We thank the reviewers for the positive and insightful feedback. All comments will be addressed in the manuscript.

R1+ R2 *EI vs TS, why large batches?*: The computational cost of TS scales only *linearly* with the batch size, contrasting batch EI. Moreover, TS maintains effectiveness even for large batches (see Sect. 3.7). Large batches are important for large sample budgets and, in our experience, TS scales better than batch EI. Note that TS is natural for TuRBO since we need to solve a bandit problem over the TRs. We will emphasize these points.

R1+ R2 *Previous work on local BO*: Thank you for the pointers to a variety of interesting papers! We will revise the section on related work accordingly. We want to point out that the paper on Global Optimization with Sparse and Local Gaussian Process Models is particularly relevant, but will struggle to compete with TuRBO in high-dimensional spaces. We will mention BADS in the related work section as they also consider large evaluation budgets. However, note that several of the related papers do not consider the large-scale high-dimensional setting.

R1 *Distribution [of optimum] known and tractable?*: This distribution is not tractable, but we can sample efficiently.

R1 *Which of TR and TS are (most) responsible for improvement over the state of the art?*: In §3 we contrast TuRBO-1 with GP+TS that uses a global GP model. That TuRBO-1 consistently outperforms GP+TS indicates that the local modeling is most responsible for the large improvement.

R1 *Why not compare to Regis and Shoemaker?*: Our comprehensive selection of baseline methods includes COBYLA, which is arguably one of the best well-known derivative-free TR methods.

R1 *What does "outputscales" mean?* We meant “signal variance” and have revised the manuscript.

R1 *Definition and evolution of TR, curse of dimensionality:* Note that TuRBO does not attempt to fill the space with smaller hypercubes. Instead, it runs several local searches simultaneously and allocates samples between them in a principled way. The domain is scaled to $[0, 1]^d$, so an initial TR with side length 1 covers the whole space.

R1 *Lmin* = $(1/2)^6$*: [what is] the underlying rationale?*: Halving the size of the trust region is standard, see, e.g., Andréasson et al. An introduction to continuous optimization. The choice $(1/2)^6$ corresponds to a side length that is slightly larger than 1% of the original length.

R2 *Why compare to COBYLA instead of BOBYQA?*: Thank you for the suggestion! We ran BOBYQA (using `nlopt`) on robot pushing (mean 9.22) and rover (mean 1.40) and it indeed performs better than COBYLA. Thus, we will replace COBYLA by BOBYQA in all experiments. Note that TuRBO still outperforms both by a large margin.

R2 *The results of EBO for the robot experiments are quite different from the results on the original paper*: The results are consistent and we used the code from the authors. The plots in the EBO paper show standard deviation instead of standard error and median instead of mean. They also use a larger number of initial random points.

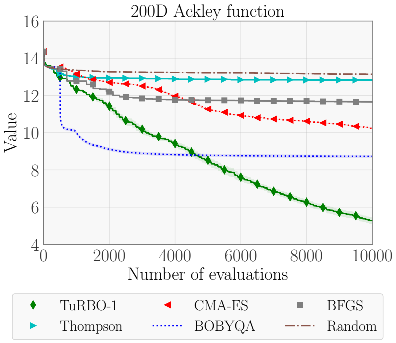
R2 *Previous work on non-stationarity in Bayesian optimization*: Thank you for the references, we will refer to them in the revision. Note that these works do not consider our large-scale high-dimensional setting.

R2 *It would be interesting to see the results of a standard GP+EI, maybe limiting the results to few hundred evals*: We ran GP+EI with botorch for 500 evals: the final performance is (mean 5.4) for robot pushing and (mean -6.0) on rover. This is competitive with, e.g., COBYLA, but far from the optimum. We will report the results for GP+EI to the text.

R2 *TuRBO gets a different number of initial samples*: The plots show the performance as a function of the total number of evaluations, which includes the initial data. Note that TuRBO needs initial points for each TR to build each local GP model. We tested a different number of initial point for the baselines to maximize their performance.

R2 *The bandit equation is purely based on the function sample and not the information/uncertainty on that region*: TS trades off exploration and exploitation effectively. To see this, note that the “function sample” can vary drastically in unexplored areas and thus will take its optimum there with a good probability.

R3 *Benchmarks are not high-dimensional enough*: We ran TuRBO-1 and a few baselines on the 200-dimensional Ackley function (10 runs, batch 100, domain $[-5, 10]^{200}$) to illustrate that TuRBO performs well for high-dimensional problems and will add this result to the manuscript. Apart from rover, we were not aware of any non-synthetic problems in the related work with more than 50 dimensions. The 70-dimensional robot tuning problem in “Local Bayesian Optimization of Motor Skills”, pointed out by R2, is worth exploring in future work.



R3 *Can [TuRBO] be viewed as regular BO?*: We agree that this is a very interesting question for future work.