# Decentralized Personalized Federated Learning: Lower Bounds and Optimal Algorithm for All Personalization Modes

**Abdurakhmon Sadiev** SADIEV.AA@PHYSTECH.EDU
*Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation*

**Ekaterina Borodich** BORODICH.ED@PHYSTECH.EDU
*Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation*

**Aleksandr Beznosikov** ANBEZNOSIKOV@GMAIL.COM
*Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation*
*Higher School of Economics (HSE University), Moscow, Russian Federation*
*Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), Masdar City, Abu Dhabi, UAE*

**Darina Dvinskikh** DARINA.DVINSKIKH@WIAS-BERLIN.DE
*Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation*
*Weierstrass Institute for Applied Analysis and Stochastics (WIAS), Berlin, Germany*
*Institute for Information Transmission Problems RAS (IITP RAS), Moscow, Russia*

**Alexander Gasnikov** GASNIKOV@YANDEX.RU
*Moscow Institute of Physics and Technology (MIPT), Moscow, Russian Federation*
*Higher School of Economics (HSE University), Moscow, Russian Federation*
*Institute for Information Transmission Problems RAS (IITP RAS), Moscow, Russia*

## Abstract

In this paper, we consider the formulation of the federated learning problem that is relevant to both decentralized personalized federated learning and multi-task learning. This formulation is widespread in the literature and represents the minimization of local losses with regularization taking into account the communication matrix of the network. First of all, we give lower bounds for the considered problem in different regularization regimes. We also constructed an optimal algorithm that matches these lower bounds.

## 1. Introduction

Over the past few years, there has been a great interest in minimizing the average of convex functions (local losses)

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{k=1}^{n} f_k(x). \tag{1}$$

The interest is due to this problem arises in many machine learning and statistical applications, e.g., empirical risk minimization, maximum likelihood estimation and etc. Recently, a new direction of distributed optimization - Federated Learning (FL) [7, 10], has appeared. Unlike classical distributed learning, the federated approach assumes that data is not stored on the same device within a computing cluster, but on client devices, such as laptops, phones, or tablets. This formulation of the training problem gives rise to many additional questions, ranging from the privacy of client's data to a high heterogeneity of data stored on local devices.

In the simplest setting of distributed and federated learning, our goal is to find a solution for (1). In this problem we find global model based on all local data. On the other hand, improving the local models using the knowledge of the global model may need a more careful balance, taking into account a possible discrepancy between data splits the local models were trained on. Attempts to find the balance between personalization and globalization have resulted in a series of works united by a common name – Personalized Federated Learning (PFL). See surveys [6, 8] for more details and explanations of different techniques.

In this paper, we also cover the topic of PFL and consider the following problem:

$$\min_{\mathbf{x}=[x_1,\ldots,x_n]\in\mathbb{R}^{nd}} F(\mathbf{x}) \triangleq \underbrace{\frac{1}{n}\sum_{k=1}^{n} f_k(x_k)}_{f(\mathbf{x})} + \underbrace{\frac{\lambda}{2}\langle \mathbf{x}, \mathbf{W}\mathbf{x}\rangle}_{g(\mathbf{x})}. \tag{2}$$

This problem consists of two parts: the main part $f$ - local losses with their independent variables $x_i$ and a regularizer $g$. We assume that our devices are connected to some kind of communication network. We want models $x_i$ and $x_j$ to take into account each other if their devices are connected in the network. For this, it seems natural to penalize the difference between $x_i$ and $x_j$. To regularize all such pairs of models for connected devices, $g$ with matrix $W$ is used . This matrix reflects the properties of the connection graph (a more formal definition will be given below). From the point of view of personalization, the issues of choosing $\lambda$ are very important. The smaller this parameter, the more personalized these models are.

### 1.1. Brief literature review

The idea of such a penalty is not new and has been encountered in the literature in various contexts.

**Classical decentralized minimization** Problem (2) was considered before it became interesting from the point of view of FL. For example, it appeared in the decentralized optimization. Intuition suggests that with $\lambda \to \infty$, the problem (2) become closer and closer to another problem:

$$\min_{[x_1=\ldots=x_n]} \frac{1}{n}\sum_{k=1}^{n} f_k(x_k),$$

which is equivalent to (1). For the first time, the idea of reformulating decentralized optimization in a penalized form appeared in [9]. The work [2] continues this idea and sheds light on the issue of optimal selection of $\lambda$, and also proposes new algorithms.

**Centralized PFL** In the centralized case (when the matrix $W$ corresponds to the complete graph), problem (2) is considered in [3–5]. In particular, work [4] gives lower bounds on the number of communications and local iterations for solving (2), as well as optimal algorithms that match these bounds.

**Multi-task Learning** For classic decentralized optimization, problem (2) is interesting with a large $\lambda$, and with a small $\lambda$, it got popular in multi-task learning. This is also a new direction in FL. Often, in multi-task learning, not only models $x$, but also configurations of connection network $W$ are optimized [12]. But there are works about multi-task learning where matrix $W$ is fixed [14]

### 1.2. Our contribution

**Lower bounds** In our paper, we present the obtained lower bounds for the tasks of personalized federated learning (PFL) (2) in the distributed decentralized case. Moreover, they are true for any mode of the constant $\lambda$, both small and large. The lower bounds obtained in the work [4] are a special case of our lower bounds, when communication network is represented by a fully connected graph.

**Optimal algorithm** Also in our work, we propose an algorithm based on Accelerated Meta-Algorithm [1]. Our algorithm for solving the problem 2 has a convergence rate that coincides with the lower bounds with the accuracy of the logarithmic factors.

See summary of our contribution in table 1.2.

| | **Lower** | **Upper** |
|---|---|---|
| local | $\widetilde{\Omega}\left(\sqrt{\frac{L}{\mu}}\right)$ | $\widetilde{\mathcal{O}}\left(\sqrt{\frac{L}{\mu}}\right)$ |
| comm | $\widetilde{\Omega}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\chi}\right\}\right)$ | $\widetilde{\mathcal{O}}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\chi}\right\}\right)$ |

Table 1: Summary of the contribution

## 2. Main results

Before present out theoretical results we introduce some notations and assumptions.

