Faster Convergence with Multiway Preferences

Aadirupa Saha
Apple

Vitaly Feldman
Apple

Tomer Koren
Tel Aviv University
& Google

Yishay Mansour
Tel Aviv University
& Google

Abstract

We address the problem of convex optimization with preference feedback, where the goal is to minimize a convex function given a weaker form of comparison queries. Each query consists of two points and the dueling feedback returns a (noisy) single-bit binary comparison of the function values of the two queried points. Here we consider the sign-function-based comparison feedback model and analyze the convergence rates with batched and multiway (argmin of a set queried points) comparisons. Our main goal is to understand the improved convergence rates owing to parallelization in sign-feedback-based optimization problems. Our work is the first to study the problem of convex optimization with multiway preferences and analyze the optimal convergence rates. Our first contribution lies in designing efficient algorithms with a convergence rate of $\widetilde{O}(\frac{d}{\min\{m,d\}\epsilon})$ for m-batched preference feedback where the learner can query m-pairs in parallel. We next study a m-multiway comparison ('battling') feedback, where the learner can get to see the argmin feedback of m-subset of queried points and show a convergence rate of $\widetilde{O}(\frac{d}{\min\{\log m,d\}\epsilon})$. We show further improved convergence rates with an additional assumption of strong convexity. Finally, we also study the convergence lower bounds for batched preferences and multiway feedback optimization showing the optimality of our convergence rates w.r.t. m.

1 Introduction

Studying the problem of convex optimization presents a unique opportunity to delve deep into a practical field of vast applications and make a lasting impact in both academia and industry. Most commonly, convex optimization is studied in a first-order gradient oracle model, where the optimization algorithm may query gradients of the objective function; or a more limited model of zero-order oracle access, where the optimization algorithm may only query function values. Such optimization frameworks are well-studied in the literature [see, e.g., Nesterov, 2003, Hazan, 2019, Bubeck, 2014].

One major limitation of the above optimization frameworks lies in the feedback model: In many practical applications, obtaining complete gradient information or even access to a function value (zeroth-order) oracle could be difficult. E.g. in recommender systems, online shopping, search engine data, the only data available to the learning algorithm specifies only the preference feedback of their users across multiple choices. Similar problems may arise in other real-world domains including crowd-sourcing surveys, drug testing, tournament ranking, social surveys, etc.

Learning with Preference Feedback. The above line of problems gave rise to a new field of research where the system aims to optimize its performance based on only choice data or relative preferences across multiple items. The problem has been widely studied in the bandit community as Dueling Bandit problems which is an online sequential game where at each round the learner (algorithm) repeatedly selects a pair of items to be compared to each other in a "duel," and consequently observes a binary stochastic preference feedback of the winning item in this duel [Ailon et al., 2014, Wu and Liu, 2016, Sui et al., 2018, Saha and Gopalan, 2020b]. The goal of the learner is to identify the 'best item' with the least possible number of pairwise queries.

Need of Optimization Algorithms with Preference Feedback. The classical problem of dueling bandits, although received wide attention in the learning theory community, most of the studies were limited to finite decision space, which allowed the learner to examine each item one-by-one before identifying the best one Sui et al. [2017], Ghoshal and Saha [2022], Bengs et al. [2021]. This again becomes unrealistic for large-scale real-world problems where decision spaces could be extremely large. Precisely, the litera-

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

ture lacks optimization methods based on preference feedback, where the decision set could be potentially infinite.

Related Works and Limitations. Two earlier works that address the problem of convex optimization with preference feedback are Yue and Joachims [2009], Jamieson et al. [2012]. However, the first work yields suboptimal convergence bounds and Jamieson et al. [2012] only deals with strongly convex and smooth functions. Another recent work by Saha et al. [2021] addresses the problem of convex optimization with pairwise preference setting which is closest to our framework: The setting assumes an underlying (convex) loss function $f : \mathbb{R}^d \mapsto R$ and at each round the learner can see the relative ordering of the function values at the two queried points: Precisely, upon querying a duel $(\mathbf{x}_t, \mathbf{y}_t)$ at round t, the learner can observe sign$(f(\mathbf{x}_t) - f(\mathbf{y}_t))$, and the objective of the learner is to find a 'near-minimizer' of f as fast as possible. Despite their setup having an interesting angle of optimization with preference feedback, their feedback model is limited to only pairwise/ dueling queries.

Motivation of Our Work: Optimization with Multiway Preferences. While pairwise comparison feedback is perhaps the simplest to model and analyze, in most real-world systems, users get to make a choice from a set of options, be that in online marketplaces, Youtube recommendations, Google maps, restaurant selection, and many more. This raises a natural question about the relative power of multiway comparison feedback in terms of the query complexity. Further, in many settings, it is not feasible to update the model's predictions after every comparison feedback provided by the user, for example, due to communication delays. Instead, the system can ask a number of comparison queries in parallel and then update its state (and generate the next set of queries). In such settings, it is natural to ask how many such rounds of m queries would be necessary to identify the (approximate) minimizer?

Some Negative Results on Multiway Preferences in Bandits Literature: The setting of multiway preferences was studied as a generalization of the dueling bandit framework, however for finite decision spaces and for a very type of Multinomial Logit (or Plackett Luce) based preference model Saha and Gopalan [2019b,a], Ren et al. [2018]. However, their specific feedback model was not able to exploit the power of multiway queries, precisely, they show multiway feedback may not yield faster convergence results, even for finite decision space settings (when $\mathcal{D}$ is finite).

But in this work, we answer the above questions in the affirmative and studied two specific types of multiway preference models which can indeed yield faster convergence rates with larger strength of multiway queries (m). *It is important to note that our results do not contradict the negative results with Multinomial Logit (MNL) models [Chen et al., 2017, Ren et al., 2018, Saha and Gopalan, 2020a] as we use a different 'argmin' based preference model as opposed to the MNL model. One of our main strengths lies in identifying such a subsetwise preference feedback model which could exploit the strength of multiway preferences.* The noisy-winner feedback in the MNL model increases the variability which nullifies the strengths of querying larger subsets Saha and Gopalan [2019b], Chen et al. [2018], but our proposed algorithms show how to exploit the latter with our multiway preference models (Sections 3 and 4). To the best of our knowledge, our work is the first to study the problem of convex optimization (on infinite decision space) with multiway preferences and analyze the optimal convergence rates.

Our contributions. The specific contributions of our are listed below:

1. Our first contribution is to propose two multiway preference feedback models for optimizing fixed convex functions $f : \mathcal{D} \mapsto \mathbb{R}$: (1) *Batched Sign-Feedback-Optimization*: In this setting, the learner can get to query a subset of $m \leq d$ distinct pair of points and receives the sign (or comparison) feedback of each queried pairs. This can be seen as a batched preference feedback model where the learner can simultaneously query m pairs of duels (2) *Battling-Feedback-Optimization*: Unlike the previous model, in this case, the learner gets to query a subset of m points in the decision space $\mathcal{D}$ and only gets to see the minimizer of the queried set of m-points. We called this as *Battling (or Multiwise Winner)* feedback model, inspired from Saha and Gopalan [2018] (see Section 2 for details).

2. We first consider the batched feedback model. Assuming f is β-smooth we apply an 'aggregated normalized gradient descent' based routine that is shown to yield $O\left(\frac{d\beta D}{\epsilon \min\{m,d\}}\right)$ convergence rate for finding an ϵ-optimal point in $\mathcal{D}$ (Algorithm 1, Theorem 3). Following this we also propose an 'epochwise warm start with smooth optimization blackbox' idea to yield a faster convergence rate of $O\left(\frac{d\beta}{\min\{m,d\}\alpha} \log_2\left(\frac{\alpha}{\epsilon}\right)\right)$ with an additional assumption of strong convexity on f (Algorithm 2, Theorem 4). We also suggest how to deal with noisy preference feedback for these settings in Remark 1 (Section 3).

3. In Section 4, we propose optimization algorithms for the *Battling-Feedback-Optimization* (with Multiwise-winner) problem. In this case, we first design a novel convergence routine (Algorithm 3) that yields $O\left(\frac{d\beta D}{\min\{\log m,d\}\epsilon}\right)$ convergence rate for the class of smooth convex functions (Theorem 7). The key novelty in the algorithm lies in querying structured m subsets at each round, exploiting which we show how to extract $\log m$ distinct pairwise sign feedback and again use an aggregated normalized gradient descent method to yield the desired convergence bound. Following this we also show a faster $O\left(\frac{d\beta}{\min\{\log m,d\}\alpha} \log_2\left(\frac{\alpha}{\epsilon}\right)\right)$ convergence rate with strong convexity in Theorem 8. We also

remark on how to deal with noisy feedback models in this case as well.

4. Finally we show matching convergence lower bounds for both the multiway preference feedback models (resp. in Theorem 5 Theorem 9), which shows our dependencies on the multiway parameter m are indeed optimal, i.e. our algorithms are able to exploit the strength of multiway queries (m) optimally.

5. We provide empirical evaluations to corroborate our theoretical findings in Section 5.

6. Finally another minor contribution lies in dealing with bounded decision space throughout, unlike Saha et al. [2021] which assumes the decision space to be unbounded (Appendix C).

2 Preliminaries and Problem Statement

Notation. Let $[n] = \{1, \ldots n\}$, for any $n \in \mathbb{N}$. Given a set S and two items $x, y \in S$, we denote by $x \succ y$ the event x is preferred over y. For any $r > 0$, let $\mathcal{B}_d(r)$ and $\mathcal{S}_d(r)$ denote the ball and the surface of the sphere of radius r in d dimensions respectively. $\mathbf{I}_d$ denotes the $d \times d$ identity matrix. For any vector $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_2$ denotes the ℓ_2 norm of vector $\mathbf{x}$. $\mathbf{1}(\varphi)$ is generically used to denote an indicator variable that takes the value 1 if the predicate φ is true and 0 otherwise. $\text{sign}(x) = +1$ if $x \geq 0$ or -1 otherwise, $\forall x \in \mathbb{R}$. Unif(S) denotes a uniform distribution over any set S. We write $\tilde{O}$ for the big O notation up to logarithmic factors.

2.1 Useful Concepts for Convex Functions

Definition 1 (β-Smooth Convex Function)**.** *Assume $\mathcal{D} \subseteq \mathbb{R}^d$ be any convex and bounded decision space. Then any differential and convex function $f : \mathcal{D} \mapsto \mathbb{R}$ is also called β-smooth (any $\beta > 0$) if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$,*

$$f(\mathbf{x}) - f(\mathbf{y}) \leq \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\beta}{2}\|\mathbf{x} - \mathbf{y}\|^2.$$

Definition 2 (α-Strongly Convex Function)**.** *Assume $\mathcal{D} \subseteq \mathbb{R}^d$ be any convex and bounded decision space. Then any differential and convex function $f : \mathcal{D} \mapsto \mathbb{R}$ is also called α-strongly convex (any $\alpha > 0$) if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$,*

$$f(\mathbf{x}) - f(\mathbf{y}) \geq \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\alpha}{2}\|\mathbf{x} - \mathbf{y}\|^2.$$

2.2 Problem Setup.

We address the convex optimization problem with binary-valued sign preference feedback: Assume $f : \mathcal{D} \mapsto \mathbb{R}$ be any convex map defined on a convex set $\mathcal{D} \subseteq \mathbb{R}^d$. At every iteration t, the goal of the learner is to pick a pair of points $(\mathbf{x}_t, \mathbf{y}_t)$, upon which it gets to see a binary $0-1$ bit noisy comparison feedback o_t s.t.:

$$Pr[o_t = \text{sign}(f(\mathbf{x}_t) - f(\mathbf{y}_t))] = 1 - \nu,$$

where $\nu \in [0, 1/2)$ is the (unknown) noise-parameter, $\nu = 0$ corresponds to pure sign feeedback (without any noise).

We consider the following two generalizations of sign-feedback considered in Saha et al. [2021].

Batched-Sign Feedback: In this setting, at any round t, the learner can query m-parallel (batched) pair of points $\{(\mathbf{x}_t^i, \mathbf{y}_t^i)\}_{i=1}^m$ and gets to see the sign feedback for each pair of points, i.e. the learner receive m-bits of sign feedback $\{o_t^i\}_{i=1}^m$ such that $Pr[o_t^i = \text{sign}(f(\mathbf{x}_t^1) - f(\mathbf{y}_t^i))] = 1 - \nu,\ i \in [m]$.

Battling (Multiwise-Winner) Feedback: In this setting, at any round t, the learner can query a set S_t of m points $S_t = (\mathbf{x}_t^1, \mathbf{x}_t^2, \ldots, \mathbf{x}_t^m)$ and gets to see the $\arg\min$ feedback of the m-points – i.e., the learner receive only 1-bit of $\arg\min$ feedback $\{o_t \in [m]\}$ such that: $Pr[o_t = \arg\min(f(\mathbf{x}_t^1), f(\mathbf{x}_t^2), \ldots, f(\mathbf{x}_t^m))] = 1 - \nu$.

Objective. We consider the objective of minimizing the function sub-optimality gap: So if $\mathbf{x}_{T+1}$ is the point suggested by the algorithm after T rounds, then the goal is to

$$\min_{\mathbf{x} \in \mathcal{D}} \big(\mathbf{E}[f(\mathbf{x}_{T+1})] - f(\mathbf{x}^*)\big),$$

with least number of queries (T) possible.

3 Batched Dueling Convex Optimization with Sign Feedback

We first analyze the *Batched Sign-Feedback-Optimization* problem, where at each iteration the learner can query m-parallel (batched) pair of points $\{(\mathbf{x}_t^i, \mathbf{y}_t^i)\}_{i=1}^m$ and gets to see the sign feedback for each pair of points, $\{o_t^i\}_{i=1}^m$ such that $Pr[o_t^i = \text{sign}(f(\mathbf{x}_t^1) - f(\mathbf{y}_t^i))] = 1 - \nu,\ i \in [m]$, $(m \leq d)$. We present two algorithms for this setup, respectively for smooth and strongly convex optimization settings and show an $O(1/m)$ improved convergence rate in both settings compared to the single pair sign-feedback setting addressed in [Saha and Krishnamurthy, 2022] (see Theorem 3 and Theorem 4). We analyze the above algorithms for the noiseless setting (i.e. $\nu = 0$), but Remark 1 discusses how they can be easily extended to the noisy sign-feedback setup for any arbitrary $\nu \in [0, 0.5)$. Following this we also prove a convergence lower bound for the batched feedback setting which shows our $1/m$ rate of improvement with the batch size m is indeed optimal (see Theorem 5).

3.1 Proposed Algorithm: Batched-NGD

The main idea in this setup is to estimate gradient directions (normalized gradient estimates) in m different directions and take an aggregated descent step. Formally, at each round t, we can query m iid random unit directions, say

$\mathbf{u}_t^1, \ldots, \mathbf{u}_t^m \overset{\text{iid}}{\sim} \text{Unif}(\mathcal{S}_d(1))$, and find the normalized gradient estimates $g_t^i = o_t^i \mathbf{u}_t^i$ along each direction, where $o_t^i = \text{sign}\big(f(\mathbf{x}_t^i) - f(\mathbf{y}_t^i)\big)$ is the sign feedback of the i-th pair of queried duel (x_t^i, y_t^i). Subsequently, we update the running prediction as $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \mathbf{g}_t$, where $\mathbf{g}_t = \frac{1}{m}\sum_{i=1}^m \mathbf{g}_t^i$ denotes the aggregated normalized gradient estimate. The algorithm also maintains a running minimum $\mathbf{m}_t$ which essentially keeps track of $\min\{\mathbf{w}_1, \ldots, \mathbf{w}_t\}$[1]. The complete algorithm is given in Algorithm 1.

