Optimal and Practical Algorithms for Smooth and Strongly Convex Decentralized Optimization

Dmitry Kovalev　　**Adil Salim**　　**Peter Richtárik**

King Abdullah University of Science and Technology
Thuwal, Saudi Arabia

Abstract

We consider the task of decentralized minimization of the sum of smooth strongly convex functions stored across the nodes of a network. For this problem, lower bounds on the number of gradient computations and the number of communication rounds required to achieve ε accuracy have recently been proven. We propose two new algorithms for this decentralized optimization problem and equip them with complexity guarantees. We show that our first method is optimal both in terms of the number of communication rounds and in terms of the number of gradient computations. Unlike existing optimal algorithms, our algorithm does not rely on the expensive evaluation of dual gradients. Our second algorithm is optimal in terms of the number of communication rounds, without a logarithmic factor. Our approach relies on viewing the two proposed algorithms as accelerated variants of the Forward Backward algorithm to solve monotone inclusions associated with the decentralized optimization problem. We also verify the efficacy of our methods against state-of-the-art algorithms through numerical experiments.

1 Introduction

In this paper we are concerned with the design and analysis of new efficient algorithms for solving optimization problems in a *decentralized* storage and computation regime. In this regime, a network of agents/devices/workers, such as mobile devices, hospitals, wireless sensors, or smart home appliances, collaborates to solve a single optimization problem whose description is stored across the nodes of the network. Each node can perform computations using its local state and data, and is only allowed to communicate with its neighbors.

Problems of this form have been traditionally studied in the signal processing community (Xu et al., 2020), but are attracting increasing interest from the machine learning and optimization community as well (Scaman et al., 2017). Indeed, the training of supervised machine learning models via empirical risk minimization from training data stored across a network is most naturally cast as a decentralized optimization problem. Finally, while current federated learning (Konečný et al., 2016; McMahan et al., 2017) systems rely on a star network topology, with a trusted server performing aggregation and coordination placed at the center of the network, advances in decentralized optimization could be useful in new generation federated learning formulations that would rely on fully decentralized computation (Li et al., 2019). In summary, decentralized optimization is of direct relevance to machine learning, present and future.

1.1 Formalism

Formally, given an undirected connected network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with nodes/vertices $\mathcal{V} = \{1, \ldots, n\}$ and edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, we consider optimization problems of the form

$$\min_{x \in \mathbb{R}^d} \sum_{i \in \mathcal{V}} f_i(x), \tag{1}$$

where the data describing functions $f_i : \mathbb{R}^d \to \mathbb{R}$ is stored on node i and not directly available to any other node. Decentralized algorithms for solving this problem need to respect the network structure of the problem, which is to say that computation can only be made on the nodes $i \in \mathcal{V}$ from data and information available on the nodes, and communication is constrained to only happen along the edges $e \in \mathcal{E}$.

1.2 Computation and communication

Several decentralized gradient-type algorithms have been proposed to solve (1) in the smooth and strongly convex regime. Two key efficiency measures used to compare such methods are: i) the *number of gradient evaluations* (where one gradient evaluation refers to computing $\nabla f_i(x_i)$ for all $i \in \mathcal{V}$ for some input vectors x_i), and ii) the *number of communication rounds*, where one round allows each node to send $\mathcal{O}(1)$ vectors of size d to their neighbors. If computation is costly, the first comparison metric is more important, and if communication is costly, the second is more important.

Note that problem (1) poses certain intrinsic difficulties each method designed for it needs to address. Clearly, more information can be communicated in each communication round if the network $\mathcal{G}$ is "more highly" connected. By χ we denote the *condition number* associated with (the connectivity of) the graph $\mathcal{G}$; a formal definition is given later. Likewise, more computation will be needed if the functions f_i are "more complicated". We will entirely focus on problems where all functions f_i are L-smooth and μ-strongly convex, which naturally leads to the quantity $\kappa := L/\mu$ as a condition number associated with computation.

Much of decentralized optimization research is focused on designing decentralized algorithms with computation and communication guarantees which have as good as possible dependence on the intrinsic properties of the problem, i.e., on the condition numbers κ and χ.

2 Related Work and Contributions

In this section we first briefly review some of the key results on decentralized optimization, and subsequently provide a brief summary of our key contributions.

2.1 Related work

Existing gradient-type decentralized methods for solving problem (1) can be informally classified into three classes: *non-accelerated* algorithms, *accelerated* algorithm and *optimal* algorithms.

Non-accelerated methods. Loosely speaking, a method is non-accelerated if it has at least a linear dependence on the condition numbers κ and χ, i.e., $\mathcal{O}(\kappa)$ and $\mathcal{O}(\chi)$. Please refer to (Xu et al., 2020, Table 1) for a summary of many such methods, see also (Alghunaim et al., 2019; Li and Lin, 2020). Xu et al. (2020) provide a tight unified analysis of many of these nonaccelerated algorithms, and relies on similar tools as those used in this paper, such as operator splitting and Chebyshev acceleration.

Accelerated methods. Accelerated methods have an improved (sublinear) dependence on the condition numbers, typically $\mathcal{O}(\sqrt{\kappa})$ and $\mathcal{O}(\sqrt{\chi})$. Accelerated algorithms include accelerated DNGD of Qu and Li (2020) and accelerated EXTRA of Li and Lin (2020); the latter using the Catalyst (Lin et al., 2017) framework to accelerate EXTRA (Shi et al., 2015). Additional accelerated methods include, the Accelerated Penalty Method of Li et al. (2018); Dvinskikh et al. (2019), SSDA and MSDA of Scaman et al. (2017) and Accelerated Dual Ascent of Uribe et al. (2020).

Optimal algorithms. Scaman et al. (2017) provide *lower bounds* for the gradient computation and communication complexities of finding an ε-accurate solution; see Section 3.2 below. There have

been several attempts to match these lower bounds, which include algorithms summarized in Table 1. Note, that gradient computation complexity is left as N/A for SSDA and MSDA. This is because they rely on the computation of the gradient of the Fenchel conjugate of f_i, called *dual gradients* in the sequel, which can be intractable. Indeed, computing a dual gradient can be as hard as minimizing f_i. Finally, we remark that Scaman et al. (2018) provide lower bounds in the nonsmooth regime as well, and an algorithm matching this lower bound is called MSPD. MSPD is primal dual Chambolle and Pock (2011), similarly to the algorithms developed in this paper.

Table 1: Comparison of existing state of the art decentralized algorithms with our results in terms of gradient computation and communication complexity of finding x such that $\|x - x^*\|^2 \le \varepsilon$, where x^* is a solution to Problem (1)

Algorithm	Gradient computation complexity	Communication complexity
Existing State of the art Decentralized Algorithms		
Accelerated Dual Ascent Uribe et al. (2020)	$\mathcal{O}\left(\kappa\sqrt{\chi}\log^2\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$
Single/Multi Step Dual Ascent Scaman et al. (2017)	N/A	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$
Accelerated Penalty Method Li et al. (2018); Dvinskikh et al. (2019)	$\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log^2\frac{1}{\varepsilon}\right)$
Accelerated EXTRA Li and Lin (2020)	$\mathcal{O}\left(\sqrt{\kappa\chi}\log(\kappa\chi)\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log(\kappa\chi)\log\frac{1}{\varepsilon}\right)$
Our Results		
Algorithm 1 this paper, Theorem 2	$\mathcal{O}\left(\left(\sqrt{\kappa\chi}+\chi\right)\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\left(\sqrt{\kappa\chi}+\chi\right)\log\frac{1}{\varepsilon}\right)$
Algorithm 2 this paper, Corollary 1	$\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$
Algorithm 3 this paper, Appendix	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$
Lower bounds Scaman et al. (2017)	$\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$

2.2 Summary of contributions

The starting point of this paper is the realization that, *to the best of our knowledge, in the class of algorithms not relying on the computation of the dual gradients, there is no algorithm optimal in communication complexity, and as a result, no algorithm optimal in both gradient computation and communication complexity.* To remedy this situation, we do the following:

- We propose a new accelerated decentralized algorithm not relying on dual gradients: Accelerated Proximal Alternating Predictor-Corrector (APAPC) method (Algorithm 1). We show that in order to obtain x for which $\|x - x^*\|^2 \le \varepsilon$, where x^* is the solution of (1), this method only needs

$$\mathcal{O}((\sqrt{\kappa\chi}+\chi)\log(1/\varepsilon))$$

  gradient computations and communication rounds (Theorem 2). When combined with Chebyshev acceleration, similarly to the trick used in (Scaman et al., 2017, Section 4.2), we show that our method, which we then call Optimal Proximal Alternating Predictor-Corrector (OPAPC) method (Algorithm 2), leads to an *optimal* decentralized method both in terms of gradient computation and communication complexity (Corollary 1). In particular, OPAPC finds an ε-solution in at most

$$\mathcal{O}\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$$

  gradient computations and at most

$$\mathcal{O}\left(\sqrt{\kappa\chi}\log(1/\varepsilon)\right)$$

3

communication rounds. Algorithm 2 reaches the lower bounds (Theorem 1), and hence it is indeed optimal.

- We also propose another accelerated algorithm (Algorithm 3) not relying on dual gradients, one that is optimal in communication complexity (this algorithm is presented in the appendix only). Compared to the above development, this algorithm has the added advantage that it requires the computation of a single gradient per communication step. This can have practical benefits when communication is expensive.

3 Background

3.1 Basic formulation of the decentralized problem

Problem (1) can be reformulated as a lifted (from $\mathbb{R}^d$ to $\mathbb{R}^{dn}$) optimization problem via consensus constraints:

$$\min_{\substack{x_1,\ldots,x_n\in\mathbb{R}^d \\ x_1=\ldots=x_n}} \sum_{i\in\mathcal{V}} f_i(x_i). \tag{2}$$

Consider the function $F : (\mathbb{R}^d)^{\mathcal{V}} \to \mathbb{R}$ defined by $F(x_1,\ldots,x_n) = \sum_{i\in\mathcal{V}} f_i(x_i)$, where $x_1,\ldots,x_n \in \mathbb{R}^d$. Then, F is μ-strongly convex and L-smooth since the individual functions f_i are. Consider also any linear operator (equivalently, any matrix) $\mathbf{W} : (\mathbb{R}^d)^{\mathcal{V}} \to (\mathbb{R}^d)^{\mathcal{V}}$ such that $\mathbf{W}(x_1,\ldots,x_n) = 0$ if and only if $x_1 = \ldots = x_n$. Denoting $x = (x_1,\ldots,x_n) \in (\mathbb{R}^d)^{\mathcal{V}}$, Problem (2) is equivalent to

$$\min_{x\in\mathrm{ker}(\mathbf{W})} F(x). \tag{3}$$

Many optimization algorithms converge exponentially fast (i.e., linearly) to a minimizer of Problem (3), e.g. the projected gradient algorithm. However, only few of them are decentralized. A decentralized algorithm typically relies on multiplication by $\mathbf{W}$, in cases where $\mathbf{W}$ is a *gossip matrix*. Consider a $n \times n$ matrix $\hat{\mathbf{W}}$ satisfying the following properties: 1) $\hat{\mathbf{W}}$ is symmetric and positive semi definite, 2) $\hat{\mathbf{W}}_{i,j} \neq 0$ if and only if $i = j$ or $(i,j) \in \mathcal{E}$, and 3) $\mathrm{ker}\,\hat{\mathbf{W}} = \mathrm{span}(\mathbb{1})$, where $\mathbb{1} = (1,\ldots,1)^{\top}$. Such a matrix is called a *gossip matrix*. A typical example is the Laplacian of the graph $\mathcal{G}$. Denoting $\mathbf{I}$ the $d \times d$ identity matrix and $\otimes$ the Kronecker product, consider $\mathbf{W} : (\mathbb{R}^d)^{\mathcal{V}} \to (\mathbb{R}^d)^{\mathcal{V}}$ the $nd \times nd$ matrix defined by $\mathbf{W} := \hat{\mathbf{W}} \otimes \mathbf{I}$. This matrix can be represented as a block matrix $\mathbf{W} = (\mathbf{W}_{i,j})_{(i,j)\in\mathcal{V}^2}$, where each block $\mathbf{W}_{i,j} = \hat{\mathbf{W}}_{i,j}\mathbf{I}$ is a $d \times d$ matrix proportional to $\mathbf{I}$. In particular, if $d = 1$, then $\mathbf{W} = \hat{\mathbf{W}}$. Moreover, $\mathbf{W}$ satisfy similar properties to $\hat{\mathbf{W}}$:

1. $\mathbf{W}$ is symmetric and positive semi definite,

2. $\mathbf{W}_{i,j} \neq 0$ if and only if $i = j$ or $(i,j) \in \mathcal{E}$,

3. $\mathrm{ker}\,\mathbf{W}$ is the consensus space, $\mathrm{ker}(\mathbf{W}) = \{(x_1,\ldots,x_n) \in (\mathbb{R}^d)^{\mathcal{V}}, x_1 = \cdots = x_n\}$,

4. $\lambda_{\max}(\mathbf{W}) = \lambda_{\max}(\hat{\mathbf{W}})$ and $\lambda_{\min}^+(\mathbf{W}) = \lambda_{\min}^+(\hat{\mathbf{W}})$, where $\lambda_{\max}$ (resp. $\lambda_{\min}^+$) denotes the largest (resp. the smallest positive) eigenvalue.

Throughout the paper, we denote $\mathbf{W}^{\dagger} : \mathrm{range}(\mathbf{W}) \to \mathrm{range}(\mathbf{W})$ the inverse of the map $\mathbf{W} : \mathrm{range}(\mathbf{W}) \to \mathrm{range}(\mathbf{W})$. The operator $\mathbf{W}^{\dagger}$ is positive definite over $\mathrm{range}(\mathbf{W})$ and we denote $\|y\|_{\mathbf{W}^{\dagger}}^2 = \langle \mathbf{W}^{\dagger}y, y \rangle$ for every $y \in \mathrm{range}(\mathbf{W})$. With a slight abuse of language, we shall say that $\mathbf{W}$ is a gossip matrix. Note that decentralized communication can be represented as a multiplication of $\mathbf{W}$ by a vector $x \in (\mathbb{R}^d)^{\mathcal{V}}$. Indeed, the ith component of $\mathbf{W}x$ is a linear combination of x_j, where j is a neighbor of i (we shall write $j \sim i$). In other words, one matrix vector multiplication involving $\mathbf{W}$ is equivalent to one communication round.

