R1Q1: *Results didn't normalize .... across different methods.* R1Q2: *Only show params and updates-adjusted results on several envs in supplementary...* **A:** Thanks for the suggestion. We'll re-organize the paper and present all results controlling for parameter count and gradient-steps. We are conducting the rest of these experiments now. From the results, we can see that there is no consistent improvement in performance for the baselines, and MC's good performance cannot be replicated simply by a corresponding increase in params and update steps of the baseline.

R1Q3: *Organize the paper better.* **A:** A great suggestion. We understand your concerns clearly this time. We have moved Sec 3.1 and Line 200 to a re-organized RL preliminaries section. It will be much clearer in the final paper.

R1Q4: *Discussion is unclear.* **A:** This discussion attempts to give a high-level intuition for how MC could benefit the vanilla actor-critic (AC) baseline. MC optimizes learning progress (Eq 5) as measured by a 'validation' set (off-policy). A conjecture about how this could improve conventional AC return maximisation is to increase visits to states with low episodic-return (contrary to vanilla AC), but which are informative for *learning* (Eq 5), and thus longer term return.

R2Q1: *Whether/when/where to use meta-critic?* **A:** Thanks. (1) Most fundamentally, MC is relevant to *off-policy*, *single-task*, *derivative-based* RL. If on-policy or evolutionary learning is desired, or the application is multi-task, then other meta-RL methods are more suitable. (2) MC can potentially be used with any OffP-AC method since results show similar or better performance than several baselines. (3) Our cost is 15-30% compute and 10% parameter count above the baselines (latter is neglectable as small compared to replay buffer) during training, with no overhead at testing-time. (4) Alternative auxiliary losses span from low-cost entropy (already in SAC) to hand-crafted unsupervised reconstruction losses e.g., (Jaderberg ICLR'17, 'Reinforcement learning with unsupervised auxiliary tasks') that are primarily relevant for pixel inputs; and meta-learned (LIRPG [41]) which should impose comparable overhead to ours.

R2Q2: *The meta-loss is myopic. Is it shortsighted? Useful to look ahead more than one gradient update step?* **A:** Thanks for the suggestion. We agree it is myopic, and using more than one gradient step is potentially valuable in principle. However in practice this would require back-propagating through a longer inner loop, which raises several challenges: (1) Additional higher-order gradient calculation, and associated memory use. (2) Risk of vanishing or unstable high-variance gradients. (Challenges are as discussed in other papers (iMAML NeurIPS'19, Taming-MAML ICML'19).) Some other meta-RL studies consider longer episode length such as EPG (Houthooft NeurIPS'18), but use zero-order optimization and on-policy learning. Nevertheless it is significant that we are able improve off-policy learning with online meta-RL, even myopically. Designing an effective longer-horizon extension is left to future work.

R2Q3: *Related work and minor points.* **A:** Thanks. We will address these points.

R3Q1: *Fair comparison... number of gradient steps?* **A:** Thanks. Consider our policy (ϕ) and auxiliary loss (ω) modules. The policy module always takes exactly as many gradient steps and data samples as the baselines. The loss module takes an additional gradient step and sees an additional validation batch from the replay buffer. But this data is not directly accessible to the policy module. So we do not see it as more data or more gradient steps for the policy per-se. The experiment in the supplementary is motivated by controlling for total compute time. We agree that investigating the impact of gradient-steps-per-environment-step is an interesting topic, but this is an orthogonal question getting out of the scope of our work because (i) it is a hyperparameter of interest to study for all the base RL algorithms without meta-learning, (ii) the same hyperparameter can be varied for both the baselines and the policy module in meta-critic.

R3Q2: *Arbitrary meta-critic loss design?* **A:** The meta-critic architecture h_ω is motivated by the need to input at minimum parameters π and states s_i, but π is high dimensional (70k param), making ω easy to overfit if $[\pi, s_i]$ is input. The trick of inputting $\bar{\pi}(s_i)$ means that both inputs are available but low-dimensional (300 param). Our alternatives were partly inspired by the form of $v(s)$ and $q(s, a)$. Including the Q-value itself may suffer value estimation uncertainty (e.g., overestimation). But we conducted this suggested experiment $h_\omega(\bar{\pi}(s_i), q(s_i, a_i), s_{i+1})$ as per Walker-2d/Table 2. The Max Avg. Ret. is 5935.1 ± 648.4 and Sum Avg. Ret. is $52, 098, 042$. So including Q-value performs a bit worse.

R4Q1: *Some figures are misleading.* **A:** Thanks for the careful reading. We can confirm there are no code bugs. For curves like ant-rllab, the concern may be because we smoothed the curve (using window_size=30, following TD3). In addition, there's the STD but overlapped by the orange line. TORCS is not widely included in the benchmark suite of existing algorithms, so they may be less well tuned for it in terms of stability (leading to mid-learning performance drops). However, interestingly we did not have to tune MC to improve performance here. We'll resize smooth window, adjust line color/transparency, and importantly include more seeds in the final paper to make the true variation clearer.

R4Q2: *Connections being made to meta-learning tenuous.* **A:** We presume the reviewer is taking meta-learning to refer specifically to multi-task meta-learning as per MAML, RL2, PEARL, etc. We note that the term meta-learning is also applied in single-task RL (when higher-order meta-gradients are used to train some aspect of the learning algorithm by backprop through inner learning steps). For example [39], [13], (IJCAI'19, '*Meta-gradient Descent For Reinforcement Learning Control*'), (ICML'18, '*Learning To Explore With Meta-Policy Gradient*'). IE: The commonality is the use of meta-gradients, rather than the multi-task setting specifically. We use meta-gradients to train our auxiliary loss online.