Algorithm 1 Batched-NGD (B-NGD)

1: **Input:** Initial point: $\mathbf{w}_1 \in \mathcal{D}$, Learning rate η, Perturbation parameter γ, Query budget T (depends on error tolerance ϵ), Batch-size m
2: **Initialize** Current minimum $\mathbf{m}_1 = \mathbf{w}_1$
3: **for** $t = 1, 2, 3, \ldots, T$ **do**
4: Sample $\mathbf{u}_t^1, \mathbf{u}_t^2, \ldots \mathbf{u}_t^m \sim \text{Unif}(\mathcal{S}_d(1))$
5: Set $\mathbf{x}_t^i := \mathbf{w}_t + \gamma \mathbf{u}_t^i,\ \mathbf{y}_t^i := \mathbf{w}_t - \gamma \mathbf{u}_t^i$
6: Play the duel $(\mathbf{x}_t^i, \mathbf{y}_t^i)$, and observe $o_t^i \in \pm 1$ such that $o_t^i = \text{sign}\big(f(\mathbf{x}_t^i) - f(\mathbf{y}_t^i)\big)$.
7: Update $\tilde{\mathbf{w}}_{t+1} \leftarrow \mathbf{w}_t - \eta \mathbf{g}_t$, where $\mathbf{g}_t = \frac{1}{m}\sum_{i=1}^m \mathbf{g}_t^i$, $\mathbf{g}_t^i = o_t^i \mathbf{u}_t^i$
8: Project $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w} \in \mathcal{D}} \|\mathbf{w} - \tilde{\mathbf{w}}_{t+1}\|$
9: Query the pair $(\mathbf{m}_t, \mathbf{w}_{t+1})$ and receive $\text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big)$.
10: $\mathbf{m}_{t+1} \leftarrow \begin{cases} \mathbf{m}_t & \text{if } \text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big) < 0 \\ \mathbf{w}_{t+1} & \text{otherwise} \end{cases}$
11: **end for**
12: Return $\mathbf{m}_{T+1}$

Theorem 3 (Convergence Analysis of Algorithm 1 for β-Smooth Functions)**.** *Consider f to be β smooth. Suppose Algorithm 1 is run with $\eta = \frac{m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ and $T_\epsilon = O\Big(\frac{d\beta D}{m\epsilon}\Big)$, where $\|\mathbf{w}_1 - \mathbf{x}^*\|^2 \le D$ (is an assumed known upper bound). Then Algorithm 1 returns $\mathbf{E}[f(\tilde{\mathbf{w}}_{T+1})] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $2T_\epsilon$, for any $m \le d$.*

Proof Sketch of Theorem 3. We start by noting that by definition: $\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \le \|\tilde{\mathbf{w}}_{t+1} - \mathbf{x}^*\|^2 = \|\mathbf{w}_t - \frac{\eta}{m}\sum_{i=1}^m \mathbf{g}_t^i - \mathbf{x}^*\|^2$, where the first inequality holds since projection reduces the distance to optimal $\mathbf{x}^*$. This further leads to

$$
\begin{aligned}
&m^2\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 = m(\|\mathbf{w}_t - \mathbf{x}^*\|^2 + \eta^2) \\
&- 2\eta \sum_{i=1}^m (\mathbf{w}_t - \mathbf{x}^*)^\top \mathbf{g}_t^i - 2\eta \sum_{i=1}^{m-1}\sum_{j=i+1}^m \big(\mathbf{w}_t - \mathbf{x}^*\big)^\top (\mathbf{g}_t^i + \mathbf{g}_t^j) \\
&+ 2\frac{m(m-1)}{2}\|\mathbf{w}_t - \mathbf{x}^*\|^2 + 2\eta \sum_{i=1}^{m-1}\sum_{j=i+1}^m {\mathbf{g}_t^i}^\top \mathbf{g}_t^j.
\end{aligned}
$$

Let us denote by $n_t = \frac{\nabla f(\mathbf{w}_t)}{\|\nabla f(\mathbf{w}_t)\|}$ the normalized gradient at point $\mathbf{w}_t$. Also let $\mathcal{H}_t$ the history $\{\mathbf{w}_\tau, U_\tau, \mathbf{o}_\tau\}_{\tau=1}^{t-1} \cup \mathbf{w}_t$ till time t and $U_t := \{\mathbf{u}_t^1, \ldots, \mathbf{u}_t^m\}$. Then *one important observation is that* the estimated gradients are nearly independent (their inner products are small): More precisely, for any $i \ne j$, since $\mathbf{u}_t^i$ and $\mathbf{u}_t^j$ are independent, from Theorem 21 we get:

$$
\begin{aligned}
\mathbf{E}_{U_t}[{\mathbf{g}_t^i}^\top \mathbf{g}_t^j \mid \mathcal{H}_t] &= \mathbf{E}_{\mathbf{u}_i}[{\mathbf{g}_t^i}^\top \mathbf{E}_{\mathbf{u}_j}[\mathbf{g}_t^j \mid \mathbf{u}_t^i] \mid \mathcal{H}_t] \\
&\le \frac{1}{\sqrt{d}}\left(\frac{n_t^\top n_t}{\sqrt{d}}\right) + 4\lambda_t = \frac{1}{d} + 4\lambda_t,
\end{aligned}
$$

where recall from Theorem 21 and Lemma 22, $\lambda_t \le \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\right)$. Combining this with the main equation, and further applying Theorem 21, with a bit of algebra one can get:

$$
\begin{aligned}
&m^2\mathbf{E}_{U_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t] = m(\|\mathbf{w}_t - \mathbf{x}^*\|^2 + \eta^2) \\
&\quad - 2\eta \sum_{i=1}^{m-1}\sum_{j=i+1}^m \big(\mathbf{w}_t - \mathbf{x}^*\big)^\top \mathbf{E}_{U_t}[(\mathbf{g}_t^i + \mathbf{g}_t^j) \mid \mathcal{H}_t] \\
&\quad + 2\frac{m(m-1)}{2}\|\mathbf{w}_t - \mathbf{x}^*\|^2 - 2\eta\sum_{i=1}^m (\mathbf{w}_t - \mathbf{x}^*)^\top \mathbf{g}_t^i \\
&\quad + 2\eta^2 \sum_{i=1}^{m-1}\sum_{j=i+1}^m \mathbf{E}_{U_t}[{\mathbf{g}_t^i}^\top \mathbf{g}_t^j \mid \mathcal{H}_t] \\
&= m^2\|\mathbf{w}_t - \mathbf{x}^*\|^2 + \eta^2(m + \frac{m(m-1)}{d}) \\
&\quad - 2\eta m^2[(\mathbf{w}_t - \mathbf{x}^*)^\top \mathbf{n}_t + 4m^2\sqrt{d}\eta\|\mathbf{w}_t - \mathbf{x}^*\|\lambda_t \\
&\qquad + 4m^2\sqrt{d}\eta\|\mathbf{w}_t - \mathbf{x}^*\|\lambda_t.
\end{aligned}
$$

Further from Claim-2 of Lemma 19, and from the fact that $m < d$, we can derive:

$$
\begin{aligned}
&m^2\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t] \le m^2\|\mathbf{w}_t - \mathbf{x}^*\|^2 \\
&\quad + m^2(-2\eta\frac{c\sqrt{2\epsilon}}{\sqrt{d\beta}} + 8\eta\lambda_t\sqrt{d}\|\mathbf{w}_t - \mathbf{x}^*\|) + 2m\eta^2,
\end{aligned}
$$

and choosing $\gamma \le \frac{\|\nabla f(\mathbf{w}_t)\|}{960\beta d\sqrt{d}\|\mathbf{w}_t - \mathbf{x}^*\|\sqrt{\log 480}}\sqrt{\frac{2\epsilon}{\beta}}$, we get:

$$
\begin{aligned}
\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2] \mid \mathcal{H}_t] &\le \|\mathbf{w}_t - \mathbf{x}^*\|^2 \\
&- \frac{\eta\sqrt{2\epsilon}}{10\sqrt{d\beta}} + \frac{\eta\sqrt{2\epsilon}}{20\sqrt{d\beta}} + \frac{2\eta^2}{m}.
\end{aligned}
$$

One possible choice of γ is $\gamma = \frac{\epsilon^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ (since $\|\nabla f(\mathbf{x})\| \ge \frac{\epsilon}{D}$ for any $\mathbf{x}$ s.t. $f(\mathbf{x}) - f(\mathbf{x}^*) > \epsilon$ by Lemma 20). Then following from above, we further get:

$$
\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t]] \le \|\mathbf{w}_t - \mathbf{x}^*\|^2,
$$

[1] Interested readers can check the analysis of Projected-Normalized Gradient Descent algorithms for single sign-Feedback in Appendix C. This is unlike the version studied in the literature which considered unconstrained optimization (i.e. $\mathcal{D} = \mathbb{R}^d$) [Saha et al., 2021], although the analysis is quite similar, except we have to account for the projection step additionally. This also leads to simpler tuning of the perturbation parameter γ in our case.

$$- \frac{(\sqrt{2}-1)m\epsilon}{400d\beta} \Big(\text{setting } \eta = \frac{m\sqrt{\epsilon}}{20\sqrt{d\beta}}\Big)$$
$$\implies \mathbf{E}_{\mathcal{H}_T}[\|\mathbf{w}_{T+1} - \mathbf{x}^*\|^2] \le \|\mathbf{w}_1 - \mathbf{x}^*\|^2 - \frac{(\sqrt{2}-1)m\epsilon T}{400d\beta}, \big(\text{summing } t = 1, \ldots T\big).$$

Above implies, if indeed $f(\mathbf{w}_\tau) - f(\mathbf{x}^*) > \epsilon$ continues to hold for all $\tau = 1, 2, \ldots T$, then $\mathbf{E}[\|\mathbf{w}_{T+1} - \mathbf{x}^*\|^2] \le 0$, for $T \ge \frac{400md\beta}{(\sqrt{2}-1)\epsilon}(\|\mathbf{w}_1 - \mathbf{x}^*\|^2)$, which basically implies $\mathbf{w}_{T+1} = \mathbf{x}^*$ (i.e. $f(\mathbf{w}_{T+1}) = f(\mathbf{x}^*)$). Otherwise there must have been a time $t \in [T]$ such that $f(\mathbf{w}_t) - f(\mathbf{x}^*) < \epsilon$. The complete proof is given in Appendix A.1. □

3.2 Improved Convergence Rates with Strong Convexity

We now show how to obtain a better convergence rate with an additional assumption of α-strong convexity on $f : \mathcal{D} \mapsto \mathbb{R}$ by simply reusing any optimal optimization algorithm for β-smooth convex functions (and hence we can use our B-NGD Algorithm 1, proposed earlier). Our proposed method Improved Batched-NGD (Alg. Algorithm 2) adapts a phase-wise iterative optimization approach, where inside each phase we use B-NGD as a blackbox to locate a ϵ_k-optimal point in that phase, say $\mathbf{w}_{k+1}$, with exponentially decaying $\epsilon_k = O(\frac{1}{2^{k-1}})$. We then warm start the B-NGD algorithm in the next phase from $\mathbf{w}_{k+1}$ and repeat – the idea is adapted from the similar warm starting idea proposed by Saha et al. [2021]. The method yields improved $O(\log \frac{1}{\epsilon})$ convergence due to the nice property of strong convexity where nearness in function values implies nearness from the optimal $\mathbf{x}^*$ in ℓ_2-norm, unlike the case for only β-smooth functions (see Lemma 10). Algorithm 2 gives the complete detail.

Algorithm 2 Improved Batched-NGD with Strong Convexity (ImpB-NGD)

1: **Input:** Error tolerance $\epsilon > 0$, Batch size m
2: **Initialize** Initial point: $\mathbf{w}_1 \in \mathbb{R}^d$ such that $D := \|\mathbf{w}_1 - \mathbf{x}^*\|^2$ (assume known).
Phase counts $k_\epsilon := \lceil \log_2 \left(\frac{\alpha}{\epsilon}\right) \rceil$, $t \leftarrow \frac{800d\beta}{(\sqrt{2}-1)\alpha}$
$\eta_1 \leftarrow \frac{m\sqrt{\epsilon_1}}{20\sqrt{d\beta}}, \epsilon_1 = \frac{400d\beta D}{(\sqrt{2}-1)t_1} = 1, t_1 = t\|\mathbf{w}_1 - \mathbf{x}^*\|^2$
$\gamma_1 \leftarrow \frac{\epsilon_1^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}, \mathbf{m}_1 = \mathbf{w}_1$
3: Update $\mathbf{w}_2 \leftarrow$ **B-NGD**$\big(\mathbf{w}_1, \eta_1, \gamma_1, t_1\big)$
4: **for** $k = 2, 3, \ldots, k_\epsilon$ **do**
5: $\eta_k \leftarrow \frac{m\sqrt{\epsilon_k}}{20\sqrt{d\beta}}, \epsilon_k = \frac{400d\beta}{(\sqrt{2}-1)t_k}, t_k = 2t$
$\gamma_k \leftarrow \frac{\epsilon_k^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$.
6: Update $\mathbf{w}_{k+1} \leftarrow$ **B-NGD**$\big(\mathbf{w}_k, \eta_k, \gamma_k, t_k, m\big)$
7: **end for**
8: Return $\mathbf{m}_\epsilon = \mathbf{w}_{k_\epsilon+1}$

Theorem 4 (Convergence Analysis of Algorithm 2 for α-strongly convex and β-Smooth Functions)**.** *Consider f to be α-strongly convex and β-smooth. Then Algorithm 2 returns $\mathbf{E}[f(\mathbf{m}_\epsilon)] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $O\Big(\frac{d\beta}{m\alpha}(\log_2\left(\frac{\alpha}{\epsilon}\right) + \|\mathbf{x}_1 - \mathbf{x}^*\|^2)\Big)$, for any $m \le d$.*

Due to space constraints, the proof is moved to Appendix A.2.

Remark 1. [Noisy Feedback $\nu \in (0, 1/2)$)]. *Note Algorithm 1 and Algorithm 2 (and consequently Theorem 3 and Theorem 4) work only for the noiseless feedback setting, when $\nu = 0$. However, it is easy to extend the above two algorithms for the noisy sign-feedback setting (for any $\nu \in (0, 0.5)$) by the resampling trick proposed in Saha et al. [2021]: Precisely, the idea is to query any pair of point (x_t, y_t) for $O(\frac{1}{\nu^2})$ times to recover the true sign feedback sign$(f(\mathbf{x}_t) - f(\mathbf{y}_t))$ with high confidence, and rest of the algorithm remains as is. Clearly, this would lead to the convergence bounds of $O\Big(\frac{d\beta D}{m(0.5-\nu)^2\epsilon}\Big)$ and $O\Big(\frac{d\beta}{m\alpha(0.5-\nu)^2}(\log_2\left(\frac{\alpha}{\epsilon}\right)\Big)$ respectively for settings of Theorem 3 and Theorem 4, where the additional $O(1/(0.5-\nu)^2)$-multiplicative factor is accounted for resampling of every pairwise query in Algorithm 1 and Algorithm 2.*

3.3 Lower Bound for *Batched Sign-Feedback-Optimization*

In this section, we show the convergence lower bounds for the *Batched Sign-Feedback-Optimization* problem. Theorem 5 shows indeed our $1/m$ rate of improvement with m-batch size is optimal.

Theorem 5 (Convergence Lower Bound: *m-Batched Sign-Feedback-Optimization* Problem)**.** *Assume the noiseless setting $\nu = 0$ and $f : \mathcal{D} \mapsto R$ be any smooth and strongly convex function. Then the ϵ-convergence bound for any algorithm for the m-*Batched Sign-Feedback-Optimization *problem is at least $\Omega(\frac{d}{m}\log\frac{1}{\epsilon})$.*

Proof of Theorem 5. We first require to show $\Omega(d\log\frac{1}{\epsilon})$ convergence lower bound for the *Sign-Feedback-Optimization* problem for smooth and strongly convex functions, as we prove below [2]:

Lemma 6 (Convergence Lower Bound for *Sign-Feedback-Optimization* Problem)**.** *Let $f : \mathcal{D} \mapsto R$ be any smooth and strongly convex function. Then the*

[2]It is important to note that Theorem 1 of [Jamieson et al., 2012] claims to yield an $\Omega(d\log\frac{1}{\epsilon})$ lower bound for the same problem (for their setting $\kappa = 1$). However, they still need to assume their noise parameter μ, which is equivalent to $\mu = 1/2 - \nu$ in our case, satisfies $\mu \le 1/2$, which is equivalent to assuming $\gamma > 0$ in our case. So their lower bound is information-theoretic owning to the noisy sign feedback, but not an optimization-based lower bound. Precisely, their lower bound does not apply in the noiseless setting $\mu = 1/2$ (or $\nu = 0$).

ϵ-convergence bound for any algorithm for the Sign-Feedback-Optimization *problem is at least* $\Omega(d \log \frac{1}{\epsilon})$.