We use $\langle x, y\rangle := \sum_{i=1}^{d} x_i y_i$ to define standard inner product of $x, y \in \mathbb{R}^d$. It induces $\ell_2$-norm in $\mathbb{R}^d$ in the following way $\|x\| := \sqrt{\langle x, x\rangle}$.

**Assumption 1** *Function $f(\mathbf{x})$ from problem (2) is $L$-smooth w.r.t $\ell_2$-norm $\|\cdot\|_2$, i.e. for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^{nd}$ we have*

$$\|\nabla f(\mathbf{x}_2) - \nabla f(\mathbf{x}_1)\| \le L\,\|\mathbf{x}_2 - \mathbf{x}_1\|\,.$$

**Assumption 2** *Function $f(\mathbf{x})$ from problem (2) is $\mu$-strongly-convex w.r.t. $\ell_2$-norm $\|\cdot\|_2$, i.e. for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^{nd}$ we have*

$$f(\mathbf{x}_2) - f(\mathbf{x}_1) \ge \langle \nabla f(\mathbf{x}_1), \mathbf{x}_2 - \mathbf{x}_1\rangle + \frac{\mu}{2}\,\|\mathbf{x}_2 - \mathbf{x}_1\|^2\,.$$

**Assumption 3** *The communication matrix $\mathbf{W}$ in (2) is defined as the Kronecker product of matrix $\hat{W}$ (to be defined further) and the identity matrix $I_d$ to take into the consideration that all $x_k \in \mathbb{R}^d$ ($k = 1, \ldots, n$): $\mathbf{W} = \hat{W} \otimes I_d$. The gossip matrix $\hat{W}$ satisfies the following conditions (see [11]):*

1. *$\hat{W}$ is symmetric positive semi-definite;*
2. *The kernel $\hat{W}$ consists of vector $\mathbf{1} = (1, \ldots, 1)^\top$;*
3. *$\hat{W}$ is defined on the edges of the communication network: $\hat{w}_{i,j} \neq 0$ if and only if $i = j$ or $(i, j) \in E$.*

*The simplest choice of $\hat{W}$ is the Laplace matrix.*

**Remark 1** *It is easy to show that the matrix $\mathbf{W}$ is a gossip matrix and also $\frac{\lambda_{\max}(\hat{W})}{\lambda^{+}_{\min}(\hat{W})} = \frac{\lambda_{\max}(\mathbf{W})}{\lambda^{+}_{\min}(\mathbf{W})}$. It is denoted as $\chi = \frac{\lambda_{\max}(\mathbf{W})}{\lambda^{+}_{\min}(\mathbf{W})}$*

We divide our contribution into two parts: lower estimates for the number of communication and local calls for problem 1 and the optimal algorithm that matches these estimates.

### 2.1. Lower bounds

Before presenting the lower bounds on the problem 2, we need to enter an assumption on the class of algorithms for which they will be correct.

**Assumption 4** *Let $\{\mathbf{x}^k\}_{k=1}^{\infty}$ be iterates generated by algorithm $\mathcal{A}$. For each nodes of graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ we define sequence of local memory $\{\mathcal{M}_{i,k}\}_{k=1}^{\infty}$ for $1 \le i \le n$:*

$$\begin{aligned} \mathcal{M}_{i,0} &= \text{span}\left(x_i^0\right) \\ \mathcal{M}_{i,k+1} &= \begin{cases} \text{span}\left(\mathcal{M}_{i,k}, \{x_i^k, \nabla f(\mathbf{x}^k)\}\right), & \text{if local oracle at the iteration } k \\ \text{span}\left(\cup_{j:(j,i)\in\mathcal{E}} \mathcal{M}_{j,k}\right), & \text{if consensus oracle at the iteration } k \end{cases} \end{aligned}$$

**Remark 2** *Local oracle commonly corresponds to the calculation of the gradient $\nabla f(\mathbf{x}^k)$ and the implementation of a single gradient step. Consensus oracle corresponds to a communication round, where information about the each vector $x_i^k$ stored on each node $i$ is transmitted between neighboring nodes $j$ ($(i, j) \in \mathcal{E}$).*

Now we are ready to provide you with the main theorem of this section.

**Theorem 3** *Let $k \ge 0$, $L \ge \mu$, $\lambda\lambda^{+}_{\min}(\mathbf{W}) \ge \mu$, $\chi \ge 6$. Then, there exist graph $\mathcal{G}$, which matrix $\mathbf{W}$ satisfies Assumption 3, $L$-smooth $\mu$-strongly convex functions $f_1, f_2, \dots, f_n : \mathbb{R}^d \to \mathbb{R}$ (see Assumption 1, 2) and a starting point $\mathbf{x}_0 \in \mathbb{R}^{nd}$ such that the sequence of iterates $\{\mathbf{x}^k\}_{k=1}^{N}$ generated by any algorithm $\mathcal{A}$ satisfying Assumption 4*

$$\|\mathbf{x}^N - \mathbf{x}^*\|^2 \ge \left(1 - 10\max\left\{\sqrt{\frac{\mu}{\lambda\lambda_{\max}(\mathbf{W})}}, \sqrt{\frac{\mu}{(L-\mu)\chi}}\right\}\right)^{N_{comm}} \frac{\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{4} \tag{3}$$

**Corollary 4** *Let $k \ge 0$, $L \ge \mu$, $\lambda\lambda^{+}_{\min}(\mathbf{W}) \ge \mu$, $\chi \ge 6$. Then, there exist graph $\mathcal{G}$, which matrix $\mathbf{W}$ satisfies Assumption 3, $L$-smooth $\mu$-strongly convex functions $f_1, f_2, \dots, f_n : \mathbb{R}^d \to \mathbb{R}$ (see Assumption 1, 2) and a starting point $\mathbf{x}_0 \in \mathbb{R}^{nd}$ such that for any algorithm $\mathcal{A}$ satisfying Assumption 4 the number of communications to achieve $\varepsilon$-solution to problem (2) is lower bounded by*

$$\widetilde{\Omega}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\chi}\right\}\right) \tag{4}$$

