An Optimal Algorithm for Strongly Convex Minimization under Affine Constraints

Adil Salim
Microsoft Research

Laurent Condat
KAUST, Saudi Arabia

Dmitry Kovalev
KAUST, Saudi Arabia

Peter Richtárik
KAUST, Saudi Arabia

Abstract

Optimization problems under affine constraints appear in various areas of machine learning. We consider the task of minimizing a smooth strongly convex function $F(x)$ under the affine constraint $\mathbf{K}x = b$, with an oracle providing evaluations of the gradient of F and multiplications by $\mathbf{K}$ and its transpose. We provide lower bounds on the number of gradient computations and matrix multiplications to achieve a given accuracy. Then we propose an accelerated primal–dual algorithm achieving these lower bounds. Our algorithm is the first optimal algorithm for this class of problems.

1 Introduction

We consider the convex optimization problem

$$\min_{x \in \mathsf{X}} F(x) \quad \text{s.t.} \quad \mathbf{K}x = b, \tag{1}$$

where F is a smooth and strongly convex function over $\mathsf{X} := \mathbb{R}^d$, $b \in \mathsf{Y} := \mathbb{R}^p$ is a vector and $\mathbf{K}$ is a nonzero $p \times d$ matrix, for some integers $d \geq 1$, $p \geq 1$. We adopt the matrix-vector setting for simplicity of the notations, but the formalism holds more generally with arbitrary real Hilbert spaces X and Y and bounded linear operator $\mathbf{K} : \mathsf{X} \to \mathsf{Y}$. We suppose that b is in the range of $\mathbf{K}$; then the sought solution to (1), denoted by $x^\star$, exists and is unique, by strong convexity.

Problem (1) covers a large number of applications in machine learning (Sra et al., 2011; Bach et al., 2012; Polson et al., 2015) and beyond (Bauschke et al., 2010; Stathopoulos et al., 2016; Glowinski et al., 2016). Examples include inverse problems in imaging (Chambolle and Pock, 2016), and recovering a model from partial measurements b on the model, in compressed sensing (Goldstein and Zhang, 2016) or sketched learning-type applications (Keriven et al., 2018). In optimal transport, one often looks for measures with fixed marginals, which can be written as an affine equality constraint (Peyré and Cuturi, 2019). Network flow optimization takes the form of Problem (1), where b contains the incoming and outgoing rates at source and sink nodes of a network, and $\mathbf{K}$ is the edge-node incidence matrix (Zargham et al., 2013). Decentralized optimization is a well-known instance of Problem (1), with $\mathbf{K}$ a gossip matrix (or its square root), and $b = 0$ (Shi et al., 2015; Scaman et al., 2017; Gorbunov et al., 2019; Li et al., 2020a,b; Ye et al., 2020; Arjevani et al., 2020; Kovalev et al., 2020; Dvinskikh and Gasnikov, 2021). If additional affine constraints are added to the decentralized optimization problem, for instance that some elements or linear measurements of the sought model $x^\star$ are fixed, decentralized optimization reverts to Problem (1) with nonzero b.

For large-scale convex optimization problems like (1), primal–dual splitting algorithms (Boţ et al., 2014; Komodakis and Pesquet, 2015; Condat et al., 2019, 2022) are well suited, as they are easy to implement and typically show state-of-the-art performance. The fully-split algorithms do not require the ability to project onto the constraint space $\{x \in \mathsf{X} : \mathbf{K}x = b\}$, and are therefore particularly adequate in the applications mentioned above. Precisely, we say that an iterative algorithm is fully split if it produces a sequence of iterates $(x^k)_{k \geq 0} \in \mathsf{X}^{\mathbb{N}}$ converging to the solution $x^\star$ of (1), using only computations of ∇F and multiplications by $\mathbf{K}$ and $\mathbf{K}^T$, the transpose of $\mathbf{K}$.