To proof above, assume $\mathcal{D} = \mathcal{B}_d(1)$ is the unit ball in dimension d and let $N(\mathcal{D}, \epsilon, \|\|_2)$ be the ϵ packing of $\mathcal{D}$ w.r.t. ℓ_2-norm [Wu, 2016]. Let $f(x) = \|\mathbf{x} - \mathbf{x}^*\|_2^2$, and the adversary can select $\mathbf{x}^*$ arbitrarily as any $\mathbf{x}^* \in N(\mathcal{D}, \epsilon, \|\|_2)$. Then note any single pair of sign feedback can allow the learner to remove at most half of the point in decision space $\mathcal{D}$, so after t number of pairwise sign feedback, the adversary still has the choice to select $\mathbf{x}^*$ from $\frac{1}{2^t}|N(\mathcal{D}, \epsilon, \|\|_2)| \le \frac{1}{2^t}\left(\frac{3}{\epsilon}\right)^d$ many numbers of points. This yields the desired ϵ-convergence sample complexity lower bound of $\Omega(d \log \frac{1}{\epsilon})$, as the learner would need to make at least $t \ge d \log \frac{1}{\epsilon}$ many pairwise sign queries before the adversary would be left with atmost $\frac{1}{2^t}\left(\frac{3}{\epsilon}\right)^d \le 1$ choice for $\mathbf{x}^*$. The above derivation is inspired by the lower bound proof of Blum et al. [2024].

Having equipped with the lower bound of Lemma 6, this immediately implies the desired lower bound of Theorem 5 as one can hope to get an improved convergence bound of at most $1/m$-multiplicative factor, even when m rounds are merged into a single round. □

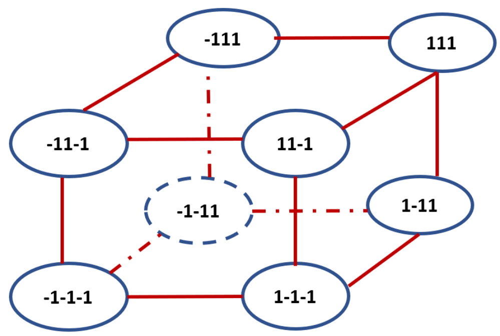
Remark 2. *It is worth noting that the lower bound above assumes f to be both strongly convex and smooth which yields a convergence lower bound of $\Omega(d \log \frac{1}{\epsilon})$ in the first place. However, it remains an open problem if we can obtain $\Omega(\frac{d}{\epsilon})$ lower bound for the class of just smooth functions to match the upper bound of Theorem 3 for only smooth convex functions (without strong convexity).*

4 Battling (Multiwise-Winner) Convex Optimization with Sign Feedback

In this section, we investigate the *Battling-Feedback-Optimization* problem. Recall from Section 2, in this case at each iteration t, the learner can query a set S_t of m points $S_t = (\mathbf{x}_t^1, \mathbf{x}_t^2, \ldots, \mathbf{x}_t^m)$ and gets to see the $\arg\min$ feedback of the m-points: $\{o_t \in [m]\}$ such that: $Pr[o_t = \arg\min(f(\mathbf{x}_t^1), f(\mathbf{x}_t^2), \ldots, f(\mathbf{x}_t^m))] = 1 - \nu, \quad \nu \in [0, 1/2)$. As before, We consider the noiseless case ($\nu = 0$) first, and present two algorithms for smooth and strongly convex optimization settings. The interesting fact is, in this case, we could only show $O(\frac{1}{\log m})$ improved convergence rate in this feedback model compared to the $O(\frac{1}{m})$ in the *Batched Sign-Feedback-Optimization* setting (see Theorem 7 and Theorem 8). In fact, a more interesting fact is we also show that $O(\frac{1}{\log m})$ improvement is the best we can hope for in this feedback model, proving a matching convergence lower bound (see Theorem 9).

4.1 Proposed Algorithm: Battling-NGD

Useful Notations. We denote by $V_n = \{(\pm 1)^n\}$, for any $n \in \mathbb{N}_+$. Clearly $|V_n| = 2^n$. Let $\mathcal{G}(V)$ be the graph with vertex set $V_n \subseteq \{\pm 1\}^n$ and there exists an (undirected) edge between two nodes $\mathbf{v}$ and $\tilde{\mathbf{v}}$ iff $\mathbf{v}$ and $\tilde{\mathbf{v}}$ only differs sign in one of the n coordinates, i.e. $\exists k \in [n],\ v(k) = \tilde{v}(k)$ and $v(k') = \tilde{v}(k')$ for any $k' \neq k$. Clearly the number of neighboring nodes of any vertex $\mathbf{v} \in V_n$ in graph $\mathcal{G}$ is $|\mathcal{N}(\mathbf{v}, \mathcal{G})| = n$. In other words, the degree of any node in graph $\mathcal{G}$ is n. We show an example for $n = 3$ in the right figure. Also, let us define $\ell_m = \lfloor \log m \rfloor$.



Algorithm Description. There are three novelties in our algorithmic idea: (i). Structure of the query sets, (ii). *One-vs-All* feedback idea and (iii). Extracting $\log m$ batched sign feedback. We explain them in more detail below.

(i). **Structured Query Sets:** As before, the algorithm maintains a current point $\mathbf{w}_t$ (initialized to $\mathbf{w}_1 \in \mathcal{D}$). At each time t, it queries a set S_t of m points around $\mathbf{w}_t$ such that for every point $\mathbf{x} \in S_t$, there exists exactly ℓ_m neighboring points which are *symmetrically opposite to* $\mathbf{x}$ *in exactly one of the realization of* $\mathbf{u}_t^i$*s*: More precisely, at each time t, the algorithm first samples ℓ_m vectors $\mathbf{u}_t^i \overset{\text{iid}}{\sim} \text{Unif}(\mathcal{S}_d(\frac{1}{\sqrt{\ell_m}}))$ independently, $i \in [\ell_m]$. Let $U_t = [\mathbf{u}_t^1, \ldots, \mathbf{u}_t^{\ell_m}] \in \mathbb{R}^{d \times \ell_m}$, and define $S_t = \{\mathbf{w}_t + \gamma U_t \mathbf{v} \mid \mathbf{v} \in V_{\ell_m}\}$. Note that by construction indeed $S_t = 2^{\ell_m} \le m$. Further, note for any point $\mathbf{x} = \mathbf{w}_t + \gamma U_t \mathbf{v} \in S_t$ there exists exactly ℓ_m symmetrically opposing points $\mathbf{x}_i' = \mathbf{w}_t + \gamma U_t \mathbf{v}_i' \in S_t$, for all $\mathbf{v}_i' \in \mathcal{N}(\mathbf{v}, \mathcal{G})$ such that $\frac{(\mathbf{x} - \mathbf{x}_i')}{2\gamma v_i} = \mathbf{u}_t^i,\ i \in [\ell_m]$. Given any such point $\mathbf{x}_\mathbf{v} := \mathbf{w}_t + \gamma U_t \mathbf{v}$, let us denote by the set $\mathcal{N}(\mathbf{x}_\mathbf{v}) = \{\mathbf{w}_t + \gamma U_t \mathbf{v}_i' \mid \mathbf{v}_i' \in \mathcal{N}(\mathbf{v}, \mathcal{G})\}$ of all symmetrically opposing points of $\mathbf{x}$ in S_t around $\mathbf{w}_t$ which differs in exactly one of the realization of $\mathbf{u}_t^i$s. This property will play a very crucial role in our analysis, as we will see in the convergence proof of Algorithm 3 (see proof of Theorem 7).

Upon constructing the set S_t, the algorithm queries the m-subset S_t and receives the winner feedback $o_t \in [m]$.

(ii). **One-vs-All Feedback Idea:** Note by definition of the winner feedback model, $f(\mathbf{x}_t^{o_t}) < f(\mathbf{x}_t^i),\ \forall i \in [m], i \neq$

o_t. Thus clearly, $\text{sign}\big(f(\mathbf{x}_t^{o_t}) < f(\mathbf{x}_t^i)\big) = -1$. So one may essentially recover exactly $m-1$ pairwise sign-feedback.

(iii). **Extracting** $\log m$ **Batched Sign Feedback:** However, there are inherent dependencies among these pair of points and most of these extracted sign feedback is redundant. We precisely identify $O(\log m)$ specific winner-vs-loser pairs and use their pairwise sign feedback to obtain a normalized gradient estimate. Let us denote by $\mathbf{x}_t^{o_t} = \mathbf{w}_t + \gamma U_t \mathbf{v}$ for some $\mathbf{v} \in V_{\ell_m}$. Then we choose all the symmetrically opposing pairs $(\mathbf{x}_t^{o_t}, \mathbf{y}_t^i)$ for all $\mathbf{y}_t^i \in \mathcal{N}(\mathbf{x}_t^{o_t})$ (as described in $\#(i)$ above), and extract the corresponding ℓ_m pairwise sign feedback. The setting then can simply reduce back to the $O(\log m)$-batched sign feedback setting and one use the similar algorithmic idea of Algorithm 1.

More precisely, the algorithm finds the ℓ_m normalized gradient estimates $g_t^i = o_t^i \mathbf{v}_t^i$ for all $i \in [\ell_m]$, where $o_t^i = \text{sign}\big(f(\mathbf{x}_t^{o_t}) - f(\mathbf{y}_t^i)\big) = -1$ is the sign feedback of the i-th (winner-loser) pair and $\mathbf{v}_t^i = v_i \mathbf{u}_t^i \sim \text{Unif}(\mathcal{S}_d(\frac{1}{\sqrt{\ell_m}}))$ is the corresponding 'scaled-unit' direction. Finally, we update the running prediction using the normalized gradient descent technique $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta \mathbf{g}_t$, using the aggregated descent direction $\mathbf{g}_t = \frac{1}{\ell_m}\sum_{i=1}^{\ell_m} \mathbf{g}_t^i$. As before, we also maintain a running minimum $\mathbf{m}_t$ which keeps track of $\min\{\mathbf{w}_1, \ldots, \mathbf{w}_t\}$. The complete algorithm is given in Algorithm 3.

One important thing to note is that from m-argmin feedback one can extract actually $m-1$ *i.e. all 'winner-vs-loser comparison pair'*, then why are we exploiting only specific $\log m$ pairs out of them? - The answer is on a high level, at any round t, our goal is to find out the maximum number of **'independent' descent directions** or directional gradients around the current optimizer $\mathbf{w}_t$ and then simply take an η-step along the aggregated descent direction (Line 8, Alg 3). The number ℓ_m is carefully chosen since, following the argument in Sec 4.1, that is the maximum number of "*consistent winner-vs-rest type of sign feedback*" one can extract such that the resulting normalized gradients $\mathbf{g}_t^i,\ i \in [\ell_m]$ are still independent. Even if one derives more normalized gradients (from the rest $m-1-\ell_m$ pairs), that will only result in linearly dependent gradients and hence could be expressed in terms of the linear combination of $\mathbf{g}_t^1, \ldots, \mathbf{g}_t^{\ell_m}$, thus yielding no additional information. For a formal justification, please see the proof of Theorem 7 to see how the independence assumption comes into play. Moreover, our matching lower bound argument in Theorem 9 corroborates that one could only hope for an $O(1/\ell_m)$-multiplicative factor improved convergence with m-argmin feedback (w.r.t. single sign feedback), which further ensures the tightness of our algorithmic approach and analysis.

Theorem 7 (Convergence Analysis of Algorithm 3 for β-Smooth Functions)**.** *Consider f to be β smooth. Suppose Alg. 3 is run with $\eta = \frac{\ell_m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{960\beta d\ell_m\sqrt{d\ell_m}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ and $T_\epsilon = O\Big(\frac{d\beta D}{\epsilon \ell_m}\Big)$, where $\|\mathbf{w}_1 - \mathbf{x}^*\|^2 \le D$, $\ell_m = \lfloor \log m \rfloor \le d$. Then Algorithm 3 returns $\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $2T_\epsilon$.*

Algorithm 3 Battling-NGD

1: **Input:** Initial point: $\mathbf{w}_1 \in \mathcal{D}$, Learning rate η, Perturbation parameter γ, Query budget T (depends on error tolerance ϵ), Batch-size m. Define $\ell_m := \lfloor \log m \rfloor$ and $\tilde{m} := 2^{\ell_m} \le m$.
2: **Initialize** Current minimum $\mathbf{m}_1 = \mathbf{w}_1$
3: **for** $t = 1, 2, 3, \ldots, T$ **do**
4: Sample $\mathbf{u}_t^1, \mathbf{u}_t^2, \ldots \mathbf{u}_t^{\ell_m} \overset{\text{iid}}{\sim} \text{Unif}(\mathcal{S}_d(\frac{1}{\sqrt{\ell_m}}))$. Denote $U_t := [\mathbf{u}_t^1, \ldots, \mathbf{u}_t^{\ell_m}] \in \mathbb{R}^{d \times \ell_m}$
5: Define $S_t := \{\mathbf{w}_t + \gamma U_t \mathbf{v} \mid \mathbf{v} \in V_{\ell_m}\}$ (see definition of V_{ℓ_m} in the description)
6: Play the m-subset S_t
7: Receive the winner feedback $o_t = \arg\min(f(\mathbf{x}_t^1), f(\mathbf{x}_t^2), \ldots, f(\mathbf{x}_t^{\tilde{m}}))$
8: Update $\tilde{\mathbf{w}}_{t+1} \leftarrow \mathbf{w}_t - \eta \mathbf{g}_t$, where $\mathbf{g}_t = \frac{1}{\ell_m}\sum_{i=1}^{\ell_m} \mathbf{g}_t^i$, $\mathbf{g}_t^i = -v_i \mathbf{u}_t^i$
9: Project $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w} \in \mathcal{D}} \|\mathbf{w} - \tilde{\mathbf{w}}_{t+1}\|$
10: Query the pair $(\mathbf{m}_t, \mathbf{w}_{t+1})$ and receive $\text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big)$.
11: $\mathbf{m}_{t+1} \leftarrow \begin{cases} \mathbf{m}_t & \text{if } \text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big) < 0 \\ \mathbf{w}_{t+1} & \text{otherwise} \end{cases}$
12: **end for**
13: Return $\mathbf{m}_{T+1}$

Proof Sketch of Theorem 7. Due to space limitations, the complete proof is deferred to Appendix B.1. The key idea relies on the idea of constructing the structured query set S_t which allows us to derive $O(\log m)$ winner-vs-loser sign feedback from the m-multiwise winner o_t, each of which results in an *independent* (normalized) gradient estimate $\mathbf{g}_t^i, \forall i \in [\ell_m]$. Note here the independence of $\mathbf{g}_t^i$s is crucial, which was possible due to the special structure of the query set S_t. We prove the formal statement in Theorem 11, Appendix B.2. The final stretch of the proof relies on exploiting the aggregated normalized gradient $\mathbf{g}_t = \frac{1}{\ell_m}\sum_{i=1}^{\ell_m} \mathbf{g}_t^i$, similar to the batched feedback model that yields the final $O(\frac{1}{\ell_m})$ factor improvement in the sample complexity of the convergence analysis. □

4.2 Improved Convergence Rates with Strong Convexity

It is easy to argue that, with the additional assumption of strong convexity, we can again obtain an improved convergence rate of $O(\frac{d}{\log m}\log\frac{1}{\epsilon})$ similar to batched setting (Section 3.2). Following the same 'phase-wise progress with warm starting' idea of Algorithm 2 with now using Battling-NGD (Algorithm 3) as the underlying black-box for an algorithm of *Battling-Feedback-Optimization* the problem for smooth convex functions, we can design an algorithm, say **Improved Battling-NGD**, to achieve improved $O(\log \frac{1}{\epsilon})$ convergence rate with strong convexity. We omit the specific details for brevity which can be easily inferred by combining Algorithm 2 and Algorithm 3. Theorem 8 gives the convergence rate of the above approach. Further details are deferred to Appendix B.3.