In the rest of the paper, our goal is to solve the equivalent problem (3) *with* $\mathbf{W}$ *being a gossip matrix via an optimization algorithm which uses only evaluations of* ∇F *and multiplications by* $\mathbf{W}$.

3.2 Lower bounds

Linearly converging decentralized algorithms using a gossip matrix $\mathbf{W}$ often have a linear rate depending on the condition number of the f_i, $\kappa := \frac{L}{\mu}$ and the condition number (or spectral gap) of

$\mathbf{W}$, $\chi(\mathbf{W}) := \frac{\lambda_{\max}(\mathbf{W})}{\lambda_{\min}^+(\mathbf{W})}$. Indeed, the spectral gap of the Laplacian matrix is known to be a measure of the connectivity of the graph.

In this paper, we define *the class of (first order) decentralized algorithms* as the subset of black box optimization procedure (Scaman et al., 2017, Section 3.1) not using dual gradients, i.e. a decentralized algorithm is not allowed to compute ∇f_i^* (a formal definition is given in the Supplementary material). Complexity lower bounds for solving Problem (1) by a black-box optimization procedure are given by Scaman et al. (2017). These lower bounds relate the number of gradient computations (resp. number of communication rounds) to achieve ε accuracy to the condition numbers κ and $\chi(\mathbf{W})$. Since a decentralized algorithm is a black-box optimization procedure, these lower bounds apply to decentralized algorithms. Therefore, we obtain our first result as a direct application of (Scaman et al., 2017, Corollary 2).

Theorem 1 (Scaman et al. (2017)). *Let $\chi \geq 1$. There exist a gossip matrix $\mathbf{W}$ with condition number χ, and a family of smooth strongly convex functions $(f_i)_{i \in \mathcal{V}}$ with condition number $\kappa > 0$ such that the following holds: for any $\varepsilon > 0$, any decentralized algorithm requires at least $\Omega\left(\sqrt{\kappa\chi}\log(1/\varepsilon)\right)$ communication rounds, and at least $\Omega\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$ gradient computations to output $x = (x_1, \ldots, x_n)$ such that $\|x - x^*\|^2 < \varepsilon$, where $x^* = \arg\min F$.*

Although the lower bounds of Theorem 1 are obvious consequences of (Scaman et al., 2017, Corollary 2), their tightness is not. Indeed, the lower bounds of Theorem 1 are tight on the class of black-box optimization procedures since they are achieved by MSDA Scaman et al. (2017). However, MSDA uses dual gradients and whether these lower bounds are tight on the class of decentralized algorithms is not known. In this paper, we propose decentralized algorithms achieving these lower bounds, showing in particular that they are tight.

3.3 Operator splitting

Recall that in this paper, any optimization algorithm solving Problem (3) by using evaluations of ∇F and multiplications by the gossip matrix $\mathbf{W}$ only is a decentralized algorithm. Such algorithms can be obtained in several ways, e.g., by applying operator splitting methods to primal dual reformulations of Problem (3), see Condat et al. (2019). This is the approach we chose in this work.

We now provide some minimal background knowledge on the Forward Backward algorithm involving monotone operators. We restrict ourselves to single valued, continuous monotone operators. For the general case of set valued monotone operators, the reader is referred to Bauschke and Combettes (2011).

Let E be an Euclidean space and denote $\langle \cdot, \cdot \rangle_\mathsf{E}, \|\cdot\|_\mathsf{E}$ its inner product and the associated norm. Given $\nu \in \mathbb{R}$, a map $A : \mathsf{E} \to \mathsf{E}$ is *ν-monotone* if for every $x, y \in \mathsf{E}$,

$$\langle A(x) - A(y), x - y \rangle_\mathsf{E} \geq \nu \|x - y\|_\mathsf{E}^2.$$

If $\nu < 0$, A is *weakly monotone*, if $\nu > 0$, A is *strongly monotone* and if $\nu = 0$ then A is *monotone*. In this paper, a monotone operator is defined as a monotone continuous map. For every monotone operator and every $\gamma > 0$, the map $I + \gamma A : \mathsf{E} \to \mathsf{E}$ is one-to-one and its inverse $J_{\gamma A} = (I + \gamma A)^{-1} : \mathsf{E} \to \mathsf{E}$, called *resolvent*, is well defined. Let F be a *smooth* convex function, i.e., F is differentiable and its gradient is Lipschitz continuous. Then ∇F is a monotone operator, and the resolvent $J_{\gamma \nabla F}$ is the proximity operator of γF. However, there exist monotone operators which are not gradients of convex functions. For instance, a skew symmetric operator S on E defines the linear map $x \mapsto Sx$ which is not a gradient. This map is a monotone operator since $\langle Sx, x \rangle_\mathsf{E} = 0$. The *set of zeros* of A, defined as $Z(A) := \{x \in \mathsf{E}, A(x) = 0\}$, is often of interest in optimization. For instance, $Z(\nabla F) = \arg\min F$.

Forward Backward.   In order to find an element in $Z(A + B)$, where B is another monotone operator, the *Forward Backward algorithm* iterates

$$x^{k+1} = J_B(x^k - A(x^k)). \tag{4}$$

Note that if $A = \nabla F$ and $B = \nabla G$, where G is another differentiable convex function, the Forward Backward algorithm boils down to the proximal gradient algorithm. In this particular case, Nesterov acceleration can be applied to (4) and leads to faster convergence rates compared to the proximal gradient algorithm (Nesterov, 1983; Beck and Teboulle, 2009).

Generalized Forward Backward.   For every positive definite operator $\mathbf{P}$ on E, the algorithm

$$x^{k+1} = J_{\mathbf{P}^{-1}B}(x^k - \mathbf{P}^{-1}A(x^k)), \tag{5}$$

called the *Generalized Forward Backward* method, can be seen as an instance of (4) because $Z(\mathbf{P}^{-1}A + \mathbf{P}^{-1}B) = Z(A + B)$ and $\mathbf{P}^{-1}A$, $\mathbf{P}^{-1}B$ are monotone operators under the inner product induced by $\mathbf{P}$ on E. For example, the gradient of F under this inner product is $\mathbf{P}^{-1}\nabla F$. A primal dual optimization algorithm is an algorithm solving a primal dual formulation of a minimization problem, see below. Many primal dual algorithms can be seen as instances of (5), with general monotone operators A, B, for a well chosen parameter $\mathbf{P}$, see (Condat et al., 2019).

4   New Decentralized Algorithms

4.1   An accelerated primal dual algorithm

Before presenting our algorithm, we introduce an accelerated decentralized algorithm which we then use to motivate the development of our method.

In this section, E is the Euclidean space $\mathsf{E} = (\mathbb{R}^d)^{\mathcal{V}} \times \text{range}(\mathbf{W})$ endowed with the norm $\|(x, y)\|_{\mathsf{E}}^2 := \|x\|^2 + \|y\|_{\mathbf{W}^\dagger}^2$.

Using the first order optimality conditions, a point x^* is a solution to Problem (3) if and only if $\nabla F(x^*) \in \text{range}(\mathbf{W})$ and $x^* \in \text{ker}(\mathbf{W})$. Solving Problem (3) is thus equivalent to finding $(x^*, y^*) \in \mathsf{E}$ such that

$$0 = \nabla F(x^*) + y^*,$$
$$0 = \mathbf{W}x^*. \tag{6}$$

Indeed, the first line of (6) is equivalent to $\nabla F(x^*) = -y^* \in \text{range}\,\mathbf{W}$, because $(x^*, y^*) \in \mathsf{E} = (\mathbb{R}^d)^{\mathcal{V}} \times \text{range}(\mathbf{W})$. The second line of (6) is just a definition of $x^* \in \text{ker}\,\mathbf{W}$. Consider the maps $M, A, B : \mathsf{E} \to \mathsf{E}$ defined by

$$M(x, y) := \begin{bmatrix} \nabla F(x) + y \\ -\mathbf{W}x \end{bmatrix}, \quad A(x, y) := \begin{bmatrix} \nabla F(x) \\ 0 \end{bmatrix}, \quad B(x, y) := \begin{bmatrix} y \\ -\mathbf{W}x \end{bmatrix}.$$

Then M, A and B are monotone operators. Indeed, A is the gradient of the convex function $(x, y) \mapsto F(x)$, B satisfies

$$\langle B(x, y), (x, y) \rangle_{\mathsf{E}} = \langle x - \mathbf{W}^\dagger \mathbf{W}x, y \rangle = 0$$

for every $(x, y) \in \mathsf{E}$ (since $y \in \text{range}(\mathbf{W})$), and $M = A + B$. Moreover, $M(x^*, y^*) = 0$, i.e., (x^*, y^*) is a zero of M.

One idea to solve (6) is therefore to apply Algorithm (4) to the sum $A + B$. However, computing the resolvent J_B in a decentralized way across the network $\mathcal{G}$ is notably challenging. Another idea is to apply (5) using the symmetric positive definite operator $P : \mathsf{E} \to \mathsf{E}$ defined by

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\eta}\mathbf{I} & 0 \\ 0 & \frac{1}{\theta}\mathbf{I} - \eta\mathbf{W} \end{bmatrix}.$$

Indeed, for every $(x, y) \in \mathsf{E}$, $(x', y') = J_{P^{-1}B}(x, y)$ implies $x' = x - \eta y'$ and $\frac{1}{\theta}(y' - y) - \eta\mathbf{W}(y' - y) = \mathbf{W}x' = \mathbf{W}(x - \eta y')$. Therefore, $y' = y + \theta\mathbf{W}(x - \eta y)$, and the computation of $J_{P^{-1}B}$ only requires one multiplication by $\mathbf{W}$, i.e., one local communication round. The resulting algorithm is

$$y^{k+1} = y^k + \theta\mathbf{W}(x^k - \eta\nabla F(x^k) - \eta y^k),$$
$$x^{k+1} = x^k - \eta\nabla F(x^k) - \eta y^{k+1}. \tag{7}$$

Remark 1. *The Proximal Alternating Predictor–Corrector (PAPC) algorithm, a.k.a. Loris–Verhoven (Loris and Verhoeven, 2011; Drori et al., 2015; Chen et al., 2013; Condat et al., 2019) is a primal dual algorithm that can tackle Problem* (3). *Up to a change of variable, Algorithm* (7) *can be shown to be equivalent to PAPC applied to* (3). *Moreover, it was already noticed that the PAPC can be represented as a Forward Backward algorithm* (5) *(Condat et al., 2019).*

Invoking a complexity result on the PAPC from Salim et al. (2020), the complexity of Algorithm (7) is $\mathcal{O}\left((\kappa + \chi(\mathbf{W}))\log(1/\varepsilon)\right)$, both in communication and gradient computations. This complexity is equivalent to that of the best performing non accelerated algorithm proposed recently, such as Exact diffusion, NIDS and EXTRA (see Li and Lin (2020); Xu et al. (2020)). In spite of this, we are able to accelerate the convergence of Algorithm (7).

In particular, we propose a new algorithm that can be seen as an accelerated version of Algorithm (7). The proposed algorithm (APAPC) is defined in Algorithm 1 , and its complexity is given in Theorem 2. We prove that the complexity of APAPC is $\mathcal{O}((\sqrt{\kappa\chi(\mathbf{W})}+\chi(\mathbf{W}))\log(1/\varepsilon))$, both in communication rounds and gradient computations. The proposed algorithm is accelerated because its dependence on the condition number κ is $\mathcal{O}(\sqrt{\kappa})$ instead of $\mathcal{O}(\kappa)$.

Algorithm 1 Accelerated PAPC (APAPC)

1: **Parameters:** $x^0 \in \mathbb{R}^{nd}, y^0 \in \text{range}\mathbf{W}, \eta, \theta, \alpha > 0, \tau \in (0,1)$
2: Set $x_f^0 = x^0$
3: **for** $k = 0, 1, 2, \ldots$ **do**
4:     $x_g^k = \tau x^k + (1-\tau)x_f^k$
5:     $x^{k+1/2} = (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + y^k))$
6:     $y^{k+1} = y^k + \theta\mathbf{W}x^{k+1/2}$
7:     $x^{k+1} = (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + y^{k+1}))$
8:     $x_f^{k+1} = x_g^k + \frac{2\tau}{2-\tau}(x^{k+1} - x^k)$
9: **end for**

Theorem 2 (Accelerated PAPC). *Set the parameters $\eta, \theta, \alpha, \tau$ to $\eta = \frac{1}{4\tau L}$, $\theta = \frac{1}{\eta\lambda_{\max}(\mathbf{W})}$, $\alpha = \mu$, and $\tau = \min\left\{1, \frac{1}{2}\sqrt{\frac{\chi(\mathbf{W})}{\kappa}}\right\}$. Then,*

$$\frac{1}{\eta}\left\|x^k - x^*\right\|^2 + \frac{2(1-\tau)}{\tau}\text{D}_F(x_f^k, x^*) \leq \left(1 + \frac{1}{4}\min\left\{\frac{1}{\sqrt{\kappa\chi(\mathbf{W})}}, \frac{1}{\chi(\mathbf{W})}\right\}\right)^{-k}C,$$

where D_F is the Bregman divergence of F and $C := \frac{1}{\eta}\left\|x^0 - x^*\right\|^2 + \frac{1}{\theta}\|y^0 - y^*\|_{\mathbf{W}^\dagger}^2 + \frac{2(1-\tau)}{\tau}\text{D}_F(x_f^0, x^*)$. Moreover, for every $\varepsilon > 0$, APAPC finds x^k for which $\|x^k - x^*\|^2 \leq \varepsilon$ in at most $\mathcal{O}\left(\left(\sqrt{\kappa\chi(\mathbf{W})} + \chi(\mathbf{W})\right)\log(1/\varepsilon)\right)$ computations (resp. communication rounds).