There exist several fully-split primal–dual algorithms well suited to solve Problem (1) and even more general problems (Combettes and Pesquet, 2012; Condat, 2013; Vũ, 2013; Yan, 2018; Mishchenko and Richtárik, 2019; Salim et al., 2020). In particular, we can mention the algorithm first proposed in Loris and Verhoeven (2011), and rediscovered independently as the PDFP2O algorithm (Chen et al., 2013) and the Prox-

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

imal Alternating Predictor-Corrector (PAPC) algorithm (Drori et al., 2015). For simplicity, we name it the PAPC algorithm. When applied to Problem (1), with F strongly convex, the PAPC has been proved to converge linearly by Salim et al. (Salim et al., 2020).

In this paper, we focus on the complexity of fully split algorithms to solve Problem (1), which is of primary importance in large-scale applications. That is, we study the number of gradient computations and matrix multiplications necessary to reach a given accuracy. We first derive lower bounds for these two quantities. No algorithm is known, matching these lower bounds, although nearly optimal algorithms exist in the case $b = 0$ (Dvinskikh and Gasnikov, 2021). Then, we propose a new accelerated primal–dual algorithm, which matches the lower bounds, and thus is optimal. Our algorithm can be viewed as an accelerated version of the PAPC algorithm.

In summary, our main **contributions** are the following:

- We provide complexity lower bounds for solving Problem (1) within the class of algorithms performing evaluations of ∇F and multiplications by $\mathbf{K}$ and $\mathbf{K}^T$, by a reduction to the technique in Scaman et al. (2017).
- We propose a new algorithm for solving Problem (1).
- We prove that the complexity of our algorithm matches the lower bounds, and therefore it is optimal.

The complexity results presented in this paper are dimension-independent. Under the additional assumption that the dimension is small, one can imagine alternative strategies to solve Problem (1) more efficiently. Our algorithm is meant to be applied on high-dimensional problems.

This paper is organized as follows. In Section 2, we introduce the notations and assumptions. Then, we summarize our contributions in the light of prior work in Section 3. In Section 4, we define the class of algorithms under study and we derive the corresponding complexity lower bounds for solving (1). Our main algorithm and our main result about its convergence and complexity are given in Section 5. Our approach for deriving and analyzing this algorithm is provided in Section 6. We illustrate our convergence results by numerical experiments in Section 7. The technical proofs are postponed to the Supplementary Material.

2 Mathematical Setting

Let us make the formulation of the problem (1) more precise. The convex function $F : \mathsf{X} \to \mathbb{R}$ is an L-smooth and μ-strongly convex function, for some $\mu > 0$ and $L > 0$; that is, F is differentiable and satisfies the strong convexity inequality

$$F(x) + \langle \nabla F(x), x' - x \rangle + \frac{\mu}{2}\|x - x'\|^2 \leq F(x'),$$

and the smoothness inequality

$$F(x') \leq F(x) + \langle \nabla F(x), x' - x \rangle + \frac{L}{2}\|x' - x\|^2,$$

for every $(x, x') \in \mathsf{X}^2$. That is, ∇F is L-Lipschitz continuous and $F - \frac{\mu}{2}\|\cdot\|^2$ is convex. Moreover, the Bregman divergence of F is denoted by $\mathrm{D}_F(x, x') \coloneqq F(x) - F(x') - \langle \nabla F(x'), x - x' \rangle \geq 0$. We have $0 < \mu \leq L$ and we denote by

$$\kappa \coloneqq \frac{L}{\mu} \geq 1$$

the condition number of F.

The kernel of the matrix $\mathbf{K}$ is denoted by $\ker(\mathbf{K})$ and its range by $\mathrm{range}(\mathbf{K})$. We define the symmetric positive semidefinite matrix $\mathbf{W} \coloneqq \mathbf{K}^T\mathbf{K}$. The largest eigenvalue of $\mathbf{W}$ is denoted by $\lambda_{\max}(\mathbf{W})$ and its smallest *positive* eigenvalue by $\lambda^+_{\min}(\mathbf{W})$. We have $0 < \lambda^+_{\min}(\mathbf{W}) \leq \lambda_{\max}(\mathbf{W})$ and we denote by

$$\chi(\mathbf{W}) \coloneqq \frac{\lambda_{\max}(\mathbf{W})}{\lambda^+_{\min}(\mathbf{W})} \geq 1$$

the condition number of $\mathbf{W}$.

