We thank the reviewers for their careful reading and thoughtful feedback regarding our manuscript. Many of these comments will be incorporated in our latest revision. We appreciate the opportunity to clarify some points and address some of the reviewers' concerns, which substantially consist of requests for more extensive *background comparisons* (in text) and *baseline comparisons* (in tests). We will make several changes on revision to address these comments, including adding discussion of more relevant citations and clarifying that we are already performing the appropriate baseline comparisons asked for by the reviewers (which we see was unclear in the writing of our first draft). Most importantly, we will clarify the specific problem space we are targeting as being both 1) completely unsupervised, and 2) requiring non-identity transformations, as is assumed by many other papers in the literature. This clarification will make apparent that we are comparing against all appropriate baselines. We will summarize our contributions as follows and planned revisions to address reviewer comments in the context of these contributions. Please note that we draw our references from R1's citations [1-10] below.

Contribution 1: our paper is the first to combine OT Procrustes [1-4] with hierarchical OT [5-10]. Specifically, we perform alignment in an **unsupervised setting where cluster pairing/ordering is not required**, which is a substantially more challenging problem than that addressed by group-based methods such as Courty et al. and [10] (where group labels *are* required). Since our proposed algorithm (e.g., HiWA-SSC as described in Fig. 1e-f) is a *completely* unsupervised method, using R1's suggested group-based *semi-supervised* methods as a baseline comparison is inappropriate. We see how this confusion has arisen from our manuscript and our revision will include a more detailed discussion of these comparisons to substantially clarify the challenging problem space that we uniquely address.

Contribution 2: novel distributed ADMM numerical algorithm for solving the OT Procrustes and hierarchical OT problems jointly. Although Algorithm 1 has certain elements of [1-4] (as correctly noted by R1), we make a substantial advance over [1-4] because our work also jointly solves the hierarchical OT problem [5-10] in Eq. (5) (which easily converges to a local minima with naïve approaches). Our proposal of using distributed ADMM to solve the joint problem both effectively finds solutions and is computationally efficient (discussed below). Indeed, [10] also uses ADMM, but in an entirely different way: within each conditional gradients iteration, ADMM is employed to find the correspondence matrix. Unfortunately, the formulation in [10] does not admit a distributed approach. In contrast, our primary approach is ADMM, where splitting (distribution) occurs across all cluster pairs: within each ADMM iteration, we use alternating minimization to find the correspondences (letting us exploit Sinkhorn, which is efficient and fast). It is important to point out that our distributed optimization approach represents a *novel* way of numerically tackling problems in hierarchical OT settings. We will clarify this distinction in the revised document.

Contribution 3: a novel analysis framework. We provide a first analysis (specific to our formulation) of the dataset conditions required to solve cluster-based alignment, in addition to providing perturbation and failure mode analyses.

Ablation studies. R1 and R4 point to our lack of a baseline comparison against OT Procrustes [1-4] types of methods and recommend an ablation study to test the utility of the hierarchical component of our algorithm. In fact, we **are performing exactly this comparison**, but we see how our imprecise description of the "Wasserstein Alignment (WA)" method (in Fig. 1e-f) has led to this misunderstanding. In revision, we will clarify that WA indeed solves Eq. (4) (similar to methods proposed in [1-4]), serving as an ablation study that jointy finds transformation and correspondences without any cluster structure. We appreciate R1's suggestion of an ablation using just the hierarchical component (with an identity transformation), but we believe this has arisen from a lack of clarity our description of the problem space we target. While identity transformations have been used in cases where target and source domains are already similar (e.g., USPS and MNIST digits of [9, §5.2]), the literature has clearly identified that it is generally required to find non-identity transformations in many cases of interest (e.g., [1-4], especially the discussion in [3]). In the revised manuscript, we will substantially clarify our focus on the setting where invariant transformations are necessary, therefore making a comparison with hierarchical OT (with identity transformations) vacuous.

Speed/complexity results. Although we state the runtime (per-iteration) complexity at the end of section 3, no formal derivation was given due to space limitations. Following the suggestion of R1, we will, in the supplementary material of the revision, (i) compare the runtime of our algorithm with and without parallelism, and (ii) give our derivation of the runtime complexity.

Data generation. A misunderstanding in the data generation procedure has arisen due to our unclear explanation. We will clarify that we are using exactly the data generation procedure described by R4.

Error in equation. We are grateful to R3 for pointing out a typo in our augmented Lagrangian. We were indeed missing an additional $-\frac{\mu}{2D}\|\Lambda_{ij}\|_F^2$ term – expanding our augmented Lagrangian with this additional term would result in a similar form as the one suggested by R3, with only a scaling difference. In our revision, we will express the augmented Lagrangian in the classical form as suggested by R3, with the presence of a $\frac{\mu}{D}$ scaling on the Lagrange multiplier so that Algorithm 1 can remain unchanged.

ADMM convergence. We will revise to reflect R3's note that [38] (*our* citation) may not be immediately applicable.