The proposed algorithm 1 provably accelerates Algorithm (7). The proof intuitively relies on viewing Algorithm 1 as an accelerated version of (5), although Nesterov's acceleration does not apply to general monotone operators *a priori*.

4.2 A decentralized algorithm optimal both in communication and computation complexity

As mentioned before, while APAPC is accelerated, it is not optimal. We now derive a variant which is optimal both in gradient computations and communication rounds. Following Scaman et al. (2017, Section 4.2), our main tool to derive the new decentralized optimal algorithm is the Chebyshev acceleration (Scaman et al., 2017; Arioli and Scott, 2014).

In particular, there exists a polynomial P such that

   (i) $P(\mathbf{W})$ is a Gossip matrix,

(ii) multiplication by $P(\mathbf{W})$ requires $\left\lfloor \sqrt{\chi(\mathbf{W})} \right\rfloor$ multiplications by $\mathbf{W}$ (i.e., communication rounds) and is described by the subroutine ACCELERATEDGOSSIP proposed in (Scaman et al., 2017, Algorithm 2) and recalled in Algorithm 2 for the ease of reading,

(iii) $\chi(P(\mathbf{W})) \leq 4$.

Therefore, one can replace $\mathbf{W}$ by $P(\mathbf{W})$ in Problem (3) to obtain an equivalent problem. Applying APAPC to the equivalent problem leads to a linearly converging decentralized algorithm. This new algorithm, called Optimal PAPC (OPAPC), is formalized as Algorithm 2.

Algorithm 2 Optimal PAPC (OPAPC)

1: **Parameters:** $x^0 \in \mathbb{R}^{nd}, y^0 \in \text{range} P(\mathbf{W}), T \in \mathbb{N}^*, c_1, c_2, c_3, \eta, \theta, \alpha > 0, \tau \in (0,1)$
2: Set $x_f^0 = x^0$
3: **for** $k = 0, 1, 2, \dots$ **do**
4: $\quad x_g^k = \tau x^k + (1-\tau)x_f^k$
5: $\quad x^{k+1/2} = (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + y^k))$
6: $\quad y^{k+1} = y^k + \theta\text{ACCELERATEDGOSSIP}(\mathbf{W}, x^{k+1/2}, T)$
7: $\quad x^{k+1} = (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + y^{k+1}))$
8: $\quad x_f^{k+1} = x_g^k + \frac{2\tau}{2-\tau}(x^{k+1} - x^k)$
9: **end for**
10: **procedure** ACCELERATEDGOSSIP($\mathbf{W}, x, T$)
11: $\quad$ Set $a_0 = 1, a_1 = c_2, x_0 = x, x_1 = c_2(I - c_3\mathbf{W})x$
12: $\quad$ **for** $i = 1, \dots, T-1$ **do**
13: $\quad\quad a_{i+1} = 2c_2 a_i - a_{i-1}$
14: $\quad\quad x_{i+1} = 2c_2(I - c_3\mathbf{W})x_i - x_{i-1}$
15: $\quad$ **end for**
$\quad\quad$ **return** $x - \frac{x_T}{a_T}$
16: **end procedure**

Using the properties of $P(\mathbf{W})$ mentioned above, we obtain the following corollary of Theorem 2.

Corollary 1 (Optimal PAPC). *Set the parameters* $T, c_1, c_2, c_3, \eta, \theta, \alpha, \tau$ *to*

$$T = \left\lfloor \sqrt{\chi(\mathbf{W})} \right\rfloor, \quad c_1 = \frac{\sqrt{\chi(\mathbf{W})} - 1}{\sqrt{\chi(\mathbf{W})} + 1}, \quad c_2 = \frac{\chi(\mathbf{W}) + 1}{\chi(\mathbf{W}) - 1}, \quad c_3 = \frac{2\chi(\mathbf{W})}{(1 + \chi(\mathbf{W}))\lambda_{\max}(\mathbf{W})},$$

$$\eta = \frac{1}{4\tau L}, \quad \theta = \frac{1 + c_1^{2T}}{\eta(1 + c_1^T)^2}, \quad \alpha = \mu, \quad \tau = \min\left\{1, \frac{1 + c_1^T}{2\sqrt{\kappa}(1 - c_1^T)}\right\}.$$

Then, there exists $C \geq 0$ *such that*

$$\frac{1}{\eta}\left\|x^k - x^*\right\|^2 + \frac{2 - \tau}{\tau}\mathrm{D}_F(x_f^k, x^*) \leq \left(1 + \frac{1}{16}\min\left\{\frac{2}{\sqrt{\kappa}}, 1\right\}\right)^{-k} C.$$

Moreover, for every $\varepsilon > 0$, *OPAPC finds* x^k *for which* $\|x^k - x^*\|^2 \leq \varepsilon$ *in at most* $\mathcal{O}\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$ *gradient computations and at most* $\mathcal{O}\left(\sqrt{\kappa\chi(\mathbf{W})}\log(1/\varepsilon)\right)$ *communication rounds.*

The Algorithm 2 achieves both the lower bounds of Theorem 1. In particular, the lower bounds of Theorem 1 are tight.

5 Numerical Experiments

In this section, we perform experiments with logistic regression for binary classification with ℓ^2 regularizer, where our loss function has the form

$$f_i(x) = \frac{1}{m}\sum_{j=1}^{m}\log(1 + \exp(-b_{ij}a_{ij}^\top x)) + \frac{r}{2}\|x\|^2,$$

where $a_{ij} \in \mathbb{R}^d$, $b_{ij} \in \{-1, +1\}$ are data points, r is the regularization parameter, m is the number of data points stored on each node.

In our experiments we used $10,000$ data samples randomly distributed to the nodes of network of size $n = 100$, $m = 100$ samples per each node. We used 2 networks: 10×10 grid and Erdös-Rényi random graph of average degree 6. Same setup was tested by Scaman et al. (2017).

We use three LIBSVM[1] datasets: a6a, w6a, ijcnn1. The regularization parameter was chosen so that $\kappa \approx 10^3$. Additional experiments with synthetic data are given in the Supplementary material.

(a) Communication complexity. (b) Gradient computation complexity.
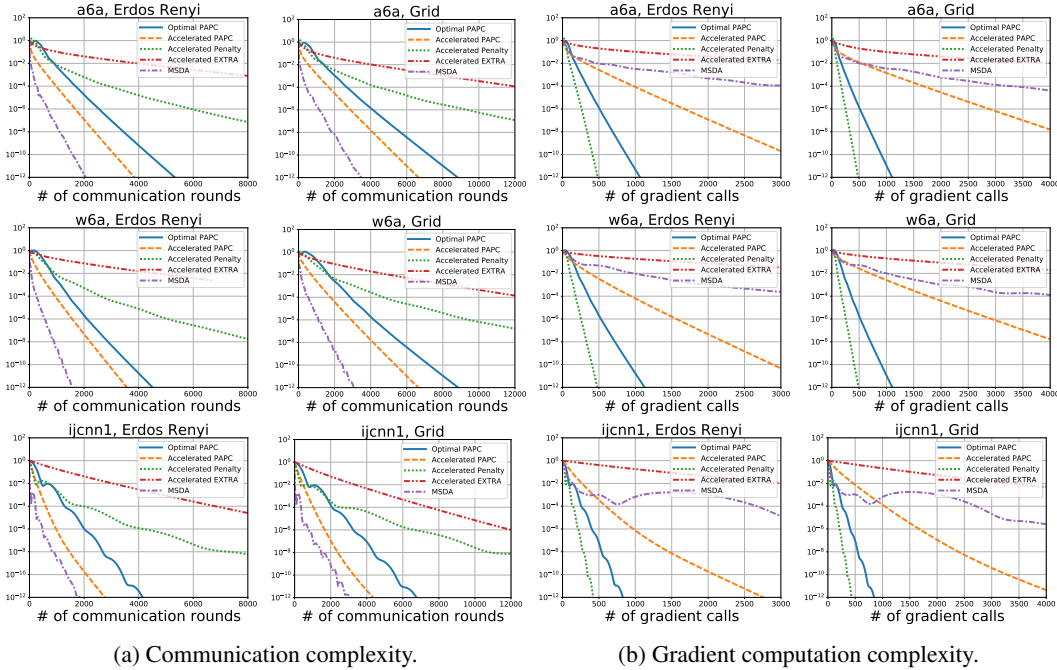
Figure 1: Linear convergence of decentralized algorithms in number of communication rounds and gradient computations.

Figure 1 compares Algorithm 1 (Accelerated PAPC) and Algorithm 2 (Optimal PAPC) with three state-of-the-art accelerated benchmarks: Accelerated Penalty (Li et al., 2018; Dvinskikh et al., 2019), Accelerated Extra (Li and Lin, 2020) and MSDA Scaman et al. (2017), where we used the subroutine of Uribe et al. (2020) to compute the dual gradients. This subroutine uses primal gradients ∇f_i, and the resulting algorithm can be shown to have an optimal communication complexity. We represent the squared distance to the solution as a function of the number of communication rounds and (primal) gradient computations.

The theory developed in this paper concerns the value of the linear rates of the proposed algorithms, i.e., the slope of the curves in Figure 1. In communication complexity, one can see that our Algorithms 1 and 2 have similar rate and perform better than the other benchmarks except MSDA. MSDA performs slightly better in communication complexity. However, MSDA uses dual gradients and has much higher iteration complexity. In gradient computation complexity, one can see that our main Algorithm 2 is, alongside Accelerated Penalty, the best performing method. Accelerated Penalty performs slightly better in gradient computation complexity. However, the theory of Accelerated Penalty does not predict linear convergence in the number of communication rounds and we see that this algorithm converges sublinearly. Overall, Optimal PAPC is the only universal method which performs well both in communication rounds and gradient computations.

---

[1]The LIBSVM dataset collection is available at `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

6 Broader Impact

Our paper is of a fundamental theoretical nature. We designed new decentralized optimization algorithms, and proved that they are optimal in a certain rigorous mathematical sense. We trust that our methods will have impact wherever decentralized optimization is needed and used, at present (e.g., sensor networks) and in the future (e.g., fully decentralized federated learning). Having said that, our methods are generic as we did not investigate any particular application. Hence, we do not expect any immediate societal impact beyond impact on the research community developing the foundational tools for AI. We hope, however, that AI practitioners will be inspired by this work and will use the fruits of our labor to benefit humanity through concrete applications and machine learning models.

References

Alghunaim, S. A., Ryu, E. K., Yuan, K., and Sayed, A. H. (2019). Decentralized proximal gradient algorithms with linear convergence rates. *arXiv preprint arXiv:1909.06479*.

Arioli, M. and Scott, J. (2014). Chebyshev acceleration of iterative refinement. *Numerical Algorithms*, 66(3):591–608.

Bauschke, H. H. and Combettes, P. L. (2011). *Convex analysis and monotone operator theory in Hilbert spaces*. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer, New York.

Beck, A. and Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202.

Chambolle, A. and Pock, T. (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145.

Chen, P., Huang, J., and Zhang, X. (2013). A primal–dual fixed point algorithm for convex separable minimization with applications to image restoration. *Inverse Problems*, 29(2):025011.

Condat, L., Kitahara, D., Contreras, A., and Hirabayashi, A. (2019). Proximal splitting algorithms: Relax them all! *arXiv preprint arXiv:1912.00137*.

Drori, Y., Sabach, S., and Teboulle, M. (2015). A simple algorithm for a class of nonsmooth convex–concave saddle-point problems. *Operations Research Letters*, 43(2):209–214.

Dvinskikh, D. and Gasnikov, A. (2019) Decentralized and parallelized primal and dual accelerated methods for stochastic convex programming problems. *arXiv preprint arXiv:1904.09015*.

Konečný, J., McMahan, H. B., Yu, F., Richtárik, P., Suresh, A. T., and Bacon, D. (2016). Federated learning: strategies for improving communication efficiency. In *NIPS Private Multi-Party Machine Learning Workshop*.

Li, H., Fang, C., Yin, W., and Lin, Z. (2018). A sharp convergence rate analysis for distributed accelerated gradient methods. *arXiv preprint arXiv:1810.01053*.

Li, H. and Lin, Z. (2020). Revisiting extra for smooth distributed optimization. *arXiv preprint arXiv:2002.10110*.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2019). Federated learning: challenges, methods, and future directions. *arXiv preprint arXiv:1908.07873*.

Lin, H., Mairal, J., and Harchaoui, Z. (2017). Catalyst acceleration for first-order convex optimization: from theory to practice. *The Journal of Machine Learning Research*, 18(1):7854–7907.

Loris, I. and Verhoeven, C. (2011). On a generalization of the iterative soft-thresholding algorithm for the case of non-separable penalty. *Inverse Problems*, 27(12):125007.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and Agüera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*.

Nesterov, Y. E. (1983). A method for solving the convex programming problem with convergence rate $1/k^2$. In *Dokl. Akad. Nauk SSSR*, volume 269, pages 543–547.

Qu, G. and Li, N. (2020). Accelerated distributed nesterov gradient descent. *IEEE Transactions on Automatic Control*, 65(6):2566–2581.

Salim, A., Condat, L., Mishchenko, K., and Richtárik, P. (2020). Dualize, split, randomize: Fast nonsmooth optimization algorithms. *arXiv preprint arXiv:2004.02635*.

Scaman, K., Bach, F., Bubeck, S., Lee, Y. T., and Massoulié, L. (2017). Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3027–3036.

Scaman, K., Bach, F., Bubeck, S., Massoulié, L., and Lee, Y. T. (2018). Optimal algorithms for non-smooth distributed optimization in networks. In *Advances in Neural Information Processing Systems*, pages 2740–2749.

Shi, W., Ling, Q., Wu, G., and Yin, W. (2015). Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966.

