We thank the reviewers for their time, feedback and highly encouraging comments. **It was acknowledged that our algorithm is intuitive and principled (R4), achieves significantly better results (R1, R3), is clearly presented and situated (by all), and is novel and relevant to the community (by all).** We will incorporate suggested improvements to the paper regarding punctuation, notation, algorithm box and typos. We address here the remaining concerns:

**R2: Discrepancy in baselines' numbers, lacking experimental information**: We appreciate the reviewer bringing up this point since it is important for the reader to understand how our comparisons are made, which we will make clearer. Regarding the disparity, the cited works use a higher resource allocation than ours: a large replay buffer of size 5120 [Ref4]/5000 [Ref2 (results from AGEM)] for GEM, while our MNIST [Rotation/Permutation] experiments have a buffer containing only 200 samples. For EWC, the network used is 20 times larger [Ref2], there are only 10 tasks [Ref1, Ref2] (we have 20 tasks) and the setting is not single-pass [Ref1] (they train each task for 20 epochs). It should be noted that the low memory regime is where the performance trends of many CL methods are most pronounced and meaningful. Given unlimited memory/compute, all the methods perform comparably to ER training (as also noted by MER). We will expand the experimental description in the main paper to highlight these details, which we have currently outlined in Appendix F and G due to space constraints.

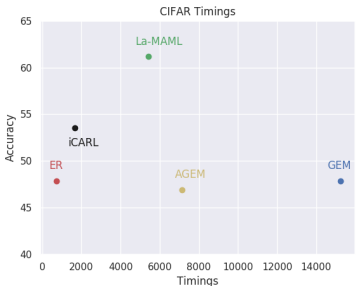


Figure 1: Accuracy vs Timing comparison on CIFAR (time in seconds)

**R2: RHS $\neq$ LHS in Eq. 1**: While we believe the equation is technically correct, we acknowledge that it might be confusing since we have clubbed the implied data arguments into $(X^n, Y^n)$ for brevity. We will separate these for each of the task-specific loss functions in the camera ready for correctness.

**R3: Timing comparison**: This is a good point, and we will include a plot for our *Multi-Pass* experiments on CIFAR, showing the total running time for La-MAML, GEM, AGEM, iCARL and ER, as shown here in Fig.1.

**R3, All: Multi-headed Problem Setting**: We thank R3 for raising this point and make a correction to our experimental description: While our real-world vision experiments are multi-headed, our MNIST experiments are in the single-headed domain-incremental setup (since as mentioned in Section 5.1, the output space for all tasks is the same set of 10 classes while the common transformation to the digits varies with each task). The paper thus contains **both task-aware and task-agnostic** experiments since our algorithm's working is task-agnostic. We omit Class-IL settings since they have many of their challenges arising primarily from the bias imbalance in the classification layer. Many class-IL works specifically focus on this issue since it is tricky to isolate it to study the general forgetting problem in CL (R3 Ref. [7]).

**R3: Choice of backbone**: We have tried to use an architecture (3-4 conv layers + 2 FCs) that is commonly used in meta-learning works for its simplicity [1] [2] [3], and have used it to run all our baselines. In the CL setting with ever-increasing tasks, any model will eventually be under-parameterised. As long as the model performs decently in the *i.i.d* setting and there is a gap between the *i.i.d*-trained model and other CL methods, it should be valid to use it to study the CL problem. If the reviewer recommends, we will add a sensitivity analysis for network sizes to the Appendix.

**R3: Relevance of LLL setting**: We agree that the issue of how setup constraints are commonly chosen in CL works is worthy of debate. However, we reiterate that the LLL setting is challenging yet realistic in many cases where it is not feasible to store all the within-task data points, and is also studied in many prior works like AGEM , MER. It should also be noted that we take multiple gradient steps (*glances*) over each sample in the LLL setup (described in Appendices E,F), thus making enough updates to the parameters.

**R3, R1: Lookahead plot**: We had hoped to show through Fig.3 in the paper, how the performance varies as more tasks are added to the problem (as asked by R1). Note that the average accuracy stays roughly the same across an increasing number of tasks. We shall remove this figure if it is not considered informative by the reviewers.

**R1: RL experiments**: We agree and think it would be particularly interesting to test our algorithm for Model-Based RL, where models learnt online from a temporal data stream should undergo considerable *forgetting*. However, given the ambiguities and lack of benchmarks for properly defining a *continual* setup in RL, we are pursuing it as an extension and it is out of scope for this work.

**R1: Lookahead search**: We added the following: *"In optimisation literature, lookahead search usually evaluates the fitness of proposed parameter updates based on an auxiliary criterion evaluated after hypothetically applying them. These proposals are then modified based on evaluated fitness to make an actual update. In our case, the LRs act as the modifiers of the update, and their values result from the evaluation of two criterions: the losses on old and new tasks".*

[1] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, Sergey Levine. Online Meta-Learning: Proceedings of the 36th International Conference on Machine Learning, PMLR 97:1920-1930, 2019.

[2] Clemens Rosenbaum, Tim Klinger, and Matthew Riemer. Routing networks: Adaptive selection of non-linear functions for multi-task learning. InInternational Conference on Learning Represen-tations, 2018.

[3] Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., and Finn, C. Gradient surgery for multi-task learning.arXiv preprint arXiv:2001.06782, 2020.