We thank reviewers for appreciating the originality of our work and providing constructive feedback. All typos and grammatical mistakes will be corrected in the final version. We address specific concerns below.

Review 1: 1. Yes, no algorithm can be minimax optimal for all α without additional assumptions. **2.** Minimax optimality could still be achieved by Alg. 3 with $\mu_\star$ being mis-specified up to error $O(1/\sqrt{T})$; and similar empirical performance is obtained under mis-specification. **3.** Maintaining $O(\log T)$ subroutines hurts the empirical performance of Alg. 3. Designing an empirically superior algorithm that uses knowledge of $\mu_\star$ remains an open question.

Review 2: 1. For any chosen hyper-parameter $\beta \in [1/2, 1]$, Alg. 1 is Pareto optimal, and no algorithm can be strictly better in terms of adaptivity. **2.** Our setting can be generalized to case with n being infinite with a bit care (thanks for pointing this out): in the infinite arm setting, m is infinite as well (what matters is the ratio n/m: one can consider embedding arms into $[0, 1]$ with the set of best arms having positive measure, but *without* additional structure assumptions). **3.** The algorithm is *valid* in the sense that it selects an arm A_t for any $t \in [T]$, i.e., it does not terminate before time T.

Review 3: 1. If n is infinite, then there are indeed cases where the maximum is undefined. To avoid the potential problem of empty $S_\star$, we advocate defining $S_\star$ in terms of ϵ-good arms. **2.** Although T was incorporated in lower bounds, to the best of our knowledge, we are the first to incorporate T into the cumulative regret minimization problem $\mathcal{R}(n, m, T)$, and quantify corresponding hardness level α. Previous work [4, 30] developed hardness parameters in terms of the reservoir distribution of arms (T not included; they additionally require those parameters to be *known*) and thus cannot be directly related to α. **3.** We defined ψ as in Section 2 to avoid the trivial case with all best arms: any algorithm is optimal in such case and achieves 0 regret. **4.** Random selection in Alg. 1 means sampling *uniformly at random without replacement*. **5.** The virtual arm could be mathematically defined as $\widetilde{\nu}_i = \sum_{j=1}^{n} \widehat{p}_i(j) \cdot \nu_j$, where $\widehat{p}_i(j)$ denotes the j-th element of the empirical sampling frequency $\widehat{p}_i$. **6.** The intuition behind Thm. 2 in explained in the paragraph above it. But to interpret Thm. 2 alone: for any algorithm considered, if B (or more precisely $\sup_{\omega \in \mathcal{H}_T(\alpha')} R_T$) is large, we directly know that the algorithm is not optimal on the easy problem within $\mathcal{H}_T(\alpha')$; if B is small, the RHS of Eq.(2) is large and then the algorithm cannot be optimal on the hard problem within $\mathcal{H}_T(\alpha)$. **7.** Whether sharper lower bounds are possible when given extra information about the optimal mean value is an open problem, as is the question of the minimal additional assumption/information needed to fully adapt to α. All we know is no algorithm is simultaneously minimax optimal for all values of α without additional assumptions, and that given the optimal mean value it is possible to be more adaptive to α than without it. **8.** We use ΔT_i to represent the length of the i-th iteration *within* the total horizon T, and it really should have been defined as $\Delta T_i = \min\{2^{p+i}, T\}$ so that $\Delta T_i \leq T$ always holds. **9.** We assume $T \geq 2$ on line 139. **10.** There is no missing factor of 2 in Eq.(28) and Eq.(26) is correct since we focus on the $(1/4)-$sub-Gaussian case. **11.** The factor of 2^{-5} in the definition of Δ on line 544 is to make sure $\sqrt{2\Delta B/K} \leq 1/4$ in Eq.(31) so that we can lower bound the averaged regret. **12.** Note that β in the proof of Thm. 3 is really just a symbol, and one could replace β with $\theta(0)$. Another way to understand the proof of Thm. 3 is as following: for *any* Pareto optimal rate θ, it satisfies the lower bound in Eq.(35); meanwhile, the rate on the RHS of Eq.(35) is achieved by Alg. 1 with input $\beta = \theta(0)$. Alg. 1 is thus Pareto optimal.

Review 4: 1. Although Alg. 1 uses `MOSS` (explained in detail in [2, 14]) as a subroutine, it is *very* different from simply fine-tuning `MOSS` , which fails arbitrarily when n is large, or applying `MOSS` on a subset, which will not lead to Pareto optimal algorithms, as discussed in Remark 2. The innovative core of Alg. 1 lies in summarizing information obtained in iteration i as a virtual arm $\widetilde{\nu}_i$. **2.** The setting with $(1/4)-$sub-Gaussian is only for convenience in calculations and could be generalized to the σ^2-sub-Gaussian case, for any σ. **3.** Eq. after line 115 defines the hardness level of a given problem, and Eq. after line 120 classifies problems in terms of their hardness levels. **4.** Alg. 1 is different from the Distilled Sensing by Haupt et al 2009 since the latter only applies to very special sparse settings where optimal arms are those with non-zero means and all other arms have zero means. **5.** Our algorithms achieves the state-of-the-art performance in adapting to *unknown* α. **6.** `RestartingEmp` (on line 256-259) represents the empirical version of Alg. 1 by allowing the reuse of statistics. Note that we are also comparing to an algorithm, i.e., QRM2, that allows the reuse of statistics [12].

Review 5: 1. Our setting could be generalized to the case with multiple ϵ-good arms without modification in algorithms and (as long as $\epsilon \leq 1/\sqrt{T}$) the theoretical results hold up to negligible factors (see line 98-103; $\epsilon \leq 1/\sqrt{T} \Rightarrow \epsilon T \leq \sqrt{T}$). **2.** The lower bound in Section 2 is in the minimax sense, so it suffices to reduce to the single-best arm case. A lower bound of the order $\Omega(\sqrt{T(n-m)/m})$ ($\approx \Omega(T^{(1+\alpha)/2})$ as long as $T^\alpha \geq 2$) for the m-best arms case could be obtained following similar analysis in Chapter 15 of [21]. **3.** Our results in Thm. 3 show that, in the minimax sense over $\mathcal{H}_T(\alpha)$, suffering a rate of 1 over a certain range of α is *unavoidable* for algorithms on the Pareto frontier. Better bounds might be obtained when restricting ourselves on a subset of $\mathcal{H}_T(\alpha)$, but not in general. **4.** When prior knowledge on α is unavailable, we recommend setting $\beta = 0.5$ and applying `RestartingEmp` in practice since it achieves performance very close to the oracle algorithm with *known* hardness level. Increasing β provides worse performance on small α but better performance on larger α. **5.** The setting with knowledge of the value $\mu_\star$ was previously studied in [23]. Besides, we allow mis-specification in $\mu_\star$ (see point 2 in response to Reviewer 1). **6.** Similar experimental results are obtained after averaging over 500 trials. $T = 50000$ is intentionally chosen to create the tension between n, m and T. **7.** The algorithm is *valid* in the sense that it selects an arm A_t for any $t \in [T]$, i.e., it does not terminate before time T.