median [Q1, Q3]
42	D2	42/288 (14.58%)	0.461 [0.396, 0.674]
42	D3	43/288 (14.93%)	0.440 [0.370, 0.694]
43	D2	35/288 (12.15%)	0.538 [0.503, 0.630]
43	D3	53/288 (18.40%)	0.437 [0.394, 0.585]
44	D2	36/288 (12.50%)	0.440 [0.379, 0.623]
44	D3	45/288 (15.62%)	0.477 [0.406, 0.630]

E.11 Functional Support and Release Diagnostics

Scope and configuration. We report the existing MIMIC-IV mechanism audit for all three audit seeds 42, 43, and 44. These checkpoints are a separate diagnostic set, rather than the selected runs underlying every performance table. In particular, seed 44 is not among the MIMIC-IV COSTAR-MARL training seeds 42, 43 and 45 of Table 6, so these diagnostics do not describe exactly the runs behind the performance tables. The configured matching threshold is $\tau_f = 0.35$; the implementation accepts absolute correlations $\mathsf{R} \geq \tau_f$. This is a fixed algorithmic threshold, not a calibrated significance level. Native matching uses at most 8,192 aligned training rows per cohort pair (minimum 256), keyed by ICU stay and time block. Support is refreshed at the incoming-stage boundary, using the incoming cohorts; this is the start-of-stage refresh of the two per stage described in Appendix E.12. Matching operates on three layers (shared adapter, first actor layer, and first critic layer), each with 32 retained directions per agent. Thus support rates below have 288 unique agent–layer–direction entries per seed and stage; a direction matched by two teammates is counted once. The remaining filtered layers are included in allocation diagnostics, but not in this support-rate denominator.

Support prevalence and scores. Table 17 reports the fraction with $s_k > 0$ and the distribution conditional on $s_k > 0$. The remaining directions have $s_k = 0$, so the unconditional median and interquartile range are all zero in every seed–stage combination. Native support was reconstructed from the original checkpoint and input tables, with exact agreement with the archived matching records; the resulting scores also reproduce every archived directional cap.

Eligibility and release. Each incoming cohort independently restarts from its preceding-stage checkpoint and takes eight native minibatch updates. Only $\beta = 0.15$ advances these diagnostic networks; counterfactual $\beta = 0, 1$ allocations are excluded here. There are 336 agent–layer–batch problems before D2 and 168 before D3 per seed, each with 32 directions. The native eligibility rule is $b_k > 0$, $c_k > 10^{-8}$, and $e_k > 10^{-8}$. Supported and unsupported caps are 0.15 and 0.0495. Table 18 summarizes $|\mathcal{V}|$ per problem and allocation over all direction exposures. Releases at most 10^{-7} are classified as numerical zero; cap equality uses $\max(10^{-7}, 10^{-5}\bar{\lambda}_k)$. Partial release is the remainder after the zero and cap categories. No problem has more than one partial fill. These are short continuations, not a trajectory over the original full training run.

Table 18: Eight-batch allocation diagnostics. $|\mathcal{V}|$ is median [Q1, Q3]. Empty denotes the percentage of problems with no eligible directions. Zero/cap percentages use direction exposures. L_{S} and L_{U} are mean release for supported and unsupported exposures across all filtered layers.

Seed	Stage	$\lvert\mathcal{V}\rvert$	Empty	Zero	Cap	L_{S}	L_{U}
42	D2	6.0 [2.0, 16.0]	5.7	79.7	17.4	0.0217	0.0090
42	D3	11.0 [3.0, 16.0]	2.4	77.5	19.5	0.0271	0.0101
43	D2	6.5 [2.0, 15.0]	8.0	80.0	17.2	0.0257	0.0088
43	D3	12.0 [2.8, 16.0]	7.1	76.6	20.5	0.0357	0.0104
44	D2	6.5 [2.0, 16.0]	4.8	79.3	17.7	0.0262	0.0091
44	D3	12.0 [4.0, 16.0]	10.7	75.3	21.9	0.0356	0.0112

What full-training records retain. The original full runs retain stage–agent averages of layer-level eligibility fractions and release, but not per-step $|\mathcal{V}|$. Table 19 reports these existing full-run aggregates separately: within each stage, agent means are weighted by their actual minibatch counts; the native within-batch average gives each filtered layer equal weight. These aggregates show nonzero eligibility and release over the run, but cannot establish or exclude the late-stage contraction discussed in the allocator analysis. Full-training per-step monitoring remains unreported, and the eight-batch diagnostics do not fill that gap.