Theorem 8 (Improved Convergence Rate for α-strongly convex and β-Smooth Functions)**.** *Consider f to be α-strongly convex and β-smooth and let $\ell_m = \lfloor \log m \rfloor \le d$. Then* Improved Battling-NGD *returns an ϵ-optimal point within $O\Big(\frac{d\beta}{\alpha \ell_m}(\log_2\left(\frac{\alpha}{\epsilon}\right) + \|\mathbf{x}_1 - \mathbf{x}^*\|^2)\Big)$ many multiwise queries.*

Noisy Argmin Feedback $\nu \in (0, 1/2)$. This setting can be handled using the same '*resampling idea*' explained in Remark 1. This would respectively yield convergence bounds of $O\Big(\frac{d\beta D}{(0.5-\nu)^2 \epsilon \log m}\Big)$ and $O\Big(\frac{d\beta}{\alpha(0.5-\nu)^2 \log m}(\log_2\left(\frac{\alpha}{\epsilon}\right) + \|\mathbf{x}_1 - \mathbf{x}^*\|^2)\Big)$ for the settings of Theorem 7 and Theorem 8.

4.3 Lower Bound: *Battling-Feedback-Optimization*

Theorem 9 (Convergence Lower Bound for *Battling-Feedback-Optimization* Problem)**.** *Let $f : \mathcal{D} \mapsto R$ be any smooth and strongly convex function. Then the ϵ-convergence bound for any algorithm for the m-*Battling-Feedback-Optimization *problem is at least $\Omega(\frac{d}{\log m} \log \frac{1}{\epsilon})$.*

Proof. We appeal back to the lower bound derivation idea of Theorem 5 to derive the lower bound of Theorem 9. Same as before, let us assume $\mathcal{D} = \mathcal{B}_d(1)$ and $N(\mathcal{D}, \epsilon, \|\|_2)$ be the ϵ packing of $\mathcal{D}$ w.r.t. ℓ_2-norm. Also assume $f(x) = \|\mathbf{x} - \mathbf{x}^*\|_2^2$, and the adversary can select $\mathbf{x}^*$ arbitrarily as any $\mathbf{x}^* \in N(\mathcal{D}, \epsilon, \|\|_2)$.

Then in this battling (multiwise-winner) feedback model, note that a single m-point subsetwise query can allow the learner to remove at most $1/m$ portions the points in decision space $\mathcal{D}$. Then after t number of such queries, the adversary will still have the choice to select $\mathbf{x}^*$ from $\frac{1}{m^t}|N(\mathcal{D}, \epsilon, \|\|_2)| \le \frac{1}{2^t}\left(\frac{3}{\epsilon}\right)^d$ many numbers of points. This immediately yields the desired $\Omega(\frac{d}{\log m} \log \frac{1}{\epsilon})$ sample complexity lower bound for the *Battling-Feedback-Optimization* setting, as the learner would need to make at least $t \ge \frac{d}{\log m} \log \frac{1}{\epsilon}$ many m-multiwise queries before the adversary would be left with at most $\frac{1}{m^t}\left(\frac{3}{\epsilon}\right)^d \le 1$ choice for $\mathbf{x}^*$. □

Note, Theorem 7 shows that our proposed algorithms actually yield optimal convergence rate in terms of m, but it is still an open problem to see if one can prove a matching $\Omega(\frac{d}{\epsilon \log m})$ convergence lower bound for the class of smooth functions (without strong convexity).

5 Experiments

In this section, we provide an empirical evaluation of our proposed methods to compare the convergence rates with different types of feedback models (1) Single sign Feedback, (2) m-Batched sign Feedback, and (3) m-Argmin Feedback. We run experiments in the following settings:

Algorithms. We compare three algorithms, (1) NGD, (2) Batched-NGD (m-NGD) and (3) Battling-NGD (s-NGD) for the above three different types of feedback.

Experiment Tradeoff between the Query complexity (T) vs SubOptimality Gap ($f(\mathbf{w}_t) - f(\mathbf{x}^*)$) for different types of multiway preference feedback:

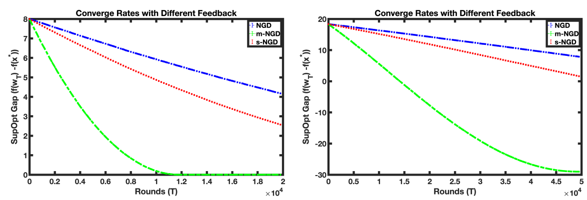


Figure 1: Query complexity (T) vs SubOptimality Gap ($f(\mathbf{w}_t) - f(\mathbf{x}^*)$) for (1) Single sign Feedback (NGD), (2) m-Batched sign Feedback (m-NGD), and (3) m-Argmin sign Feedback (s-NGD). First plot uses $f(x) = \|\mathbf{x}\|_2^2$. The second plot uses $f(x) = 3 + \sum_{i=1}^d sin(x_i)$. We set $d = 32$, $m = 6$ and initialized $x_0(i) = 0.5,\ \forall i \in [d]$.

Observations: Figure 1 corroborates our results showing that indeed we get the fastest convergence for m-Batched sign Feedback, followed by m-Multiway Feedback and single sign feedback leads to the slowest convergence in each case. Further, note $f(x^*)$ is respectively 0 and -29 for the above two functions, and the algorithms indeed tend to converge to the true minimum over time. Since m-NGD converges the fastest, note it already converged to the true minimum in both settings.

We report some additional experiments to reflect the tradeoff between convergence rate vs size of multiway preferences, for both m-Batched sign and m-Argmin feedback. We use the function $f : \mathbb{R}^{32} \mapsto \mathbb{R}$ such that $f(x) = \|\mathbf{x}\|_2^2 + \frac{1}{2}\|\mathbf{x}\|_1^2$. So we have $d = 32$.

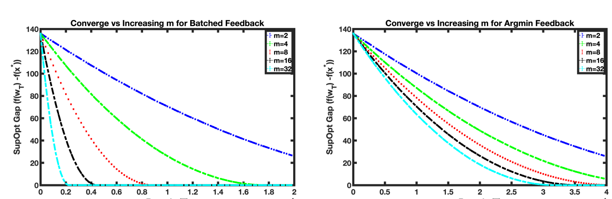


Figure 2: (left) Convergence rate vs time with increasing m (right) Convergence rate vs time with increasing m. We use $f(x) = \|\mathbf{x}\|_2^2 + \frac{1}{2}\|\mathbf{x}\|_1^2$ for both cases, with $d = 32$, and initialized $x_0(i) = 0.5,\ \forall i \in [d]$.

Observations: Figure 2 again corroborates with our inferences on the tradeoff between convergence rates vs increasing m for the m-batched and m-argmin feedback. Indeed the rate of convergence decays linearly as $O(1/m)$ for m-Batched sign Feedback, as derived in Theorem 3, Theorem 4; whereas with m-argmin feedback the decay is only logarithmic $O(\frac{1}{\log m})$ in m, as derived in Theorem 7, Theorem 8. The algorithms converge to the true minimum $f(\mathbf{x}^*) = 0$ in both settings, although the rate of convergence is much faster in the left case (i.e. with m-batched sign feedback).

6 Perspective

We address the problem of convex optimization with multiway preference feedback, where the learner can only receive relative feedback of a subset of queried points and design gradient descent-based algorithms with fast convergence rates for smooth and strongly convex functions. In particular, we worked with batched and argmin-type m-multiway preferences and designed algorithms with optimal convergence dependencies on m. Our work is the first to study and analyze the problem of convex optimization with multiway preferences.

Future Works. A natural extension of this work could be to understand if one can work with the class of any arbitrary convex functions (beyond the smoothness assumption which is crucially used to derive the normalized gradient estimates in the current algorithms). Investigating our problem setup to a more general class of preference functions could be useful to understand what is the right rate of improvement one can hope for with parallelism Ren et al. [2018]. Another interesting direction could be to generalize the setting to an online model, where the underlying functions can vary across time. Can we even hope to achieve sublinear convergence (or regret bounds) for such cases?

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreements No. 882396; 101078075). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. This project has also received funding from the Israel Science Foundation (ISF, grant numbers 2549/19; 2250/22), the Tel Aviv University Center for AI and Data Science (TAD), the Len Blavatnik and the Blavatnik Family foundation, and from the Adelis Foundation.

References

N. Ailon, Z. S. Karnin, and T. Joachims. Reducing dueling bandits to cardinal bandits. In *ICML*, volume 32, pages 856–864, 2014.

V. Bengs, R. Busa-Fekete, A. El Mesaoudi-Paul, and E. Hüllermeier. Preference-based online learning with dueling bandits: A survey. *Journal of Machine Learning Research*, 2021.

A. Blum, M. Gupta, G. Li, N. S. Manoj, A. Saha, and Y. Yang. Dueling optimization with a monotone adversary. In *International Conference on Algorithmic Learning Theory*, 2024.

S. Bubeck. Convex optimization: Algorithms and complexity. *arXiv preprint arXiv:1405.4980*, 2014.

X. Chen, S. Gopi, J. Mao, and J. Schneider. Competitive analysis of the top-k ranking problem. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1245–1264. SIAM, 2017.

X. Chen, Y. Li, and J. Mao. A nearly instance optimal algorithm for top-k ranking under the multinomial logit model. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2504–2522. SIAM, 2018.

S. Ghoshal and A. Saha. Exploiting correlation to achieve faster learning rates in low-rank preference bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 456–482. PMLR, 2022.

E. Hazan. Introduction to online convex optimization. *arXiv preprint arXiv:1909.05207*, 2019.

K. G. Jamieson, R. Nowak, and B. Recht. Query complexity of derivative-free optimization. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2012.

Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

W. Ren, J. Liu, and N. B. Shroff. PAC ranking from pairwise and listwise queries: Lower bounds and upper bounds. *arXiv preprint arXiv:1806.02970*, 2018.

A. Saha and A. Gopalan. Battle of bandits. In *Uncertainty in Artificial Intelligence*, 2018.

A. Saha and A. Gopalan. Active ranking with subset-wise preferences. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019a.

A. Saha and A. Gopalan. PAC battling bandits in the plackett-luce model. In *Algorithmic Learning Theory*, pages 700–737, 2019b.

A. Saha and A. Gopalan. From pac to instance-optimal sample complexity in the plackett-luce model. In *International Conference on Machine Learning*, pages 8367–8376. PMLR, 2020a.

A. Saha and A. Gopalan. Best-item learning in random utility models with subset choices. In *International Conference on Artificial Intelligence and Statistics*, pages 4281–4291. PMLR, 2020b.

A. Saha and A. Krishnamurthy. Efficient and optimal algorithms for contextual dueling bandits under realizability. In *International Conference on Algorithmic Learning Theory*, pages 968–994. PMLR, 2022.

A. Saha, T. Koren, and Y. Mansour. Dueling convex optimization. In *International Conference on Machine Learning*, pages 9245–9254. PMLR, 2021.

Y. Sui, V. Zhuang, J. Burdick, and Y. Yue. Multi-dueling bandits with dependent arms. In *Conference on Uncertainty in Artificial Intelligence*, UAI'17, 2017.

Y. Sui, M. Zoghi, K. Hofmann, and Y. Yue. Advancements in dueling bandits. In *IJCAI*, pages 5502–5510, 2018.

H. Wu and X. Liu. Double Thompson sampling for dueling bandits. In *Advances in Neural Information Processing Systems*, pages 649–657, 2016.

Y. Wu. Packing, covering, and consequences on minimax risk. *Course Lecture notes for ECE598: Information-theoretic methods for high-dimensional statistics*, 2016.

Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1201–1208, 2009.

Checklist

The checklist follows the references. For each question, choose your answer from the three possible options: Yes, No, Not Applicable. You are encouraged to include a justification to your answer, either by referencing the appropriate section of your paper or providing a brief inline description (1-2 sentences). Please do not modify the questions. Note that the Checklist section does not count towards the page limit. Not including the checklist in the first submission won't result in desk rejection, although in such case we will ask you to upload it during the author response period and include it in camera ready (if accepted).

In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes/No/Not Applicable] -Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes/No/Not Applicable] -Yes
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes/No/Not Applicable] -Yes
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes/No/Not Applicable] -Yes
   (b) Complete proofs of all theoretical results. [Yes/No/Not Applicable] -Yes
   (c) Clear explanations of any assumptions. [Yes/No/Not Applicable] -Yes
3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes/No/Not Applicable] -Yes
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes/No/Not Applicable] -Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes/No/Not Applicable] -Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes/No/Not Applicable] -Yes

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes/No/Not Applicable] -Not Applicable
   (b) The license information of the assets, if applicable. [Yes/No/Not Applicable] -Not Applicable
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/Not Applicable] -Not Applicable
   (d) Information about consent from data providers/curators. [Yes/No/Not Applicable] -Not Applicable
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/Not Applicable] -Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Yes/No/Not Applicable] -Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/Not Applicable] -Not Applicable
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/Not Applicable] -Not Applicable

Supplementary: Faster Convergence with Multiway Preferences

A Appendix for Section 3

A.1 Proof of Theorem 3

Theorem 3 (Convergence Analysis of Algorithm 1 for β-Smooth Functions)**.** *Consider f to be β smooth. Suppose Algorithm 1 is run with $\eta = \frac{m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ and $T_\epsilon = O\Big(\frac{d\beta D}{m\epsilon}\Big)$, where $\|\mathbf{w}_1 - \mathbf{x}^*\|^2 \le D$ (is an assumed known upper bound). Then Algorithm 1 returns $\mathbf{E}[f(\tilde{\mathbf{w}}_{T+1})] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $2T_\epsilon$, for any $m \le d$.*

Proof of Theorem 3. We start by noting that in this case:

$$\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \le \|\tilde{\mathbf{w}}_{t+1} - \mathbf{x}^*\|^2 = \|\mathbf{w}_t - \frac{\eta}{m}\sum_{i=1}^{m}\mathbf{g}_t^i - \mathbf{x}^*\|^2, \tag{1}$$

where the first inequality holds since projection reduces the distance to optimal $\mathbf{x}^*$. This further leads to

$$\begin{aligned}
m^2\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 &= \sum_{i=1}^{m}\|\mathbf{w}_t - \eta\mathbf{g}_t^i - \mathbf{x}^*\|^2 + 2\sum_{1\le i<j\le K}\big(\mathbf{w}_t - \eta\mathbf{g}_t^i - \mathbf{x}^*\big)^\top\big(\mathbf{w}_t - \eta\mathbf{g}_t^i - \mathbf{x}^*\big)\\
&= m(\|\mathbf{w}_t - \mathbf{x}^*\|^2 + \eta^2) - 2\eta\sum_{i=1}^{m}(\mathbf{w}_t - \mathbf{x}^*)^\top\mathbf{g}_t^i - 2\eta\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\big(\mathbf{w}_t - \mathbf{x}^*\big)^\top(\mathbf{g}_t^i + \mathbf{g}_t^j) + 2\frac{m(m-1)}{2}\|\mathbf{w}_t - \mathbf{x}^*\|^2\\
&\qquad + 2\eta\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}{\mathbf{g}_t^i}^\top\mathbf{g}_t^j.
\end{aligned}$$

Let us denote by $\mathcal{H}_t$ the history $\{\mathbf{w}_\tau, U_\tau, \mathbf{o}_\tau\}_{\tau=1}^{t-1} \cup \mathbf{w}_t$ till time t. Then conditioning on the history $\mathcal{H}_t$ till time t, and taking expectation over $U_t := \{\mathbf{u}_t^1, \ldots, \mathbf{u}_t^m\}$ we further get:

$$\begin{aligned}
&m^2\mathbf{E}_{U_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t]\\
&= m(\|\mathbf{w}_t - \mathbf{x}^*\|^2 + \eta^2) - 2\eta\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\big(\mathbf{w}_t - \mathbf{x}^*\big)^\top\mathbf{E}_{U_t}[(\mathbf{g}_t^i + \mathbf{g}_t^j) \mid \mathcal{H}_t] + 2\frac{m(m-1)}{2}\|\mathbf{w}_t - \mathbf{x}^*\|^2\\
&\qquad - 2\eta\sum_{i=1}^{m}(\mathbf{w}_t - \mathbf{x}^*)^\top\mathbf{g}_t^i + 2\eta^2\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\mathbf{E}_{U_t}[{\mathbf{g}_t^i}^\top\mathbf{g}_t^j \mid \mathcal{H}_t].
\end{aligned}$$