*and the number of local computations to achieve $\varepsilon$-solution to problem (2) is lower bounded by*

$$\widetilde{\Omega}\left(\sqrt{\frac{L}{\mu}}\right) \tag{5}$$

**Algorithm 1:** Accelerated Meta-Algorithm (AM), $p = 1$, [1] for convex composite optimization problem. AM$(z_0, h, r, H, K)$.

```
Input: $p \in \mathbb{N}$, number of iterations $K$, starting point $z_0$, parameter $H > 0$.
$A_0 = 0$
$y_0 = z_0$
$\lambda_0 = \frac{1}{2H}$
for $k = 0, \dots, K-1$ do
    $a_{k+1} = \frac{\lambda_0 + \sqrt{\lambda_0^2 + 4\lambda_{k+1}A_k}}{2}$
    $A_{k+1} = A_k + a_{k+1}$
    $w_k = \frac{A_k}{A_{k+1}} y_k + \frac{a_{k+1}}{A_{k+1}} z_k$
    $y_{k+1} = \text{argmin}_{y \in \mathbb{R}^d} \left\{ h(w_k) + \langle \nabla h(w_k), y - w_k \rangle + r(y) + \frac{H}{2}\|y - w_k\|^2 \right\}$
    $z_{k+1} := z_k - a_{k+1}\nabla h(y_{k+1}) - a_{k+1}\nabla r(y_{k+1})$
end
Output: AM$(z_0, K) := y_K$.
```

**Algorithm 2:** Restarted Accelerated Meta-Algorithm [1], for strongly convex composite optimization problem. RAM$(z_0, h, r, \mu, H, s)$.

```
Input: starting point $z_0$, $H > 0, \mu > 0$, number of restarts $s \geqslant 1$.
for $k = 0, \dots, s-1$ do
    $N_k = \max\left\{ \left\lceil 4 \cdot \sqrt{\frac{2H}{\mu}} \right\rceil, 1 \right\}$
    $z_{k+1} :=$ AM$(z_k, N_k)$    // the output of Algorithm 1, with starting
     point $z_k$ and $N_k$ iterations)
end
Output: $z_s$
```

### 2.2. Optimal Algorithm

In this section, we present near-optimal upper bounds for the problem 2. However, before we give the main theorem of this section, we will briefly describe a universal algorithm for the composite optimization problem

$$\min_{x \in \mathbb{R}^d} \{h(x) + r(x)\} \tag{6}$$

co-called Accelerated Meta-Algorithm 1 and its restarted version 2 [1]. This method has a linear convergence rate under the assumptions that $h$ is $L(h)$-smooth, $\mu$-strongly convex and $r$ is $L(r)$-smooth and convex ($\mu = 0$-strongly convex). But the main question is which function will play the role of the function $h$, and which will play the role of the function $r$ in the problem 2. Our answer to this question is as follows: **it depends on the value of the parameter** $\lambda$.

**Theorem 5** *Let each functions $f_k(\cdot)$ satisfy the Assumptions 1, 2 and let $\delta > 0$ be accuracy of solution auxiliary problem 1. Then to achieve $\varepsilon$-solution to problem 2 solving by Algorithm 1 it* *needs the number of communications*

$$N^{comm} = \mathcal{O}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\chi}\right\}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^\star\|^2}{\varepsilon}\log\frac{1}{\delta}\right),$$

*where* $\chi = \frac{\lambda_{\max}(\mathbf{W})}{\lambda^+_{\min}(\mathbf{W})}$*, and the number of local oracle calls (gradient of each functions* $f_k(\cdot)$*)*

$$N^{loc} = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^\star\|^2}{\varepsilon}\log\frac{1}{\delta}\right)$$

*where we take* $\delta$ *as follows*

$$\delta = \frac{\varepsilon\mu}{864^2(L + \lambda\lambda_{\max}(\mathbf{W}) + H)^2}.$$

## References

[1] Alexander Gasnikov, Darina Dvinskikh, Pavel Dvurechensky, Dmitry Kamzolov, Vladislav Matykhin, Dmitry Pasechnyk, Nazarii Tupitsa, and Alexei Chernov. Accelerated meta-algorithm for convex optimization. *arXiv preprint arXiv:2004.08691*, 2020.

[2] Eduard Gorbunov, Darina Dvinskikh, and Alexander Gasnikov. Optimal decentralized distributed algorithms for stochastic convex optimization. *arXiv preprint arXiv:1911.07363*, 2019.

[3] Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020.

[4] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. Lower bounds and optimal algorithms for personalized federated learning. *arXiv preprint arXiv:2010.02372*, 2020.

[5] Filip Hanzely, Boxin Zhao, and Mladen Kolar. Personalized federated learning: A unified framework and universal optimization techniques, 2021.

[6] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[7] Jakub Konečnỳ, H Brendan McMahan, Daniel Ramage, and Peter Richtárik. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016.

[8] Viraj Kulkarni, Milind Kulkarni, and Aniruddha Pant. Survey of personalization techniques for federated learning. In *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*, pages 794–797. IEEE, 2020.

[9] Huan Li, Cong Fang, Wotao Yin, and Zhouchen Lin. Decentralized accelerated gradient methods with increasing penalty parameters. *IEEE Transactions on Signal Processing*, 68:4855–4870, 2020.

[10] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[11] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. *arXiv preprint arXiv:1702.08704*, 2017.

[12] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet Talwalkar. Federated multi-task learning. *arXiv preprint arXiv:1705.10467*, 2017.

[13] Vladislav Tominin, Yaroslav Tominin, Ekaterina Borodich, Dmitry Kovalev, Alexander Gasnikov, and Pavel Dvurechensky. On accelerated methods for saddle-point problems with composite structure. *arXiv preprint arXiv:2103.09344*, 2021.