Uribe, C. A., Lee, S., Gasnikov, A., and Nedić, A. (2020). A dual approach for optimal algorithms in distributed optimization over networks. *Optimization Methods and Software*, pages 1–40.

Xu, J., Tian, Y., Sun, Y., and Scutari, G. (2020). Distributed algorithms for composite optimization: Unified and tight convergence analysis. *arXiv preprint arXiv:2002.11534*.

Appendix

Contents

A   Experiments with synthetic data

In this section, we present additional experiments. The experimental setup is the same as before, with only one difference: we use randomly generated dataset with the following choice of the number of features d: 40, 60, 80, 100. The results, which are shown in Figure 2, are similar to the previous results, and the same conclusions can be made.
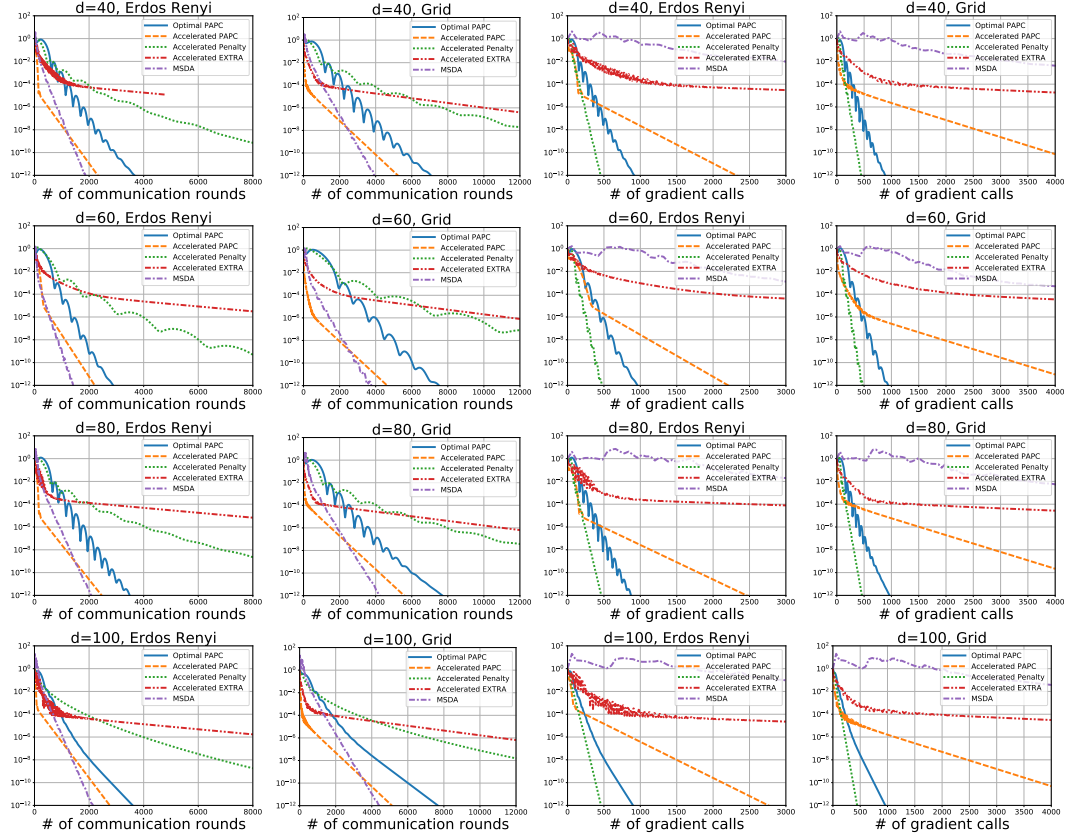
(a) Communication complexity.

(b) Gradient computation complexity.

Figure 2: Linear convergence of decentralized algorithms in number of communication rounds and gradient computations.

B Formal Definition of Decentralized Algorithms

In this paper, we considered the resolution of (1) distributively across the nodes of the network G. Each node $i \in \mathcal{V}$ is associated with a computing agent that only have access to the local function f_i. The goal of the network of computing agent is to minimize the function (1) by performing local computations involving f_i at each node i and by communicating vectors along the edges, i.e., with neighbors $j \sim i$.

More precisely, we considered the class of decentralized algorithms, similarly to (Scaman et al., 2017, Section 3.1). In this paper, a decentralized algorithm is formally defined as an algorithm satisfying the following constraints. At time k, each node i possesses a local internal memory $M_i^k \subset \mathbb{R}^d$ and outputs an estimation $x_i^k \in M_i^k$ of the solution to Problem (1). This internal memory is updated via gradient computations and communication rounds i.e.,

$$M_i^{k+1} \subset \mathrm{Span}(\mathrm{Comm}_i^{k+1} \bigcup \mathrm{Comp}_i^{k+1}),$$

where Comm_i^{k+1} is the communication component and Comp_i^{k+1} the computation component. The communication component is updated by combining the elements of the local memories of nodes $j \sim i$ at time k: $\mathrm{Comm}_i^{k+1} = \mathrm{Span}(\bigcup_{j \sim i} M_j^k)$. The computation component is updated by combining the elements of the local memory of i at time k along with the gradients of the local functions f_i at these elements: $\mathrm{Comp}_i^{k+1} = \mathrm{Span}(\{x, \nabla f_i(x), x \in M_i^k\})$. Compared to the class of black-box optimization procedures of (Scaman et al., 2017), the class of decentralized algorithm is smaller (i.e., included). Indeed, black-box optimization procedures use dual gradients. In other words, they use the following definition of the computation component:

$$\widetilde{\mathrm{Comp}}_i^{k+1} = \{x, \nabla f_i(x), \nabla f_i^*(x), x \in M_i^k\}$$

(where f_i^* is the Fenchel transform of f_i), which is a set containing Comp_i^{k+1}. Recall that computing the dual gradient $\nabla f_i^*(0)$ is equivalent to minimizing f_i.

Finally, as in Scaman et al. (2017), we say that a decentralized algorithm uses the gossip matrix $\mathbf{W}$ if the local communication is achieved by multiplication of a vector by $\mathbf{W}$.

C Proof of Theorem 2 (APAPC)

For every $p \geq 0$, we denote by $\|\cdot\|_\mathbf{P}$ the (semi)-norm induced by any positive (semi)-definite matrix $\mathbf{P} : \mathbb{R}^p \to \mathbb{R}^p$.

Lemma 1. *Let $\mathbf{P} \in \mathbb{R}^{2nd \times 2nd}$ be the following matrix:*

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\eta}\mathbf{I} & 0 \\ 0 & \frac{1}{\theta}\mathbf{W}^\dagger - (1+\eta\alpha)^{-1}\eta\mathbf{I} \end{bmatrix}. \tag{8}$$

If parameters η and θ satisfy

$$\eta\theta\lambda_{\max}(\mathbf{W}) \leq 1, \tag{9}$$

then for all $x \in \mathbb{R}^{nd}$, $y \in \text{range}\mathbf{W}$ the following inequality holds:

$$\frac{1}{\eta}\|x\|^2 \leq \left\| \begin{bmatrix} x \\ y \end{bmatrix} \right\|_\mathbf{P}^2 \leq \frac{1}{\eta}\|x\|^2 + \frac{1}{\theta}\|y\|_{\mathbf{W}^\dagger}^2. \tag{10}$$

Proof. Note that under our assumptions, the matrix $\frac{1}{\theta}\mathbf{W}^\dagger - (1+\eta\alpha)^{-1}\eta\mathbf{I}$ is positive semi-definite on range$\mathbf{W}$. $\square$

Lemma 2. *Let α satisfy $0 \leq \alpha \leq \mu$. Then the following inequality holds:*

$$-\frac{1}{2\eta}\|x^{k+1} - x^k\|^2 \leq -\frac{\eta}{4}\|y^{k+1} - y^*\|^2 + \eta\alpha^2\|x^{k+1} - x^*\|^2 + 2\eta L D_f(x_g^k, x^*). \tag{11}$$

Proof. From line (7) of Algorithm 1 and optimality condition (6) it follows that

$$\|x^{k+1} - x^k\|^2 = \|\eta(y^{k+1} - y^*) + \eta(\nabla F(x_g^k) - \nabla F(x^*) - \alpha(x_g^k - x^*)) + \eta\alpha(x^{k+1} - x^*)\|^2$$
$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2$$
$$\quad - 2\eta^2\|\nabla F(x_g^k) - \nabla F(x^*) - \alpha(x_g^k - x^*)\|^2.$$

Since $f(x) - \frac{\alpha}{2}\|x\|^2$ is a convex and $(L - \alpha)$-smooth function, we can lower bound the last term and get

$$\|x^{k+1} - x^k\|^2 = \|\eta(y^{k+1} - y^*) + \eta(\nabla F(x_g^k) - \nabla F(x^*) - \alpha(x_g^k - x^*)) + \eta\alpha(x^{k+1} - x^*)\|^2$$
$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2 - 4\eta^2(L - \alpha)D_{f - \frac{\alpha}{2}\|\cdot\|^2}(x_g^k, x^*)$$
$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2 - 4\eta^2 L D_f(x_g^k, x^*).$$

Rearranging and dividing by 2η concludes the proof. $\square$

Lemma 3. *Let $\mathbf{P}$ be the matrix defined by (8):*

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\eta}\mathbf{I} & 0 \\ 0 & \frac{1}{\theta}\mathbf{W}^\dagger - (1+\eta\alpha)^{-1}\eta\mathbf{I} \end{bmatrix}. \tag{8}$$

Then the following equality holds:

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \mathbf{W}\mathbf{W}^\dagger x^{k+1} \end{bmatrix}. \tag{12}$$

Proof. From the definition of $\mathbf{P}$ it follows that

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \frac{1}{\eta}(x^{k+1} - x^k) \\ \frac{1}{\theta}\mathbf{W}^\dagger(y^{k+1} - y^k) - (1+\eta\alpha)^{-1}\eta(y^{k+1} - y^k) \end{bmatrix}.$$

From line (7) of Algorithm 1 it follows that

$$\frac{1}{\eta}(x^{k+1} - x^k) = \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}),$$

and hence,

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \frac{1}{\theta}\mathbf{W}^\dagger(y^{k+1} - y^k) - (1+\eta\alpha)^{-1}\eta(y^{k+1} - y^k) \end{bmatrix}.$$

From line (6) of Algorithm 1 it follows that

$$y^{k+1} - y^k = \theta\mathbf{W}x^{k+1/2},$$

and hence,

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \mathbf{W}\mathbf{W}^\dagger x^{k+1/2} - (1+\eta\alpha)^{-1}\eta(y^{k+1} - y^k) \end{bmatrix}.$$

Since $y^k \in \mathrm{range}\mathbf{W}$ for all $k = 0, 1, 2, \ldots$, we have

$$\mathbf{W}\mathbf{W}^\dagger(y^{k+1} - y^k) = y^{k+1} - y^k,$$

and hence we obtain

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \mathbf{W}\mathbf{W}^\dagger\left[x^{k+1/2} - (1+\eta\alpha)^{-1}\eta(y^{k+1} - y^k)\right] \end{bmatrix}.$$

Finally, from lines 5 and 7 of Algorithm 1 it follows that

$$x^{k+1} = x^{k+1/2} + (1+\eta\alpha)^{-1}\eta(y^k - y^{k+1}),$$

and hence,

$$\mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \mathbf{W}\mathbf{W}^\dagger x^{k+1} \end{bmatrix}.$$

$\square$

Lemma 4. *Let parameter η be defined by*

$$\eta = \frac{1}{4\tau L}. \tag{13}$$

Let parameter θ be defined by

$$\theta = \frac{1}{\eta\lambda_{\max}(\mathbf{W})}. \tag{14}$$

Let parameter α be defined by

$$\alpha = \mu. \tag{15}$$

Let parameter τ be defined by

$$\tau = \min\left\{1, \frac{1}{2}\sqrt{\frac{\mu}{L}\frac{\lambda_{\max}(\mathbf{W})}{\lambda_{\min}^+(\mathbf{W})}}\right\}. \tag{16}$$

16

Let Ψ^k be the following Lyapunov function:

$$\Psi^k = \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} + \frac{2(1-\tau)}{\tau} D_F(x_f^k, x^*), \tag{17}$$

where $\mathbf{P}$ is defined by (8):

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\eta}\mathbf{I} & 0 \\ 0 & \frac{1}{\theta}\mathbf{W}^\dagger - (1+\eta\alpha)^{-1}\eta\mathbf{I} \end{bmatrix}. \tag{8}$$

Then the following inequality holds:

$$\Psi^{k+1} \leq \left(1 + \frac{1}{4} \min\left\{ \sqrt{\frac{\mu}{L} \frac{\lambda^+_{\min}(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda^+_{\min}(\mathbf{W})}{\lambda_{\max}(\mathbf{W})} \right\} \right)^{-1} \Psi^k.$$

Proof.