Let us denote by $n_t = \frac{\nabla f(\mathbf{w}_t)}{\|\nabla f(\mathbf{w}_t)\|}$ the normalized gradient at point $\mathbf{w}_t$. Now note for any $i \neq j$, since $\mathbf{u}_t^i$ and $\mathbf{u}_t^j$ are independent, from Theorem 21 we get:

$$\begin{aligned}
\mathbf{E}_{U_t}[{\mathbf{g}_t^i}^\top\mathbf{g}_t^j \mid \mathcal{H}_t] &= \mathbf{E}_{\mathbf{u}_i}[\mathbf{E}_{\mathbf{u}_j}[{\mathbf{g}_t^i}^\top\mathbf{g}_t^j \mid \mathbf{u}_t^i] \mid \mathcal{H}_t]\\
&\le \mathbf{E}_{\mathbf{u}_i}\bigg[{\mathbf{g}_t^i}^\top\frac{n_t}{\sqrt{d}} + 2\lambda_t \mid \mathcal{H}_t\bigg]\\
&\le \frac{1}{\sqrt{d}}\bigg(\frac{n_t^\top n_t}{\sqrt{d}}\bigg) + 4\lambda_t = \frac{1}{d} + 4\lambda_t.
\end{aligned}$$

where recall from Theorem 21 and Lemma 22, $\lambda_t \le \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\bigg(1 + 2\sqrt{\log\frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\bigg)$. Combining this with the main equation, and further applying Theorem 21, we get:

$$
\begin{aligned}
& m^2\mathbf{E}_{U_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2 \mid \mathcal{H}_t] \\
& \le m(\|\mathbf{w}_t-\mathbf{x}^*\|^2+\eta^2) - 2\eta\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\big(\mathbf{w}_t-\mathbf{x}^*\big)^\top \mathbf{E}_{U_t}[(\mathbf{g}_t^i+\mathbf{g}_t^j)\mid \mathcal{H}_t] + 2\frac{m(m-1)}{2}\|\mathbf{w}_t-\mathbf{x}^*\|^2 \\
& \qquad\qquad - 2\eta\sum_{i=1}^{m}(\mathbf{w}_t-\mathbf{x}^*)^\top\mathbf{g}_t^i + 2\eta^2\frac{m(m-1)}{2}\left(\frac{1}{d}+4\lambda_t\right), \\
& \le m(\|\mathbf{w}_t-\mathbf{x}^*\|^2+\eta^2) - 2m\eta\|\mathbf{w}_t-\mathbf{x}^*\|\sum_{i=1}^{m}\frac{\big(\mathbf{w}_t-\mathbf{x}^*\big)^\top}{\|\mathbf{w}_t-\mathbf{x}^*\|}\mathbf{E}_{U_t}[\mathbf{g}_t^i\mid\mathcal{H}_t] + 2\frac{m(m-1)}{2}\|\mathbf{w}_t-\mathbf{x}^*\|^2 \\
& \qquad\qquad + 2\eta^2\frac{m(m-1)}{2}\left(\frac{1}{d}+4\lambda_t\right), \\
& = m(\|\mathbf{w}_t-\mathbf{x}^*\|^2+\eta^2) - 2m\eta\|\mathbf{w}_t-\mathbf{x}^*\|\sum_{i=1}^{m}[\frac{\big(\mathbf{w}_t-\mathbf{x}^*\big)^\top}{\|\mathbf{w}_t-\mathbf{x}^*\|}\mathbf{n}_t+2\lambda_t] + 2\frac{m(m-1)}{2}\|\mathbf{w}_t-\mathbf{x}^*\|^2 \\
& \qquad\qquad + 2\eta^2\frac{m(m-1)}{2}\left(\frac{1}{d}+4\lambda_t\right), \\
& = m(\|\mathbf{w}_t-\mathbf{x}^*\|^2+\eta^2) - 2m\eta\|\mathbf{w}_t-\mathbf{x}^*\|\sum_{i=1}^{m}[\frac{\big(\mathbf{w}_t-\mathbf{x}^*\big)^\top}{\|\mathbf{w}_t-\mathbf{x}^*\|}\mathbf{n}_t+2\lambda_t] + 2\frac{m(m-1)}{2}\|\mathbf{w}_t-\mathbf{x}^*\|^2 \\
& \qquad\qquad + 2\eta^2\frac{m(m-1)}{2}\left(\frac{1}{d}+4\lambda_t\right), \\
& = m^2\|\mathbf{w}_t-\mathbf{x}^*\|^2 + \eta^2(m+\frac{m(m-1)}{d}) - 2\eta m^2[(\mathbf{w}_t-\mathbf{x}^*)^\top\mathbf{n}_t \\
& \qquad\qquad\qquad\qquad + 4\eta^2m^2\lambda_t + 4m^2\eta\|\mathbf{w}_t-\mathbf{x}^*\|\lambda_t, \\
& = m^2\|\mathbf{w}_t-\mathbf{x}^*\|^2 + \eta^2(m+\frac{m(m-1)}{d}) - 2\eta m^2[(\mathbf{w}_t-\mathbf{x}^*)^\top\mathbf{n}_t \\
& \qquad\qquad\qquad\qquad + 4m^2\sqrt{d}\eta\|\mathbf{w}_t-\mathbf{x}^*\|\lambda_t + 4m^2\sqrt{d}\eta\|\mathbf{w}_t-\mathbf{x}^*\|\lambda_t
\end{aligned}
$$

where the last inequality follows by a choice of η such that $\frac{\eta}{\|\mathbf{w}_t-\mathbf{x}^*\|\sqrt{d}} \le 1$ (will see shortly below why this is true). Further from Claim-2 of Lemma 19, and from the fact that $m < d$, we can derive:

$$
m^2\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2 \mid \mathcal{H}_t] \le m^2\|\mathbf{w}_t-\mathbf{x}^*\|^2 + m^2(-2\eta\frac{c\sqrt{2\epsilon}}{\sqrt{d\beta}} + 8\eta\lambda_t\sqrt{d}\|\mathbf{w}_t-\mathbf{x}^*\|) + 2m\eta^2.
$$

Now, similar to the derivation followed in Saha et al. [2021] (see proof of Lem 6, Saha et al. [2021]), choosing $\gamma \le \frac{\|\nabla f(\mathbf{w}_t)\|}{960\beta d\sqrt{d}\|\mathbf{w}_t-\mathbf{x}^*\|\sqrt{\log 480}}\sqrt{\frac{2\epsilon}{\beta}}$, we can get:

$$
\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2] \mid \mathcal{H}_t] \le \|\mathbf{w}_t-\mathbf{x}^*\|^2 - \frac{\eta\sqrt{2\epsilon}}{10\sqrt{d\beta}} + \frac{\eta\sqrt{2\epsilon}}{20\sqrt{d\beta}} + \frac{2\eta^2}{m}.
$$

One possible choice of γ is $\gamma = \frac{\epsilon^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ (since $\|\nabla f(\mathbf{x})\| \ge \frac{\epsilon}{D}$ for any $\mathbf{x}$ s.t. $f(\mathbf{x}) - f(\mathbf{x}^*) > \epsilon$ by Lemma 20). Then following from the above equation, we further get:

$$
\begin{aligned}
\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2 \mid \mathcal{H}_t]] &\le \|\mathbf{w}_t-\mathbf{x}^*\|^2 - \eta\frac{\sqrt{2\epsilon}}{20\sqrt{d\beta}} + \frac{2\eta^2}{m}, \\
&= \|\mathbf{w}_t-\mathbf{x}^*\|^2 - \frac{(\sqrt{2}-1)m\epsilon}{400d\beta}, \quad \left(\text{setting } \eta = \frac{m\sqrt{\epsilon}}{20\sqrt{d\beta}}\right)
\end{aligned}
$$

$$\implies \mathbf{E}_{\mathcal{H}_T}[\|\mathbf{w}_{T+1} - \mathbf{x}^*\|^2] \le \|\mathbf{w}_1 - \mathbf{x}^*\|^2 - \frac{(\sqrt{2}-1)m\epsilon T}{400d\beta},$$
$$\big(\text{summing over} t = 1, \ldots T \text{ and laws of iterated expectation}\big).$$

Above implies, if indeed $f(\mathbf{w}_\tau) - f(\mathbf{x}^*) > \epsilon$ continues to hold for all $\tau = 1, 2, \ldots T$, then $\mathbf{E}[\|\mathbf{w}_{T+1} - \mathbf{x}^*\|^2] \le 0$, for $T \ge \frac{400md\beta}{(\sqrt{2}-1)\epsilon}(\|\mathbf{w}_1 - \mathbf{x}^*\|^2)$, which basically implies $\mathbf{w}_{T+1} = \mathbf{x}^*$ (i.e. $f(\mathbf{w}_{T+1}) = f(\mathbf{x}^*)$). Otherwise there must have been a time $t \in [T]$ such that $f(\mathbf{w}_t) - f(\mathbf{x}^*) < \epsilon$.

The last bit of the proof lies in ensuring that indeed $\frac{\eta}{\|\mathbf{w}_t - \mathbf{x}^*\|\sqrt{d}} \le 1$ in all those rounds where $f(\mathbf{w}_{t+1}) - f(\mathbf{x}^*) > \epsilon$. This is easy to note given β-smoothness as:

$$\frac{\eta}{\|\mathbf{w}_t - \mathbf{x}^*\|\sqrt{d}} \le \frac{m\sqrt{\epsilon}}{20d\sqrt{\beta}\|\mathbf{w}_t - \mathbf{x}^*\|} \le \frac{m\sqrt{f(\mathbf{w}_t) - f(\mathbf{x}^*)}}{20d\sqrt{\beta}\|\mathbf{w}_t - \mathbf{x}^*\|} \le \frac{m\sqrt{\beta}\|\mathbf{w}_t - \mathbf{x}^*\|}{20d\sqrt{\beta}\|\mathbf{w}_t - \mathbf{x}^*\|} \le \frac{m}{20d} < 1.$$

This concludes the proof with $T_\epsilon = T$, which gives an $O(m)$-factor improvement over the convergence bounds with single-sign feedback (as derived in Theorem 14). ☐

A.2 Proof of Theorem 4

Theorem 4 (Convergence Analysis of Algorithm 2 for α-strongly convex and β-Smooth Functions)**.** *Consider f to be α-strongly convex and β-smooth. Then Algorithm 2 returns $\mathbf{E}[f(\mathbf{m}_\epsilon)] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $O\Big(\frac{d\beta}{m\alpha}(\log_2\big(\frac{\alpha}{\epsilon}\big) + \|\mathbf{x}_1 - \mathbf{x}^*\|^2)\Big)$, for any $m \le d$.*

Proof of Theorem 4. Let $\mathcal{H}_k := \{\mathbf{w}_{k'}, (\mathbf{w}_{t'}, U_{t'}, o_{t'})_{t' \in t_{k'}}\}_{k'=0}^{k} \cup \{\mathbf{w}_{k+1}\}$ denotes the complete history till the end of phase k for all $k \in [k_\epsilon]$. By Theorem 4 we know that, for any fixed $T > 0$, when Algorithm 1 is run with $\eta = \frac{m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{960\beta d\sqrt{d}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ and $\epsilon = \frac{400d\beta D}{(\sqrt{2}-1)T}$ ($D := \|\mathbf{w}_1 - \mathbf{w}^*\|^2$), Algorithm 1 returns

$$\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \le \epsilon = \frac{400d\beta\|\mathbf{w}_1 - \mathbf{x}^*\|^2}{(\sqrt{2}-1)T}$$

with sample complexity (number of pairwise comparisons) $2T$.

However, in this case since f is also α-strongly convex Lemma 10 further implies

$$\mathbf{E}[{}^{\alpha}/_{2}\|\mathbf{m}_{T+1} - \mathbf{x}^*\|^2] \le \mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \le \frac{400d\beta\|\mathbf{w}_1 - \mathbf{x}^*\|^2}{(\sqrt{2}-1)T} \tag{2}$$
$$\implies \mathbf{E}[\|\mathbf{m}_{T+1} - \mathbf{x}^*\|^2] \le \frac{800d\beta\|\mathbf{w}_1 - \mathbf{x}^*\|^2}{(\sqrt{2}-1)\alpha T}$$

Now initially for $k = 1$, clearly applying the above result for $T = t\|\mathbf{w}_1 - \mathbf{x}^*\|^2$, we get

$$\mathbf{E}[\|\mathbf{w}_2 - \mathbf{x}^*\|^2] \le \frac{800d\beta\|\mathbf{w}_1 - \mathbf{x}^*\|^2}{(\sqrt{2}-1)\alpha T} = 1$$

Thus, for any $k = 2, 3, \ldots k_\epsilon - 1$, given the initial point $\mathbf{w}_k$, if we run Algorithm 3 with $T = 2t = \frac{1600d\beta}{(\sqrt{2}-1)\alpha}$, we get from (2)

$$\mathbf{E}_{\mathcal{H}_k}[\|\mathbf{w}_{k+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_{k-1}] \le \frac{800d\beta\|\mathbf{w}_k - \mathbf{x}^*\|^2}{(\sqrt{2}-1)\alpha T} = \frac{\|\mathbf{w}_k - \mathbf{x}^*\|^2}{2}$$

This implies given the history till phase $k-1$, using Equation (2) and our choice of t_k,

$$\mathbf{E}_{\mathcal{H}_k}[f(\mathbf{w}_{k+1}) - f(\mathbf{x}^*) \mid \mathcal{H}_{k-1}] \le \mathbf{E}_{\mathcal{H}_k}[\frac{1}{4\alpha}\|\mathbf{w}_k - \mathbf{x}^*\|^2 \mid \mathcal{H}_{k-1}] \le \frac{1}{4\alpha}(\frac{1}{2})^{k-1}\|\mathbf{w}_1 - \mathbf{x}^*\|^2 \le \frac{1}{\alpha 2^{k+1}}.$$

Thus, to ensure at $k = k_\epsilon$, $\mathbf{E}[f(\mathbf{w}_{k_\epsilon+1}) - f(\mathbf{x}^*)] \le \epsilon$, this demands $(1/2)^{k_\epsilon+1}\alpha \le \epsilon$, or equivalently $\frac{\alpha}{2\epsilon} \le 2^{k_\epsilon+1}$, which justifies the choice of $k_\epsilon = \log_2\left(\frac{\alpha}{\epsilon}\right)$. By Theorem 3, recall running the subroutine B-NGD$(\mathbf{w}_k, \eta_k, \gamma_k, t_k, m)$ actually requires a query complexity of $2t_k = 4t$, and hence the total query complexity (over k_ϵ phases) of Algorithm 3 becomes $4tk_\epsilon + t_1 = O\Big(\frac{800d\beta}{(\sqrt{2}-1)\alpha}(\log_2\left(\frac{\alpha}{\epsilon}\right) + D)\Big)$, where recall $D := \|\mathbf{w}_1 - \mathbf{x}^*\|^2$. ☐

Lemma 10 ([Hazan, 2019, Bubeck, 2014]). *If $f : \mathcal{D} \mapsto \mathbb{R}$ is an α-strongly convex function, with $\mathbf{x}^*$ being the minimizer of f. Then for any $\mathbf{x} \in \mathcal{R}$, $\frac{\alpha}{2}\|\mathbf{x}^* - \mathbf{x}\|^2 \le f(\mathbf{x}) - f(\mathbf{x}^*)$.*

Proof. This simply follows by the properties of α-strongly convex function. Note by definition of α-strong convexity, for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}$,

$$f(\mathbf{x}) - f(\mathbf{y}) \ge \nabla f(\mathbf{y})^\top(\mathbf{x} - \mathbf{y}) + \frac{\alpha}{2}\|\mathbf{x} - \mathbf{y}\|^2.$$

The proof simply follows setting $\mathbf{y} = \mathbf{x}^*$. ☐

B Appendix for Section 4

B.1 Proof of Theorem 7

Theorem 7 (Convergence Analysis of Algorithm 3 for β-Smooth Functions). *Consider f to be β smooth. Suppose Alg. 3 is run with $\eta = \frac{\ell_m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{960\beta d\ell_m\sqrt{d\ell_m}D^2\,\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ and $T_\epsilon = O\Big(\frac{d\beta D}{\epsilon\ell_m}\Big)$, where $\|\mathbf{w}_1 - \mathbf{x}^*\|^2 \le D$, $\ell_m = \lfloor \log m \rfloor \le d$. Then Algorithm 3 returns $\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \le \epsilon$ with sample complexity $2T_\epsilon$.*

Proof of Theorem 7. We start by noting that at each round t, the the algorithm receives the winner feedback $o_t = \arg\min(f(\mathbf{x}_t^1), f(\mathbf{x}_t^2), \ldots, f(\mathbf{x}_t^{\tilde{m}}))$, upon playing the subset $S_t = \{\mathbf{x}_t^1, \mathbf{x}_t^2, \ldots, \mathbf{x}_t^{\tilde{m}}\}$, we recall $\tilde{m} = 2^{\ell_m}$.