[14] Weiran Wang, Jialei Wang, Mladen Kolar, and Nathan Srebro. Distributed stochastic multi-task learning with graph regularization. *arXiv preprint arXiv:1802.03830*, 2018.

## Appendix A. Proof Theorem 3

In this section, we prove lower convergence bounds of algorithms for this class of problems 2 similar to the work [4]. As in many papers we give an example of a ”bad” function on which algorithms satisfying the assumption 4 converge at least at a rate that coincides with the lower estimates. We consider a linear graph with the number of nodes equal to $n = 3\left\lfloor \frac{\chi}{3} \right\rfloor$, where $\chi$ is condition number of communication network. Let’s divide the nodes into three types: the first type includes $\mathcal{V}_1 = \left\{1, 2, \ldots, \frac{n}{3}\right\}$, the second type includes $\mathcal{V}_2 = \left\{\frac{n}{3}+1, \frac{n}{3}+2, \ldots, \frac{2n}{3}\right\}$, the third type includes $\mathcal{V}_3 = \left\{\frac{2n}{3}+1, \frac{2n}{3}+2, \ldots, n\right\}$. Let $d = 2T$ dimension and $T$ more than $n$.

$$f_i(x) = \begin{cases} \frac{\mu}{2}\|x\|^2 + ax_1 + \frac{c\lambda}{2}\left(\sum_{i=1}^{T-1}(x_{2i} - x_{2i+1})\right) + \frac{b\lambda}{2}x_{2T}, & \text{if } i \in \mathcal{V}_1 \\ \frac{\mu}{2}\|x\|^2, & \text{if } i \in \mathcal{V}_2 \\ \frac{\mu}{2}\|x\|^2 + \frac{c\lambda}{2}\left(\sum_{i=1}^{T-1}(x_{2i+1} - x_{2i+2})\right), & \text{if } i \in \mathcal{V}_3 \end{cases} \tag{7}$$

For the gossip matrix, we take the Laplacian of a linear graph. Then, for our problem (2), we get that the matrix $\mathbf{W}$ will have the following form:

$$\mathbf{W} = \hat{\mathbf{W}} \otimes I_d,$$

where matrix $\hat{W}$ has folowing form

$$\hat{\mathbf{W}} = \frac{1}{2n} \cdot \begin{pmatrix} 1 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ & -1 & 2 & -1 & & & & \\ & & -1 & 2 & -1 & & & \\ & & & -1 & 2 & -1 & & \\ & & & & & \cdots & & \\ & & & & & -1 & 2 & -1 \\ & & & & & & -1 & 1 \end{pmatrix}.$$

It is easy to make sure that this matrix $W$ satisfies the assumption of the gossip matrix.

According to the definition of functions, now we can write the form of the objective function of problem 2:

$$\frac{n}{\lambda}F(x) = \frac{1}{2}x^\top \mathbf{M}x + \frac{a}{\lambda}\sum_{i\in\mathcal{V}_1} x_i(1), \tag{8}$$

where $M$ looks like

$$\mathbf{M} \overset{\text{def}}{=} n\mathbf{W} + \frac{\mu}{\lambda}\mathbf{I}_{nd} + \begin{pmatrix} \mathbf{M}_1 & 0 & 0 \\ 0 & \mathbf{0}_{d|\mathcal{V}_2|} & 0 \\ 0 & 0 & \mathbf{M}_2 \end{pmatrix}, \text{ where}$$

$$\mathbf{M}_1 \overset{\text{def}}{=} \mathbf{I}_{d|\mathcal{V}_1|} \otimes \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & \begin{pmatrix} c & -c \\ -c & c \end{pmatrix} & 0 & \ddots & \vdots \\ 0 & 0 & \begin{pmatrix} c & -c \\ -c & c \end{pmatrix} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & b \end{pmatrix} \text{ and}$$

$$\mathbf{M}_2 \overset{\text{def}}{=} \mathbf{I} \otimes \begin{pmatrix} \begin{pmatrix} c & -c \\ -c & c \end{pmatrix} & 0 & \dots \\ 0 & \begin{pmatrix} c & -c \\ -c & c \end{pmatrix} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$

Due to the fact that the functions of each type are similar, their minima coincide. Therefore, we denote them as follows

$$\underset{x}{\operatorname{argmin}} f_i(x) = \begin{cases} x^\star, & \text{if } i \in \mathcal{V}_1 \\ y^*, & \text{if } i \in \mathcal{V}_2 \\ z^*, & \text{if } i \in \mathcal{V}_3 \end{cases} \tag{9}$$

Now we give a proof of the lemma that indicates a recursive connection

**Lemma 6 (see [4])** *Let*

$$w_i \overset{\textit{def}}{=} \begin{cases} \begin{pmatrix} z_i^\star \\ x_i^\star \end{pmatrix} & \textit{if } i \textit{ is even} \\ \begin{pmatrix} x_i^\star \\ z_i^\star \end{pmatrix} & \textit{if } i \textit{ is odd} \end{cases}. \tag{10}$$

*Then, we have*

$$w_{i+1} = \mathbf{Q}w_i \tag{11}$$

*where*

$$\mathbf{Q} \overset{def}{=} \begin{pmatrix} -\frac{1}{2c\left(1+\frac{2\mu}{\lambda}\right)} & \frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} \\ -\frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} & \left(1+\frac{2\mu}{\lambda}\right)\left(\frac{2\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)^2}{c}-2c\right) \end{pmatrix} \tag{12}$$

**Proof** Let's write down the first-order optimality conditions for the problem (2)

$$\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)x^\star_{2i+1} - cx^\star_{2i} - \frac{1}{2}y^*_{2i+1} = 0, \quad \text{for } 0 \le i \le T-1 \tag{13}$$

$$\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)x^\star_{2i} - cx^\star_{2i+1} - \frac{1}{2}y^*_{2i} = 0, \quad \text{for } 0 \le i \le T-1 \tag{14}$$

$$\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)z^*_{2i-1} - cz^*_{2i} - \frac{1}{2}y^*_{2i-1} = 0, \quad \text{for } 1 \le i \le T-1 \tag{15}$$

$$\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)z^*_{2i} - cz^*_{2i-1} - \frac{1}{2}y^*_{2i} = 0, \quad \text{for } 1 \le i \le T-1 \tag{16}$$

$$\left(1+\frac{2\mu}{\lambda}\right)y^*_i - x^\star_i = 0, \quad \text{for } 1 \le i \le 2T-1 \tag{17}$$

$$\left(1+\frac{2\mu}{\lambda}\right)y^*_i - z^*_i = 0, \quad \text{for } 1 \le i \le 2T-1 \tag{18}$$