The condition number κ (resp. $\chi(\mathbf{W})$) measures the regularity of F (resp. $\mathbf{K}$). The complexity results obtained in this paper are (nondecreasing) functions of κ (resp. $\chi(\mathbf{W})$).

We can note that $\ker(\mathbf{K}) = \ker(\mathbf{W})$. If $\ker(\mathbf{W}) = \{0\}$, the solution $x^\star$ to the linear system $\mathbf{K}x = b$ is unique, so that F does not play any role and Problem (1) reverts to solving this linear system. We allow for this case, but it is of course not the focus of this paper.

Finally, we denote by $\iota_{\{b\}}$ the indicator function of $\{b\}$; that is, $\iota_{\{b\}} : y \in \mathsf{Y} \mapsto \{0 \text{ if } y = b, +\infty \text{ otherwise}\}$. This function is convex and lower semi-continuous over Y. Denoting by ∂ the subdifferential operator (Bauschke and Combettes, 2017, Section 16), we recall that $\partial\iota_{\{b\}}(y) \neq \emptyset$ if and only if $y = b$.

3 Related Works and Summary of the Contributions

Most algorithms able to solve Problem (1) using evaluations of ∇F and multiplications by $\mathbf{K}$ and $\mathbf{K}^T$ can

be viewed as primal–dual algorithms. For instance, the Condat-Vũ algorithm (Condat, 2013; Vũ, 2013) and its variants, the PAPC algorithm (Loris and Verhoeven, 2011; Chen et al., 2013; Drori et al., 2015) can be applied to Problem (1). The PAPC algorithm applied to Problem (1) consists in iterating

$$\left\lfloor \begin{array}{l} x^{k+\frac{1}{2}} := x^k - \eta \nabla F(x^k) - \eta \mathbf{K}^T y^k \\ y^{k+1} := y^k + \theta(\mathbf{K} x^{k+\frac{1}{2}} - b) \\ x^{k+1} := x^k - \eta \nabla F(x^k) - \eta \mathbf{K}^T y^{k+1} \end{array} \right. \tag{2}$$

for some parameters $\eta, \theta > 0$; it converges in general if $\eta \in (0, \frac{2}{L})$ and $\eta\theta\|\mathbf{K}\|^2 \leq 1$, see (Condat et al., 2019), but this particular instance converges linearly (Salim et al., 2020), see Table 1. To our knowledge, the first algorithm solving Problem (1), for which linear convergence was proved, has been proposed in Mishchenko and Richtárik (2019), see Table 1.

Most of the progress in solving Problem (1), *with* $b = 0$, at an accelerated or (nearly) optimal rate have been made recently in the particular case of decentralized optimization (Scaman et al., 2017; Gorbunov et al., 2019; Li et al., 2020a,b; Ye et al., 2020; Arjevani et al., 2020; Kovalev et al., 2020; Dvinskikh and Gasnikov, 2021). In this case, $\mathbf{K}$ is typically the square root of a gossip matrix, i.e. a symmetric positive semidefinite matrix supported by a graph, whose kernel is the consensus space. In particular, optimal decentralized algorithms have been proposed using acceleration techniques (Nesterov, 2004; Auzinger, 2011; Allen-Zhu, 2017) in (Scaman et al., 2017; Kovalev et al., 2020; Li et al., 2020b). In particular, the algorithm of Scaman et al. (2017) relies on the computation of ∇F^*, where F^* is the Fenchel transform of F. Since evaluating ∇F^* is equivalent to minimizing F, Kovalev et al. (2020) proposed an algorithm relying on ∇F only. In Machine Learning applications, 'full' gradients are often intractable, therefore Li et al. (2020b) introduced a method relying on stochastic estimates of ∇F only. Each of these three methods is optimal for the class of algorithms they belong to. Our approach can be seen as an extension of (Kovalev et al., 2020) to the general setting of linearly constrained minimization, with an arbitrary right hand side b.[1]