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} = \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} - \left\| \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} \right\|^2_{\mathbf{P}} + 2 \left\langle \mathbf{P} \cdot \begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\rangle$$

Note, that stepsize η defined by (13) and stepsize θ defined by (14) satisfy (9), hence inequality (10) holds. Using (10) and (12) we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} - \frac{1}{\eta}\|x^{k+1} - x^k\|^2$$

$$+ 2 \left\langle \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + y^{k+1}) \\ \mathbf{W}\mathbf{W}^\dagger x^{k+1} \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\rangle$$

$$= \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^{k+1}, x^{k+1} - x^* \rangle$$

$$- 2\langle \nabla F(x_g^k) + y^{k+1}, x^{k+1} - x^* \rangle + 2\langle \mathbf{W}\mathbf{W}^\dagger x^{k+1}, y^{k+1} - y^* \rangle.$$

Since $\mathbf{W}\mathbf{W}^\dagger x^* = 0$ and $\mathbf{W}\mathbf{W}^\dagger(y^{k+1} - y^*) = y^{k+1} - y^*$, we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|^2_{\mathbf{P}} - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^{k+1}, x^{k+1} - x^* \rangle$$

$$- 2\langle \nabla F(x_g^k) + y^{k+1}, x^{k+1} - x^* \rangle + 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

Since $\nabla F(x^*) + y^* = 0$ (optimality condition (6)), we get

$$\left\|\begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 \leq \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^{k+1}, x^{k+1} - x^*\rangle$$
$$- 2\langle \nabla F(x_g^k) - \nabla F(x^*) + y^{k+1} - y^*, x^{k+1} - x^*\rangle + 2\langle x^{k+1} - x^*, y^{k+1} - y^*\rangle$$
$$= \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 - 2\alpha\|x^{k+1} - x^*\|^2$$
$$- 2\alpha\langle x_g^k - x^*, x^{k+1} - x^*\rangle - 2\langle \nabla F(x_g^k) - \nabla F(x^*), x^{k+1} - x^*\rangle.$$

Using Young's inequality $2\langle a, b\rangle \leq \|a\|^2 + \|b\|^2$ we get

$$\left\|\begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 \leq \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 - 2\alpha\|x^{k+1} - x^*\|^2$$
$$+ \alpha\|x_g^k - x^*\|^2 + \alpha\|x^{k+1} - x^*\|^2 - 2\langle \nabla F(x_g^k) - \nabla F(x^*), x^{k+1} - x^*\rangle$$
$$= \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 - \alpha\|x^{k+1} - x^*\|^2 + \alpha\|x_g^k - x^*\|^2$$
$$- 2\langle \nabla F(x_g^k) - \nabla F(x^*), x^{k+1} - x^*\rangle.$$

Now, we use lines 4 and 8 of Algorithm 1 and get

$$\left\|\begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 \leq \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \alpha\|x^{k+1} - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{1}{2\eta}\|x^{k+1} - x^k\|^2$$
$$- \frac{2-\tau}{\tau}\left(\langle \nabla F(x_g^k) - \nabla F(x^*), x_f^{k+1} - x_g^k\rangle + \frac{1}{2\eta}\frac{(2-\tau)}{4\tau}\|x_f^{k+1} - x_g^k\|^2\right)$$
$$- 2\langle \nabla F(x_g^k) - \nabla F(x^*), x_g^k - x^*\rangle + \frac{2(1-\tau)}{\tau}\langle \nabla F(x_g^k) - \nabla F(x^*), x_f^k - x_g^k\rangle.$$

Since parameter η defined by (13) satisfy $\eta \leq \frac{2-\tau}{4\tau L}$, we get

$$\left\|\begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 \leq \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \alpha\|x^{k+1} - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{1}{2\eta}\|x^{k+1} - x^k\|^2$$
$$- \frac{2-\tau}{\tau}\left(\langle \nabla F(x_g^k) - \nabla F(x^*), x_f^{k+1} - x_g^k\rangle + \frac{L}{2}\|x_f^{k+1} - x_g^k\|^2\right)$$
$$- 2\langle \nabla F(x_g^k) - \nabla F(x^*), x_g^k - x^*\rangle + \frac{2(1-\tau)}{\tau}\langle \nabla F(x_g^k) - \nabla F(x^*), x_f^k - x_g^k\rangle.$$

Using μ-strong convexity and L-smoothness of $f(x)$ we get

$$\left\|\begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 \leq \left\|\begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix}\right\|_{\mathbf{P}}^2 - \alpha\|x^{k+1} - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{1}{2\eta}\|x^{k+1} - x^k\|^2$$
$$- \frac{2-\tau}{\tau}\left(\mathrm{D}_F(x_f^{k+1}, x^*) - \mathrm{D}_F(x_g^k, x^*)\right) + \frac{2(1-\tau)}{\tau}\left(\mathrm{D}_F(x_f^k, x^*) - \mathrm{D}_F(x_g^k, x^*)\right)$$

$$- 2 \left(\mathrm{D}_F(x_g^k, x^*) + \frac{\mu}{2} \|x_g^k - x^*\|^2 \right)$$

$$= \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \alpha \|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*)$$

$$+ (\alpha - \mu) \|x_g^k - x^*\|^2 - \frac{1}{2\eta} \|x^{k+1} - x^k\|^2 - \mathrm{D}_F(x_g^k, x^*).$$

Now, we define $\delta = \min \left\{ 1, \frac{1}{2\eta L} \right\}$. Since α defined by (15) satisfies conditions of Lemma 2, we can use (11) and get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \alpha \|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*)$$

$$+ (\alpha - \mu) \|x_g^k - x^*\|^2 - \frac{\delta}{2\eta} \|x^{k+1} - x^k\|^2 - \mathrm{D}_F(x_g^k, x^*)$$

$$\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \alpha \|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*)$$

$$- \frac{\eta\delta}{4} \|y^{k+1} - y^*\|^2 + \eta\alpha^2\delta \|x^{k+1} - x^*\|^2 + 2\eta L\delta \mathrm{D}_f(x_g^k, x^*)$$

$$+ (\alpha - \mu) \|x_g^k - x^*\|^2 - \mathrm{D}_F(x_g^k, x^*)$$

$$\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \alpha \|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*)$$

$$- \frac{\eta\delta}{4} \|y^{k+1} - y^*\|^2 + \frac{\alpha^2}{2L} \|x^{k+1} - x^*\|^2 + (\alpha - \mu) \|x_g^k - x^*\|^2$$

$$= \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \left(\alpha - \frac{\alpha^2}{2L} \right) \|x^{k+1} - x^*\|^2 - \frac{\eta\delta}{4} \|y^{k+1} - y^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*) + (\alpha - \mu) \|x_g^k - x^*\|^2.$$

Using parameter α defined by (15) we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \frac{\mu}{2} \|x^{k+1} - x^*\|^2 - \frac{\eta\delta}{4} \|y^{k+1} - y^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*).$$

Since $y^k, y^* \in \mathrm{range}\mathbf{W}$, we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \frac{\mu}{2} \|x^{k+1} - x^*\|^2 - \frac{\eta\delta\lambda_{\min}^+(\mathbf{W})}{4} \|y^{k+1} - y^*\|_{\mathbf{W}^\dagger}^2$$

$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*).$$

19

Using (10) we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min\left\{ \frac{\eta\mu}{2}, \frac{\eta\theta\delta\lambda_{\min}^+(\mathbf{W})}{4} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$
$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*).$$

Using parameter θ defined by (14) and definition of δ we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min\left\{ \frac{\eta\mu}{2}, \frac{\lambda_{\min}^+(\mathbf{W})}{4\lambda_{\max}(\mathbf{W})}, \frac{\lambda_{\min}^+(\mathbf{W})}{8\eta L \lambda_{\max}(\mathbf{W})} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$
$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*).$$

Plugging parameter η defined by (13) we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min\left\{ \frac{\mu}{8\tau L}, \frac{\lambda_{\min}^+(\mathbf{W})}{4\lambda_{\max}(\mathbf{W})}, \frac{\tau\lambda_{\min}^+(\mathbf{W})}{2\lambda_{\max}(\mathbf{W})} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$
$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \frac{2-\tau}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*)$$
$$\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min\left\{ \frac{\mu}{8\tau L}, \frac{\lambda_{\min}^+(\mathbf{W})}{4\lambda_{\max}(\mathbf{W})}, \frac{\tau\lambda_{\min}^+(\mathbf{W})}{2\lambda_{\max}(\mathbf{W})} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$
$$+ \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^k, x^*) - \left(1 + \frac{\tau}{2}\right) \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^{k+1}, x^*).$$

After rearranging and using definition of Ψ^k (17) we get

$$\Psi^k \geq \left(1 + \min\left\{ \frac{\tau}{2}, \frac{\mu}{8\tau L}, \frac{\lambda_{\min}^+(\mathbf{W})}{4\lambda_{\max}(\mathbf{W})}, \frac{\tau\lambda_{\min}^+(\mathbf{W})}{2\lambda_{\max}(\mathbf{W})} \right\}\right) \Psi^{k+1}.$$

Plugging parameter τ defined by (16) we get

$$\Psi^k \geq \left(1 + \frac{1}{4} \min\left\{ \sqrt{\frac{\mu}{L} \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})} \right\}\right) \Psi^{k+1}.$$

$\square$

Proof of Theorem 2 (APAPC). Conditions of Lemma 4 are satisfied, hence the following inequality holds for all k:

$$\Psi^{k+1} \leq \left(1 + \frac{1}{4} \min\left\{ \sqrt{\frac{\mu}{L} \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})} \right\}\right)^{-1} \Psi^k.$$

After doing telescoping we get

$$\Psi^k \leq \left(1 + \frac{1}{4} \min\left\{ \sqrt{\frac{\mu}{L} \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})} \right\}\right)^{-k} \Psi^0.$$

Inequality (10) implies $\Psi^0 \leq C$, where $C := \frac{1}{\eta} \left\| x^0 - x^* \right\|^2 + \frac{1}{\theta} \|y^0 - y^*\|_{\mathbf{W}^\dagger}^2 + \frac{2(1-\tau)}{\tau} \mathrm{D}_F(x_f^0, x^*)$. Hence, we obtain.

$$\Psi^k \leq \left(1 + \frac{1}{4} \min\left\{ \sqrt{\frac{\mu}{L} \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})} \right\}\right)^{-k} C.$$

It remains to lower bound Ψ^k using (10) one more time:

$$\frac{1}{\eta}\left\|x^k - x^*\right\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^*) \leq \Psi^k \leq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}}, \frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}\right\}\right)^{-k}C.$$

Finally, choosing number of iterations

$$k \geq \left(1 + 4\max\left\{\sqrt{\frac{L\lambda_{\max}(\mathbf{W})}{\mu\lambda_{\min}^+(\mathbf{W})}}, \frac{\lambda_{\max}(\mathbf{W})}{\lambda_{\min}^+(\mathbf{W})}\right\}\right)\log\left(\frac{\eta C}{\varepsilon}\right).$$

implies $\|x^k - x^*\|^2 \leq \varepsilon$. $\qquad\square$

D Proof of Corollary 1 (OPAPC)

First, Theorem 2 still holds true by replacing $\lambda_{\max}(\mathbf{W})$ by an upper bound λ_1, $\lambda_{\min}^+(\mathbf{W})$ by a lower bound $\lambda_2 > 0$ and $\chi(\mathbf{W})$ by the upper bound $\chi = \lambda_1/\lambda_2$.[2]

The proof of Corollary 1 is similar to the proof of Theorem 4 of Scaman et al. (2017).

Denote $\tilde{\mathbf{W}} = \frac{2\chi(\mathbf{W})}{(1+\chi(\mathbf{W}))\lambda_{\max}(\mathbf{W})}\mathbf{W}$. Let I be the interval $I = [1 - \frac{1}{c_2}, 1 + \frac{1}{c_2}] \subset (0, 2)$. Then, $\mathrm{Sp}(\tilde{\mathbf{W}}) \setminus \{0\} \subset I$, where Sp denotes the spectrum. Moreover, using Scaman et al. (2017), the polynomial P satisfies $P(0) = 0$ and $\max_{t \in I} |1 - P(t)| = \frac{2c_1^T}{1+c_1^{2T}} < 1$. Therefore,

$$\mathrm{Sp}(I - P(\tilde{\mathbf{W}})) \setminus \{1\} \subset \left[-\frac{2c_1^T}{1+c_1^{2T}}, \frac{2c_1^T}{1+c_1^{2T}} \right] \subset (-1, 1).$$

Consequently,

$$\lambda_{\max}(P(\tilde{\mathbf{W}})) \leq \lambda_1 := 1 + \frac{2c_1^T}{1+c_1^{2T}} < 2, \quad \lambda_{\min}^+(P(\tilde{\mathbf{W}})) \geq \lambda_2 := 1 - \frac{2c_1^T}{1+c_1^{2T}} > 0.$$

Moreover, by replacing c_1 and T by their values, $\chi := \frac{\lambda_1}{\lambda_2} \leq 4$, see (Scaman et al., 2017, Equation 34).

Applying APAPC with the gossip matrix $P(\tilde{\mathbf{W}})$ leads to OPAPC. Then, we apply Theorem 2 to OPAPC. More precisely, we apply Theorem 2 by replacing $\mathbf{W}$ by $P(\tilde{\mathbf{W}})$ and $\lambda_{\max}(\mathbf{W})$ (resp. $\lambda_{\min}^+(\mathbf{W})$) by the upper bound (resp. the lower bound) λ_1 (resp. λ_2) of $\lambda_{\max}(P(\tilde{\mathbf{W}}))$ (resp. $\lambda_{\min}^+(P(\tilde{\mathbf{W}}))$). Denoting x^k the iterates of OPAPC, we obtain

$$\frac{1}{\eta}\left\| x^k - x^* \right\|^2 + \frac{2-\tau}{\tau}D_F(x_f^k, x^*) \leq \left(1 + \frac{1}{16}\min\left\{ \frac{2}{\sqrt{\kappa}}, 1 \right\} \right)^{-k} C.$$

Finally, the gradient computation complexity of OPAPC is $\mathcal{O}(\sqrt{\kappa}\log(1/\varepsilon))$. One multiplication by $P(\tilde{\mathbf{W}})$ is equivalent to one application of the procedure AcceleratedGossip$(\mathbf{W}, \cdot, T)$, which requires exactly T communication rounds. Therefore, the communication complexity of OPAPC is $T\mathcal{O}(\sqrt{\kappa}\log(1/\varepsilon)) = \mathcal{O}(\sqrt{\kappa\chi(\mathbf{W})}\log(1/\varepsilon))$.

---

[2]The proof is the same by replacing $\lambda_{\max}(\mathbf{W})$ by λ_1 and $\lambda_{\min}^+(\mathbf{W})$ by λ_2.

E   A Loopless Algorithm Optimal in Communication Complexity

We propose another accelerated Forward Backward algorithm to solve Problem (3). More precisely, we first provide a reformulation of Problem (3), different from the reformulation (6). Then, we design an accelerated Forward Backward algorithm associated with this reformulation. Remarkably, the matrix $\mathbf{W}$ is only involved in the operator A of this new Forward Backward algorithm. This leads to an acceleration compared to APAPC, and to an optimal communication complexity.