Table 19: Original full-training aggregates, distinct from the short diagnostic continuations. Nonzero follows the native logger's strict $\lambda > 0$ rule.

Seed	Stage	Mean eligible (%)	Mean λ	Mean nonzero (%)
42	D2	34.97	0.0138	28.53
42	D3	33.30	0.0126	25.56
43	D2	33.72	0.0131	27.05
43	D3	32.78	0.0136	26.17
44	D2	34.02	0.0127	26.07
44	D3	34.47	0.0141	28.14

Complete-ICU-stay permutation audit. For each seed and incoming stage, the preceding checkpoint and bases are held fixed. A separate sample contains complete incoming-cohort training stays; it differs from the native row sample used above. Stays are stratified by the joint full observed grid of time blocks and elapsed times across agents. Within each stratum, one agent is anchored and the others' complete trajectories are permuted, with one coherent mapping per agent across all pairs and layers. Singleton strata are excluded. Calendar origins are ignored. Pairwise correlations use common time blocks, preserving the whole stay as the exchange unit. Table 20 gives the sample sizes; the same sampled stays are used for all seeds.

Each permutation recomputes all component-pair correlations and their stage-wide maximum over $3 \times 3 \times 32^2 = 9{,}216$ comparisons: three cohort–agent-pair contexts, three matching layers, and all direction pairs. With 999 permutations, $\widehat{p} = (1 + \#\{T^{(b)} \geq T^{\mathrm{obs}}\})/1000$. The observed and null maxima are reported in Table 21 without pooling individual correlations. Each seed–stage is a separate family; there is no correction across seeds or stages.

Interpretation and remaining limits. The audit detects cross-agent association under its conditional exchangeability assumptions. It does not validate every native support label. For seed 43 before D2, the null maximum's 95th percentile is 0.3592, above $\tau_f = 0.35$; 138/999 null maxima

Table 20: Permutation samples. Rows are pairwise aligned observations, not independent patients. For the three-agent cohort, row counts are ordered AKI–RF, AKI–Sepsis, RF–Sepsis. Unique natural-person counts are unavailable to this audit.

Cohort	ICU stays	Strata	Paired rows
Sepsis+AKI	505	22	5668
Sepsis+RF	510	62	6508
AKI+RF	493	125	5921
Sepsis+AKI+RF	495	62	6347, 6941, 6347

Table 21: Conditional stage-wide maximum-statistic diagnostic with $\tau_f = 0.35$. The fifth column counts null maxima reaching the algorithmic threshold; it is not the omnibus p-value in the final column.

Seed	Stage	Observed max	Null 95th pct.	Null max $\geq \tau_f$	$\widehat{p}$
42	D2	0.8064	0.33846	17/999	0.001
42	D3	0.7977	0.24048	0/999	0.001
43	D2	0.8254	0.35918	138/999	0.001
43	D3	0.8069	0.21037	0/999	0.001
44	D2	0.8247	0.34996	49/999	0.001
44	D3	0.7676	0.21739	0/999	0.001

reach 0.35. Of 56 pair–layer direction endpoints exceeding this threshold on the complete-stay sample, 52 also pass the global-null max-statistic comparison at 0.05. These endpoint counts can count one direction in multiple pairs and are not the unique-direction counts in Table 17. The audit metadata provide ICU-stay identifiers, not subject identifiers, so dependence across multiple stays of the same person is unresolved. This is an ICU-stay-block diagnostic, not a completed patient-level test of the stronger design described in the theoretical appendix. Time-grid stratification also does not control calendar-time effects or all shared clinical inputs. Accordingly, correlation is evidence of association rather than proof of redundant clinical knowledge, safe release, or a causal adaptation benefit.

Additional reading of the audit. The audit concerns the support mechanism only and enters no performance table.

