We now show a simple lower bound for optimisation over non-convex objective functions. We reduce while there remains a small gap between our main lower bound of Theorem 3 and the deterministic quantised gradient descent of Section 6, we can show that the gap cannot be closed by improved deterministic algorithms where the coordinator learns value of objective function $F(x)$ in addition to the minimiser $x$. That is, our quantised gradient descent is the communication-optimal deterministic algorithm for variant (1) for objectives with constant condition number.

Recall that in the $N$-player equality over universe of size $d$, denoted by $\text{EQ}_{d,N}$, each player $i$ is given an input $b_i \in \{0, 1\}^d$, and the task is to decide if all players have the same input. That is, $\text{EQ}_{d,N}(b_1, \ldots, b_N) = 1$ if all inputs are equal, and 0 otherwise. It is known [33] that the deterministic communication complexity of $\text{EQ}_{d,N}$ is $C(\text{EQ}_{d,N}) = \Omega(Nd)$.

**Theorem 8.** Given parameters $N, d, \varepsilon, \beta_0$ and $\beta = \beta_0N$ satisfying $d\beta/\varepsilon = \Omega(1)$, any deterministic protocol solving (1) for quadratic input functions $x \mapsto \beta_0\|x - x_0\|^2$ has communication complexity $\Omega\left(Nd \log(\beta d/\varepsilon)\right)$, if the coordinator is also required to output estimate $r \in \mathbb{R}$ for the minimum function value such that $\sum_{i=1}^{N} f_i(z) \leq r \leq \sum_{i=1}^{N} f_i(z) + \varepsilon$.

Proof. Assume $\Pi$ is a deterministic protocol solving (1) with communication complexity $C_{\Pi}$. We show that $\Pi$ can then solve $N$-party equality over a universe of size $D = \Omega(d \log(\beta d/\varepsilon))$, implying

$$C_{\Pi} = \Omega(ND) = \Omega\left(Nd \log(\beta d/\varepsilon)\right).$$

More specifically, let $S$ be the set given by Lemma 2 with $\delta = (2\varepsilon/\beta)^{1/2}$, and let $D = \lceil \log |S| \rceil = \Theta(d \log(\beta d/\varepsilon))$. Note that since we assume $d\beta/\varepsilon = \Omega(1)$, the set $S$ has at least two elements and $D \geq 1$. For technical convenience, assume $|S| = 2^D$, and identify each binary string $b \in \{0, 1\}^D$ with an element $\tau(b) \in S$.

Next, assume that each node $i$ is given a binary string $b_i \in \{0, 1\}^D$ as input, and we want to compute $\text{EQ}_{D,N}(b_1, b_2, \ldots, b_N)$. The nodes simulate protocol $\Pi$ with input function $f_i$ for node $i$, where $f_i(x) = \beta_0\|x - \tau(b_i)\|^2$. Let us denote $F = \sum_{i=1}^{d} f_i$. Upon termination of the protocol, the coordinator learns a point $y \in [0, 1]^d$ satisfying $F(y) \leq F(x^*) + \varepsilon$ and an estimate $r \in \mathbb{R}$ satisfying $r \leq F(y) + \varepsilon$, where $x^*$ is the true global minimum. The coordinator can now adjudicate equality based on $F(y)$ as follows:

1. If all inputs $b_i$ are equal, then the functions $f_i$ are also equal, and $F(x^*) = 0$. In this case, we have $F(y) \leq 2\varepsilon$, and the coordinator outputs 1.

2. If there are nodes $i$ and $j$ such that $i \neq j$, then for all points $x \in [0, 1]^d$, we have $f_i(x) + f_j(x) > 2\varepsilon$ by the definition of $S$, and thus $F(x^*) > 2\varepsilon$. In this case, we have $r > 2\varepsilon$, and the coordinator outputs 0.

Since communication is only used for the simulation of $\Pi$, this computes $\text{EQ}_{D,N}(b_1, b_2, \ldots, b_N)$ with $C_{\Pi}$ total communication, completing the proof. \hfill $\square$

**B Lower bound for non-convex functions**

We now show a simple lower bound for optimisation over non-convex objective functions. We reduce from the $N$-player set disjointness over universe of size $d$, denoted by $\text{DISJ}_{d,N}$: each player $i$ is given an input $b_i \in \{0, 1\}^d$, and the coordinator needs to output 0 if there is a coordinate $\ell \in [d]$ such that $b_i(\ell) = 1$ for all $i \in \{N\}$, and 1 otherwise.