In this section, E is the Euclidean space $\mathsf{E} = (\mathbb{R}^d)^{\mathcal{V}} \times (\mathbb{R}^d)^{\mathcal{V}} \times \mathrm{range}(\mathbf{W})$ endowed with the norm $\|(x,y,z)\|_{\mathsf{E}}^2 := \|x\|^2 + \|y\|^2 + \|z\|_{\mathbf{W}^\dagger}^2$.

Using the first order optimality conditions, a point x^* is a solution to Problem (3) if and only if $\nabla F(x^*) \in \mathrm{range}(\mathbf{W})$ and $x^* \in \mathrm{ker}(\mathbf{W})$. Solving Problem (3) is therefore equivalent to finding $(x^*, y^*, z^*) \in \mathsf{E}$ such that

$$0 = \nabla F(x^*) - \frac{\mu}{2}x^* - y^*, \tag{18}$$

$$0 = x^* + \frac{2}{\mu}(y^* + z^*), \tag{19}$$

$$0 = \frac{2}{\mu}\mathbf{W}(y^* + z^*). \tag{20}$$

Indeed, if (18)–(20) holds, then using (18), $y^* = \nabla F(x^*) - \frac{\mu}{2}x^*$ and using (19) $z^* = -\nabla F(x^*) \in \mathrm{range}(\mathbf{W})$. Since $z^* \in \mathrm{range}(\mathbf{W})$ and $y^* + z^* = -\frac{\mu}{2}x^*$, we have $\nabla F(x^*) \in \mathrm{range}(\mathbf{W})$ and $x^* \in \mathrm{ker}(\mathbf{W})$. On the other hand, if $\nabla F(x^*) \in \mathrm{range}(\mathbf{W})$ and $x^* \in \mathrm{ker}(\mathbf{W})$, then $\mathbf{W}x^* = 0$ and setting $y^* = \nabla F(x^*) - \frac{\mu}{2}x^*$ and $z^* = -\nabla F(x^*) \in \mathrm{range}(\mathbf{W})$ leads to (18)–(20).

Consider the map $M : \mathsf{E} \to \mathsf{E}$

$$M(x,y,z) := \begin{bmatrix} \nabla F(x) - \frac{\mu}{2}x & -y & \\ x & +\frac{2}{\mu}y & +\frac{2}{\mu}z \\ & \frac{2}{\mu}\mathbf{W}y & +\frac{2}{\mu}\mathbf{W}z \end{bmatrix}.$$

Similary to Section 4.1, one can show that M is a monotone operator. Moreover, $M(x^*, y^*, z^*) = 0$, i.e., (x^*, y^*, z^*) is a zero of M.

Consider the maps $A, B : \mathsf{E} \to \mathsf{E}$ defined by

$$A(x,y,z) = \begin{bmatrix} \nabla F(x) - \frac{\mu}{2}x \\ \frac{2}{\mu}(y+z) + \nu y \\ \frac{2}{\mu}\mathbf{W}(y+z) \end{bmatrix}, \quad B(x,y,z) = \begin{bmatrix} -y \\ x - \nu y \\ 0 \end{bmatrix}.$$

Then, $M = A + B$. Note that there is a term νy, where $\nu > 0$ in $A(x,y,z)$ and a term $-\nu y$ in $B(x,y,z)$, which cancel out in the sum $A(x,y,z) + B(x,y,z)$. This additional term makes the operator $A(x,y,z)$ strongly monotone. Indeed, A is the gradient of the strongly convex function (in E) $\mathsf{E} \ni (x,y,z) \mapsto r(x) + h(y,z)$ defined by

$$r(x) := F(x) - \frac{\mu}{4}\|x\|^2, \quad h(y,z) := \frac{1}{\mu}\|y+z\|^2 + \frac{\nu}{2}\|y\|^2.$$

In other words, operator $A(x,y,z)$ can be written as

$$A(x,y,z) = \begin{bmatrix} \nabla r(x) \\ \nabla_y h(y,z) \\ \mathbf{W}\nabla_z h(y,z) \end{bmatrix},$$

23

and one can check that A is strongly monotone. However, the operator $B(x, y, z)$ is not monotone in general. Indeed, B is only weakly monotone since B satisfies

$$\left\langle B(x, y, z) - B(x^*, y^*, z^*), \begin{bmatrix} x - x^* \\ y - y^* \\ z - z^* \end{bmatrix} \right\rangle_{\mathsf{E}} = -\nu \|y - y^*\|^2.$$

One idea to solve (18)–(20) is to apply Algorithm (4) to the sum $A + B$, although B is not monotone. Note that B is linear and, although B is not monotone, the resolvent of B is still well defined while $1 - \gamma\nu + \gamma^2 \neq 0$. Indeed, $(x', y') = J_{\gamma B}(x, y)$ implies $x' = x + \gamma y'$, and $(1 - \gamma\nu + \gamma^2)y' = y - \gamma x$.

In particular, we propose a new algorithm that can be seen as an accelerated version of the Forward Backward Algorithm (4) to find a zero of $A + B$. The proposed algorithm is defined in Algorithm 3 and its complexity is given in Theorem 3. We show that the complexity of Algorithm 3 is $\mathcal{O}(\sqrt{\kappa\chi(\mathbf{W})}\log(1/\varepsilon))$, both in communication rounds and gradient computations. The proposed algorithm is therefore optimal in communication complexity, see Section 3.2. Moreover, Algorithm 3 uses only one gradient computation by communication round.

Algorithm 3

1: **Parameters:** $x^0, y^0 \in \mathbb{R}^{nd}, z^0 \in \mathrm{range}\mathbf{W}, \eta, \theta, \lambda, \alpha, \beta, \gamma, \nu > 0, \tau, \sigma \in (0, 1)$
2: Set $x_f^0 = x^0$
3: Set $y_f^0 = y^0$
4: Set $z_f^0 = z^0$
5: **for** $k = 0, 1, 2, \ldots$ **do**
6:     $x_g^k = \tau x^k + (1 - \tau)x_f^k$
7:     $y_g^k = \sigma y^k + (1 - \sigma)y_f^k$
8:     $z_g^k = \sigma z^k + (1 - \sigma)z_f^k$
9:     $x^{k+1} = x^k + \eta\alpha(x_g^k - x^{k+1}) - \eta\nabla r(x_g^k) + \eta y^{k+1}$
10:    $y^{k+1} = y^k + \theta\beta(y_g^k - y^{k+1}) - \theta\nabla_y h(y_g^k, z_g^k) + \theta\nu y^{k+1} - \theta x^{k+1}$
11:    $z^{k+1} = z^k + \lambda\gamma(z_g^k - z^{k+1}) - \lambda\mathbf{W}\nabla_z h(y_g^k, z_g^k)$
12:    $x_f^{k+1} = x_g^k + \frac{2\tau}{2-\tau}(x^{k+1} - x^k)$
13:    $y_f^{k+1} = y_g^k + \sigma(y^{k+1} - y^k)$
14:    $z_f^{k+1} = z_g^k + \sigma(z^{k+1} - z^k)$
15: **end for**

Theorem 3 (Algorithm 3). *Set the parameters* $\eta, \theta, \lambda, \alpha, \beta, \gamma, \nu > 0, \tau, \sigma \in (0, 1)$ *to*

$$\eta = \left[2\sqrt{L\mu} + \mu\right]^{-1}, \qquad\qquad \alpha = \frac{\mu}{3}, \qquad \tau = \frac{1}{2}\sqrt{\frac{\mu}{L}},$$

$$\theta = \left[\frac{1}{4}\sqrt{\frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})\mu L}} + \frac{5}{96L}\right]^{-1}, \qquad \beta = \frac{1}{96L}, \qquad \sigma = \frac{1}{20}\sqrt{\frac{\lambda_{\min}^+(\mathbf{W})}{\lambda_{\max}(\mathbf{W})}\frac{\mu}{L}},$$

$$\lambda = \left[\frac{1}{4}\sqrt{\frac{\lambda_{\min}^+(\mathbf{W})\lambda_{\max}(\mathbf{W})}{\mu L}} + \frac{\lambda_{\min}^+(\mathbf{W})}{96L}\right]^{-1}, \quad \gamma = \frac{\lambda_{\min}^+(\mathbf{W})}{96L}, \quad \nu = \frac{1}{24L}.$$

Then, the sequence (x^k) *converges linearly to* x^*. *Moreover, for every* $\varepsilon > 0$, *Algorithm 3 finds* x^k *for which* $\|x^k - x^*\|^2 \leq \varepsilon$ *in at most* $\mathcal{O}\left(\sqrt{\kappa\chi(\mathbf{W})}\log(1/\varepsilon)\right)$ *gradient computations (resp. communication rounds).*

The Algorithm 3 achieves the communication lower bound of Theorem 1. The proof of Theorem 3 intuitively relies on viewing Algorithm 3 as an accelerated version of (4), although Nesterov's acceleration does not apply to general monotone operators and even less to non monotone operators.

F Proof of Theorem 3 (Algorithm 3)

Lemma 5. *Let α satisfy*

$$\alpha \leq \frac{\mu}{2}. \tag{21}$$

Let δ be defined by

$$\delta = \min\left\{1, \frac{1}{2\eta L}\right\}. \tag{22}$$

Then the following inequality holds:

$$-\frac{1}{2\eta}\|x^{k+1} - x^*\|^2 \leq -\frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \frac{\alpha}{4}\|x^{k+1} - x^*\|^2 + D_r(x_g^k, x^*).$$

Proof. From line 9 of Algortihm 3 it follows that

$$x^{k+1} - x^k = \eta\alpha(x_g^k - x^{k+1}) - \eta\nabla r(x_g^k) + \eta y^{k+1}.$$

From optimality condition (18) it follows that $\nabla r(x^*) = y^*$ and hence

$$\|x^{k+1} - x^k\|^2 = \eta^2\|y^{k+1} - y^* - \alpha(x^{k+1} - x^*) - (\nabla r(x_g^k) - \nabla r(x^*) - \alpha(x_g^k - x^*))\|^2$$

$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - \eta^2\|\alpha(x^{k+1} - x^*) + (\nabla r(x_g^k) - \nabla r(x^*) - \alpha(x_g^k - x^*))\|^2$$

$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2$$

$$- 2\eta^2\|\nabla r(x_g^k) - \nabla r(x^*) - \alpha(x_g^k - x^*)\|^2.$$

From (21) it follows that function $r(x) - \frac{\alpha}{2}\|x\|^2 = F(x) - \frac{\mu+2\alpha}{4}\|x\|^2$ is convex and L-smooth, hence we can bound the last term:

$$\|x^{k+1} - x^k\|^2 \geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2 - 4\eta^2 LD_{r(\cdot) - \frac{\alpha}{2}\|\cdot\|^2}(x_g^k, x^*)$$

$$= \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2 - 4\eta^2 LD_r(x_g^k, x^*) + 2\eta^2 L\alpha\|x_g^k - x^*\|^2$$

$$\geq \frac{\eta^2}{2}\|y^{k+1} - y^*\|^2 - 2\eta^2\alpha^2\|x^{k+1} - x^*\|^2 - 4\eta^2 LD_r(x_g^k, x^*).$$

Multiplying by $\frac{1}{2\eta}$ and rearranging gives

$$-\frac{1}{2\eta}\|x^{k+1} - x^*\|^2 \leq -\frac{\eta}{4}\|y^{k+1} - y^*\|^2 + \eta\alpha^2\|x^{k+1} - x^*\|^2 + 2\eta LD_r(x_g^k, x^*).$$

Using δ defined by (22) we obtain

$$-\frac{1}{2\eta}\|x^{k+1} - x^*\|^2 \leq -\frac{\delta}{2\eta}\|x^{k+1} - x^*\|^2$$

$$\leq -\frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \delta\eta\alpha^2\|x^{k+1} - x^*\|^2 + 2\delta\eta LD_r(x_g^k, x^*)$$

$$\leq -\frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \frac{\eta\alpha^2}{2\eta L}\|x^{k+1} - x^*\|^2 + D_r(x_g^k, x^*)$$

$$\leq -\frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \frac{\alpha\mu}{4L}\|x^{k+1} - x^*\|^2 + D_r(x_g^k, x^*)$$

$$\leq -\frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \frac{\alpha}{4}\|x^{k+1} - x^*\|^2 + D_r(x_g^k, x^*).$$

$\square$

Lemma 6. *Let α satisfy*

$$\alpha \leq \frac{\mu}{2}. \tag{21}$$

Let η satisfy

$$\eta \le \frac{1}{4\tau L}. \tag{23}$$

Then the following inequality holds:

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \le \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau}D_r(x_f^k, x^*) - \frac{2-\tau}{\tau}D_r(x_f^{k+1}, x^*) \tag{24}$$

$$- \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle.$$

Proof. Using line 9 of Algorithm 3 we get

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 = \frac{1}{\eta}\|x^k - x^*\|^2 + \frac{2}{\eta}\langle x^{k+1} - x^k, x^{k+1} - x^*\rangle - \frac{1}{\eta}\|x^{k+1} - x^k\|^2$$

$$= \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^{k+1}, x^{k+1} - x^*\rangle$$

$$- 2\langle \nabla r(x_g^k) - y^{k+1}, x^{k+1} - x^*\rangle$$

$$= \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^*, x^{k+1} - x^*\rangle - 2\alpha\|x^{k+1} - x^*\|^2$$

$$- 2\langle \nabla r(x_g^k) - y^{k+1}, x^{k+1} - x^*\rangle$$

$$\le \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + \alpha\|x_g^k - x^*\|^2 - \alpha\|x^{k+1} - x^*\|^2$$

$$- 2\langle \nabla r(x_g^k) - y^{k+1}, x^{k+1} - x^*\rangle.$$

From optimality condition (18) it follows that $\nabla r(x^*) = y^*$ and hence

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \le \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \alpha\|x^{k+1} - x^*\|^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2$$

$$- 2\langle \nabla r(x_g^k) - \nabla r(x^*), x^{k+1} - x^*\rangle + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle.$$

