We would like to thank the reviewers for their valuable feedback. We have carefully thought through your comments.

- **Reviewer 1**

  Thank you for your review. We appreciate your point on demonstrating merit in higher dimensional settings. We have run additional synthetic experiments in higher dimensions since our submission that show our proposed algorithms outperform the comparison algorithms or are competitive with the best ones. We agree with your point that our algorithm can be applied in other situations besides the ones we emphasize. We meant for the term "offline" to differentiate our setting from the traditional one where context is decided by the state of nature. In our opinion, any situation in which nature can be bypassed and the context can be explicitly set could be considered offline. We will clarify this in the final paper.

  On whether the context variables are continuous or discrete, the context variables themselves can be either discrete or continuous variables, but we only consider the case where the number of contexts is finite. This is because the current algorithm requires observations to have been made in all contexts. One could make an alteration to the algorithm that leverages posterior mean rather than past observations in order to allow infinite contexts; however, we chose not to talk about that adaptation here.

- **Reviewer 2** Thank you for your review. We appreciate your suggestions for related work, and we will incorporate them into our final work. You suggested comparing to Toscano-Palmerin and Frazier's work in our experiments; however, we have reviewed these works and have noted that they are studying a different problem. Instead of finding the best action for each task, they find the best *single* action averaged over tasks.

  To address your questions about the Gibbs kernel, we picked this kernel because it was easy to choose the lengthscale function such that the lengthscales only changed as a function of the context/task variables. We did this to demonstrate that it is sometimes essential for lengthscales to change based on task (but not necessarily based on action). That being said, you are correct that there are probably other kernels that would allow us to avoid the problem of misjudging difficulty in the tasks.

  Regarding your last remark about previous methods' compatibility with non-stationary kernels, these methods are flexible to the choice of kernel; however, the authors chose to only use stationary kernels.

- **Reviewer 3**

  Thank you for your review. We will work on improving the figures to be more readable in black and white as well as our description of related methods.

- **Reviewer 4**

  Thank you for your review. You had questioned how our setting was different from standard Bayesian optimization and existing work in multi-task Bayesian optimization. What is crucial to note in our setting is that the optimal action may be different for each task. In particular, if one was to apply standard BO techniques on the joint task-action space, the algorithm would only identify a single optimal action. In contrast, our algorithm was designed to seek out the (most likely different) optimal actions corresponding to each task.

  This is also the key distinction between our work and that of Swersky et al. and Perrone et al. Their works aim to find a *single* action that either works well across all tasks or works well in a particular task. Therefore, these algorithms are not applicable here. Moreover, while having methods that trade off between cost and information gain makes sense in their setting, we believe that it does not make sense for our problem because one must adequately explore every task in order to find its corresponding best action.

  In response to your questions under the clarity section: it is true that approximations may be necessary to account for the cubic scaling of the GP; however, we did not require such approximations in our experiments, and such approximation methods are tangential to the goal of this paper. For the reasoning behind the Gibbs kernel, we wanted a locally stationary kernel (only stationary within a fixed task) in order to demonstrate the importance of having varying lengthscales as tasks change. You are correct in that there may be other non-stationary kernels that sufficiently model joint task-action space well, and we will expand on this point in the final paper.