**Theorem 11 (6).** For $\delta > 0$, $N \geq 1$ and $d = \omega(\log N)$, the randomised communication complexity of set disjointness is $\text{RCC}(\text{DISJ}_{d,N}) = \Omega(Nd)$.

Again consider for fixed $\varepsilon, d$ and $\beta$ the set $S$ given by Lemma 2 with $\delta = 2\varepsilon/\beta$. This gives a set $S$ with size at least $(\beta d^{1/2}/2C\varepsilon)^d = \exp(\Omega(d \log(\beta d/\varepsilon)))$. Let us identify the points in $S$ with indices in $|S|$. For a binary string $b \in \{0, 1\}^{|S|}$, define the function $f_b$ by

$$f_b(x) = \begin{cases} \beta \|x - s\|^2 & \text{if } \|x - s\| < \varepsilon/\beta \\ \varepsilon & \text{otherwise} \end{cases}$$

for $s$ with $b_s = 1$, or $f_b(x) = 0$ otherwise.
Since the distance between points in $S$ is at least $2\varepsilon/\beta$, the functions $f_T$ are well-defined, continuous and $\beta$-Lipschitz.

**Theorem 12.** Given parameters $N$, $d$, $\varepsilon$ and $\beta$ satisfying $d\beta/\varepsilon = \Omega(1)$ and $(\beta d^{1/2}/2C\varepsilon)^d = \omega(\log N)$, any protocol solving $I$ with error probability $\delta > 0$ when the inputs are guaranteed to be functions $f_b$ for $b \in \{0, 1\}^{|S|}$ has communication complexity $N \exp(\Omega(d \log(\beta d)/\varepsilon))$.

**Proof.** Assume there is a protocol $\Pi$ with the properties stated in the claim, and worst-case communication cost $C_{11}$. We now show that we can use $\Pi$ to solve set disjointness over universe of size $|S|$ with $C_{11}$ total communication, which implies

$$C_{11} \geq \text{RCC}^3(\text{DISJ}_{|S|, N}) = \Omega(N \exp(\Omega(d \log(\beta d)/\varepsilon)))$$

yielding the claim.

First, we note that after running $\Pi$, the coordinator can send the final estimate $\hat{z}$ of the optimum to all nodes, and receive approximations of the local function values $f_i(z)$ from all nodes with additive $O(Nd \log d \beta/\varepsilon)$ overhead, e.g. using quantisation of Corollary 9. We can without loss of generality assume that this does not exceed the total communication cost of $\Pi$.

For $b_1, b_2, \ldots, b_N \in \{0, 1\}^{|S|}$ that all contain 1 in some position $s$, then we have $\sum_{i=1}^N f_{b_i}(x) = 0$. Otherwise, for any point $x \in [0, 1]^d$, consider the closest point $s \in S$ to $x$; there is at least one $b_i$ with $b_i = 0$, and for that function $f_{b_i}(x) = \varepsilon$ by definition. Thus, if $b_1, b_2, \ldots, b_N$ are a YES-instance for set disjointness, then $\inf_{x \in [0, 1]^d} \sum_{i=1}^N f_{b_i}(x) \geq \varepsilon$, and if $b_1, b_2, \ldots, b_N$ are a NO-instance, then $\inf_{x \in [0, 1]^d} \sum_{i=1}^N f_{b_i}(x) = 0$.

By definition, $\Pi$ can be used to distinguish between the two cases, and thus to solve set disjointness. 

\[ \square \]

### C Communication-optimal quantised gradient descent, full version

We now describe in detail our deterministic upper bound. Our algorithm uses quantised gradient descent, loosely following the outline of Magnússon et al. [25]. However, there are two crucial differences. First, we use a carefully-calibrated instance of the quantisation scheme of Davies, et al. [11] to remove a log $d$ factor from the communication cost, and second, we use two-step quantisation to avoid all-to-all communication.

**Preliminaries on gradient descent.** We will assume that the input functions $f_i : [0, 1]^d \to \mathbb{R}$ are $\alpha_0$-strongly convex and $\beta_0$-strongly smooth. This implies that $F = \sum_{i=1}^N f_i$ is $\alpha$-strongly convex and $\beta$-strongly smooth for $\alpha = N\alpha_0$ and $\beta = N\beta_0$. Consequently, the functions $f_i$ and $F$ have condition number bounded by $\kappa = \beta/\alpha$.