[1] We do not assume the knowledge of a solution $\tilde{x}$ to the linear system $\mathbf{K}x = b$, otherwise one could get back to the case $b = 0$ using a change of variable. One could think of solving this linear system approximately as a preprocessing step: the typical Conjugate Gradient Method yields $\hat{x}$ with $\|\mathbf{K}\hat{x} - b\|^2 \leq \epsilon\|b\|^2$ with $\mathcal{O}(\sqrt{\chi}\log(1/\epsilon))$ complexity, where $\chi = \chi(\mathbf{W})$. But, assuming for simplicity that $\|\mathbf{K}\| = 1$, to guarantee that $\|\hat{x} - \tilde{x}\|^2 \leq \epsilon\|b\|^2$ for some $\tilde{x}$ with $\mathbf{K}\tilde{x} = b$, using the inequality $\|\hat{x} - \tilde{x}\|^2 \leq \chi\|\mathbf{K}\hat{x} - b\|^2$, the complexity becomes $\mathcal{O}(\sqrt{\chi}\log(\chi/\epsilon))$. Thus, there is an additional $\log(\chi)$ factor appearing in the complexity, which is not optimal, contrary to the proposed approach.

In the case where projecting onto the constraint space $\{x \in \mathsf{X} : \ \mathbf{K}x = b\}$ is possible, FISTA (Beck and Teboulle, 2009; Chambolle and Dossal, 2015) is an optimal algorithm for solving Problem (1). FISTA can be seen as Nesterov's acceleration (Nesterov, 2004) of the classical projected gradient algorithm.

In a nutshell, our approach consists in a rigorous combination of Nesterov's acceleration (Nesterov, 2004) to minimize a smooth and strongly convex function, and the Chebyshev iteration method (Flanders and Shortley, 1950; Golub and Loan, 1983; Auzinger, 2011; Gutknecht and Röllin, 2002) for linear system solving. Our approach allows us to accelerate the PAPC algorithm and, for the first time, to achieve the asymptotic complexity lower bounds. Our results and the most relevant results of the literature are summarized in Table 1.

4 First-Order Algorithms for the Problem

We now define the family of algorithms considered to solve Problem (1). Informally, this is the family of algorithms using gradient computations and matrix multiplications. Since no particular structure is assumed on $\mathbf{K}$, any multiplication of the iterates by $\mathbf{K}$ must be followed by a multiplication by $\mathbf{K}^T$ in order to map the iterates back into the optimization space X, before an application of ∇F. Hence, we consider the wide class of Black-Box First Order algorithms using ∇F, $\mathbf{K}$ and $\mathbf{K}^T$, denoted by BBFO$(\nabla F, \mathbf{K})$, which generate a sequence of vectors $(x^n)_{n\in\mathbb{N}} \in \mathsf{X}^{\mathbb{N}}$ such that

$$\begin{aligned} x^{n+1} \in \mathrm{Span}\Big(& x^0, \ldots, x^n, \nabla F(x^0), \ldots, \nabla F(x^n), \\ & \mathbf{K}^T \mathrm{Span}\big(b, \mathbf{K}x^0, \ldots, \mathbf{K}x^n, \mathbf{K}\nabla F(x^0), \ldots, \\ & \mathbf{K}\nabla F(x^n)\big)\Big) \end{aligned}$$

and do not apply the operators ∇F, $\mathbf{K}$ and $\mathbf{K}^T$ to other vectors. It is important to note that the index n need not coincide with the iteration counter of an iterative algorithm: each x^n can correspond to an intermediate vector in X obtained after any computation or sequence of computations during the course of the algorithm.