Moreover by definition, $f(\mathbf{x}_t^{o_t}) < f(\mathbf{x}_t^i)$, $\forall i \ne o^t, i \in [\ell_m]$, and hence $\text{sign}\big(f(\mathbf{x}_t^{o_t}) - f(\mathbf{x}_t^i)\big) = -1$ for all i. Now let us denote by $\mathbf{y}_t^i = \mathbf{w}_t + \gamma U_t \mathbf{v}_i'$, where $v_i' = -v_i$ and $v_j' = v_j, \forall j \in [\ell_m]$. Note $\mathbf{v}_i' \in \mathcal{N}(\mathbf{v}, \mathcal{G})$, i.e. $\mathbf{v}_i'$ is a neighboring node of $\mathbf{v}$ in the graph $\mathcal{G}(V_{\ell_m})$, and also $\mathbf{y}_t^i \in S_t$ by construction. Hence

$$\text{sign}\big(f(\mathbf{x}_t^{o_t}) - f(\mathbf{y}_t^i)\big) = -1.$$

Combining the above and the definition of $\mathbf{g}_t^i$, this actually implies

$$\text{sign}\big(f(\mathbf{x}_t^{o_t}) - f(\mathbf{y}_t^i)\big)\mathbf{u}_t^i = v_i\mathbf{u}_t^i = g_t^i.$$

But then by Theorem 11 we have, for any d-dimensional unit norm vector $\mathbf{b} \in \mathcal{S}_d(1)$:

$$\frac{c}{\sqrt{d}}\frac{\nabla f(\mathbf{w})^\top}{\|\nabla f(\mathbf{w})\|}\mathbf{b} - 2\lambda_t \le \mathbf{E}_{\mathbf{u}_t^i}[-v_i{\mathbf{u}_t^i}^\top\mathbf{b}] \le \frac{c}{\sqrt{d}}\frac{\nabla f(\mathbf{w})^\top}{\|\nabla f(\mathbf{w})\|}\mathbf{b} + 2\lambda_t, \tag{3}$$

$\lambda_t \le \frac{\beta\gamma\sqrt{d\ell_m}}{\|\nabla f(\mathbf{w}_t)\|}\left(1 + 2\sqrt{\log\frac{\|\nabla f(\mathbf{w}_t)\|}{\beta\gamma\sqrt{d\ell_m}}}\right)$.

Now note, by the update rule:

$$\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \le \|\tilde{\mathbf{w}}_{t+1} - \mathbf{x}^*\|^2 = \|\mathbf{w}_t - \frac{\eta}{\ell_m}\sum_{i=1}^{\ell_m}\mathbf{g}_t^i - \mathbf{x}^*\|^2,$$

since we had $\mathbf{g}_t := \frac{1}{\ell_m}\sum_{i=1}^{\ell_m}\mathbf{g}_t^i$ and the first inequality holds since projection reduces the distance to optimal $\mathbf{x}^*$. Note the update of $\mathbf{w}_{t+1}$ appears to be the same update by ℓ_m-batched sign feedback. Then following the same derivation of Theorem 3 (as it proceeds from Equation (1) in the proof of Theorem 3), with $\eta = \frac{\ell_m\sqrt{\epsilon}}{20\sqrt{d\beta}}$, and $\gamma = \frac{\epsilon^{3/2}}{960\beta d\ell_m\sqrt{d\ell_m}D^2\,\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ yields the desired result. ☐

B.2 Key Lemmas to Prove Theorem 7

Theorem 11. *Let f is β-smooth function. Assume $\mathbf{u}_t^1, \mathbf{u}_t^2, \ldots \mathbf{u}_t^{\ell_m} \overset{iid}{\sim}$ Unif$(\mathcal{S}_d(\frac{1}{\sqrt{\ell_m}}))$. Denote $U_t := [\mathbf{u}_t^1, \ldots, \mathbf{u}_t^{\ell_m}] \in \mathbb{R}^{d\times \ell_m}$. Let $\mathbf{x} = \mathbf{w} + \gamma U_t \mathbf{v}$ for any $\mathbf{v} \in V_{\ell_m}$, and $\mathbf{y} = \mathbf{w} + \gamma U_t \mathbf{v}'$, for any $\mathbf{v}' \in \mathcal{N}(\mathbf{v}, \mathcal{G})$, i.e. $\mathbf{v}'$ is any neighboring node of $\mathbf{v}$ in the graph $\mathcal{G}(V_{\ell_m})$. In particular, let $v_i' = -v_i$ and $v_j' = v_j, \forall j \in [\ell_m]$. Then*

$$\frac{c}{\sqrt{d}}\frac{\nabla f(\mathbf{w})^\top}{\|\nabla f(\mathbf{w})\|}\mathbf{b} - 2\lambda \leq \mathbf{E}_{\mathbf{u}_i}[\mathit{sign}(f(\mathbf{x}) - f(\mathbf{y}))\mathbf{u}_i^\top \mathbf{b}] \leq \frac{c}{\sqrt{d}}\frac{\nabla f(\mathbf{w})^\top}{\|\nabla f(\mathbf{w})\|}\mathbf{b} + 2\lambda,$$

for some universal constant $c \in [\frac{1}{20}, 1]$, $\lambda \leq \frac{\beta\gamma\sqrt{d\ell_m}}{\|\nabla f(\mathbf{w})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{w})\|}{\beta\gamma\sqrt{d\ell_m}}}\right)$ and $\mathbf{b} \in S_d(1)$ being any unit vector of dimension d.

The proof of Theorem 11 follows combining the results of Lemma 12 and Lemma 13 (from [Saha et al., 2021]), proved below.

Lemma 12. *Let f is β-smooth function. Assume $\mathbf{u}_t^1, \mathbf{u}_t^2, \ldots \mathbf{u}_t^{\ell_m} \overset{iid}{\sim}$ Unif$(\mathcal{S}_d(\frac{1}{\sqrt{\ell_m}}))$. Denote $U_t := [\mathbf{u}_t^1, \ldots, \mathbf{u}_t^{\ell_m}] \in \mathbb{R}^{d\times \ell_m}$. Let $\mathbf{x} = \mathbf{w} + \gamma U_t \mathbf{v}$ for any $\mathbf{v} \in V_{\ell_m}$, and $\mathbf{y} = \mathbf{w} + \gamma U_t \mathbf{v}'$, for any $\mathbf{v}' \in \mathcal{N}(\mathbf{v}, \mathcal{G})$, i.e. $\mathbf{v}'$ is any neighboring node of $\mathbf{v}$ in the graph $\mathcal{G}(V_{\ell_m})$. In particular, let $v_i' = -v_i$ and $v_j' = v_j, \forall j \in [\ell_m]$. Then for any unit dimension d vector $\mathbf{b} \in \mathcal{S}_d(1)$ we have:*

$$\left|\mathbf{E}_{\mathbf{u}_i}[\mathit{sign}(f(\mathbf{x}) - f(\mathbf{y}))\mathbf{u}_i^\top \mathbf{b}] - \mathbf{E}_{\mathbf{u}_i}[\mathit{sign}(\nabla f(\mathbf{w}) \cdot \mathbf{u}_i)\mathbf{u}_i^\top \mathbf{b}]\right| \leq 2\lambda,$$

where $\lambda \leq \frac{\beta\gamma\sqrt{d\ell_m}}{\|\nabla f(\mathbf{w})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{w})\|}{\beta\gamma\sqrt{d\ell_m}}}\right)$.

Proof. Without loss of generality, assume $\mathbf{v} = (1, 1, \ldots, 1) \in \{0, 1\}^{\ell_m}$ and $\mathbf{v}' = (-1, 1, \ldots, 1)$, i.e. $i = 1$. Thus $\mathbf{x} = \mathbf{w} + \gamma(\mathbf{u}_1 + \mathbf{u}_2 + \ldots + \mathbf{u}_{\ell_m})$, and $\mathbf{y} = \mathbf{w} + \gamma(-\mathbf{u}_1 + \mathbf{u}_2 + \ldots + \mathbf{u}_{\ell_m})$. Also let us denote by $\mathbf{u} = (\mathbf{u}_1 + \mathbf{u}_2 + \ldots + \mathbf{u}_{\ell_m})$, $\mathbf{u}' = (-\mathbf{u}_1 + \mathbf{u}_2 + \ldots + \mathbf{u}_{\ell_m})$.

From smoothness we have

$$\gamma\mathbf{u}_i \cdot \nabla f(\mathbf{w}) - \tfrac{1}{2}\beta\gamma^2 \leq f(\mathbf{w} + \gamma\mathbf{u}) - f(\mathbf{w}) \leq \gamma\mathbf{u} \cdot \nabla f(\mathbf{w}) + \tfrac{1}{2}\beta\gamma^2;$$
$$\gamma\mathbf{u}' \cdot \nabla f(\mathbf{w}) - \tfrac{1}{2}\beta\gamma^2 \leq f(\mathbf{w} + \gamma\mathbf{u}') - f(\mathbf{w}) \leq \gamma\mathbf{u}' \cdot \nabla f(\mathbf{w}) + \tfrac{1}{2}\beta\gamma^2.$$

Subtracting the inequalities, we get

$$|f(\mathbf{w} + \gamma\mathbf{u}) - f(\mathbf{w} + \gamma\mathbf{u}') - 2\gamma\mathbf{u}_1 \cdot \nabla f(\mathbf{x})| \leq \beta\gamma^2.$$

Therefore, if $\beta\gamma^2 \leq \gamma|\mathbf{u}_1 \cdot \nabla f(\mathbf{w})|$, we will have that $\text{sign}(f(\mathbf{w} + \gamma\mathbf{u}) - f(\mathbf{w} + \gamma\mathbf{u}')) = \text{sign}(\mathbf{u}_1 \cdot \nabla f(\mathbf{w}))$. Let us analyse $\Pr_{\mathbf{u}_1}(\beta\gamma \geq |\mathbf{u}_1 \cdot \nabla f(\mathbf{w})|)$. We know for $\mathbf{v} \sim \mathcal{N}(\mathbf{0}_d, \mathcal{I}_d)$, $\tilde{\mathbf{v}} := \mathbf{v}/\|\mathbf{v}\| \sim \mathcal{S}_d(1)$, i.e. $\tilde{\mathbf{v}}$ is uniformly distributed on the unit sphere, and hence $\dfrac{\mathbf{v}}{\|\mathbf{v}\|\sqrt{\ell_m}} \sim \mathcal{S}_d(1/\sqrt{\ell_m})$. Then can write:

$$\begin{aligned}
\mathbf{P}_{\mathbf{u}_1}\big(|\mathbf{u}_1 \cdot \nabla f(\mathbf{w})| \leq \beta\gamma\big) &= \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v} \cdot \nabla f(\mathbf{w})| \leq \beta\gamma\|\mathbf{v}\|\sqrt{\ell_m}\big)\\
&\leq \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v} \cdot \nabla f(\mathbf{w})| \leq 2\beta\gamma\sqrt{d\ell_m \log(1/\gamma')}\big) + \mathbf{P}_{\mathbf{v}}(\|\mathbf{v}\| \geq 2\sqrt{d\ell_m \log(1/\gamma')})\\
&\leq \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v} \cdot \nabla f(\mathbf{w})| \leq 2\beta\gamma\sqrt{d\ell_m \log(1/\gamma')}\big) + \gamma',
\end{aligned}$$

where the final inequality is since $\mathbf{P}_{\mathbf{v}}(\|\mathbf{v}\|^2 \leq 2d\ell_m \log(1/\gamma')) \geq 1 - \gamma'$ for any γ' (see Lemma 23). On the other hand, since $\mathbf{v} \cdot \nabla f(\mathbf{w}) \sim \mathcal{N}(0, \|\nabla f(\mathbf{w})\|^2)$, we have for any $a > 0$ that

$$\Pr(|\mathbf{v} \cdot \nabla f(\mathbf{w})| \leq a) \leq \frac{2a}{\|\nabla f(\mathbf{w})\|\sqrt{2\pi}} \leq \frac{a}{\|\nabla f(\mathbf{w})\|}.$$

Setting, $a = 2\beta\gamma\sqrt{d\ell_m \log(1/\gamma')}$, and combining the inequalities, we have that $\text{sign}(f(\mathbf{w} + \gamma\mathbf{u}) - f(\mathbf{w} - \gamma\mathbf{u})) = \text{sign}(\mathbf{u} \cdot \nabla f(\mathbf{w}))$ except with probability at most

$$\inf_{\gamma' > 0}\left\{\gamma' + \frac{2\beta\gamma\sqrt{d\ell_m \log(1/\gamma')}}{\|\nabla f(\mathbf{w})\|}\right\} = \lambda \ \text{(say)},$$

and further choosing $\gamma' = \frac{\beta\gamma\sqrt{d\ell_m}}{\|\nabla f(\mathbf{w})\|}$, we get that $\lambda \leq \frac{\beta\gamma\sqrt{d\ell_m}}{\|\nabla f(\mathbf{w})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{w})\|}{\beta\gamma\sqrt{d\ell_m}}}\right)$. As for the claim about the expectation, note that for any vector $\mathbf{b} \in \mathcal{S}_d(1)$,

$$\left|\mathbf{E}_{\mathbf{u}_1}[\text{sign}(f(\mathbf{w}+\gamma\mathbf{u}) - f(\mathbf{w}+\gamma\mathbf{u}'))\mathbf{u}_1{}^\top\mathbf{b}] - \mathbf{E}_{\mathbf{u}_1}[\text{sign}(\nabla f(\mathbf{w})\cdot\mathbf{u}_1)\mathbf{u}_1{}^\top\mathbf{b}]\right| \leq 2\lambda,$$

as with probability $1-\lambda$ the two expectations are equal, and otherwise, they differ by at most 2. □

Lemma 13 (Saha et al. [2021])**.** *For a given vector $\mathbf{g} \in \mathbb{R}^d$ and a random unit vector $\mathbf{u}$ drawn uniformly from $\mathcal{S}_d(1)$, we have*

$$\mathbf{E}[\textit{sign}(\mathbf{g}\cdot\mathbf{u})\mathbf{u}] = \frac{c}{\sqrt{d}}\frac{\mathbf{g}}{\|\mathbf{g}\|},$$

for some universal constant $c \in [\frac{1}{20}, 1]$.

