# Paper ID 974 Rebuttal

We sincerely thank all reviewers for their time and thoughtful feedback. We address concerns sequentially.

## Reviewer 1.

**1. Actions being broadcast**: Apologies for the confusion! Indeed, the actions (and rewards) broadcast are different from those taken. For example, if communication occurs every round, then the agent broadcasts a perturbed Gram matrix of *all rewards* up to that instant, (i.e., even at $t = 1$, the rank-1 matrix broadcasted is additionally perturbed as well). The original actions or rewards are not transmitted anywhere. We will be sure to clarify this in the final version.

**2. Lower Bound.** Thanks for the catch! We overlook a detailed comparison of the lower bound in the draft, which is crucial since the comparison is not straightforward, thanks again for catching this. The bound presented in Shariff and Sheffet (2018) is for the case when the arm rewards are separated by a gap $\Delta$ (and hence the $\mathcal{O}(\log T/\varepsilon)$ bound). Our $\varepsilon$-dependent bound in the same case admits a dependence of $\mathcal{O}((\log T/\varepsilon)^{3/2})$, which is an excess of $1/\sqrt{\varepsilon}$. We will definitely address lower bounds in more detail in the full paper, apologies for the confusion!

## Reviewer 2.

Thank you for your review and positive appraisal of the paper! Apologies for the detailed experiment information – essentially all experiments utilize the identical setting of Section 4.4, and should be reproducible from Section 4.4. (up to randomness of the environment). The algorithms have been implemented in Python using NumPy, following the library `contextualbandits` as reference.

## Reviewer 3.

**1. Privacy Angle.** Thank you for the question! We apologize for the confusion. Contextual bandit algorithms are most relevant in recommendation systems, where the context vectors $\boldsymbol{x}_t$ usually refer to a (random) user's description at time $t$, which often includes sensitive information about the user (e.g., in online retail, it will include a vector of websites visited, etc.), and hence this information is desired to be kept private. Moreover, in our setting, several agencies cooperate to solve the problem (in the decentralized setting), which requires privacy mechanisms to be set in place. For example, in medical imaging, a group of hospitals may be interested in training a joint model, however, none wish to share their data as per regulations. Our approach can enable joint learning in this setting.

**Privacy parameters.** Apologies for the unclear exposition! Approximate differential privacy assumes a noise threshold ($\varepsilon$) and a failure probability ($\delta$); both these parameters are fixed during the design of the algorithm. Now, our algorithm builds on the idea of changing regularizers, and has 3 crucial parameters $\rho_{\min}, \rho_{\max}$ and $\kappa$. Proposition 4 in the paper describes how, for any $(\varepsilon, \delta)$ one can obtain the required values of $\rho_{\min}$ and $\rho_{\max}$ (since $\kappa$ only shows up in the regret, and not the algorithm itself). These quantities, in turn, provide a bound on the group regret as per Theorem 1. Simply replacing these quantities in terms of $\varepsilon$ and $\delta$ gives us a regret bound in terms of the privacy parameters themselves, as in Corollary 1.

In a nutshell, in the experiments, we vary $\rho_{\min}$ directly (following the protocol in Shariff and Sheffet (2018)), but we will include a comparison with the privacy parameter $\varepsilon$ directly, as that will definitely improve understanding of the algorithm. Thank you for this question! we will also include a brief comment on the privacy parameters to clarify the setting.

## Reviewer 4.

Thank you for your positive appraisal and catching the typo!