Theorem 1 (Lower bounds)**.** *Let $\chi \geq 1$. There exist a vector b_0, a matrix $\mathbf{K}_0$ such that the condition number of $\mathbf{K}_0^T\mathbf{K}_0$ is χ, and a smooth and strongly convex function F_0 with condition number κ, such that the following holds: for any $\varepsilon > 0$, any BBFO$(\nabla F_0, \mathbf{K}_0)$ algorithm requires at least*

- $\Omega(\sqrt{\kappa\chi}\log(1/\varepsilon))$ *multiplications by* $\mathbf{K}_0$,
- $\Omega(\sqrt{\kappa\chi}\log(1/\varepsilon))$ *multiplications by* $\mathbf{K}_0^T$,

Table 1: Comparison of the complexity of state-of-the-art algorithms with our results, in terms of gradient computations and matrix multiplications to find $x \in \mathsf{X}$ such that $\|x - x^\star\|^2 \leq \varepsilon$. The condition number of F is denoted by κ and the condition number of $\mathbf{K}^T\mathbf{K}$ is denoted by χ.

Algorithm	Gradient computations	Matrix multiplications
PAPC algorithm (Salim et al., 2020)	$\mathcal{O}\left((\kappa+\chi)\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left((\kappa+\chi)\log\frac{1}{\varepsilon}\right)$
(Mishchenko and Richtárik, 2019)	$\mathcal{O}\left((\kappa+\chi)\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left((\kappa+\chi)\log\frac{1}{\varepsilon}\right)$
(Dvinskikh and Gasnikov, 2021) (case $b=0$)	$\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log^2\frac{1}{\varepsilon}\right)$
Algorithm 1 (This paper, Theorem 2)	$\boldsymbol{\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)}$	$\boldsymbol{\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)}$
Lower bound (This paper, Theorem 1)	$\mathcal{O}\left(\sqrt{\kappa}\log\frac{1}{\varepsilon}\right)$	$\mathcal{O}\left(\sqrt{\kappa\chi}\log\frac{1}{\varepsilon}\right)$

- $\Omega(\sqrt{\kappa}\log(1/\varepsilon))$ *computations of* ∇F_0,

to output a vector x *such that* $\|x - x^\star\|^2 < \varepsilon$, *where* $x^\star = \arg\min_{\{x \,:\, \mathbf{K}_0 x = b_0\}} F_0(x)$.

Theorem 1 provides lower bounds on the number of gradient computations and matrix multiplications needed to reach ε accuracy, which here means that $\|x - x^\star\|^2 \leq \varepsilon$.

Proof. We follow the ideas of Scaman et al. (2017), in the context of decentralized optimization, to exhibit worst-case function F_0 and matrix $\mathbf{K}_0$. Let $\chi \geq 1$.

"Bad" function F_0 and "bad" matrix $\mathbf{K}_0$. Consider the family of smooth and strongly convex functions $(f_i)_{i=1}^n$ and the matrix $\mathbf{W}$ with condition number χ given by (Scaman et al., 2017, Corollary 2). Denote by κ the common condition number of f_i. Set $F_0(x_1,\dots,x_n) \coloneqq \sum_{i=1}^n f_i(x_i)$, $\mathbf{K}_0 \coloneqq \sqrt{\mathbf{W}}$ and $b_0 \coloneqq 0$. Then, the condition number of F is κ and the condition number of $\mathbf{W} = \mathbf{K}_0^T\mathbf{K}_0$ is χ. Moreover, $\mathbf{W}$ is a gossip matrix (Scaman et al., 2017, Section 2.2).