**Gradient descent** optimises the sum $\sum_{i=1}^N f_i(x)$ by starting from an arbitrary point $x^{(0)} \in [0, 1]^d$, and applying the update rule

$$x^{(t+1)} = x^{(t)} - \gamma \sum_{i=1}^N \nabla f_i(x^{(t)})$$

where $\gamma > 0$ is a parameter.

Let $x^*$ denote the global minimum of $F$. We use the following standard result on the convergence of gradient descent; see e.g. Bubeck [8].

**Theorem 13.** For $\gamma = 2/(\alpha + \beta)$, we have that $\|x^{(t+1)} - x^*\|_2 \leq \frac{\kappa_1}{\kappa+1} \|x^{(t)} - x^*\|_2$.

**Preliminaries on quantisation.** For compressing the gradients the nodes will send to coordinator, we use the recent quantisation scheme of Davies et al. [11]. Whereas the original uses randomised selection of the quantisation point to obtain an unbiased estimator, we can use a deterministic version that picks an arbitrary feasible quantisation point (e.g. the closest one). This gives the following guarantees:

**Corollary 14 ([11]).** Let $R$ and $\varepsilon$ be fixed positive parameters, and $q \in \mathbb{R}^d$ be an estimate vector, and $B \in \mathbb{N}$ be the number of bits used by the quantisation scheme. Then, there exists a deterministic quantisation scheme, specified by a function $Q_{\varepsilon, R} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$, an encoding function
enc_{ε,R}: \mathbb{R}^d \rightarrow \{0, 1\}^B, and a decoding function dec_{ε,R}: \mathbb{R}^d \times \{0, 1\}^B \rightarrow \mathbb{R}^d, with the following properties:

1. (Validity.) \ \text{dec}_{ε,R}(q, \text{enc}_{ε,R}(x)) = Q_{ε,R}(x, q) for all x, q \in \mathbb{R}^d with \|x - q\|_2 \leq R.

2. (Accuracy.) \ \|Q_{ε,R}(x, q) - x\|_2 \leq ε for all x, q \in \mathbb{R}^d with \|x - q\|_2 \leq R.

3. (Cost.) If ε = λR for any λ < 1, the bit cost of the scheme satisfies B = O(d \log \lambda^{-1}).

C.1 Algorithm description

We now describe the algorithm, and overview its guarantees. We assume that the constants \(\alpha\) and \(\beta\) are known to all nodes, so the parameters of the quantised gradient descent can be computed locally, and use \(W\) to be an upper bound on the diameter on the convex domain \(\mathbb{D}\), e.g. \(W = d^{1/2}\) if \(\mathbb{D} = [0, 1]^d\). We assume that the initial iterate \(x^{(0)}\) is arbitrary, but the same at all nodes, and set the initial quantisation estimate \(q^{(0)}_i\) at each \(i\) as the origin.

We define the following parameters for the algorithm. Let \(γ = 2/(\alpha + \beta)\) and \(ξ = \frac{\kappa - 1}{\kappa + 1}\) be the step size and convergence rate of gradient descent, and let \(W\) be such that \(\|x^{(0)} - x^*\| \leq W\). We define

\[μ = 1 - \frac{1}{\kappa + 1}, \quad \delta = ξ(1 - ξ)/4, \quad R^{(t)} = \frac{2β}{ξ}μ^tW,\]

where \(μ\) will be the convergence rate of our quantised gradient descent, and \(δ\) and \(R^{(t)}\) will be parameters controlling the quantisation at each step. For the purposes of analysis, we assume that \(κ ≥ 2\). Note that this implies that \(1/3 ≤ ξ < 1, μ < 1, \) and \(0 < δ < 1\).

