Thank you to all reviewers for taking the time to give us feedback on our submission. We hope the following addresses your concerns.

Reviewer 1:

- We would like to clarify, the *greedy* policy (exploitation) is not the one learned by the advisor. That is the task specific policy. The exploration policy (the epsilon part) is the one learned by the advisor, which as shown in Figure 6, is not simply a general policy to solve many tasks.

- We will expand our explanation on when we can expect this method to be useful. Intuitively, it helps when in a given state certain actions are not appropriate regardless of task variations. For example, if the pole in pole balancing is about to fall to the right, moving the cart to the left will never help the situation.

- Thank you for the recommendation on other exploration baselines. We will take them into account.

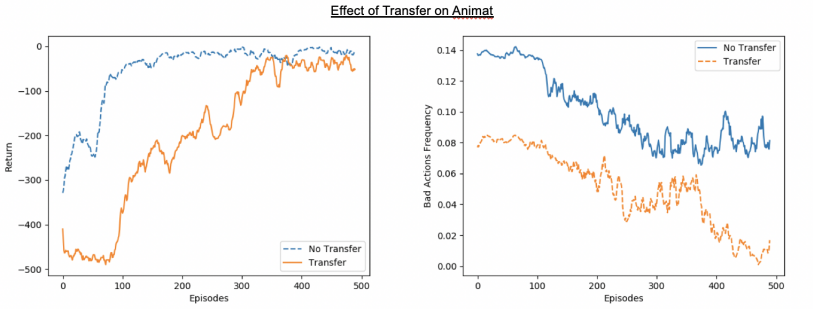
Reviewer 2:

- We also found it surprising how little work there is in this specific area. After going over your suggestion, we believe a discussion on Simsek and Barto's work is appropriate.

- We apologize if there was some confusion and we will clarify in the text. 1) Yes, we do assume that tasks come from a distribution for our theoretical formulation; however, in practice we use a finite set of tasks to solve, a small subset of which is used to learn the exploration policy. 2) No, we do not need to train in parallel. The results hold when training on different MDPs sequentially, but when possible, updating the advisor from multiple tasks in parallel makes learning an exploration policy quicker. Good point! Training on increasingly difficult problems, like in curriculum learning, would be a clear scenario where learning an exploration policy, as we propose, could lead to clear benefits.

- We did perform the experiment you are suggesting to evaluate the learned exploration policy. Figures 5 and table 1, evaluate the performance of the learned exploration policy on a set of novel tasks (different from the ones used for training), showing that it leads to clear performance improvement.

- Your comment on the Animat problem is a fair point. A simple transfer of policies would definitely reduce the number of 'non useful' actions taken. However, it would be highly biased to the task where the policy was learned, which could lead to really poor performance in a new task. We evaluated this point based on your comment. The figure on the right shows the (average) effect of a simple transfer of policy over five task variations. The right plot shows that the frequency with which the agent takes 'non useful' actions is decreased significantly with simple policy transfer; however, left plot shows that a simple transfer of policy can actually make learning more difficult.



- We will gladly add your suggestions to improve presentation. The reason why no training progress is shown for Animat is that the plot would not show anything that was not shown for pole-balancing, and we thought that using the space to analyze the action selection process in Animat would be more relevant.

Reviewer 3:

- We will make sure to clarify on the intuition behind our objective. Assuming an agent improves the expected return of the policy with each episode, maximizing the cumulative return is equivalent to maximizing the AUC. Intuitively, when plotted, the quicker an agent reaches an optimal policy, the larger the AUC will be. Because the only variable that is optimized is the exploration policy, it learns to exploit structures present on the tasks to achieve this.

- The behaviors learned for pole balancing and self-driving task were a bit more intuitive. In pole balancing, the exploration policy learns that if the pole is about to fall to the right, the cart should compensate by moving to the right, and vice versa. Similarly, in the self-driving task, if the car is driving to much to the right the appropriate action should be moving to the left by some amount and vice versa. In general, certain actions can safely be omitted from exploration in certain states. We will make a more detailed discussion regarding this result.

- Figure 5 is showing that, in the first half of the set of tasks (while the advisor has not trained sufficiently), the exploration policy is inefficient. The second half of set of tasks demonstrate that the exploration policy is improving over the initial exploration policy and leads to improving learning over random exploration. We will clarify the points discussed in section 6.1

- Thank you for pointing out REPTILE as an alternative meta learning method.