BBFO$(\nabla F_0, \mathbf{K}_0)$ are decentralized optimization algorithms. Any BBFO algorithm using these operators ∇F_0, $\mathbf{K}_0$, $\mathbf{K}_0^T$ can be rewritten as a function of ∇F_0 and $\mathbf{W} = \mathbf{K}_0^T\mathbf{K}_0$. Indeed,

$$\begin{aligned}
&\mathrm{Span}\Big(x^0,\dots,x^n,\nabla F_0(x^0),\dots,\nabla F_0(x^n),\\
&\quad \mathbf{K}_0^T\mathrm{Span}\big(b_0,\mathbf{K}_0x^0,\dots,\mathbf{K}_0x^n,\mathbf{K}_0\nabla F_0(x^0),\dots,\\
&\quad \mathbf{K}_0\nabla F_0(x^n)\big)\Big)\\
&= \mathrm{Span}\Big(x^0,\dots,x^n,\nabla F_0(x^0),\dots,\nabla F_0(x^n),\\
&\quad \mathrm{Span}\big(\mathbf{W}x^0,\dots,\mathbf{W}x^n,\mathbf{W}\nabla F_0(x^0),\dots,\\
&\quad \mathbf{W}\nabla F_0(x^n)\big)\Big).
\end{aligned}$$

Since $\mathbf{W}$ is a gossip matrix, BBFO$(\nabla F_0, \mathbf{K}_0)$ algorithms are therefore Black-box optimization procedures using $\mathbf{W}$, in the sense of (Scaman et al., 2017, Section 3.1). In other words, BBFO$(\nabla F_0, \mathbf{K}_0)$ algorithms are decentralized optimization algorithms over a network, in which communication amounts to multiplication by $\mathbf{W}$, and local computations correspond to evaluations of ∇F.

Any solution to (1) is a solution to a decentralized optimization problem. Since $\ker(\mathbf{W})$ is the consensus space, $x^\star = \arg\min_{\{x \,:\, \mathbf{W}x=0\}} F_0(x)$ can be written as $x^\star = (x_0^\star,\dots,x_0^\star)$ where $x_0^\star = \arg\min \frac{1}{n}\sum_{i=1}^n f_i$.

BBFO$(\nabla F_0, \mathbf{K}_0)$ algorithms cannot outperform the lower bounds of decentralized algorithms. As shown in Scaman et al. (2017, Corollary 2), for any $\varepsilon > 0$, any Black-box optimization procedure using $\mathbf{W}$ 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 $x = (x_1,\dots,x_n)$ such that $\|x - x^\star\|^2 < \varepsilon$, where $x^\star = \arg\min F_0$. *In particular*, for any $\varepsilon > 0$, any BBFO$(\nabla F_0, \mathbf{K}_0)$ algorithm requires at least $\Omega\left(\sqrt{\kappa\chi}\log(1/\varepsilon)\right)$ multiplications by $\mathbf{K}_0^T\mathbf{K}_0$, and at least $\Omega\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$ computations of ∇F_0 to output $x = (x_1,\dots,x_n)$ such that $\|x - x^\star\|^2 < \varepsilon$, where $x^\star = \arg\min F_0$.

Finally, one multiplication by $\mathbf{W}$ is equivalent to one multiplication by $\mathbf{K}_0$ followed by one multiplication by $\mathbf{K}_0^T$. □

5 Proposed Algorithm

In this section, we present our main algorithm, Algorithm 1, and our main convergence result, Theorem 2. The derivations and proofs are deferred to Section 6. Algorithm 2 implements the classical Chebyshev iteration (Flanders and Shortley, 1950; Golub and Loan, 1983; Auzinger, 2011; Gutknecht and Röllin, 2002), see Section 6.3 for details. It is used as a subroutine in Algorithm 1 and denoted by Chebyshev, with its parameters passed as arguments. We stress here

Algorithm 1 Proposed algorithm

Parameters: $x^0 \in \mathsf{X}$, $N \in \mathbb{N}^*$, $\tau \in (0,1)$,
$\lambda_1, \lambda_2, \eta, \theta, \alpha > 0$
$x_f^0 \coloneqq x^0$, $u^0 \coloneqq 0_{\mathsf{X}}$
for $k = 0, 1, \ldots$ **do**
  $x_g^k \coloneqq \tau x^k + (1-\tau) x_f^k$
  $x^{k+\frac{1}{2}} \coloneqq (1+\eta\alpha)^{-1}\big(x^k - \eta(\nabla F(x_g^k)$
    $- \alpha x_g^k + u^k)\big)$
  $r^k \coloneqq \theta\big(x^{k+\frac{1}{2}}$
    $-$ Chebyshev$(x^{k+\frac{1}{2}}, \mathbf{K}, b, N, \lambda_1, \lambda_2)\big)$
  $u^{k+1} \coloneqq u^k + r^k$
  $x^{k+1} \coloneqq x^{k+\frac{1}{2}} - \eta(1+\eta\alpha)^{-1} r^k$
  $x_f^{k+1} \coloneqq x_g^k + \frac{2\tau}{2-\tau}(x^{k+1} - x^k)$
end for