Combining (15) and (16), we get for all $1 \le i \le T$

$$\begin{pmatrix} c & 0 \\ -c-\frac{1}{2}-\frac{\mu}{\lambda} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} z^\star_{2i} \\ y^\star_{2i} \end{pmatrix} = \begin{pmatrix} c+\frac{1}{2}+\frac{\mu}{\lambda} & -\frac{1}{2} \\ -c & 0 \end{pmatrix} \begin{pmatrix} z^\star_{2i-1} \\ y^\star_{2i-1} \end{pmatrix} \tag{19}$$

Rewriting this equation, we get

$$\begin{pmatrix} z^\star_{2i} \\ y^\star_{2i} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2c} & \frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} \\ -\frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} & \frac{2\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)^2}{c}-2c \end{pmatrix} \begin{pmatrix} y^\star_{2i-1} \\ z^\star_{2i-1} \end{pmatrix}.$$

Similarly, using (13), (14) we get for all $1 \le i \le T$

$$\begin{pmatrix} x^\star_{2i+1} \\ y^\star_{2i+1} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2c} & \frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} \\ -\frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} & \frac{2\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)^2}{c}-2c \end{pmatrix} \begin{pmatrix} y^\star_{2i} \\ x^\star_{2i} \end{pmatrix}.$$

Using (17) and (18), we obtain

$$\begin{pmatrix} x^\star_{2i+1} \\ z^\star_{2i+1} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2c\left(1+\frac{2\mu}{\lambda}\right)} & \frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} \\ -\frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} & \left(1+\frac{2\mu}{\lambda}\right)\left(\frac{2\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)^2}{c}-2c\right) \end{pmatrix} \begin{pmatrix} z^\star_{2i} \\ x^\star_{2i} \end{pmatrix}.$$

$$\begin{pmatrix} z^\star_{2i} \\ x^\star_{2i} \end{pmatrix} = \begin{pmatrix} -\frac{1}{2c\left(1+\frac{2\mu}{\lambda}\right)} & \frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} \\ -\frac{c+\frac{\mu}{\lambda}+\frac{1}{2}}{c} & \left(1+\frac{2\mu}{\lambda}\right)\left(\frac{2\left(c+\frac{\mu}{\lambda}+\frac{1}{2}\right)^2}{c} - 2c\right) \end{pmatrix} \begin{pmatrix} x^\star_{2i-1} \\ z^\star_{2i-1} \end{pmatrix}.$$

■

The following lemma would be very difficult to prove without using Mathematica due to very cumbersome expressions. The following statement shows a recursive relationship between coordinates in the following sense $w_i = \gamma^{i-1} w_1$, where $\gamma$ is the eigenvalue of the matrix $\mathbf{Q}$.

**Lemma 7 (see [4])** *Choose* $c \overset{def}{=} \begin{cases} \delta \frac{\mu}{\lambda\lambda_{\max}(\mathbf{W})} & \textit{if } L \geq \lambda\lambda_{\max}(\mathbf{W}) + \mu \\ \delta \frac{\mu}{\lambda\lambda_{\max}(\mathbf{W})} & \textit{if } L < \lambda\lambda_{\max}(\mathbf{W}) + \mu \end{cases}$, $\delta \geq 1$ *and*

$$b \overset{def}{=} \frac{1+\frac{2\mu}{\lambda}}{2}\left(\frac{-(1+2\frac{\mu}{\lambda})\nu}{2(1+2c+2\frac{\mu}{\lambda})} + \frac{-\alpha+2\sqrt{\alpha^2-4\beta}}{4(1+2\frac{\mu}{\lambda})(1+2c+2\frac{\mu}{\lambda})}\right) - \frac{1}{2} - \frac{\mu}{\lambda} \tag{20}$$

*Then, we have* $b \geq 0$ *and*

$$w_i = \gamma^{i-1} w_1 \neq \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \textit{for } i = 1, 2, \ldots, d$$

*where*

$$\gamma \overset{def}{=} \frac{\alpha}{8c(1+2\frac{\mu}{\lambda})} - \frac{\sqrt{(\alpha^2-4\beta)}}{8c(1+2\frac{\mu}{\lambda})} \geq 1 - 10\sqrt{\frac{1}{\delta}}, \tag{21}$$

*where*

$$\begin{aligned} \alpha \quad = \quad & -1 + 4c - 4c^2 + 8\frac{\mu}{\lambda} + 24cx - 16c^2\frac{\mu}{\lambda} + 24\left(\frac{\mu}{\lambda}\right)^2 + 48c\left(\frac{\mu}{\lambda}\right)^2 \\ & -16c^2\left(\frac{\mu}{\lambda}\right)^2 + 32\left(\frac{\mu}{\lambda}\right)^3 + 32c\left(\frac{\mu}{\lambda}\right)^3 + 16\left(\frac{\mu}{\lambda}\right)^4 \end{aligned}$$

*and*

$$\begin{aligned} \beta \quad = \quad & 2 + 8c + 24c^2 + 16\left(\frac{\mu}{\lambda}\right) + 48c\left(\frac{\mu}{\lambda}\right) + 96c^2\left(\frac{\mu}{\lambda}\right) + 48\left(\frac{\mu}{\lambda}\right)^2 \\ & +96c\left(\frac{\mu}{\lambda}\right)^2 + 96c^2\left(\frac{\mu}{\lambda}\right)^2 + 64\left(\frac{\mu}{\lambda}\right)^3 + 64c\left(\frac{\mu}{\lambda}\right)^3 + 32\left(\frac{\mu}{\lambda}\right)^4 \end{aligned}$$