This proof is same as the proof in Saha et al. [2021]. Without loss of generality we can assume $\|\mathbf{g}\| = 1$, since one can divide by $\|\mathbf{g}\|$ in both side of Lem. 13 without affecting the claim. Now to bound $\mathbf{E}[|\mathbf{g}\cdot\mathbf{u}|]$, note that since $\mathbf{u}$ is drawn uniformly from $\mathcal{S}_d(1)$, by rotation invariance this equals $\mathbf{E}[|u_1|]$. For an upper bound, observe that by symmetry $\mathbf{E}[u_1^2] = \frac{1}{d}\mathbf{E}[\sum_{i=1}^d u_i^2] = \frac{1}{d}$ and thus

$$\mathbf{E}[|u_1|] \leq \sqrt{\mathbf{E}[u_1^2]} = \frac{1}{\sqrt{d}}.$$

We turn to prove a lower bound on $\mathbf{E}[|\mathbf{g}\cdot\mathbf{u}|]$. If $\mathbf{u}$ were a Gaussian random vector with i.i.d. entries $u_i \sim \mathcal{N}(0, 1/d)$, then from standard properties of the (truncated) Gaussian distribution we would have gotten that $\mathbf{E}[|u_1|] = \sqrt{2/\pi d}$. For $\mathbf{u}$ uniformly distributed on the unit sphere, u_i is distributed as $v_1/\|\mathbf{v}\|$ where $\mathbf{v}$ is Gaussian with i.i.d. entries $\mathcal{N}(0, 1/d)$. We then can write

$$\begin{aligned}\Pr\left(|u_1| \geq \frac{\epsilon}{\sqrt{d}}\right) = \Pr\left(\frac{|v_1|}{\|\mathbf{v}\|} \geq \frac{\epsilon}{\sqrt{d}}\right) &\geq \Pr\left(|v_1| \geq \frac{1}{\sqrt{d}} \text{ and } \|\mathbf{v}\| \leq \frac{1}{\epsilon}\right) \\ &\geq 1 - \Pr\left(|v_1| < \frac{1}{\sqrt{d}}\right) - \Pr\left(\|\mathbf{v}\| > \frac{1}{\epsilon}\right).\end{aligned}$$

Since $\sqrt{d}v_1$ is a standard Normal, we have

$$\Pr\left(|v_1| < \frac{1}{\sqrt{d}}\right) = \Pr\left(-1 < \sqrt{d}v_1 < 1\right) = 2\Phi(1) - 1 \leq 0.7,$$

and since $\mathbf{E}[\|\mathbf{v}\|^2] = 1$ an application of Markov's inequality gives

$$\Pr\big(\|\mathbf{v}\| > \frac{1}{\epsilon}\big) = \Pr\big(\|\mathbf{v}\|^2 > \frac{1}{\epsilon^2}\big) \leq \epsilon^2\mathbf{E}[\|\mathbf{v}\|^2] = \epsilon^2.$$

For $\epsilon = \frac{1}{4}$ this implies that $\Pr\big(|u_1| \geq 1/4\sqrt{d}\big) \geq \frac{1}{5}$, whence $\mathbf{E}[|\mathbf{g}\cdot\mathbf{u}|] = \mathbf{E}[|u_1|] \geq 1/20\sqrt{d}$. □

B.3 Proof of Theorem 8

Theorem 8 (Improved Convergence Rate for α-strongly convex and β-Smooth Functions)**.** *Consider f to be α-strongly convex and β-smooth and let $\ell_m = \lfloor \log m \rfloor \leq d$. Then* Improved Battling-NGD *returns an ϵ-optimal point within $O\Big(\frac{d\beta}{\alpha\ell_m}(\log_2\big(\frac{\alpha}{\epsilon}\big) + \|\mathbf{x}_1 - \mathbf{x}^*\|^2)\Big)$ many multiwise queries.*

Proof of Theorem 8. The proof follows from the exactly same analysis as the proof of Theorem 4. □

Algorithm 4 Improved Battling-NGD (with Strong Convexity)

Input: Error tolerance $\epsilon > 0$, Batch size m
Initialize Initial point: $\mathbf{w}_1 \in \mathbb{R}^d$ such that $D := \|\mathbf{w}_1 - \mathbf{x}^*\|^2$ (assume known).
Phase counts $k_\epsilon := \lceil \log_2\left(\frac{\alpha}{\epsilon}\right) \rceil$, $t \leftarrow \frac{800d\beta}{(\sqrt{2}-1)\alpha}$
$\eta_1 \leftarrow \frac{m\sqrt{\epsilon_1}}{20\sqrt{d\beta}}, \epsilon_1 = \frac{400d\beta D}{(\sqrt{2}-1)t_1} = 1, t_1 = t\|\mathbf{w}_1 - \mathbf{x}^*\|^2$
$\gamma_1 \leftarrow \frac{\epsilon_1^{3/2}}{960\beta d\ell_m\sqrt{d\ell_m}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$.
Update $\mathbf{w}_2 \leftarrow$ **Battling-NGD**$\big(\mathbf{w}_1, \eta_1, \gamma_1, t_1, m\big)$
for $k = 2, 3, \ldots, k_\epsilon$ **do**
    $\eta_k \leftarrow \frac{m\sqrt{\epsilon_k}}{20\sqrt{d\beta}}, \epsilon_k = \frac{400d\beta}{(\sqrt{2}-1)t_k}, t_k = 2t$
    $\gamma_k \leftarrow \frac{\epsilon_k^{3/2}}{960\beta d\ell_m\sqrt{d\ell_m}D^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$.
    Update $\mathbf{w}_{k+1} \leftarrow$ **Battling-NGD**$\big(\mathbf{w}_k, \eta_k, \gamma_k, t_k, m\big)$
end for
Return $\mathbf{m}_\epsilon = \mathbf{w}_{k_\epsilon+1}$

C Projected Dueling Convex Optimization with Single Sign Feedback

Main Idea: Estimating Gradient Directions (Normalized Gradients): The algorithmic idea of our proposed algorithm is almost the same as what was proposed in Saha et al. [2021]. Essentially, we start with any arbitrary 'current estimate' of the function minimizer $\mathbf{w}_1 \in \mathcal{D}$, and at any round t, we compute normalized gradient estimate $\mathbf{g}_t$ such that $\mathbf{g}_t := o_t\mathbf{u}_t$, $\mathbf{u}_t \sim \text{Unif}(\mathcal{S}_d(1))$ being any random unit direction in $\mathbb{R}^d$ and $o_t = \text{sign}\Big(f(\mathbf{x}_t) - f(\mathbf{y}_t)\Big)$ is the sign feedback of the queried duel (x_t, y_t) at round t, such that $\mathbf{x}_t = \mathbf{w}_t + \gamma\mathbf{u}_t$, and $\mathbf{y}_t = \mathbf{w}_t - \gamma\mathbf{u}_t$, γ being any tunable perturbation step size. Subsequently, the algorithm takes a step along the estimated descent direction $\mathbf{g}_t$ with (tunable) step-size η and updates the current estimate $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t - \eta\mathbf{g}_t$ and repeat up to any given number of T steps before outputting the final estimate of the minimizer $\mathbf{w}_{T+1}$.

C.1 Algorithm Design: Projected Normalized Gradient Descent (P-NGD)

In this section, we analyzed the *Sign-Feedback-Optimization* problem for a bounded decision space $\mathcal{D}$. We describe a normalized gradient descent based algorithm for the purpose and anlyzed its convergence guarantee in Theorem 14. It's important to note that the same problem was analyzed in [Saha et al., 2021], but their analysis was limited to limited unbounded decision spaces only, which is unrealistic for practical problems and also led to more complication tuning of the learning parameters $\gamma > 0$ and $\eta > 0$.

Algorithm 5 Projected Normalized Gradient Descent (P-NGD)

Input: Initial point: $\mathbf{w}_1 \in \mathcal{D}$, Initial distance: D s.t. $D \geq \|\mathbf{w}_1 - \mathbf{x}^*\|^2$, Learning rate η, Perturbation parameter γ, Query budget T
Initialize Current minimum $\mathbf{m}_1 = \mathbf{w}_1$
for $t = 1, 2, 3, \ldots, T$ **do**
    Sample $\mathbf{u}_t \sim \text{Unif}(\mathcal{S}_d(1))$
    Set $\mathbf{x}_t := \mathbf{w}_t + \gamma\mathbf{u}_t,\ \mathbf{y}_t := \mathbf{w}_t - \gamma\mathbf{u}_t$
    Play the duel $(\mathbf{x}_t, \mathbf{y}_t)$, and observe $o_t \in \pm 1$ such that $o_t = \text{sign}\big(f(\mathbf{x}_t) - f(\mathbf{y}_t)\big)$.
    Update $\tilde{\mathbf{w}}_{t+1} \leftarrow \mathbf{w}_t - \eta\mathbf{g}_t$, where $\mathbf{g}_t = o_t\mathbf{u}_t$
    Project $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{D}}\|\mathbf{w} - \tilde{\mathbf{w}}_{t+1}\|$
    Query the pair $(\mathbf{m}_t, \mathbf{w}_{t+1})$ and receive $\text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big)$.
    Update $\mathbf{m}_{t+1} \leftarrow \begin{cases} \mathbf{m}_t & \text{if } \text{sign}\big(f(\mathbf{m}_t) - f(\mathbf{w}_{t+1})\big) < 0 \\ \mathbf{w}_{t+1} & \text{otherwise} \end{cases}$
end for
Return $\mathbf{m}_{T+1}$

Algorithm description: P-NGD Our algorithm follows the same strategy same as the β-NGD (Algorithm 1) of [Saha et al., 2021] modulo a projection step (see Line 8) which we had to incorporate for assuming bounded decision space $\mathcal{D}$: The main idea is to estimate gradient direction of any point $\mathbf{w} \in \mathcal{D}$ (normalized gradient estimate) querying the sign feedback of two symmetrically opposite points $(\mathbf{w} + \gamma \mathbf{u}, \mathbf{w} - \gamma \mathbf{u})$, $\mathbf{u} \sim \text{Unif}(\mathcal{S}_d(1))$ being any random unit direction, and simply take a 'small-enough' step (η) in the opposite direction of the estimated gradient. More formally, at each round t the algorithm maintains a current point $\mathbf{w}_t$, initialized to any random point $\mathbf{w}_1 \in \mathcal{D}$, and query two symmetrically opposite points $(\mathbf{w}_t + \gamma \mathbf{u}_t, \mathbf{w}_t - \gamma \mathbf{u}_t)$ along a random unit direction $\mathbf{u}_t \sim \text{Unif}(\mathcal{S}_d(1))$. Following this it finds a normalized gradient estimate at $\mathbf{w}_t$, precisely $g_t = o_t \mathbf{u}_t$ based on Theorem 21, where $o_t = \text{sign}\big(f(\mathbf{w}_t + \gamma \mathbf{u}_t) - f(\mathbf{w}_t + \gamma \mathbf{u}_t)\big)$ is the sign feedback of the queried duel (x_t, y_t). Subsequently, we update the running prediction using a (normalized) gradient descent step: $\tilde{\mathbf{w}}_{t+1} \leftarrow \mathbf{w}_t - \eta \mathbf{g}_t$ followed by a projection $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w} \in \mathcal{D}} \|\mathbf{w} - \tilde{\mathbf{w}}_{t+1}\|$. The algorithm also maintains a running minimum $\mathbf{m}_t$ which essentially keeps track of $\min\{\mathbf{w}_1, \ldots, \mathbf{w}_t\}$. The complete algorithm is given in Algorithm 5.

Theorem 14 (Convergence Analysis of Algorithm 5 for β-Smooth Functions)**.** *Consider f to be β smooth, and the desired accuracy level (suboptimality gap) is given to be $\epsilon > 0$. Then if Alg. 5 is run with $\eta = \frac{\sqrt{\epsilon}}{20\sqrt{d\beta}}$, $\gamma = \frac{\epsilon^{3/2}}{480\beta d D^2 \sqrt{\log 480}} \sqrt{\frac{2}{\beta}}$ and $T = T_\epsilon = O\Big(\frac{d\beta D}{m\epsilon}\Big)$, $D \geq \|\mathbf{w}_1 - \mathbf{x}^*\|^2$ being any upper bound on the initial distance from the optimal, Alg. 5 returns an ϵ-optimal point in at most $2T_\epsilon$ pairwise queries; i.e.*

$$\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \leq \epsilon.$$

Proof of Theorem 14. The proof idea crucially relies on Lemma 15, which essentially shows that if we start from an initial point $(\mathbf{w}_1)$, which is more than ϵ-suboptimal, i.e. $f(\mathbf{w}_1) - f(\mathbf{w}^*) > \epsilon$, then we will have $\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \leq \epsilon$.

The formal statement is as follows:

Lemma 15. *Consider f is β smooth. Then in Alg. 5, if the initial point $\mathbf{w}_1$ is such that $f(\mathbf{w}_1) - f(\mathbf{x}^*) > \epsilon$ (for $\epsilon > 0$), and the tuning parameters T, γ and η is as in defined in Theorem 14, we will have $\mathbf{E}[f(\mathbf{m}_{T+1})] - f(\mathbf{x}^*) \leq \epsilon$.*

Given Lemma 15, the statement of Theorem 14 follows straightforwardly: Note $\mathbf{m}_t$ essentially keeps track of $\min_{t \in [T]} \mathbf{w}_t$. Now either $\mathbf{w}_1$ is such that $f(\mathbf{w}_1) - f(\mathbf{x}^*) < \epsilon$, in case the bound of Theorem 14 is trivially true as by definition $f(\mathbf{m}_{T+1}) \leq f(\mathbf{w}_1)$. On the other hand, if $f(\mathbf{w}_1) - f(\mathbf{x}^*) > \epsilon$, bound of Theorem 14 follows by Lemma 15. We discuss the proof of Lemma 15 below. $\square$

Proof of Lemma 15. Our main claim lies in showing that at any round t, if the iterate is at least ϵ away from the optional, i.e. $f(\mathbf{w}_t) - f(\mathbf{x}^*) > \epsilon$, then on expectation $\mathbf{w}_{t+1}$ comes closer to $\mathbf{x}^*$ (in ℓ_2 distance), compared to $\mathbf{w}_t$. The following argument proves it formally:

Consider any $t = 1, 2, \ldots T$, such that $f(\mathbf{w}_t) > f(\mathbf{x}^*) + \epsilon$. Let us denote by $n_t = \frac{\nabla f(\mathbf{w}_t)}{\|\nabla f(\mathbf{w}_t)\|}$, the normalized gradient at point $\mathbf{w}_t$. Now from the update rule, we get that:

$$\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \leq \|\tilde{\mathbf{w}}_{t+1} - \mathbf{x}^*\|^2 \leq \|\mathbf{w}_t - \mathbf{x}^*\|^2 - 2\eta \mathbf{g}_t^\top (\mathbf{w}_t - \mathbf{x}^*) + \eta^2.$$

where the first inequality holds since projection reduces distance to optimal $\mathbf{x}^*$. Let us denote by $\mathcal{H}_t$ the history $\{\mathbf{w}_\tau, \mathbf{u}_\tau, \mathbf{m}_\tau\}_{\tau=1}^{t-1} \cup \mathbf{w}_t$ till time t. Then conditioning on the history $\mathcal{H}_t$ till time t, and taking expectation over $\mathbf{u}_t$ we further get:

$$\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t] \leq \mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_t - \mathbf{x}^*\|^2 \mid \mathcal{H}_t] - 2\eta \mathbf{E}_{\mathbf{u}_t}[\mathbf{g}_t^\top \mid \mathcal{H}_t](\mathbf{w}_t - \mathbf{x}^*) + \eta^2,$$

Further applying Theorem 21, one can get:

$$\begin{aligned}
\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1} - \mathbf{x}^*\|^2 \mid \mathcal{H}_t] &\leq \|\mathbf{w}_t - \mathbf{x}^*\|^2 - 2\eta \bigg(\mathbf{E}_{\mathbf{u}_t}[\text{sign}(\nabla f(\mathbf{w}_t) \cdot \mathbf{u}) \mathbf{u}^\top \frac{(\mathbf{w}_t - \mathbf{x}^*)}{\|\mathbf{w}_t - \mathbf{x}^*\|}] - 2\lambda_t \bigg) \|\mathbf{w}_t - \mathbf{x}^*\| + \eta^2, \\
&\leq \|\mathbf{w}_t - \mathbf{x}^*\|^2 - 2\eta \frac{c}{\sqrt{d}} \mathbf{n}_t^\top (\mathbf{w}_t - \mathbf{x}^*) + 4\eta\lambda_t \|\mathbf{w}_t - \mathbf{x}^*\| + \eta^2,
\end{aligned}$$

where recall from Theorem 21 and Lemma 22, $\lambda_t \le \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\left(1+2\sqrt{\log\frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\right)$. Further from Claim-2 of Lemma 19, we have:

$$\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2 \mid \mathcal{H}_t] \le \|\mathbf{w}_t-\mathbf{x}^*\|^2 - 2\eta\frac{c\sqrt{2\epsilon}}{\sqrt{d\beta}} + 4\eta\lambda_t\|\mathbf{w}_t-\mathbf{x}^*\| + \eta^2,$$

