We thank the reviewers for their insightful comments.

[R4] Knowing $r_{\min}$ and $r_{\max}$: Jun et al. (2018) [19] relaxes the assumption of knowing $[r_{\min}, r_{\max}]$ when studying rewards attacks *only* for ε-greedy and UCB while we provide an efficient attack for any algorithms in the linear contextual case. We could also build an attack on LinUCB only without knowing $[r_{\min}, r_{\max}]$ using $\tilde{r}^2$.

[R2, R4] Bounded Rewards: We need the assumption that rewards are bounded in $(0, 1)$ to prove a formal bound on the total cost of the attacks for *any* no-regret bandit algorithm, otherwise we need more information about the attacked algorithm. In practice, the second attack on the rewards, $\tilde{r}^2$, can be used in the case of unbounded rewards for *any* algorithms. We could not prove a bound on the total cost of the attacks with $\tilde{r}^2$ because the reward process becomes non-stationary under this attack. Thus, there is no guarantee that an algorithm like LinUCB will pull a target arm as the proof relies on the environment observed by the bandit algorithm being stationary.

The assumption of bounded reward is not present in earlier works but the proofs provided by them do not address the issue of non-stationarity in the attacked reward process. We discussed it with the authors of Liu & Shroff (2019) [27], who also study attacks on rewards independent of the bandit algorithms. We were not able to find a solution to this problem for either of our theorem or theirs.

To sum up, it is possible to construct an attack which does not assume bounded rewards but this comes at the price of a formal proof of the total cost for the attacker or knowing the bandit algorithm. We observe empirically that the total cost of attack is sublinear when using $\tilde{r}^2$.

[R1] Fundamental differences between attacking the rewards and the contexts: Perturbing the contexts is fundamentally different from perturbing the rewards for the following reasons:

- The attacker only modifies the context that is *shown* to the MAB algorithm. The true context, which is used to compute the reward, remains unchanged. In other words, the attacker cannot modify the reward observed by the MAB algorithm. Instead, the attack algorithm described in Sec. 4 fools the MAB algorithm by making the rewards *appear* small relative to the contexts and requires more assumptions on the MAB algorithm.
- For the attack on rewards, the attacker modifies the reward after the MAB algorithm has chosen an arm. When the context is attacked, the MAB algorithm will choose an arm based on the attacked context. Therefore the attack may change the arm pulled at the time-step of the attack. It allows for offline attacks as studied in Sec. 5, which are not doable in the case of attacks on rewards. Studying online attacks on contexts requires to have some control over the arm that is pulled and makes this problem more complex. We needed to show that the attack algorithm described in Sec. 4 does not change the arm pulled by the bandit algorithm.

[R1] Difference with Jun et al. (2018) and attacks on contexts: Our work is indeed inspired by the work of Jun et al. (2018). The main difference is that we extend the setting of adversarial attacks in bandits to the contextual case which is in general a much more complex setting than the classical MAB setting. In addition, we also:

- show that adversarial algorithms such as EXP4 can be fooled by our attacks on rewards with a sublinear total attack cost (see App. D).
- consider the setting of attacks on contexts, which is fundamentally different as discussed above.
- introduce a more realistic attack setting where the attacker has very limited power and can only modify the context associated to *a single user*, see Sec. 5.

[R4] Norm of contexts: Clipping the norm of the attacked contexts is not beneficial for the attacker. It means that the attack for a specific context was violating the assumption used by the bandit algorithm that contexts are bounded by L. Prop. 2 is here to provide a theoretical grounding for the proposed attack, but in practice the contexts are bounded by L. We failed to convey this clearly in the paper and will correct this in the revised version. We show experimentally that, when the unattacked contexts are bounded by L and not just by $\nu L/2$, the attack algorithm enjoys a logarithmic total cost of attack and fools the bandit algorithm into pulling an arm from the target set.

[R2] Attacks on multiple users: We did not consider attacking multiple users in the setting of Sec 5. It would have interesting applications as an attacker could infect multiple users. In that case, the theoretical study of the feasibility condition would be more complex as the attacks could significantly modify the behaviour of the MAB algorithm.

[R1] Robust Algorithms: We thank the reviewer for the references around this idea. Prop. 1 shows that under Assumption 1 it is not possible to build an algorithm robust to a logarithmic perturbation of the rewards. Also, building such algorithms for attacks on context, be it a unique one or all contexts, is not something we have investigated yet.

[R1] Presentation: We thank the reviewers for their advice on how to improve the presentation of the paper. The issues mentioned will be rectified in the revised version of the paper.