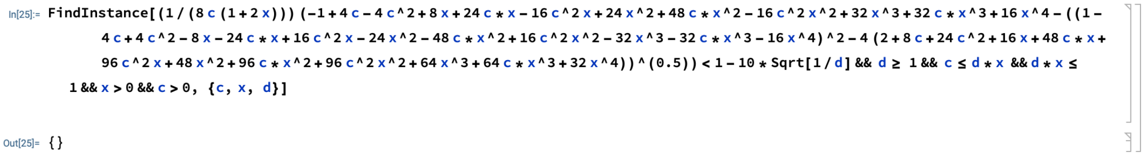
*and*

$$\nu = -1 - 4c + 4c^2 - 4\left(\frac{\mu}{\lambda}\right) - 8c\left(\frac{\mu}{\lambda}\right) - 4\left(\frac{\mu}{\lambda}\right)^2$$

**Proof** First, using the Mathematica software package, we calculate the minimum eigenvalue $\gamma$ of the matrix $\mathbf{W}$ and prove that it satisfies the inequality. For a detailed study of the proof, see the file or screenshots Using Mathematica, we find the eigenvector $v$ of the matrix $Q$ corresponding to the

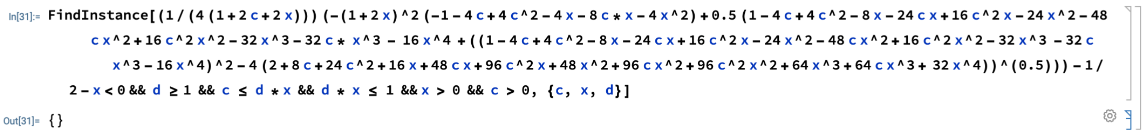
```
Eigenvalues[{{-1 / (2 * (c * (1 + 2 * x))), (c + x + 1 / 2) / c},
  {- (c + x + 1 / 2) / c, (1 + 2 * x) * (((c + x + 1 / 2) ^2) / c - 2 * c)}}]
```

$$\left\{\frac{1}{8\,c\,(1+2\,x)}\left(-1+4\,c-4\,c^2+8\,x+24\,c\,x-16\,c^2\,x+24\,x^2+48\,c\,x^2-16\,c^2\,x^2+32\,x^3+32\,c\,x^3+16\,x^4-\sqrt{\left(\left(1-4\,c+4\,c^2-8\,x-24\,c\,x+16\,c^2\,x-24\,x^2-48\,c\,x^2+16\,c^2\,x^2-32\,x^3-32\,c\,x^3-16\,x^4\right)^2-4\,\left(2+8\,c+24\,c^2+16\,x+48\,c\,x+96\,c^2\,x+48\,x^2+96\,c\,x^2+96\,c^2\,x^2+64\,x^3+64\,c\,x^3+32\,x^4\right)\right)}\right),\ \frac{1}{8\,c\,(1+2\,x)}\left(-1+4\,c-4\,c^2+8\,x+24\,c\,x-16\,c^2\,x+24\,x^2+48\,c\,x^2-16\,c^2\,x^2+32\,x^3+32\,c\,x^3+16\,x^4+\sqrt{\left(\left(1-4\,c+4\,c^2-8\,x-24\,c\,x+16\,c^2\,x-24\,x^2-48\,c\,x^2+16\,c^2\,x^2-32\,x^3-32\,c\,x^3-16\,x^4\right)^2-4\,\left(2+8\,c+24\,c^2+16\,x+48\,c\,x+96\,c^2\,x+48\,x^2+96\,c\,x^2+96\,c^2\,x^2+64\,x^3+64\,c\,x^3+32\,x^4\right)\right)}\right)\right\}$$



```
In[25]:= FindInstance[(1 / (8 c (1 + 2 x))) (-1 + 4 c - 4 c^2 + 8 x + 24 c * x - 16 c^2 x + 24 x^2 + 48 c * x^2 - 16 c^2 x^2 + 32 x^3 + 32 c * x^3 + 16 x^4 - ((1 -
           4 c + 4 c^2 - 8 x - 24 c * x + 16 c^2 x - 24 x^2 - 48 c * x^2 + 16 c^2 x^2 - 32 x^3 - 32 c * x^3 - 16 x^4)^2 - 4 (2 + 8 c + 24 c^2 + 16 x + 48 c * x +
           96 c^2 x + 48 x^2 + 96 c * x^2 + 96 c^2 x^2 + 64 x^3 + 64 c * x^3 + 32 x^4))^(0.5)) < 1 - 10 * Sqrt[1 / d] && d ≥  1 && c ≤ d * x  && d * x ≤
       1 && x > 0 && c > 0, {c, x, d}]

Out[25]= {}
```



```
In[31]:= FindInstance[(1 / (4 (1 + 2 c + 2 x))) (-(1 + 2 x)^2 (-1 - 4 c + 4 c^2 - 4 x - 8 c * x - 4 x^2) + 0.5 (1 - 4 c + 4 c^2 - 8 x - 24 c x + 16 c^2 x - 24 x^2 - 48
          c x^2 + 16 c^2 x^2 - 32 x^3 - 32 c *  x^3  -  16 x^4 + ((1 - 4 c + 4 c^2 - 8 x - 24 c x + 16 c^2 x - 24 x^2 - 48 c x^2 + 16 c^2 x^2 - 32 x^3  - 32 c
            x^3 - 16 x^4)^2 - 4 (2 + 8 c + 24 c^2 + 16 x + 48 c x + 96 c^2 x + 48 x^2 + 96 c x^2 + 96 c^2 x^2 + 64 x^3 + 64 c x^3 +  32 x^4))^(0.5))) - 1 /
       2 - x < 0 && d  ≥ 1  &&  c  ≤  d  * x  &&  d  *  x  ≤  1  && x  >  0  &&  c  >  0, {c, x, d}]
Out[31]= {}
```

eigenvalue $\gamma$

$$v = \begin{pmatrix} \frac{-(1+2\frac{\mu}{\lambda})\nu}{2(1+2c+2\frac{\mu}{\lambda})} + \frac{-\alpha+2\sqrt{\alpha^2-4\beta}}{4(1+2\frac{\mu}{\lambda})(1+2c+2\frac{\mu}{\lambda})} \\ 1 \end{pmatrix} \tag{22}$$