The algorithm proceeds in rounds \(t = 1, 2, \ldots, T\). At the beginning of round \(t+1\), each node \(i\) knows the values of the iterate \(x^{(t)}\), the global quantisation estimate \(q^{(t)}\), and its local quantisation estimate \(q^{(t)}_i\) for \(i = 1, 2, \ldots, N\). At step \(t\), nodes perform the following steps:

1. Each node \(i\) updates its iterate as \(x^{(t+1)} = x^{(t)} - γq^{(t)}\).

2. Each node \(i\) computes its local gradient over \(x^{(t+1)}\), and transmits it in quantised form to the coordinator as follows. Let \(ε_1 = δR^{(t+1)}/2N\) and \(ρ_1 = R^{(t+1)}/N\).

   a. Node \(i\) computes \(\nabla f_i(x^{(t+1)})\) locally, and sends message \(m_i = \text{enc}_{ε_1,ρ_1}(\nabla f_i(x^{(t+1)}))\) to the coordinator.

   b. The coordinator receives messages \(m_i\) for \(i = 1, 2, \ldots, N\), and decodes them as \(q^{(t+1)}_i = \text{dec}_{ε_1,ρ_1}(q^{(t+1)}_i, m_i)\). The coordinator then computes \(x^{(t+1)} = \sum_{i=1}^{N} q^{(t+1)}_i\).

3. The coordinator sends the quantised sum of gradients to all other nodes as follows. Let \(ε_2 = δR^{(t+1)}/2\) and \(ρ_2 = 2R^{(t+1)}\).

   a. The coordinator sends the message \(m = \text{enc}_{ε_2,ρ_2}(r^{(t+1)})\) to each node \(i\).

   b. Each node decodes the coordinator’s message as \(q^{(t+1)} = \text{dec}_{ε_2,ρ_2}(q^{(t)}, m)\).

After round \(T\), all nodes know the final iterate \(x^{(T)}\).

C.2 Analysis

For simplicity, we will split the analysis into two parts. The first describes and analyses the algorithm in an abstract way; the second part describes the details of implementing it in the coordinator model. For technical convenience, assume \(κ ≥ 2\); for smaller condition numbers, we can run the algorithm with \(κ = 2\).

Convergence. Let \(γ = 2/(\alpha + \beta)\), let \(x^{(0)} \in [0, 1]^d\), \(q^{(0)}_i \in \mathbb{R}^d\) and \(q^{(0)}_i \in \mathbb{R}^d\) for \(i = 1, 2, \ldots, N\) be arbitrary initial values. From the algorithm description, we see that the update rule for our quantised gradient descent is

\[x^{(t+1)} = x^{(t)} - γq^{(t)}, \quad q^{(t+1)}_i = Q_{ε_1,ρ_1}(\nabla f_i(x^{(t+1)}), q^{(t)}_i) \quad \text{for} \quad ε_1 = δR^{(t+1)}/2N \quad \text{and} \quad ρ_1 = R^{(t+1)}/N,\]
\[ r^{(t+1)} = \sum_{i=1}^{N} q_i^{(t+1)}, \]
\[ q^{(t+1)} = Q_{\varepsilon_2, \rho_2} (r^{(t+1)}, d^{(t)}) \quad \text{for} \ \varepsilon_2 = \delta R^{(t+1)}/2 \text{and} \ \rho_2 = 2R^{(t+1)}. \]

**Lemma 15.** The inequalities
\[
\| x^{(t)} - x^* \|_2 \leq \mu t W, \quad \text{(Q1)} \\
\| \nabla f_i (x^{(t)}) - q_i^{(t)} \|_2 \leq \delta R^{(t)}/2N, \quad \text{(Q2)} \\
\| \nabla F (x^{(t)}) - q^{(t)} \|_2 \leq \delta R^{(t)} \quad \text{(Q3)}
\]
hold for all \( t \), assuming that they hold for \( x^{(0)}, q^{(0)} \) and \( q^{(0)}_i \) for \( i = 1, 2, \ldots, N \).

**Proof.** We apply induction over \( t \); we assume that the inequalities (Q1-Q3) hold for \( t \), and prove that they also hold for \( t + 1 \). Since we assume the inequalities hold for \( t = 0 \), the base case is trivial. By elementary computation, the following hold:
\[ 0 < \xi < 1, \quad 0 < \delta < 1, \quad 2\delta/\xi + \xi = \mu, \quad \gamma/\beta \leq 2, \quad \mu R^{(t)} = R^{(t+1)}. \]

