We thank the reviewers for their many helpful suggestions to improve the presentation. We first give general responses to common issues raised. We hope our clarifications address concerns regarding the paper, and elevate your view of it.

1. Designing a parallel first-order $\widetilde{O}(\epsilon^{-1})$ algorithm has been a well-studied open problem in computational OT and our major contribution is its resolution. This problem has persisted despite extensive work on first-order methods and varied reductions and though we lean on this literature, our specific objective $c^\top x + 2\|c\|_\infty \|Ax - b\|_1$ and method used are new to OT and crucial to our improvement. A minor contribution is an analysis of area convexity closer to dual extrapolation, and proving the complexity of a prox step, previously used without proof (3.7 [She17], which does not follow from analysis in [Beck15]).

2. An objection to recent reduction-based algorithms with ϵ^{-1} rates was their impracticality; we would like to emphasize our experiments' goal was to give preliminary evidence that our performance extends to real data. We hope this addresses concerns that experiments did not show a practical outperformance of Sinkhorn, which we did not mean to claim.

3. Prior empirical work notes Sinkhorn outperforms its current worst-case bounds on real data [Cut13]. Common real instances may have more structure bypassing the worst-case and allowing improved runtimes; we consider theoretical guarantees with no additional assumptions. Regarding algorithms with better theoretical performance than Sinkhorn in some regimes, we found our algorithm with tuned parameters outperformed APDAMD, the state of the art.

4. Using MNIST for experiments was for consistency with recent literature on computational OT. We acknowledge additional structure of images yield further speedups, e.g. FFT, and modifications which do not improve theoretical bounds may help in practice, e.g. coordinate methods ("Greenkhorn"), adaptivity. We focus on resolving the outstanding issue of ϵ^{-1} dependence; we believe these optimizations merit interesting follow-up work, but it is outside our focus.

5. We thank the reviewers for their many suggestions regarding presentation order (introducing algorithm and regularizer earlier, more motivation, necessity of App. C), agree they help readability, and will implement these in the next version.

Reviewer 2 *Alg doesn't scale well?* Our implementation at submission time was designed only to compare iterations and was inefficient. Since submission we improved the implementation and it now scales comparably with Sinkhorn at least up to the full image dimensions 784×784. We will include experiments at this scale in the final version.

Change "improved" to "competitive" in abstract. We agree with this assessment and will implement this change.

Two tunables? In theory, the entropy constant is not an additional tunable, as the smallest satisfying B.3. It may not be tight; we acknowledge the current analysis does not predict performance with a smaller constant. In practice, we believe this should not be considered a second hyperparameter; we found a smaller constant, e.g. 3, sufficed for convergence.

Previous runtime scale invariance. We reported runtimes as claimed in prior work (e.g. Thms. 4.7, 4.9 [LHJ19]). We investigated and believe the true dependence on $\|C\|_{\max}$ is as suggested by the reviewer, and will edit accordingly.

Bit-complexity? It is $O(\log(n))$ and we will add a discussion: assume $\epsilon = n^{-O(1)}$ (else IPM suffices), and maintain variable x proportional to $\exp(v)$ for v with $O(\log(n))$ range without affecting correctness, by slightly modifying B.1.

Kernel-approx [ABRW18]? Our method is based on matrix-vector multiplies; thus we hope low-rank approximations to the costs when more structure exists (kernel matrices arising from lower dimensions) can speed up iterations. We agree the application is not immediately clear and will modify the language appropriately.

L227,L248 inequalities wrong? They are correct: $\Theta > r(u) - r(\bar{z})$ implies the divergence bound. We will clarify this.

Reviewer 3 *Lemma 2.3 does not show (3) has the same optimal value?* To the best of our knowledge, 2.3 is correct as stated. A similar penalized regression objective was used in the maximum flow literature, and the ideas behind correctness are not new (2.2 [She13]). Such a penalized objective is new to OT which has typically considered an ϵ-regularized objective, whose value is not equal; our objective is not such a regularization. The proof: for any argument to the modified objective, the value does not increase by rounding (to $A\tilde{x} = b$), so there exists optimal $\tilde{x}$ with $A\tilde{x} = b$. OPT lower bounds the modified objective at $\tilde{x}$ by definition, and the minimizing argument of the OT objective achieves it. We note x in line 154 should be $\tilde{x}$, and we do not state that the OT plan achieves OPT; we will clarify these points.

Reviewer 4 *Comparisons on more pictures? Average several trials?* We agree with the suggestion. We experimented on other digits: across multiple trials we observed similar performance, and will include additional experiments.

Is the full objective ℓ_1-reg + regularizer? Clarify constants in regularization, motivate regularizer. The objective we optimize is only the ℓ_1-regularized portion; the regularizer is used to define algorithm steps. Re: constants: κ is the largest which satisfies area convexity (3), and the entropy constant (10) is the smallest which makes the quadratic form PSD in B.3. The regularizer $x^\top A(y^2)$ captures a "local version" of the smallest-width ℓ_∞-strongly-convex regularizer, a quadratic (A.1 of [ST18]). Since we want to decrease regularizer size and $x \in \Delta^m$, this is a small regularizer which captures enough local behavior of a quadratic allowing for area-convexity. We will add this discussion.