Now, using Mathematica, we prove that $b \geq 0$ It is easy to see that when choosing the parameter $b$ according to (20), the vector $w_i$ is proportional to the eigenvector $v$ of the matrix $Q$. ■

Now we are ready to complete the proof of the theorem 3. Let $\mathbf{x}^0 = 0 \in \mathbb{R}^{nd}$.After $N$ iteration of algorithm $\mathcal{A}$ let $\mathbf{x}^N$ has $K$ non-zero coordinates(below we will establish a connection between the number of non-zero coordinates and the number of iteration of algorithm $\mathcal{A}$). Then using the equations (17), (18), we obtain

$$\begin{aligned} \frac{\|\mathbf{x}^N - \mathbf{x}^\star\|^2}{\|\mathbf{x}^0 - \mathbf{x}^\star\|} &\geq \frac{1}{2}\frac{\sum_{i=K+2}^d \|w_j\|^2 + (y_j^*)^2}{\sum_{i=1}^d \|w_j\|^2 + (y_j^*)^2} = \frac{1}{2}\frac{\sum_{i=K+2}^d \|w_j\|^2 + \left(\frac{\lambda}{\mu+\lambda}\right)^2 (y_j^*)^2}{\sum_{i=1}^d \|w_j\|^2 + \left(\frac{\lambda}{\mu+\lambda}\right)^2 (z_j^*)^2} \\ &= \frac{1}{2}\frac{\sum_{i=K+2}^d \|R\tilde{w}_j\|^2}{\sum_{i=1}^d \|R\tilde{w}_j\|^2}, \end{aligned}$$

where $R = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1+\left(\frac{\lambda}{\mu+\lambda}\right)^2} \end{pmatrix}$ and $\tilde{w}_j = (\mathbf{x}_j^\star, z_j^*)^\top$ $(\tilde{w}_j = \gamma^{j-1}\tilde{w}_1)$. It is not difficult to see that the matrix is positive definite and symmetric. Therefore, you can use it to define a new norm $\|w\|_R = \sqrt{\langle x, Rx\rangle}$ and use the properties of the norm. Then for large enough $d$ we have

$$\begin{aligned} \frac{\|\mathbf{x}^N - \mathbf{x}^\star\|^2}{\|\mathbf{x}^0 - \mathbf{x}^\star\|^2} &\geq \frac{1}{2}\frac{\sum_{i=K+2}^d \|R\tilde{w}_j\|^2}{\sum_{i=1}^d \|R\tilde{w}_j\|^2} = \frac{1}{2}\frac{\sum_{i=K+2}^d \gamma^{j-1}\|R\tilde{w}_1\|^2}{\sum_{i=1}^d \gamma^{j-1}\|R\tilde{w}_1\|^2} = \frac{1}{2}\frac{\sum_{i=K+2}^d \gamma^{j-1}}{\sum_{i=1}^d \gamma^{j-1}} \\ &= \frac{1}{2}\frac{\gamma^{K+1}\sum_{i=0}^{d-K-2}\gamma^j}{\sum_{i=0}^{d-1}\gamma^j} = \frac{1}{2}\gamma^{K+1}\frac{1-\gamma^{d-K-1}}{1-\gamma^d} \geq \frac{1}{4}\left(1-10\sqrt{\frac{1}{\delta}}\right)^K \end{aligned}$$

It is worth noting that the number of communication rounds $N_{\text{comm}}$ can be expressed in terms of $K$the number of non-zero coordinates of $\mathbf{x}^N$. Due to the fact that the communication network $\mathcal{G}$ is linear, it is necessary to conduct a number of communications equal to the diameter of the graph in order to transfer information from the first node to the last. Therefore, the number of communication rounds $N_{\text{comm}}$ is the number of non-zero coordinates of $\mathbf{x}^N$ multiplied by the diameter. In turn, the diameter is equal to $\sqrt{\chi}$. Thus, we obtain

$$\begin{aligned} \|\mathbf{x}^N - \mathbf{x}^\star\|^2 &\geq \frac{1}{4}\left(1-10\sqrt{\frac{1}{\delta}}\right)^{\frac{N_{\text{comm}}}{\sqrt{\chi}}} \|\mathbf{x}^0 - \mathbf{x}^\star\|^2 \\ &\geq \frac{1}{4}\left(1-10\sqrt{\frac{1}{\delta\chi}}\right)^{N_{\text{comm}}} \|\mathbf{x}^0 - \mathbf{x}^\star\|^2 \end{aligned}$$

Now we have to consider two cases: one when $L \geq \lambda\lambda_{\max}(\mathbf{W}) + \mu$, the other when $L < \lambda\lambda_{\max}(\mathbf{W}) + \mu$. Moreover, we must select the parameter $\delta$ in such a way that $\delta \geq 1$ but also $c \leq 1$.

