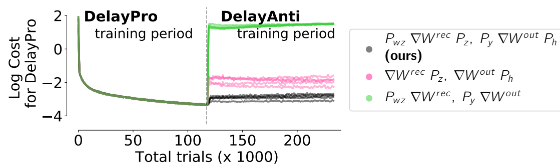
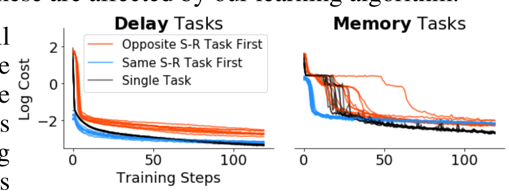
We thank the reviewers for their careful reading, feedback and helpful comments and address specific concerns below.



Novelty of contributions. The novelty of our contribution is two-fold: First, our proposed learning rule with modifications to both sides of the gradient update is novel. This feature also distinguishes our method from OWM (**R3**) and FORCE (**R2**). Following **R3**'s suggestions, we repeat the experiments of Figure 3 using one-sided projections and plot the log-cost on test trials throughout training of the 1st and 2nd task. The increase in cost after introducing the second task provides a comparison of the extent of forgetting across methods. Tasks are best retained using the double-sided approach. **R4** raised that they did not understand the motivation for the double-sided learning rule. In brief, both input and output spaces can interfere across tasks unless we project out updates in these dimensions during learning. We hope the additional results are convincing and we will further expand on the motivation offered in lines 107-111 when we revise the text. Second, a key contribution of this work is the dynamical systems analysis of how our learning algorithm shapes the organization of multiple tasks within an RNN. We find that tasks with similar input/output relationships may utilize shared dynamics in aligned subspaces, even when orthogonality was imposed using our learning rule. For example, when a `Memory` task is learned first, followed by a `Delay` task with the same input/output structure (e.g. both `Pro` tasks), then the `Delay` task dynamics can reuse structures of the `Memory` task without changing the input/output relationships in the `Memory` task subspace. This leads to alignment across both task-subspaces (Figure 6D) and the existence of a trace of a memory structure (ring attractor) during the performance of the `Delay` task (Figure 7A). **R3** felt that we spent little effort discussing training order. In the revised paper, we will make it more explicit that analyses of sections 5.3-5.5 were carried out to better understand training order results (cf lines 164-165). The novel analyses we developed contribute to a better understanding of learned representations in RNNs and how these are affected by our learning algorithm.



Transfer learning and orthogonality. In response to **R4**, we will update Figure 4 to also include the single task training setting to more clearly demonstrate transfer learning (see right). **R4** also raised the question whether orthogonality was imposed in the same way across experiments. We apply the same learning strategy in all cases, according to Eq.(3)-(5). We agree that saying "were allowed" in line 195 is confusing, since the learning algorithm itself is agnostic to any similarities across tasks. **R4** noted that the case where orthogonality was imposed but the network still used the same subspace, would mean the results of the paper are nontrivial and interesting, and we would like to note that this was indeed the case. We will revise relevant sections in the paper to be more explicit about these points.

Related work and comparisons. **R2** pointed out similarities with FORCE, which we refer to in lines 252-256 in the discussion. We plan to revise the related work section to focus more explicitly on FORCE and also include a reference to the relevant work in Beer & Barak (2019), as suggested by **R2**. **R3** asked why we focus only on regularization-based approaches in our comparisons. Our work is motivated by the question of how the *same* neural population may be involved in computations relating to multiple tasks (lines 23-24). EWC and SI are appropriate for this setting and represent well-established baselines. Furthermore, SI represents the state-of-the-art on the task-set and architecture we study here (see Yang, 2019). Replay revisits training data, while we consider the setting where training examples from previously learned tasks are inaccessible (lines 54-55). Dynamic network architectures (e.g. Li, 2020 and Cossu, 2020) solve continual learning by growing the network, which is a fundamentally different solution and the reason why we didn't include such methods as baseline comparisons here. However, we thank **R3** for pointing out the above references and other recent work on continual learning in the RNN setting. We were not aware of these recent publications, but agree that they should be discussed as related work in the revised paper.

Other concerns. **R1,3** noted that our set of tasks were limited. We focus on toy examples so that we may analyze the solutions the network obtains under different training regimes, which we emphasize is a key contribution of our work. We view this as an important first step, but agree that real-world applications (e.g. cart/pole control in multiple settings, or brain machine interface control) should be the ultimate goal. **R2** was interested in fixed point structure change after learning new tasks. Fixed point structures were highly overlapping upon visual inspection in TDR subspaces. All examined subspace angles between fixed point structures before and after learning subsequent tasks were <0.1 radians, and q values (cf Beer & Barak, 2019) remained in the same range. Such minimal change is consistent with our learning rule limiting change of dynamics within previously explored subspaces. We will add a discussion of this point to the revised manuscript. **R2** also raised concerns about biological plausibility and known task boundaries. While the motivation for our learning rule is based on orthogonal subspace structure in neural populations, the learning rule itself is not biologically plausible. In the revised manuscript we will clarify that biologically plausible learning, as well as learning without knowledge of task boundaries are interesting and important directions for future work. Regarding reproducibility, we will make code publicly available should the paper get accepted.