- *The threshold sits near the null at D2.* The fraction of null maxima reaching $\tau_f = 0.35$ is $17/999$ (1.7%), $138/999$ (13.8%) and $49/999$ (4.9%) for seeds 42, 43 and 44 before D2, and $0/999$ before D3 in every seed. Under the global null, a nonnegligible share of D2 families would therefore contain at least one comparison passing τ_f, which is why τ_f is not a calibrated significance level. The observed maxima (0.768 to 0.825) lie well above every null 95th percentile (0.210 to 0.359).
- *Resolution of $\widehat{p}$.* With 999 permutations, $\widehat{p} = 0.001$ is the smallest attainable value, a resolution floor and not an estimate of a smaller tail probability.
- *Release by support status.* In Table 18, supported exposures receive 2.4 to 3.4 times the mean release of unsupported ones ($L_{\mathrm{S}}/L_{\mathrm{U}}$), compared with a cap ratio of $0.15/0.0495 \approx 3.03$. Release also depends on eligibility and on the budget, so this ratio does not measure the benefit of support. Supported directions are 12.2% to 18.4% of the 288 entries (Table 17); most directions keep the smaller cap.

E.12 Computational Cost of Spectral Memory and Functional Support

Measurement protocol. We use the existing complete MIMIC-IV profiling runs for seeds 42, 43, and 44, executed serially on an Apple M1 Max with 64 GiB RAM, using CPU execution and one thread for PyTorch intra/inter-op and BLAS/OpenMP. Wall-clock times include profiling instrumentation; they exclude OPE and behavior-policy fitting. All nine profiled stage checkpoints have online weights identical to the corresponding original checkpoints. These measurements are separate profiling runs, not recovered timers from the original performance experiments or a cross-method speed comparison.

Stage-boundary time. Table 22 separates the optimization loop, private-memory rebuilding, shared-latent construction, and functional support. Private-memory time includes activation collection, Gram accumulation, spectral decomposition, prototype refresh, and relation bookkeeping; shared-latent time is excluded from that column. Support is refreshed twice per comorbidity stage rather than at every optimization step: once at the incoming-stage boundary, using the incoming cohorts and the parameters carried over from the previous stage, which sets the scores used throughout that stage (this is the refresh reported in Appendix E.11), and once at the end of the stage, using the final stage parameters, so that the consolidated checkpoint carries current scores. Over D2 and D3 this gives four refreshes per complete run, and the Support column sums the two for each stage. Nested timers are partitioned to avoid counting shared construction twice.

Table 22: Measured wall-clock seconds, mean ± sample standard deviation over three seeds. Support is the sum of the two stage-boundary refreshes. D1 has no support comparisons; its very short shared-memory no-op rounds to zero.

Stage	Training	Private memory	Shared latent	Support
D1	28.144 ± 2.222	2.034 ± 0.105	0.000 ± 0.000	0.000 ± 0.000
D2	139.940 ± 1.021	4.096 ± 1.170	0.325 ± 0.005	0.220 ± 0.006
D3	66.263 ± 1.171	4.284 ± 0.422	0.376 ± 0.007	0.223 ± 0.012
Total	234.346 ± 3.600	10.415 ± 1.503	0.700 ± 0.012	0.443 ± 0.011

The complete entry point, including data loading, setup, and serialization, takes 258.773 ± 5.398 seconds. Private plus shared memory construction takes 11.115 ± 1.510 seconds (4.30% of mean total time), and support takes 0.443 ± 0.011 seconds (0.17%). The narrower private covariance-finalization/decomposition timers sum to 0.114 ± 0.002 seconds; activation passes dominate the memory-building cost. The shared eigendecomposition takes less than one millisecond per D2/D3 rebuild. These percentages describe boundary work only. The per-step cost is visible in the training column: dividing by the reference update counts of Appendix E.2 gives about 1.4 ms per iteration in D1 (19,616 iterations, single-agent teams, no filtering active because no memory exists yet), against about 40 ms in D2 (3,478 iterations) and 61 ms in D3 (1,090 iterations). The increase reflects everything that becomes active once a team has more than one agent: two or three agents' forward and backward passes through the communication paths, the per-layer gradient projection, the allocation of Eq. (5) for every filtered layer at every step, and the profiling instrumentation itself. We did not profile the baselines under the same instrumentation, so we do not report a cross-method per-step comparison; the boundary work is small, but the per-step overhead of filtering and communication in multi-agent stages is not, and should be weighed against the single-agent D1 rate rather than against the 4.30% figure above.