Using lemma 5 we get

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \le \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \alpha\|x^{k+1} - x^*\|^2 - \frac{1}{2\eta}\|x^{k+1} - x^k\|^2$$

$$- \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + \frac{\alpha}{4}\|x^{k+1} - x^*\|^2 + D_r(x_g^k, x^*)$$

$$- 2\langle \nabla r(x_g^k) - \nabla r(x^*), x^{k+1} - x^*\rangle + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$\le \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2 - \frac{1}{2\eta}\|x^{k+1} - x^k\|^2$$

$$- 2\langle \nabla r(x_g^k) - \nabla r(x^*), x^{k+1} - x^*\rangle + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$+ D_r(x_g^k, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2.$$

Using lines 6 and 12 of Algorithm 3 we get

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \le \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau}\langle \nabla r(x_g^k) - \nabla r(x^*), x_f^k - x_g^k\rangle - 2\langle \nabla r(x_g^k) - \nabla r(x^*), x_g^k - x^*\rangle$$

$$- \frac{2-\tau}{\tau}\langle \nabla r(x_g^k) - \nabla r(x^*), x_f^{k+1} - x_g^k\rangle - \frac{(2-\tau)^2}{8\eta\tau^2}\|x_f^{k+1} - x_g^k\|^2$$

$$+ D_r(x_g^k, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$\le \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau}\langle \nabla r(x_g^k) - \nabla r(x^*), x_f^k - x_g^k\rangle - 2\langle \nabla r(x_g^k) - \nabla r(x^*), x_g^k - x^*\rangle$$

$$- \frac{2-\tau}{\tau}\left(\langle \nabla r(x_g^k) - \nabla r(x^*), x_f^{k+1} - x_g^k\rangle + \frac{1}{8\eta\tau}\|x_f^{k+1} - x_g^k\|^2\right)$$

$$+ D_r(x_g^k, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle.$$

Using $\frac{\mu}{2}$-strong convexity and L-smoothness of $r(x)$ and η defined by (23) we get

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \leq \frac{1}{\eta}\|x^k - x^*\|^2 + \alpha\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau}(D_r(x_f^k, x^*) - D_r(x_g^k, x^*)) - 2D_r(x_g^k, x^*) - \frac{\mu}{2}\|x_g^k - x^*\|^2$$

$$- \frac{2-\tau}{\tau}\left(\langle \nabla r(x_g^k) - \nabla r(x^*), x_f^{k+1} - x_g^k\rangle + \frac{L}{2}\|x_f^{k+1} - x_g^k\|^2\right)$$

$$+ D_r(x_g^k, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$\leq \frac{1}{\eta}\|x^k - x^*\|^2 + \left(\alpha - \frac{\mu}{2}\right)\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau}(D_r(x_f^k, x^*) - D_r(x_g^k, x^*)) - \frac{2-\tau}{\tau}(D_r(x_f^{k+1}, x^*) - D_r(x_g^k, x^*))$$

$$- D_r(x_g^k, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$= \frac{1}{\eta}\|x^k - x^*\|^2 + \left(\alpha - \frac{\mu}{2}\right)\|x_g^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2$$

$$+ \frac{2(1-\tau)}{\tau}D_r(x_f^k, x^*) - \frac{2-\tau}{\tau}D_r(x_f^{k+1}, x^*)$$

$$- \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle.$$

Using α defined by (21) we get

$$\frac{1}{\eta}\|x^{k+1} - x^*\|^2 \leq \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau}D_r(x_f^k, x^*) - \frac{2-\tau}{\tau}D_r(x_f^{k+1}, x^*)$$

$$- \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle.$$

$\square$

Lemma 7. *For all $y_1, y_2 \in \mathbb{R}^{nd}$ and $z_1, z_2 \in \text{range}\mathbf{W}$ the following inequality holds:*

$$D_h((y_1, z_1), (y_2, z_2)) \leq \left(\frac{2}{\mu} + \frac{\nu}{2}\right)\|y_1 - y_2\|^2 + \frac{2}{\mu}\|z_1 - z_2\|^2.$$

Proof. It follows from from the definition of D_h:

$$D_h((y_1, z_1), (y_2, z_2)) = \frac{1}{\mu}\|y_1 + z_1 - y_2 - z_2\|^2 + \frac{\nu}{2}\|y_1 - y_2\|^2$$

$$\leq \left(\frac{2}{\mu} + \frac{\nu}{2}\right)\|y_1 - y_2\|^2 + \frac{2}{\mu}\|z_1 - z_2\|^2.$$

$\square$

Lemma 8. *Let θ satisfy*

$$\theta \leq \left[\sigma\left(\frac{4}{\mu} + \nu\right)\right]^{-1}. \tag{25}$$

Let λ satisfy

$$\lambda \leq \left[\frac{4\sigma\lambda_{\max}(\mathbf{W})}{\mu}\right]^{-1}. \tag{26}$$

Let β satisfy

$$\beta \leq \min\left\{\frac{1}{\mu}, \frac{\nu}{3}\right\}. \tag{27}$$

Let γ satisfy

$$\gamma \leq \lambda_{\min}^+(\mathbf{W})\beta. \tag{28}$$

Then the following inequality holds:

$$\left\|\begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix}\right\|_{\mathbf{M}}^2 \leq \left\|\begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix}\right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2 \tag{29}$$

$$+ \frac{2(1-\sigma)}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle,$$

where $\mathbf{M} \in \mathbb{R}^{2nd \times 2nd}$ is a matrix defined by

$$\mathbf{M} = \begin{bmatrix} \frac{1}{\theta}\mathbf{I} & 0 \\ 0 & \frac{1}{\lambda}\mathbf{W}^\dagger \end{bmatrix}. \tag{30}$$

Proof. Using line 10 of Algorithm 3 we get

$$\frac{1}{\theta}\|y^{k+1} - y^*\|^2 = \frac{1}{\theta}\|y^k - y^*\|^2 + \frac{2}{\theta}\langle y^{k+1} - y^k, y^{k+1} - y^* \rangle - \frac{1}{\theta}\|y^{k+1} - y^k\|^2$$

$$= \frac{1}{\theta}\|y^k - y^*\|^2 - \frac{1}{\theta}\|y^{k+1} - y^k\|^2 + 2\beta\langle y_g^k - y^{k+1}, y^{k+1} - y^* \rangle$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nu y^{k+1} + x^{k+1}, y^{k+1} - y^* \rangle$$

$$= \frac{1}{\theta}\|y^k - y^*\|^2 - \frac{1}{\theta}\|y^{k+1} - y^k\|^2 + 2\beta\langle y_g^k - y^*, y^{k+1} - y^* \rangle - 2\beta\|y^{k+1} - y^*\|^2$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nu y^{k+1} + x^{k+1}, y^{k+1} - y^* \rangle$$

$$\leq \frac{1}{\theta}\|y^k - y^*\|^2 + \beta\|y_g^k - y^*\|^2 - \beta\|y^{k+1} - y^*\|^2 - \frac{1}{\theta}\|y^{k+1} - y^k\|^2$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nu y^{k+1} + x^{k+1}, y^{k+1} - y^* \rangle.$$

From optimality condition (19) it follows that $x^* = -\frac{2}{\mu}(y^* + z^*) = -\nabla_y h(y^*, z^*) + \nu y^*$ and hence

$$\frac{1}{\theta}\|y^{k+1} - y^*\|^2 \leq \frac{1}{\theta}\|y^k - y^*\|^2 + \beta\|y_g^k - y^*\|^2 - \beta\|y^{k+1} - y^*\|^2 - \frac{1}{\theta}\|y^{k+1} - y^k\|^2$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y^{k+1} - y^* \rangle$$

$$+ 2\nu\|y^{k+1} - y^*\|^2 - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle$$

$$= \frac{1}{\theta}\|y^k - y^*\|^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 + \beta\|y_g^k - y^*\|^2 - \frac{1}{\theta}\|y^{k+1} - y^k\|^2$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y^{k+1} - y^* \rangle - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

Using lines 7 and 13 of Algorithm 3 we get

$$\frac{1}{\theta}\|y^{k+1} - y^*\|^2 \leq \frac{1}{\theta}\|y^k - y^*\|^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 + \beta\|y_g^k - y^*\|^2$$

$$+ \frac{2(1-\sigma)}{\sigma}\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_f^k - y_g^k \rangle$$

$$- \frac{2}{\sigma}\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_f^{k+1} - y_g^k \rangle - \frac{1}{\theta\sigma^2}\|y_f^{k+1} - y_g^k\|^2$$

$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_g^k - y^* \rangle - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

Using θ defined by (25) we get

$$\frac{1}{\theta}\|y^{k+1}-y^*\|^2 \leq \frac{1}{\theta}\|y^k-y^*\|^2 - (\beta-2\nu)\|y^{k+1}-y^*\|^2 + \beta\|y_g^k-y^*\|^2 \tag{31}$$
$$+ \frac{2(1-\sigma)}{\sigma}\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_f^k - y_g^k\rangle$$
$$- \frac{2}{\sigma}\left(\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_f^{k+1} - y_g^k\rangle + \left(\frac{2}{\mu}+\frac{\nu}{2}\right)\|y_f^{k+1}-y_g^k\|^2\right)$$
$$- 2\langle \nabla_y h(y_g^k, z_g^k) - \nabla_y h(y^*, z^*), y_g^k - y^*\rangle - 2\langle x^{k+1}-x^*, y^{k+1}-y^*\rangle.$$

Using line 11 of Algorithm 3 we get

$$\frac{1}{\lambda}\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 = \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 + \frac{2}{\lambda}\langle z^{k+1}-z^k, \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle - \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2$$
$$= \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 - \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2 + 2\gamma\langle z_g^k - z^{k+1}, \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle$$
$$- 2\langle \mathbf{W}\nabla_z h(y_g^k, z_g^k), \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle$$
$$= \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 + 2\gamma\langle z_g^k - z^*, \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle - 2\gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2$$
$$- \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2 - 2\langle \mathbf{W}\nabla_z h(y_g^k, z_g^k), \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle$$
$$\leq \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 + \gamma\|z_g^k - z^*\|_{\mathbf{W}^\dagger}^2 - \gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 - \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2$$
$$- 2\langle \mathbf{W}\nabla_z h(y_g^k, z_g^k), \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle.$$

From optimality condition (20) it follows that $\mathbf{W}\nabla_z h(y^*, z^*) = 0$ and hence

$$\frac{1}{\lambda}\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 \leq \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 + \gamma\|z_g^k - z^*\|_{\mathbf{W}^\dagger}^2 - \gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 - \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2$$
$$- 2\langle \mathbf{W}(\nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*)), \mathbf{W}^\dagger(z^{k+1}-z^*)\rangle$$
$$= \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 + \gamma\|z_g^k - z^*\|_{\mathbf{W}^\dagger}^2 - \gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 - \frac{1}{\lambda}\|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2$$
$$- 2\langle \nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*), \mathbf{W}\mathbf{W}^\dagger(z^{k+1}-z^*)\rangle.$$

It's easy to observe that $z^k, z^* \in \mathrm{range}\mathbf{W}$ for all $k = 0, 1, 2, \ldots$, which implies

$$\mathbf{W}\mathbf{W}^\dagger(z^{k+1}-z^*) = z^{k+1}-z^* \text{ and } \|z^{k+1}-z^k\|_{\mathbf{W}^\dagger}^2 \geq \frac{1}{\lambda_{\max}(\mathbf{W})}\|z^{k+1}-z^k\|^2.$$

Hence,

$$\frac{1}{\lambda}\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 \leq \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 - \gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|^2$$
$$- \frac{1}{\lambda \cdot \lambda_{\max}(\mathbf{W})}\|z^{k+1}-z^k\|^2 - 2\langle \nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*), z^{k+1}-z^*\rangle.$$

Using lines 8 and 14 of Algorithm 3 we get

$$\frac{1}{\lambda}\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 \leq \frac{1}{\lambda}\|z^k-z^*\|_{\mathbf{W}^\dagger}^2 - \gamma\|z^{k+1}-z^*\|_{\mathbf{W}^\dagger}^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|_{\mathbf{W}^\dagger}^2$$
$$+ \frac{2(1-\sigma)}{\sigma}\langle \nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*), z_f^k - z_g^k\rangle$$
$$- \frac{2}{\sigma}\langle \nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*), z_f^{k+1} - z_g^k\rangle - \frac{1}{\lambda\sigma^2\lambda_{\max}(\mathbf{W})}\|z_f^{k+1}-z_g^k\|^2$$
$$- 2\langle \nabla_z h(y_g^k, z_g^k) - \nabla_z h(y^*, z^*), z_g^k - z^*\rangle.$$

Using λ defined by (26) we get

$$\frac{1}{\lambda}\|z^{k+1} - z^*\|^2_{\mathbf{W}^\dagger} \le \frac{1}{\lambda}\|z^k - z^*\|^2_{\mathbf{W}^\dagger} - \gamma\|z^{k+1} - z^*\|^2_{\mathbf{W}^\dagger} + \frac{\gamma}{\lambda^+_{\min}(\mathbf{W})}\|z^k_g - z^*\|^2_{\mathbf{W}^\dagger} \quad (32)$$

$$+ \frac{2(1-\sigma)}{\sigma}\langle \nabla_z h(y^k_g, z^k_g) - \nabla_z h(y^*, z^*), z^k_f - z^k_g \rangle$$

$$- \frac{2}{\sigma}\left(\langle \nabla_z h(y^k_g, z^k_g) - \nabla_z h(y^*, z^*), z^{k+1}_f - z^k_g \rangle - \frac{2}{\mu}\|z^{k+1}_f - z^k_g\|^2 \right)$$

$$- 2\langle \nabla_z h(y^k_g, z^k_g) - \nabla_z h(y^*, z^*), z^k_g - z^* \rangle.$$