**Convergence (Q1):** First, we observe that \( \frac{2\delta}{\xi} + \xi = \frac{2}{1+\xi} = 1 - \frac{1}{\alpha+1} = \mu \) and \( \gamma/\beta \leq 2 \). We now have that
\[ \| x^{(t+1)} - x^* \|_2 = \| x^{(t)} - q^{(t)} + \gamma \nabla F (x^{(t)}) - \gamma \nabla F (x^{(t)}) + x^* \|_2 \quad \text{(def.)} \\
\leq \| \gamma q^{(t)} - \gamma \nabla F (x^{(t)}) \|_2 + \| (x^{(t)} - \gamma \nabla F (x^{(t)})) - x^* \|_2 \quad \text{(triangle-i.e.)} \\
\leq \gamma \| \nabla F (x^{(t)}) - q^{(t)} \|_2 + \xi \| x^{(t)} - x^* \|_2 \quad \text{(norm, Thm. 13)} \\
\leq \gamma\delta R^{(t)} + \xi \mu W \quad \text{(by Q1, Q3 for) t} \\
= (\gamma\beta/\xi + \xi)\mu W \quad \text{(expand) R^{(t)})} \\
\leq (2\delta/\xi + \xi)\mu W = \mu^{t+1} W. \quad \text{(\( \gamma/\beta \leq 2 \))}
\]

**Local quantisation (Q2):** First, let us observe that to prove that (Q2) holds for \( t + 1 \), it is sufficient to show \( \| \nabla f_i (x^{(t+1)}) - q_i^{(t)} \|_2 \leq R^{(t+1)}/N \), as the claim then follows from the definition of \( q_i^{(t+1)} \) and Corollary 9. We have
\[ \| \nabla f_i (x^{(t+1)}) - q_i^{(t)} \|_2 = \| \nabla f_i (x^{(t+1)}) - \nabla f_i (x^{(t)}) + \nabla f_i (x^{(t)}) - q_i^{(t)} \|_2 \\
\leq \| \nabla f_i (x^{(t+1)}) - \nabla f_i (x^{(t)}) \|_2 + \| \nabla f_i (x^{(t)}) - q_i^{(t)} \|_2 \quad \text{(triangle-i.e.)} \\
\leq 2\beta \| x^{(t+1)} - x^{(t)} \|_2 + \delta R^{(t)}/N \quad \text{(smoothness, Q3)} \\
\leq 2\beta \| x^{(t+1)} - x^* \|_2 + \| x^{(t)} - x^* \|_2 + \delta R^{(t)}/N \quad \text{(triangle-i.e.)} \\
\leq 2\beta \mu W + \delta R^{(t)}/N \quad \text{(by Q1 for \( t, t+1 \))} \\
= \xi R^{(t)}/N + \delta R^{(t)}/N \quad \text{(\( \beta = \beta_0 N \))} \\
= (\xi + \delta) R^{(t)}/N \quad \text{(rearrange)} \\
\leq (\xi + 2\delta/\xi) R^{(t)}/N \quad \text{(\( 2/\xi \geq 1 \))} \\
= \mu R^{(t)}/N = R^{(t+1)}/N. \quad \text{(\( 2\delta/\xi + \xi = \mu \))}
\]

**Global quantisation (Q3):** To prove (Q3), we start by giving two auxiliary inequalities. First, we prove that \( \| \nabla F (x^{(t)}) - r^{(t+1)} \|_2 \leq \delta R^{(t+1)}/2: \)
\[ \| \nabla F (x^{(t+1)}) - r^{(t+1)} \|_2 = \| \sum_{i=1}^{N} \nabla f_i (x^{(t+1)}) - \sum_{i=1}^{N} q_i^{(t+1)} \|_2 \quad \text{(def.)} \\
\leq \sum_{i=1}^{N} \| \nabla f_i (x^{(t+1)}) - q_i^{(t+1)} \|_2 \quad \text{(triangle-i.e.)} 
\]
where the last inequality follows from smoothness of $F$. To get $O(d)$, as the claim then follows from the definition of $q^{(t+1)}$ and Corollary 9. We have

\begin{align*}
\|r^{(t+1)} - q^{(t)}\|_2 &= \|r^{(t+1)} + \nabla F(x^{(t+1)}) - \nabla F(x^{(t+1)}) - q^{(t)}\|_2 \\
&\leq \|r^{(t+1)} - \nabla F(x^{(t+1)})\|_2 + \|\nabla F(x^{(t+1)}) - q^{(t)}\|_2 \\
&\leq \delta R^{(t+1)}/2 + \beta \|x^{(t+1)} - x^{(t)}\|_2 + \delta R^{(t)} ,
\end{align*}