SVD implementation and persistent storage. For an input width d, output width o, retained rank r, and N activation rows, the implementation accumulates the uncentered Gram matrix in float64 and uses a dense symmetric eigendecomposition. Excluding neural forward passes, the main work is $O(Nd^2 + d^3 + odr)$ for Gram accumulation, decomposition, and prototypes, with an additional $O(dr^2)$ term when matching new and old directions. Thus retaining $r = 32$ does not make the dense decomposition cost $O(r^3)$. Persistent per-layer tensor storage scales as $O(d^2 + dr + or + r)$; the Gram collector processes minibatches and does not retain all Nd historical activations. The deployed model tracks seven linear layers per agent (21 in total), whereas support matching uses only three layers per agent. Shared communication uses latent width 64 and rank 16.

Table 23 counts actual tensor backing storage, deduplicated within each category. Private spectral state includes directions, eigenvalues, and prototypes; support-related component metadata reuse the existing memory objects. The byte counts exclude Python objects, optimizer/model state, raw datasets, and allocator overhead.

Support comparison count and scaling. Let $\mathcal{T}_g$ be the agents present in cohort g, L the matching layers, and $n_{gij} \le 8192$ the sampled aligned rows for pair (i, j). After alignment and neural forward passes, the support calculation costs

$$O\left(\sum_{g} \sum_{i<j \in \mathcal{T}_g} \sum_{\ell=1}^{L} n_{gij}\left[d_{i\ell} r_{i\ell} + d_{j\ell} r_{j\ell} + r_{i\ell} r_{j\ell}\right]\right). \tag{34}$$

Table 23: Persistent memory payload. Private sizes are identical at D1–D3 for all three seeds. Shared state is 768 bytes before activation after D1; the table gives its active size. The private total includes the first two rows.

Stored state	Bytes	Size
Private Gram matrices	3,250,440	3.100 MiB
Private spectral state/prototypes	963,712	0.919 MiB
Private total	4,214,152	4.019 MiB
Shared latent (after D2/D3)	37,696	36.812 KiB

The first two terms construct directional responses; the last computes all pairwise correlations. Normalization is linear in the response array size. Metadata alignment and neural forward costs are additional. The implementation collects all matching layers in one forward pass per agent–cohort-pair, then processes layers sequentially; it does not repeat the forward pass for each layer.

In D2, three cohorts each contain one agent pair; in D3, one cohort contains three pairs. Both therefore have three cohort–pair contexts, not nine. With $L = 3$ and $r = 32$, each refresh computes $3 \times 3 \times 32^2 = 9{,}216$ scalar correlations in nine matrices. At $n = 8192$, the correlation products alone require 75,497,472 multiply–accumulate operations per refresh, not just 9,216 operations. The four refreshes produce 36,864 correlation entries per complete run. Response construction, forward passes, and alignment are included in the measured support time above. The separate permutation audit is an evaluation diagnostic and is not part of training cost.

Temporary tensors and peak-memory limitation. The 32×32 float32 correlation matrix occupies only 4 KiB, but it is not the dominant temporary tensor. At $n = 8192$, the two normalized $n \times 32$ response arrays occupy 2 MiB together. For the actual matching-layer input widths $(128, 179, 179)$, cached inputs for both agents across all three layers occupy $2 \times 8192 \times (128 + 179 + 179) \times 4$ bytes, or 30.375 MiB. These are shape-based payload counts, not measured RSS or a peak bound: concatenation, normalization, indexed state copies, metadata joins, and overlapping tensor lifetimes can allocate additional memory.

Shared-latent construction also buffers the float32 latent minibatches before accumulating their statistics. The buffered payload is 54.324 MiB after D2 and 64.512 MiB after D3 (222,512 and 264,242 agent-rows at width 64), before additional workspaces. Consequently, fixed-size persistent spectral state does not imply constant temporary memory for this implementation. The archived profiling runs did not record peak CPU RSS or GPU memory; we do not report tensor payload as peak process memory. Memory construction rereads historical observations: 1,072,484 old agent-row accesses per run across private/shared passes, with no old-row optimizer updates. These are logical accesses, not unique patients or physical disk I/O.