Now, similar to the derivation followed in Saha et al. [2021] (see proof of Lem 6, Saha et al. [2021]), choosing $\gamma \le \frac{\|\nabla f(\mathbf{w}_t)\|}{480\beta d\|\mathbf{w}_t-\mathbf{x}^*\|\sqrt{\log 480}}\sqrt{\frac{2\epsilon}{\beta}}$, we can get:

$$\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2] \mid \mathcal{H}_t] \le \mathbf{E}_{\mathcal{H}_t}[\|\mathbf{w}_t-\mathbf{x}^*\|^2] - \frac{\eta\sqrt{2\epsilon}}{10\sqrt{d\beta}} + \frac{\eta\sqrt{2\epsilon}}{20\sqrt{d\beta}} + \eta^2,$$

so one possible choice of γ is $\gamma = \frac{\epsilon^{3/2}}{480\beta dD^2\sqrt{\log 480}}\sqrt{\frac{2}{\beta}}$ (since $\|\nabla f(\mathbf{x})\| \ge \frac{\epsilon}{D}$ for any $\mathbf{x}$ s.t. $f(\mathbf{x})-f(\mathbf{x}^*) > \epsilon$). Then following from the above equation, we further get:

$$\begin{aligned}
&\mathbf{E}_{\mathcal{H}_t}[\mathbf{E}_{\mathbf{u}_t}[\|\mathbf{w}_{t+1}-\mathbf{x}^*\|^2 \mid \mathcal{H}_t]] \le \mathbf{E}_{\mathcal{H}_t}[\|\mathbf{w}_t-\mathbf{x}^*\|^2] - \eta\frac{\sqrt{2\epsilon}}{20\sqrt{d\beta}} + \eta^2,\\
&= \mathbf{E}_{\mathcal{H}_t}[\|\mathbf{w}_t-\mathbf{x}^*\|^2] - \frac{(\sqrt{2}-1)\epsilon}{400d\beta}, \quad \left(\text{since } \eta = \frac{\sqrt{\epsilon}}{20\sqrt{d\beta}}\right)\\
\implies &\mathbf{E}_{\mathcal{H}_T}[\|\mathbf{w}_{T+1}-\mathbf{x}^*\|^2] \le \|\mathbf{w}_1-\mathbf{x}^*\|^2 - \frac{(\sqrt{2}-1)\epsilon T}{400d\beta}, \ \left(\text{summing } t=1,\ldots T \text{ and laws of iterated expectation}\right)
\end{aligned}$$

Above implies, if indeed $f(\mathbf{w}_\tau)-f(\mathbf{x}^*) > \epsilon$ continues to hold for all $\tau = 1,2,\ldots T$, then $\mathbf{E}[\|\mathbf{w}_{T+1}-\mathbf{x}^*\|^2] \le 0$, for $T \ge \frac{400d\beta}{(\sqrt{2}-1)\epsilon}(\|\mathbf{w}_1-\mathbf{x}^*\|^2)$, which basically implies $\mathbf{w}_{T+1} = \mathbf{x}^*$ (i.e. $f(\mathbf{w}_{T+1}) = f(\mathbf{x}^*)$). Otherwise there must have been a time $t \in [T]$ such that $f(\mathbf{w}_t)-f(\mathbf{x}^*) < \epsilon$. This concludes the proof with $T_\epsilon = T$. □

D Some Useful Results on Convex Functions

Definition 16 (Convex Function)**.** *Assume $\mathcal{D} \subseteq \mathbb{R}^d$ be any convex and bounded decision space. Then any differential function $f : \mathcal{D} \mapsto \mathbb{R}$ is called convex if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$,*

$$f(\mathbf{x}) - f(\mathbf{y}) \ge \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}).$$

Definition 17 (β-Smooth Convex Function)**.** *Assume $\mathcal{D} \subseteq \mathbb{R}^d$ be any convex and bounded decision space. Then any differential and convex function $f : \mathcal{D} \mapsto \mathbb{R}$ is also called β-smooth (any $\beta > 0$) if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$,*

$$f(\mathbf{x}) - f(\mathbf{y}) \le \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\beta}{2}\|\mathbf{x}-\mathbf{y}\|^2.$$

Definition 18 (α-Strongly Convex Function)**.** *Assume $\mathcal{D} \subseteq \mathbb{R}^d$ be any convex and bounded decision space. Then any differential and convex function $f : \mathcal{D} \mapsto \mathbb{R}$ is also called α-strongly convex (any $\alpha > 0$) if for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$,*

$$f(\mathbf{x}) - f(\mathbf{y}) \ge \nabla f(\mathbf{y})^\top(\mathbf{x}-\mathbf{y}) + \frac{\alpha}{2}\|\mathbf{x}-\mathbf{y}\|^2.$$

Lemma 19. *Suppose $f : \mathcal{D} \mapsto \mathbb{R}$ is a convex function such that $f(\mathbf{y}) < f(\mathbf{x})$. Then $\left(\frac{\nabla f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|_2}\right)^\top(\mathbf{y}-\mathbf{x}) \le 0$. Further if $\mathbf{z}$ is a point such that $f(\mathbf{z}) - f(\mathbf{x}^*) > \epsilon$, then one can show that $-\frac{\nabla f(\mathbf{z})}{\|\nabla f(\mathbf{z})\|}^\top(z-\mathbf{x}^*) \le -\sqrt{\frac{2\epsilon}{\beta}}$.*

Proof. **Proof of Claim-1:** To show the first part of the claim, note that since f is convex,

$$f(\mathbf{y}) \ge f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \implies \nabla f(\mathbf{x})^\top(\mathbf{y}-\mathbf{x}) \le f(\mathbf{y}) - f(\mathbf{x}) \le 0,$$

which proves the claim by dividing both sides with $\|\nabla f(\mathbf{x})\|_2$.

Proof of Claim-2: Assume another point $\tilde{\mathbf{z}} := \mathbf{x}^* + \sqrt{\frac{2\epsilon}{\beta}}\mathbf{n}_z$, where we denote by $n_{\mathbf{z}} := \frac{\nabla f(\mathbf{z})}{\|\nabla f(\mathbf{z})\|}$. Now using β-smoothness of f we have: $f(\tilde{\mathbf{z}}) \le f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)(\tilde{\mathbf{z}} - \mathbf{x}^*) + \frac{\beta}{2}\|\tilde{\mathbf{z}} - \mathbf{x}^*\|^2 = f(\mathbf{x}^*) + \epsilon$. Thus we have $f(\tilde{\mathbf{z}}) < f(\mathbf{x}^*) + \epsilon < f(\mathbf{z})$, and hence from Lemma 19, we get $\mathbf{n}_{\mathbf{z}}^\top(\tilde{\mathbf{z}} - \mathbf{z}) \le 0$. But note this further implies $\mathbf{n}_{\mathbf{z}}^\top\left(\mathbf{x}^* + \sqrt{\frac{2\epsilon}{\beta}}\mathbf{n}_{\mathbf{z}} - \mathbf{z}\right) \le 0 \implies -\mathbf{n}_{\mathbf{z}}^\top(\mathbf{z} - \mathbf{x}^*) \le -\sqrt{\frac{2\epsilon}{\beta}}$. □

Lemma 20. *Suppose $f : \mathcal{D} \mapsto \mathbb{R}$ is a convex function for some convex set $\mathcal{D} \subseteq \mathbb{R}^d$ such that for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, $f(\mathbf{x}) - f(\mathbf{y}) > \epsilon$. Then this implies $\|\nabla f(\mathbf{x})\| > \frac{\epsilon}{\|\mathbf{x}-\mathbf{y}\|}$. Further, assuming $D := \max_{\mathbf{x},\mathbf{y} \in \mathbb{R}^d} \|\mathbf{x} - \mathbf{y}\|_2$, we get $\|\nabla f(\mathbf{x})\| > \frac{\epsilon}{D}$ for any $\mathbf{x} \in \mathcal{D}$.*

Proof. The proof simply follows using convexity of f as:

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{y}) > \epsilon &\implies \epsilon < f(\mathbf{x}) - f(\mathbf{y}) \le \nabla f(\mathbf{x})(\mathbf{x} - \mathbf{y}) \le \|\nabla f(\mathbf{x})\|_2 \|\mathbf{x} - \mathbf{y}\|_2 \\ &\implies \|\nabla f(\mathbf{x})\| \ge \frac{\epsilon}{\|\mathbf{x} - \mathbf{y}\|}. \end{aligned}$$

□

As shown in Saha et al. [2021], using the above result one can obtain the normalized gradient estimate of f at any given point $\mathbf{x}$, as described below:

Theorem 21 (Adapted from Saha et al. [2021] with Slight Modifications)**.** *If f is β-smooth, for any $\mathbf{u} \sim$ Unif$(\mathcal{S}_d(1))$, $\delta \in (0, 1)$ and vector $\mathbf{b} \in \mathcal{S}_d(1)$:*

$$\mathbf{E}_{\mathbf{u}}[\mathit{sign}(f(\mathbf{x} + \delta\mathbf{u}) - f(\mathbf{x} - \delta\mathbf{u}))\mathbf{u}^\top\mathbf{b}] \le \frac{c}{\sqrt{d}} \frac{\nabla f(\mathbf{x})^\top}{\|\nabla f(\mathbf{x})\|}\mathbf{b} + 2\lambda,$$

for some universal constant $c \in [\frac{1}{20}, 1]$, and $\lambda \le \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\right)$.

Proof of Theorem 21. The proof mainly lies on the following lemma that shows how to the comparison feedback of two close points, $\mathbf{x} + \gamma\mathbf{u}$ and $\mathbf{x} - \gamma\mathbf{u}$, can be used to recover a directional information of the gradient of f at point $\mathbf{x}$.

Lemma 22. *If f is β-smooth, for any $\mathbf{u} \sim$ Unif$(\mathcal{S}_d(1))$, and $\gamma \in (0, 1)$, then with probability at least $1 - \lambda$ where $\lambda = \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\left(1 + 2\sqrt{\log \frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\right)$, we have*

$$\mathit{sign}(f(\mathbf{x} + \gamma\mathbf{u}) - f(\mathbf{x} - \gamma\mathbf{u}))\mathbf{u} = \mathit{sign}(\nabla f(\mathbf{x}) \cdot \mathbf{u})\mathbf{u}.$$

Consequently, for any vector $\mathbf{b} \in \mathcal{S}_d(1)$ we have $\left|\mathbf{E}_{\mathbf{u}}[\mathit{sign}(f(\mathbf{x}+\gamma\mathbf{u}) - f(\mathbf{x}-\gamma\mathbf{u}))\mathbf{u}^\top\mathbf{b}] - \mathbf{E}_{\mathbf{u}}[\mathit{sign}(\nabla f(\mathbf{x}) \cdot \mathbf{u})\mathbf{u}^\top\mathbf{b}]\right| \le 2\lambda$.

Remark 3 (Ensuring λ denotes a valid probability)**.** *It is important and assuring to note that when for any $\mathbf{x} \in \mathcal{D}$ such that $f(\mathbf{x}) - f(\mathbf{x}^*) > \epsilon$ (which in turn implies $\|\nabla f(\mathbf{x})\| \ge \frac{\epsilon}{D}$ by Lemma 20), $\lambda \in [0, 1]$ for any choice of $\gamma \in [0, \frac{\epsilon}{\beta D\sqrt{d}}]$ (Note we respect this in our choice of γ for the algorithm guarantees, e.g. Theorem 14, Theorem 3, etc).*

The result of Thm. 21 now simply follows by combining the guarantees of Lem. 13 and 22. □

Proof of Lemma 22. From smoothness we have

$$\begin{aligned} \gamma\mathbf{u} \cdot \nabla f(\mathbf{x}) - \tfrac{1}{2}\beta\gamma^2 &\le f(\mathbf{x} + \gamma\mathbf{u}) - f(\mathbf{x}) \le \gamma\mathbf{u} \cdot \nabla f(\mathbf{x}) + \tfrac{1}{2}\beta\gamma^2; \\ -\gamma\mathbf{u} \cdot \nabla f(\mathbf{x}) - \tfrac{1}{2}\beta\gamma^2 &\le f(\mathbf{x} - \gamma\mathbf{u}) - f(\mathbf{x}) \le -\gamma\mathbf{u} \cdot \nabla f(\mathbf{x}) + \tfrac{1}{2}\beta\gamma^2. \end{aligned}$$

Subtracting the inequalities, we get

$$|f(\mathbf{x} + \gamma\mathbf{u}) - f(\mathbf{x} - \gamma\mathbf{u}) - 2\gamma\mathbf{u} \cdot \nabla f(\mathbf{x})| \le \beta\gamma^2.$$

Therefore, if $\beta\gamma^2 \le \gamma|\mathbf{u}\cdot\nabla f(\mathbf{x})|$, we will have that $\text{sign}(f(\mathbf{x}+\gamma\mathbf{u}) - f(\mathbf{x}-\gamma\mathbf{u})) = \text{sign}(\mathbf{u}\cdot\nabla f(\mathbf{x}))$. Let us analyse $\Pr_{\mathbf{u}}(\beta\gamma \ge |\mathbf{u}\cdot\nabla f(\mathbf{x})|)$. We know for $\mathbf{v} \sim \mathcal{N}(\mathbf{0}_d, \mathcal{I}_d)$, $\mathbf{u} := \mathbf{v}/\|\mathbf{v}\|$ is uniformly distributed on the unit sphere. Then can write:

$$\begin{aligned}
\mathbf{P}_{\mathbf{u}}\big(|\mathbf{u}\cdot\nabla f(\mathbf{x})| \le \beta\gamma\big) &= \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v}\cdot\nabla f(\mathbf{x})| \le \beta\gamma\|\mathbf{v}\|\big)\\
&\le \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v}\cdot\nabla f(\mathbf{x})| \le 2\beta\gamma\sqrt{d\log(1/\gamma')}\big) + \mathbf{P}_{\mathbf{v}}(\|\mathbf{v}\| \ge 2\sqrt{d\log(1/\gamma')})\\
&\le \mathbf{P}_{\mathbf{v}}\big(|\mathbf{v}\cdot\nabla f(\mathbf{x})| \le 2\beta\gamma\sqrt{d\log(1/\gamma')}\big) + \gamma',
\end{aligned}$$

where the final inequality is since $\mathbf{P}_{\mathbf{v}}(\|\mathbf{v}\|^2 \le 2d\log(1/\gamma')) \ge 1-\gamma'$ for any γ' (see Lemma 23). On the other hand, since $\mathbf{v}\cdot\nabla f(\mathbf{x}) \sim \mathcal{N}(0, \|\nabla f(\mathbf{x})\|^2)$, we have for any $\gamma > 0$ that

$$\Pr(|\mathbf{v}\cdot\nabla f(\mathbf{x})| \le \gamma) \le \frac{2\gamma}{\|\nabla f(\mathbf{x})\|\sqrt{2\pi}} \le \frac{\gamma}{\|\nabla f(\mathbf{x})\|}.$$

Combining the inequalities, we have that $\text{sign}(f(\mathbf{x}+\gamma\mathbf{u}) - f(\mathbf{x}-\gamma\mathbf{u})) = \text{sign}(\mathbf{u}\cdot\nabla f(\mathbf{x}))$ except with probability at most

$$\inf_{\gamma'>0}\left\{\gamma' + \frac{2\beta\gamma\sqrt{d\log(1/\gamma')}}{\|\nabla f(\mathbf{x})\|}\right\} = \lambda(\text{say}),$$

and further choosing $\gamma' = \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{w})\|}$, we get that $\lambda \le \frac{\beta\gamma\sqrt{d}}{\|\nabla f(\mathbf{x})\|}\left(1 + 2\sqrt{\log\frac{\|\nabla f(\mathbf{x})\|}{\sqrt{d}\beta\gamma}}\right)$. As for the claim about the expectation, note that for any vector $\mathbf{b} \in \mathcal{S}_d(1)$,

$$\Big|\mathbf{E}_{\mathbf{u}}[\text{sign}(f(\mathbf{x}+\gamma\mathbf{u}) - f(\mathbf{x}-\gamma\mathbf{u}))\mathbf{u}^\top\mathbf{b}] - \mathbf{E}_{\mathbf{u}}[\text{sign}(\nabla f(\mathbf{x})\cdot\mathbf{u})\mathbf{u}^\top\mathbf{b}]\Big| \le 2\lambda,$$

since with probability $1-\lambda$ the two expectations are identical, and otherwise, they differ by at most 2. □

Lemma 23. *For $\mathbf{v} \sim \mathcal{N}(\mathbf{0}_d, \mathcal{I}_d)$ and any $\lambda > 0$, it holds that $\|\mathbf{v}\|^2 \le d + 4\log(1/\lambda)$ with probability at least $1-\lambda$.*

Proof. Let $X = \|\mathbf{v}\|^2$. Then X is distributed Chi-squared with d degrees of freedom, and so its moment generating function is $\mathbf{E}[e^{zX}] = (1-2z)^{-d/2}$ for $z < 1/2$. Using Markov's inequality we have, for all $t > 0$ and $0 < z < 1/2$,

$$\Pr(X \ge t) = \Pr(e^{zX} \ge e^{zt}) \le e^{-zt}\mathbf{E}[e^{zX}] = e^{-zt}(1-2z)^{-d/2} \le e^{-z(t-d)}.$$

Choosing $z = 1/4$ and $t = d + 4\log(1/\lambda)$ makes the right-hand side smaller than λ, as we require. □