where the last inequality follows from smoothness of $F$, equation (Q2) for $t + 1$ and equation (Q3) for $t$. It holds that

\begin{align*}
\beta \|x^{(t+1)} - x^{(t)}\|_2 + \delta R^{(t)} &\leq \beta (\|x^{(t+1)} - x^*\|_2 + \|x^{(t)} - x^*\|_2) + \delta R^{(t)} \quad \text{(triangle-i.e.)} \\
&\leq 2\beta \mu^f W + \delta R^{(t)} \quad \text{(by Q1 for $t, t + 1$)} \\
&= \xi R^{(t)} + \delta R^{(t)} \quad \text{(definition of $R^{(t)}$)} \\
&\leq (\xi + 2\delta/\xi) R^{(t)} \quad (2/\xi \geq 1) \\
&= \mu R^{(t)} = R^{(t+1)} .
\end{align*}

Combining the two previous inequalities, we have

\[ \|r^{(t+1)} - q^{(t)}\|_2 \leq \delta R^{(t+1)}/2 + R^{(t+1)} \leq 2R^{(t+1)}, \]

as desired.

Finally, putting things together, we have

\begin{align*}
\|\nabla F(x^{(t+1)}) - q^{(t+1)}\|_2 &= \|\nabla F(x^{(t+1)}) - r^{(t+1)} + r^{(t+1)} - q^{(t+1)}\|_2 \\
&\leq \|\nabla F(x^{(t+1)}) - r^{(t+1)}\|_2 + \|r^{(t+1)} - q^{(t+1)}\|_2 \\
&\leq \delta R^{(t+1)}/2 + \delta R^{(t+1)}/2 = \delta R^{(t+1)},
\end{align*}

completing the proof. \hfill \square

**Lemma 16.** For any $\varepsilon > 0$ and $t \geq (\kappa + 1) \log \frac{W}{\varepsilon}$, we have $\|x^{(t)} - x^*\|_2 \leq \varepsilon$.

**Proof.** By Lemma 15, we have $\|x^{(t)} - x^*\|_2 \leq \mu^f W = (1 - (1 - \mu)^t) W \leq e^{-(1 - \mu)t} W$. Assuming $t \geq \frac{1}{1 - \mu} \log \frac{W}{\varepsilon}$, we have

\[ e^{-(1 - \mu)t} W \leq e^{-(1 - \mu)(1 - \mu)^{-1} \log W/\varepsilon} W = e^{\log \varepsilon/\varepsilon} W = \varepsilon W/\varepsilon = \varepsilon . \]

The claim follows by observing that $\frac{1}{1 - \mu} = \kappa + 1$ by definition. \hfill \square

**Communication cost.** Finally, we analyse the distributed implementation described at the beginning of this section, and analyse its total communication cost. Recall that we assume that the parameters $\alpha$ and $\beta$ are known to all nodes, so the parameters of the quantised gradient descent can be computed locally, and use $W = d^{1/2}$. Note that $W$ is the only parameter depending on the input domain, so the algorithm also applies for arbitrary convex domain $D \subseteq \mathbb{R}^d$, setting $W$ to be the diameter of $D$.

Since $\delta < 1$, we have by Lemma 9 that the each of the messages sent by the nodes has length at most $O(d \log \delta^{-1})$ bits. Assuming $\kappa \geq 2$, we have $\xi \geq 1/3$ and

\[ \log \delta^{-1} = \log \frac{2(\kappa + 1)}{\xi} \leq \log 6(\kappa + 1) \leq \log 7\kappa . \]

Since the nodes send a total of $2N$ messages of $O(d \log \kappa)$ bits each, the total communication cost of a single round is $O(N d \log \kappa)$ bits.

To get $F(x^{(T)}) - F(x^*) \leq \varepsilon$, we need $\|x^{(T)} - x^*\|_2 \leq (\varepsilon/\beta)^2$. By Lemma 16, selecting $T = O(\kappa \log \frac{\beta W}{\varepsilon})$ is sufficient. Finally, using $W = O(d^{1/2})$, we have that the total communication cost of the optimisation is $O(N d \kappa \log \kappa \log \frac{\beta d}{\varepsilon})$. 

18
D Subsampling

In this section, we show that the condition \( \frac{\beta d}{N^2 \varepsilon} = \Omega(1) \) in our main lower bound is, to a degree, necessary.