- $L \geq \lambda\lambda_{\max}(\mathbf{W}) + \mu$: if we take $\delta = \frac{\lambda\lambda^{+}_{\min}(\mathbf{W})}{\mu}$, then we have $c \leq 1$ and

$$\|\mathbf{x}^N - \mathbf{x}^\star\|^2 \quad \geq \quad \frac{1}{4}\left(1 - 10\sqrt{\frac{\mu}{\lambda\lambda_{\max}(\mathbf{W})}}\right)^{N_{\text{comm}}} \|\mathbf{x}^0 - \mathbf{x}^\star\|^2$$

- $L < \lambda\lambda_{\max}(\mathbf{W}) + \mu$: if we take $\delta = \frac{L-\mu}{\mu}$, then we have $c \leq 1$ and

$$\|\mathbf{x}^N - \mathbf{x}^\star\|^2 \quad \geq \quad \frac{1}{4}\left(1 - 10\sqrt{\frac{\mu}{(L-\mu)\chi}}\right)^{N_{\text{comm}}} \|\mathbf{x}^0 - \mathbf{x}^\star\|^2$$

Summing up the results obtained, we obtain

$$\|\mathbf{x}^N - \mathbf{x}^\star\|^2 \quad \geq \quad \left(1 - 10\max\left\{\sqrt{\frac{\mu}{\lambda\lambda_{\max}(\mathbf{W})}}, \sqrt{\frac{\mu}{(L-\mu)\chi}}\right\}\right)^{N_{\text{comm}}} \frac{\|\mathbf{x}^0 - \mathbf{x}^\star\|^2}{4}$$

## Appendix B. Proof Theorem 5

For analysis we consider auxiliary problem 1 of Algorithm 1 with $p = 1$:

$$y_{k+1} = \underset{y \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ f(w_k) + \langle \nabla f(w_k), y - w_k \rangle + g(y) + \frac{H}{2}\|y - w_k\|^2 \right\} \tag{23}$$

**Case 1** $\lambda\lambda_{\max}(\mathbf{W}) \geq L$ Then we take $f(\mathbf{x})$ like sum component, $g(\mathbf{x})$ like $\frac{\lambda}{2}\langle \mathbf{x}, \mathbf{W}\mathbf{x} \rangle$ and we know that number of calls of gradient of $f$:

$$N_{\nabla f_k} = \mathcal{O}\left(\sqrt{\frac{L}{\mu}} \log \frac{\|\mathbf{x}_0 - \mathbf{x}^\star\|^2}{\varepsilon}\right) \tag{24}$$

Now we look carefully at auxiliary problem. We know **Ker**$(\mathbf{W})$ is not empty. And function $g(\mathbf{x})$ takes zero on this subspace **Ker**$(\mathbf{W})$. Then we can divide our problem on two subproblem: minimization of quadratic form with matrix $H \cdot I$ on **Ker**$(\mathbf{W})$ and minimization of quadratic form with matrix $\lambda\mathbf{W} + H \cdot I$ on $(\mathbf{Ker}(\mathbf{W}))^{\perp}$. Complexity of the first problem is equal to $\mathcal{O}(1)$. Complexity of the second problem is equal to

$$\mathcal{O}\left(\sqrt{\frac{H + \lambda\lambda_{\max}(\mathbf{W})}{\max\{H, \lambda\lambda^{+}_{\min}(\mathbf{W})\}}} \log \frac{1}{\delta}\right),$$

where $\delta$ is accuracy of solution to the auxiliary problem 1. Then we can say number of calls of gradient of $g$:

$$N_{\mathbf{Wx}} = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\sqrt{\frac{H + \lambda\lambda_{\max}(\mathbf{W})}{\max\{H, \lambda\lambda^{+}_{\min}(\mathbf{W})\}}} \log \frac{\|\mathbf{x}_0 - \mathbf{x}^\star\|^2}{\varepsilon} \log \frac{1}{\delta}\right) \tag{25}$$

Calculate the following

$$\sqrt{\frac{H + \lambda\lambda_{\max}(\mathbf{W})}{\max\{H, \lambda\lambda^{+}_{\min}(\mathbf{W})\}}} = \min\left\{\sqrt{\frac{H + \lambda\lambda_{\max}(\mathbf{W})}{H}}, \sqrt{\frac{H + \lambda\lambda_{\max}(\mathbf{W})}{\lambda\lambda^{+}_{\min}(\mathbf{W})}}\right\}$$

Taking $H$ be equal to $L$, we get

$$N_{\mathbf{W}\mathbf{x}} = \mathcal{O}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\frac{\lambda\lambda_{\max}(\mathbf{W})}{\lambda\lambda^{+}_{\min}(\mathbf{W})}}\right\}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\log\frac{1}{\delta}\right) \tag{26}$$

**Case 2** $\lambda\lambda_{\max}(\mathbf{W}) < L$ Then we take $g(x)$ like sum component, $f(x)$ like $\frac{\lambda}{2}\langle\mathbf{x}, \mathbf{W}\mathbf{x}\rangle$ and we know that number of calls of gradient of $f$:

$$N_{\mathbf{W}\mathbf{x}} = \mathcal{O}\left(\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\right) \tag{27}$$

we know that number of calls of gradient of $g$:

$$N_{\nabla f_k} = \mathcal{O}\left(\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}\sqrt{\frac{L+H}{\mu+H}}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\log\frac{1}{\delta}\right),$$

where $\delta$ is accuracy of solution to the auxiliary problem 1. Taking $H$ be equal to $\lambda\lambda_{\max}(\mathbf{W})$, we get

$$N_{\nabla f_k} = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\log\frac{1}{\delta}\right),$$

Overall, we have

$$N_{\mathbf{W}\mathbf{x}} = \mathcal{O}\left(\min\left\{\sqrt{\frac{\lambda\lambda_{\max}(\mathbf{W})}{\mu}}, \sqrt{\frac{L}{\mu}\frac{\lambda\lambda_{\max}(\mathbf{W})}{\lambda\lambda^{+}_{\min}(\mathbf{W})}}\right\}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\log\frac{1}{\delta}\right) \tag{28}$$

and

$$N_{\nabla f_k} = \mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\frac{\|\mathbf{x}_0 - \mathbf{x}^{\star}\|^2}{\varepsilon}\log\frac{1}{\delta}\right) \tag{29}$$

Now we consider $\delta$ accuracy of auxiliary problem 1. According to theorem from [13], we can take $\delta$ like this

$$\delta = \frac{\varepsilon\mu}{864^2(L + \lambda\lambda_{\max}(\mathbf{W}) + H)^2},$$

because function $f(x)$ is $L$-smooth and $\frac{\lambda}{2}\langle\mathbf{x}, \mathbf{W}\mathbf{x}\rangle$ is $\lambda\lambda_{\max}(\mathbf{W})$-smooth.