We would firstly like to thank the reviewers for their time. We understand many of the issues highlighted were largely caused by a lack of clarity. We humbly request for more of your time in re-evaluating scores, given our responses below.

R1: We are pleased to see you like the idea, and motivation. However, it seems the reject decision hinges on many results from a "missing" Appendix **which was indeed included**. In the supplementary materials, (DvD_Supplement.pdf) we included the proof for Lemma 4.1 (l. 584-596), as well as ablation studies for the different components of our method (l. 528-532). Given that the majority of your concerns were fully addressed in the Appendix, we would greatly appreciate it if you could please update your score to reflect this. In particular, Fig 8 shows that, while it is true the adaptive method outperforms the fixed positive diversity, in the latter case we can **still learn good policies**. This is not the case for NSR-ES (due to the cycling issues). The method with $\lambda = 0$ is already included in the main body, it is the vanilla ES. The purpose of Section 3.1 is to introduce the notion of behavioral embeddings which we use to define a similarity measure between policies and is crucial to the definition of our joint objective in equation (8). We agree with your comment (#1) that we should remove the confusing reference to Trust Region Methods in Section 3.1, this will save space for additional clarity elsewhere. Regarding a MAP-Elites comparison, typically when people use this algorithm they have a specific problem in mind so use domain knowledge to form the grid. Thus, we cannot compare against that approach. The matrix inversion is trivial as $\mathbf{K}$ is a $\mathrm{M} \times \mathrm{M}$ matrix ($\mathrm{M} \in \{3, 5\}$).

R2: Thank you for your review, and comment that our paper is novel and well written. Regarding the use of stationary policies in a finite time setting, this is a standard approach in deep RL and beyond the scope of our paper.

R3: We appreciate your positive comments, from reading them, it was a surprise to see such a negative score (reject). In particular, you say "This is a very relevant problem to the RL community... The paper is well-written, formalizes the problem clearly, and the results are encouraging." When looking at the negative comments, it appears the issues are all things we can clarify with an extra page in the camera ready. The first comment claims we do not explain the behaviors for NSR-ES and ES. However, in the paper **we explicitly said this**, see l. 268 "As we see, both vanilla ES and NSR-ES fail to get past the wall (a reward of -800)". Second, regarding the t-test, we used the distribution of maximum rewards obtained per seed. We used a similar approach to multiple other studies. Of course, we can add additional comments that no t-test is perfect, however, it seems to be a minor remark and not one specific to our work. We are surprised it's used as one of three comments to reject our paper. Finally, the computational cost of the gradients. We do agree we should have included this. We ran these experiments on a laptop computer, without a GPU, and it used on average 27% more wall-clock time to train with DvD. We believe this would be reduced on a GPU. We also note that in RL, we often care more about samples, which may be from the real world, and our results are a **new state of the art** in this context. We will absolutely include this discussion in the paper. The number of seeds is included in the caption for each figure. In this paper we focus on deterministic policies, we can include this earlier in the paper. It could be extended to stochastic policies if we used a kernel which measures similarity between distributions.

R4: Thank you for your detailed comments, we very much appreciate them and believe they open up exciting directions for future work. In particular, how do we know how many optimal policies to look for? This is currently a hyperparameter and ideally we would be able to learn the number of optimal solutions for a given problem. We could do this with an adaptive population size, for example, if there are only three possible optimal solutions we do not need a population of ten agents. However, this issue is common across all quality diversity algorithms and we believe it is beyond the scope (and length) of our paper. Next, *why* we wish to find diverse solutions (#1), there have been many different use cases for quality diversity algorithms. For example if the distribution shifts (the classic example from Cully 2015), we may wish to switch "behaviors", which means having access to a set of distinct (yet high performing) policies. In the examples we used, it is simply the case that by adding diversity you are able to explore better (motivated by Conti 2018). Outside of RL, there has been focus on creating diversity in deep ensembles, to improve uncertainty callibration on out of distribution data, an approach such as DvD may also make sense here for future work.

(#2) The inequality in the paper is correct - the argument assumes there is at least one suboptimal policy, not all of them. A well-known theorem in RL states that there is always a deterministic optimal policy. This justifies restricting ourselves to deterministic policies, albeit, with stochastic policies as future work. The reviewer is right, the assumption on Δ may not be always satisfied. Nevertheless it is a reasonable assumption for many classes of problems including sparse reward scenarios. (#4) Justifying the determinant: we feel this has been covered with the example. The determinant ensures no two solutions are the same, rather than just the **average** difference being high. Our theorem directly addresses this. Thank you for highlighting the notation issue, we will fix this. (#5) Regarding hyperparameters, we address many of these in the Appendix. We show the choice of kernel, and means to sample states for the embedding, do not have a huge impact. The key hyperparameter is the number of diverse solutions, which we previously discussed. (#6) See above for discussion of computational cost, we will include this in the paper. We feel this is not so much about DvD vs. TD3, but more that DvD can be used alongside TD3 to boost exploration.