We thank the reviewers for their detailed comments. Few questions require a more detailed analysis (lower bounds). We respond to all the comments, but postpone some further analysis to a longer version of the paper that is in preparation.

**Reviewer 1:** **1. Comparison with Emamjomeh-Zadeh and Kempe [EK18].** A table may confuse the reader since their setup differs from ours, but we will include the following discussion. [EK18] considers noise in the comparisons rather than in the similarities, and studies triplets rather than quadruplets. They show that: (i) $O(N \ln N)$ active comparisons are sufficient (we match this in Theorem 2), and (ii) $\Omega(N^3)$ passive triplets are necessary (for quadruplets, their proof leads to $\Omega(N^4)$ passive quadruplets are necessary; we obtain better guarantees, but under a planted model).

**2. Improving $N_0 = \Omega(\sqrt{N})$ condition.** Note that $N_0 = \Omega(\sqrt{N})$ implies there are $O(\sqrt{N})$ pure clusters. In the standard stochastic block model literature, there are no known poly-time algorithm that can exactly recover $\omega(\sqrt{N})$ planted clusters (see Figure 1 of Chen & Xu, arXiv:1402.1267). Hence, this condition could be optimal for exact recovery under a planted model in the sense that it is necessary for any poly-time algorithm.

**3. Can larger $\mathcal{R}$ reduce the comparison complexity?** The analysis is possible, but the improvement would be at most $\ln N$ since the active comparison complexity for exact recovery is $\Omega(N)$ to have at least one comparison per item.

**4. Does $K$ vary at different levels?** Yes, after every merge the number of clusters $K$ is reduced by one.

**5. Interpretation of Theorem 4.** (i) The condition $m = \Omega(\ln N)$ should have been $\omega(\ln N)$, which will be corrected. (ii) $m = \Omega(N)$ can be obtained through flat clustering using kernel (3). Large initial clusters are also needed for recovering a planted hierarchy using average linkage with known similarities [Cohen-Addad et al. 2018; Theorem 5.8].

**6. How is initial cluster different from pure cluster in 4–AL–I3?** The term "pure clusters" is probably confusing. The clusters at the bottom of the planted hierarchy are "pure clusters" of size $N_0 = 30$. Then, following Theorem 4, 4–AL has to be initialized with small "pure clusters" of size $m$ (pure means that they are sub-clusters of one of the bottom clusters in the planted hierarchy). For 4–AL–I3, we set $m = 3$. For 4–AL, we set $m = 1$.

**7. Inherent noise in quadruplet queries.** To deal with noise in the quadruplets (that is quadruplets that are flipped compared to an omnipotent oracle), one may assume that each query is independently incorrect with some probability, as in crowdsourcing. Then, it is possible to show that the proposed algorithms can recover the hierarchy under similar sufficient conditions. We will address this issue in the longer version of the paper. In the current submission, we focus on the problem of noise in the similarities as it brings more novelty to the field ([EK18] studies flipped comparsions).

**Reviewer 3:** **1. Theorem for average linkage similar to Theorem 1.** This comment is not clear to us. If it is about necessary $\frac{\delta}{\sigma}$, see our answer to Comment-1 of Reviewer-4. If it asks for lower bound, we respond to your Comment-3.

**2. Theoretical guarantees for Dasgupta's score.** Under a planted model, exact recovery corresponds to achieving a $(1 + o(1))$-approximation of the optimal Dasgupta's score. Obtaining a worst-case guarantee for arbitrary data would be more difficult in our comparison setting. Indeed, Dasgupta's score, and the associated theoretical results, all heavily rely on the fact that true values of the similarities can be accessed. Following this, an interesting future research direction would be to derive an ordinal variant of Dasgupta's score based on comparisons.

**3. Can lower bounds on the number of queries be given?** Yes, but we do not have a complete picture yet. We only provide insights here. In the active setting, it has to be at least $\Omega(N)$ to observe at least one comparison per item. We nearly match this bound with the $O(N \ln N)$ upper bound in Theorem 2 (in a special case). In the passive setting, there are possibly two cases depending on whether the SNR is constant or grows with $N$, which we will explore in the future.

**Reviewer 4:** **1. Necessary conditions for $\frac{\delta}{\sigma}$.** The necessity of $\frac{\delta}{\sigma} = \Omega(\sqrt{\ln N})$ is established in Theorem 1 to show that single linkage only recovers the hierarchy under a very restrictive scenario where the SNR grows with $N$. This does not happen in the subsequent theorems, which all include the special case of $\frac{\delta}{\sigma} = O(1)$. We feel this is reasonable, and hence, state the sufficient conditions in terms of $N_0$. Assuming $\frac{\delta}{\sigma} = O(1)$, the sufficient condition $N_0 = \Omega(\sqrt{N})$, stated in Theorems 2-3, is also necessary (see our answer to Comment-2 of Reviewer-1).

**2. Why average linkage performs better? Why not other linkages?** Average linkage is generally better since averaging tends to reduce the noise. For 4K–AL, there are two levels of averaging – the kernels in (1-3) are sums, and we use average linkage. Similarly, for 4–AL, the averaging in (4) is crucial to counteract the noise. In this paper, we focus on the most popular linkage methods but it would also be interesting to further study other linkages. In particular, the kernels in Equations (1-3) can be combined with similarity-based linkages. Developing a median linkage variant of 4–AL would, however, require a more extensive study.

**3. Is each comparison sampled multiple times by the algorithm?** No, in the current theoretical study, we assume that each comparison is observed exactly once (in both the active and the passive case). In the experiments, we only query the comparisons once in the active case. In the passive case (in particular on the real dataset), it might happen that the same comparison is given several times. In this case, we use a majority vote (that is, use the more frequent answer).