Dear Reviewer #1:

> Is it possible to achieve similar results without continuous exponential weight when the number of actions is finite?

Currently, we have no idea for bypassing continuous exponential weight. As mentioned around Lines 84–88 of the manuscript, any existing algorithm for finite action sets that does not rely on continuous exponential weight mixes p_t with another distribution, which hinders improved first- or second-order regret bounds. As you commented, however, bypassing continuous exponential weight would improve practical computational efficiency, which we consider as an important future work.

> 1. Is the covariance matrix of the truncated distribution $S(\tilde{p}_t)$ always invertible?

Yes, $S(\tilde{p}_t)$ is invertible. This follows from the assumption that $\mathcal{A}$ is not contained in any proper linear subspace, which is stated at Line 258 of the manuscript. Indeed, under this assumption, $\mathcal{A}'$ is a full-dimensional convex set with a positive Lebesgue measure. Combining this and Lemma 1, we can see that the domain of $\tilde{p}_t$ is full-dimensional as well. Therefore, the distribution $\tilde{p}_t$ has a density function taking positive values over a full-dimensional convex set, which implies that $S(\tilde{p}_t)$ is positive-definite. A similar argument can be found, e.g., in p.8 of [Ito et al., oracle-efficient algorithms for online linear optimization with bandit feedback, NeurIPS2019] (between Eq. (4) and (5)), and is implicitly used in [Bubeck, Lee, Eldan (2017)] as well. In the revised manuscript, we add a more clarified proof.

> 2. How to calculate/approximate the inverse of $S(\tilde{p}_t)$ efficiently?

Since $\tilde{p}_t$ is log-concave, for any $\epsilon > 0$, we can get an ϵ-approximation of $S(\tilde{p}_t)$ w.h.p. by generating $(d/\epsilon)^{O(1)}$ samples from $\tilde{p}_t$, from Corollary 2.7 of [Lovaŝz and Vempara (2007)]. Samples from $\tilde{p}_t$ can be generated with their polynomial-time sampling algorithm as mentioned in Section 4.4 of our manuscript. A similar discussion can be found in Lemma 5.17 of [Bubeck, Lee, Eldan (2017)] and around Corollary 1 of [Ito et al., oracle-efficient ..., NeurIPS2019]. This fact is implicitly used in [Hazan and Karnin (2016)] as well. We clarify this in the revised manuscript.

> - In Eq. (20), to avoid confusion, please say that this applies Lemma 1 with $S(p_t)^{-1/2}x$.

Yes. We shall state this more clearly in the revised manuscript. For more details, please see the response to Reviewer#2.

Dear Reviewer #2:

> For the unit ball the algorithm of Rakhlin and Sridharan (2013) has significantly smaller runtime..

We agree with this comment. In the revised version, we shall note these facts the reviewer pointed out.

> Also note that assumption (iii) is not an assumption due to the existence of the universal barrier.

We guess that the reviewer read the sentence "(iii) $\mathcal{A}$ has a self-concordant barrier with parameter $\theta \geq 1$" as "there exists $\theta \geq 1$ such that $\mathcal{A}$ has a θ-self-concordant barrier." We meant, however, that "for a given $\theta \geq 1$, $\mathcal{A}$ has a θ-self-concordant barrier," which is an assumption on θ and $\mathcal{A}$.

> from the main text it does not become clear how Lemma 1 is used,

As Reviewer #1 mentioned, we use Lemma 1 for $x = S(p_t)^{-1/2}y$ with $y \sim S(p_t)$. We can see that assumptions in Lemma 1 hold since we have $\mathbf{E}[xx^\top] = S(p_t)^{-1/2}\mathbf{E}[yy^\top]S(p_t)^{-1/2} = S(p_t)^{-1/2}S(p_t)S(p_t)^{-1/2} = I$ and since log-concavity is preserved under any liner transformation. Using Lemma 1 for $x = S(p_t)^{-1/2}y$, we obtain high-probability bounds for $\|x\|_2^2 = \|S(p_t)^{-1/2}y\|_2^2 = \|y\|_{S(p_t)^{-1}}^2$. We add a clear description of this in the revision.

Dear Reviewer #3:

> For example, in Line 395, I think the authors should argue more clearly why x the authors refer to is larger than -1.

We can confirm that $x > -1$ holds since x here corresponds to $x = -1 + \mathbf{E}[\exp(-\eta_t \langle \hat{\ell}_t - m_t, x \rangle)]$, as can be seen from the transformation in lines 393–395. We add a more clarified explanation in the revision.

> Also the parts in Lemma 4 using Lemma 1 is not well explained.. Another part is (20).

Please refer to the response for Reviewer #2.

> it is known that in MAB, if one uses truncated distribution, then the standard loss estimator is not unbiased anymore.

The unbiasedness is proved in the proof of Lemma 2. We guess that the standard loss estimator the reviewer refers to is the one using $S(p_t)^{-1}$ instead of $S(\tilde{p}_t)^{-1}$. This "standard" one may be indeed biased as the reviewer pointed out.

> Specifically, why is the matrix in (8) always invertible? ... Moreover, how to compute $S(p')^{-1}$ efficiently ...

Please refer to the response for Reviewer #1.