**Lemma 17.** Let \( S = \{x_1, x_2, \ldots, x_N\} \subseteq [0, 1]^d \), and let \( X_1, X_2, \ldots, X_M \) be i.i.d. random variables, each selected uniformly at random from \( S \). Writing \( \hat{x} = \frac{1}{N} \sum_{i=1}^{N} x_i \) and \( X = \frac{1}{M} \sum_{i=1}^{M} X_i \), we have

\[
\mathbb{E} \left[ \|X - \hat{x}\|_2^2 \right] = 0, \quad \text{and} \quad \text{Var} \left( \|X - \hat{x}\|_2^2 \right) = \frac{d}{M}.
\]

**Proof.** The first part follows immediately by the definition of the expectation. For the second part, we first note that since all points within \([0, 1]^d\) are at most \( \frac{d}{2} \) apart, and thus by the definition of variance, it follows that

\[
\text{Var}(\|X - \hat{x}\|_2^2) = \mathbb{E}[\|X - \hat{x}\|_2^4] - \mathbb{E}[\|X - \hat{x}\|_2]^2 = \mathbb{E}[\|X - \hat{x}\|_2^4] - 0
\]

\[
\leq \mathbb{E} \left[ \frac{1}{M^2} \sum_{i=1}^{M} \|X_i - \hat{x}\|_2^4 \right] = \frac{1}{M^2} \sum_{i=1}^{M} \mathbb{E}[\|X_i - \hat{x}\|_2^4]
\]

\[
\leq \frac{1}{M^2} Md = \frac{d}{M}.
\]

**Theorem 18.** Assume the input functions \( f_i \) of the nodes are promised to be quadratic functions \( x \mapsto \beta_0 \|x - x^*\|_2^2 \) for some constant \( \beta_0 > 0 \), let \( \beta = \beta_0 N \), and assume we can select \( M \leq N \) to be an integer satisfying \( \beta d/M \varepsilon \leq 1/8 \). The there is a randomised algorithm solving (1) using \( O \left( M d \log \frac{\beta d}{\varepsilon} \right) \) bits of communication, with probability at least \( 1/2 \).

**Proof.** We start by having the coordinator select a multiset \( I \) of \( M \) nodes uniformly at random with replacement. Let \( \hat{x} \) denote the global optimum of \( \sum_{i=1}^{N} f_i \), and let \( \hat{Y} \) be the random variable for the global optimum of \( \sum_{i \in I} f_i \). By Lemma 17, Chebyshev’s inequality and the assumption \( \beta d/N \varepsilon \leq 1/8 \), we have that

\[
\Pr \left[ \|\hat{Y} - \hat{x}\|_2 \geq \frac{1}{2} \left( \frac{\varepsilon}{\sqrt{\beta}} \right)^{1/2} \right] \leq \frac{4d \beta}{\varepsilon M} \leq 1/2.
\]

Let \( \hat{y} \) be the actualised value of \( \hat{Y} \). We now apply the algorithm of Theorem 10 to find a point \( z \) such that \( \|z - \hat{y}\|_2 \leq 1/2 (\varepsilon/\beta)^{1/2} \), where, if the multiset \( I \) contains duplicates, those nodes simulate multiple copies of themselves. This uses \( O(M d \log \beta d/\varepsilon) \) bits of communication. We now have with probability at least \( 1/2 \) that \( \|z - \hat{x}\| \leq (\varepsilon/\beta)^{1/2} \), and thus \( \sum_{i=1}^{N} f_i(z) \leq \sum_{i=1}^{N} f_i(\hat{x}) + \varepsilon. \)
Checklist

(1) For all authors...
   (1) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (2) Did you describe the limitations of your work? [Yes]
   (3) Did you discuss any potential negative societal impacts of your work? [Yes]
   (4) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

(2) If you are including theoretical results...
   (1) Did you state the full set of assumptions of all theoretical results? [Yes]
   (2) Did you include complete proofs of all theoretical results? [Yes]

(3) If you ran experiments...
   (1) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (2) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (3) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (4) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]

(4) If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (1) If your work uses existing assets, did you cite the creators? [N/A]
   (2) Did you mention the license of the assets? [N/A]
   (3) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (4) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [N/A]
   (5) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

(5) If you used crowdsourcing or conducted research with human subjects...
   (1) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (2) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (3) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]