After combining (31) and (32) we get

$$\left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|^2_{\mathbf{M}} \le \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|^2_{\mathbf{M}} - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|^2_{\mathbf{W}^\dagger}$$

$$+ \beta\|y^k_g - y^*\|^2 + \frac{\gamma}{\lambda^+_{\min}(\mathbf{W})}\|z^k_g - z^*\|^2$$

$$+ \frac{2(1-\sigma)}{\sigma}\left\langle \nabla h(y^k_g, z^k_g) - \nabla h(y^*, z^*), \begin{bmatrix} y^k_f \\ z^k_f \end{bmatrix} - \begin{bmatrix} y^k_g \\ z^k_g \end{bmatrix} \right\rangle$$

$$- \frac{2}{\sigma}\left\langle \nabla h(y^k_g, z^k_g) - \nabla h(y^*, z^*), \begin{bmatrix} y^{k+1}_f \\ z^{k+1}_f \end{bmatrix} - \begin{bmatrix} y^k_g \\ z^k_g \end{bmatrix} \right\rangle$$

$$- \frac{2}{\sigma}\left(\left(\frac{2}{\mu} + \frac{\nu}{2}\right)\|y^{k+1}_f - z^k_g\|^2 + \frac{2}{\mu}\|z^{k+1}_f - z^k_g\|^2 \right)$$

$$- 2\left\langle \nabla h(y^k_g, z^k_g) - \nabla h(y^*, z^*), \begin{bmatrix} y^k_g \\ z^k_g \end{bmatrix} - \begin{bmatrix} y^* \\ z^* \end{bmatrix} \right\rangle - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle,$$

where $\mathbf{M} \in \mathbb{R}^{2nd \times 2nd}$ is a matrix defined by (30). Using convexity of $h(y, z)$ and the fact that
$\nabla h(y, z) = \begin{bmatrix} \frac{2}{\mu}(y + z) + \nu y \\ \frac{2}{\mu}(y + z) \end{bmatrix}$ we get

$$\left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|^2_{\mathbf{M}} \le \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|^2_{\mathbf{M}} - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|^2_{\mathbf{W}^\dagger}$$

$$+ \beta\|y^k_g - y^*\|^2 + \frac{\gamma}{\lambda^+_{\min}(\mathbf{W})}\|z^k_g - z^*\|^2$$

$$+ \frac{2(1-\sigma)}{\sigma}\left[D_h((y^k_f, z^k_f), (y^*, z^*)) - D_h((y^k_g, z^k_g), (y^*, z^*)) \right]$$

$$- \frac{2}{\sigma}\left\langle \nabla h(y^k_g, z^k_g) - \nabla h(y^*, z^*), \begin{bmatrix} y^{k+1}_f \\ z^{k+1}_f \end{bmatrix} - \begin{bmatrix} y^k_g \\ z^k_g \end{bmatrix} \right\rangle$$

$$- \frac{2}{\sigma}\left(\left(\frac{2}{\mu} + \frac{\nu}{2}\right)\|y^{k+1}_f - z^k_g\|^2 + \frac{2}{\mu}\|z^{k+1}_f - z^k_g\|^2 \right)$$

$$- \frac{4}{\mu}\|y^k_g + z^k_g - y^* - z^*\|^2 - 2\nu\|y^k_g - y^*\|^2 - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

Using lemma 7 we can obtain

$$
\left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 \leq \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2
$$

$$
+ (\beta - 2\nu)\|y_g^k - y^*\|^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|^2 - \frac{4}{\mu}\|y_g^k + z_g^k - y^* - z^*\|^2
$$

$$
+ \frac{2(1-\sigma)}{\sigma}\left[\mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) - \mathrm{D}_h((y_g^k, z_g^k), (y^*, z^*)) \right]
$$

$$
- \frac{2}{\sigma}\left[\mathrm{D}_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*)) - \mathrm{D}_h((y_g^k, z_g^k), (y^*, z^*)) \right]
$$

$$
- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle
$$

$$
= \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2
$$

$$
+ (\beta - 2\nu)\|y_g^k - y^*\|^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|^2 - \frac{4}{\mu}\|y_g^k + z_g^k - y^* - z^*\|^2
$$

$$
+ \frac{2(1-\sigma)}{\sigma}\mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}\mathrm{D}_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))
$$

$$
+ 2\mathrm{D}_h((y_g^k, z_g^k), (y^*, z^*)) - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle
$$

$$
= \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2
$$

$$
+ (\beta - 2\nu)\|y_g^k - y^*\|^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|^2 - \frac{4}{\mu}\|y_g^k + z_g^k - y^* - z^*\|^2
$$

$$
+ \frac{2(1-\sigma)}{\sigma}\mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}\mathrm{D}_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))
$$

$$
+ \frac{2}{\mu}\|y_g^k + z_g^k - y^* - z^*\|^2 + \nu\|y_g^k - y^*\|^2 - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle
$$

$$
= \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2
$$

$$
+ \frac{2(1-\sigma)}{\sigma}\mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}\mathrm{D}_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))
$$

$$
+ (\beta - \nu)\|y_g^k - y^*\|^2 + \frac{\gamma}{\lambda_{\min}^+(\mathbf{W})}\|z_g^k - z^*\|^2 - \frac{2}{\mu}\|y_g^k + z_g^k - y^* - z^*\|^2
$$

$$
- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.
$$

Using γ defined by (28) and the fact that $\beta \leq \frac{1}{\mu}$ which follows from (27) we get

$$
\left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 \leq \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2
$$

$$
+ \frac{2(1-\sigma)}{\sigma}\mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}\mathrm{D}_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))
$$

$$
+ (\beta - \nu)\|y_g^k - y^*\|^2 + \beta\|z_g^k - z^*\|^2 - 2\beta\|y_g^k + z_g^k - y^* - z^*\|^2
$$

$$
- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle
$$

$$\leq \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2$$

$$+ \frac{2(1-\sigma)}{\sigma}D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$+ (\beta - \nu)\|y_g^k - y^*\|^2 + \beta\|z_g^k - z^*\|^2 - \beta\|z_g^k - z^*\|^2 + 2\beta\|y_g^k - y^*\|^2$$

$$- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle$$

$$= \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2$$

$$+ \frac{2(1-\sigma)}{\sigma}D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$+ (3\beta - \nu)\|y_g^k - y^*\|^2 - 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

Using the fact that $\beta \leq \frac{\nu}{3}$ which follows from (27) we get

$$\left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 \leq \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2$$

$$+ \frac{2(1-\sigma)}{\sigma}D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$- 2\langle x^{k+1} - x^*, y^{k+1} - y^* \rangle.$$

$\square$

Theorem 4. *Let τ be defined by*

$$\tau = \frac{1}{2}\sqrt{\frac{\mu}{L}}.$$

Let α be defined by

$$\alpha = \frac{\mu}{2}.$$

Let η be defined by

$$\eta = \frac{1}{2\sqrt{\mu L}}.$$

Let σ be defined by

$$\sigma = \frac{1}{18}\sqrt{\frac{\mu \lambda_{\min}^+(\mathbf{W})}{L \lambda_{\max}(\mathbf{W})}}.$$

Let ν be defined by

$$\nu = \frac{3}{80L}.$$

Let β be defined by

$$\beta = \frac{1}{80L}.$$

Let θ be defined by

$$\theta = \frac{18\sqrt{\mu L \lambda_{\max}(\mathbf{W})}}{5\sqrt{\lambda_{\min}^+(\mathbf{W})}}$$

Let γ be defined by

$$\gamma = \frac{\lambda_{\min}^+(\mathbf{W})}{80L}.$$

32

Let λ be defined by

$$\lambda = \frac{9\sqrt{\mu L}}{2\sqrt{\lambda_{\min}^+(\mathbf{W})\lambda_{\max}(\mathbf{W})}}.$$

where $\mathbf{P} \in \mathbb{R}^{3nd \times 3nd}$ is a matrix defined by

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\eta}\mathbf{I} & 0 & 0 \\ 0 & \frac{1}{\theta}\mathbf{I} & 0 \\ 0 & 0 & \frac{1}{\lambda}\mathbf{W}^\dagger \end{bmatrix}. \tag{33}$$

Let ρ be defined by

$$\rho = \frac{1}{18}\sqrt{\frac{\mu\lambda_{\min}^+(\mathbf{W})}{L\lambda_{\max}(\mathbf{W})}}.$$

Let Ψ^k be the following Lyapunov function:

$$\Psi^k = (1+\rho)\left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 + \frac{(2-\tau)}{\tau}D_r(x_f^k, x^*) + \frac{2}{\sigma}D_h((y_f^k, z_f^k), (y^*, z^*)). \tag{34}$$

Then the following inequality holds:

$$\Psi^{k+1} \le \left(1 - \frac{1}{1+\rho^{-1}}\right)\Psi^k.$$

Proof. One can observe that conditions of lemma 6 and lemma 8 are satisfied. Hence we can combine (24) and (29) and get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 = \frac{1}{\eta}\|x^{k+1} - y^*\|^2 + \left\| \begin{bmatrix} y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2$$

$$\le \frac{1}{\eta}\|x^k - x^*\|^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2 + \frac{2(1-\tau)}{\tau}D_r(x_f^k, x^*)$$

$$- \frac{2-\tau}{\tau}D_r(x_f^{k+1}, x^*) - \frac{\eta\delta}{4}\|y^{k+1} - y^*\|^2 + 2\langle y^{k+1} - y^*, x^{k+1} - x^*\rangle$$

$$+ \left\| \begin{bmatrix} y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{M}}^2 - (\beta - 2\nu)\|y^{k+1} - y^*\|^2 - \gamma\|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2$$

$$+ \frac{2(1-\sigma)}{\sigma}D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma}D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$- 2\langle x^{k+1} - x^*, y^{k+1} - y^*\rangle$$

$$= \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \frac{3\alpha}{4}\|x^{k+1} - x^*\|^2 - \left(\frac{\eta\delta}{4} + \beta - 2\nu\right)\|y^{k+1} - y^*\|^2$$

33

$$- \gamma \|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2 + \frac{2(1 - \tau)}{\tau} D_r(x_f^k, x^*) - \frac{2 - \tau}{\tau} D_r(x_f^{k+1}, x^*)$$

$$+ \frac{2(1 - \sigma)}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*)),$$

where $\mathbf{P} \in \mathbb{R}^{3nd \times 3nd}$ is a matrix defined by (33). From (22) it follows that

$$\frac{\eta \delta}{4} = \min \left\{ \frac{1}{8\sqrt{\mu L}}, \frac{1}{8L} \right\} = \frac{1}{8L},$$

and hence, using choice of α, β and ν, we get

$$\left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 \leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \frac{3\mu}{8} \|x^{k+1} - x^*\|^2 - \left(\frac{1}{8L} + \frac{1}{80L} - \frac{6}{80L} \right) \|y^{k+1} - y^*\|^2$$

$$- \frac{\lambda_{\min}^+(\mathbf{W})}{80L} \|z^{k+1} - z^*\|_{\mathbf{W}^\dagger}^2 + \frac{2(1 - \tau)}{\tau} D_r(x_f^k, x^*) - \frac{2 - \tau}{\tau} D_r(x_f^{k+1}, x^*)$$

$$+ \frac{2(1 - \sigma)}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min \left\{ \frac{3\eta \mu}{8}, \frac{\theta}{16L}, \frac{\lambda \cdot \lambda_{\min}^+(\mathbf{W})}{80L} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$

$$+ \left(1 - \frac{\tau}{2} \right) \frac{(2 - \tau)}{\tau} D_r(x_f^k, x^*) - \frac{2 - \tau}{\tau} D_r(x_f^{k+1}, x^*)$$

$$+ (1 - \sigma) \frac{2}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$= \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \min \left\{ \frac{3}{16} \sqrt{\frac{\mu}{L}}, \frac{9\sqrt{\mu \lambda_{\max}(\mathbf{W})}}{40\sqrt{L\lambda_{\min}^+(\mathbf{W})}}, \frac{9\sqrt{\mu \lambda_{\min}^+(\mathbf{W})}}{160\sqrt{L\lambda_{\max}(\mathbf{W})}} \right\} \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$

$$+ \left(1 - \frac{1}{4} \sqrt{\frac{\mu}{L}} \right) \frac{(2 - \tau)}{\tau} D_r(x_f^k, x^*) - \frac{2 - \tau}{\tau} D_r(x_f^{k+1}, x^*)$$

$$+ \left(1 - \frac{1}{18} \sqrt{\frac{\mu \lambda_{\min}^+(\mathbf{W})}{L\lambda_{\max}(\mathbf{W})}} \right) \frac{2}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*))$$

$$\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 - \rho \left\| \begin{bmatrix} x^{k+1} - x^* \\ y^{k+1} - y^* \\ z^{k+1} - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2$$

$$+ (1 - \rho) \frac{(2 - \tau)}{\tau} D_r(x_f^k, x^*) - \frac{2 - \tau}{\tau} D_r(x_f^{k+1}, x^*)$$

$$+ (1 - \rho) \frac{2}{\sigma} D_h((y_f^k, z_f^k), (y^*, z^*)) - \frac{2}{\sigma} D_h((y_f^{k+1}, z_f^{k+1}), (y^*, z^*)).$$

After rearranging and using definition of Ψ^k (34) we get

$$
\begin{aligned}
\Psi^{k+1} &\leq \left\| \begin{bmatrix} x^k - x^* \\ y^k - y^* \\ z^k - z^* \end{bmatrix} \right\|_{\mathbf{P}}^2 + (1-\rho)\frac{(2-\tau)}{\tau} \mathrm{D}_r(x_f^k, x^*) \\
&\quad + (1-\rho)\frac{2}{\sigma} \mathrm{D}_h((y_f^k, z_f^k), (y^*, z^*)) \\
&\leq \left(1 - \frac{1}{1+\rho^{-1}} \right) \Psi^k.
\end{aligned}
$$

$\square$