Algorithm 2 Chebyshev iteration

Parameters: $z^0 \in \mathsf{X}, \mathbf{K}, b \in \mathsf{Y}, N \in \mathbb{N}^*, \lambda_1 > 0$, $\lambda_2 > 0$.
$\rho \coloneqq \big(\lambda_1 - \lambda_2\big)^2/16$, $\nu \coloneqq (\lambda_1 + \lambda_2)/2$
$\gamma^0 \coloneqq -\nu/2$
$p^0 \coloneqq -\mathbf{K}^T(\mathbf{K}z^0 - b)/\nu$
$z^1 \coloneqq z^0 + p^0$
for $i = 1, \ldots, N-1$ **do**
  $\beta^{i-1} \coloneqq \rho/\gamma^{i-1}$
  $\gamma^i \coloneqq -(\nu + \beta^{i-1})$
  $p^i \coloneqq \big(\mathbf{K}^T(\mathbf{K}z^i - b) + \beta^{i-1}p^{i-1}\big)/\gamma^i$
  $z^{i+1} \coloneqq z^i + p^i$
end for
Output: z^N

that the Chebyshev iteration is diverted from its usual use, which is solving linear systems, and is used here as a preconditioner (a similar idea appears in Bredies and Sun (2015)). Although Algorithm 1 runs at every iteration a number N of Chebyshev iterations, there is no approximation or truncation error here: Algorithm 1 converges to the exact solution $x^\star$ of Problem (1). This is achieved without solving the full linear system $\mathbf{K}x = b$ at each iteration.

Theorem 2 (Convergence of Algorithm 1)**.** *Consider $\lambda_1 \geq \lambda_{\max}(\mathbf{W})$ and λ_2 such that $0 < \lambda_2 \leq \lambda_{\min}^+(\mathbf{W})$. Let $\chi \coloneqq \frac{\lambda_1}{\lambda_2}$ and choose $N \geq \sqrt{\chi}$.*

Set the parameters $\tau, \eta, \theta, \alpha$ as $\tau \coloneqq \min\left\{1, \frac{1}{2}\sqrt{\frac{19}{15\kappa}}\right\}$, $\eta \coloneqq \frac{1}{4\tau L}$, $\theta \coloneqq \frac{15}{19\eta}$, and $\alpha \coloneqq \mu$. Then, there exists $C \geq 0$ such that

$$\frac{1}{\eta}\left\|x^k - x^\star\right\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \leq \left(1 + \frac{1}{4}\min\left\{\frac{15}{19}, \sqrt{\frac{15}{19\kappa}}\right\}\right)^{-k} C.$$

Moreover, for every $\varepsilon > 0$, Algorithm 1 finds x^k for which $\|x^k - x^\star\|^2 \leq \varepsilon$ using $\mathcal{O}\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$ gradient computations and $\mathcal{O}\left(N\sqrt{\kappa}\log(1/\varepsilon)\right)$ matrix multiplications with $\mathbf{K}$ or $\mathbf{K}^T$.

Corollary 1 (Tight version of Theorem 2)**.** *Set the parameters $\lambda_1, \lambda_2, N, \tau, \eta, \theta, \alpha$ to $\lambda_1 = \lambda_{\max}(\mathbf{W})$, $\lambda_2 = \lambda_{\min}^+(\mathbf{W})$, $N = \lceil\sqrt{\chi(\mathbf{W})}\rceil$, $\tau = \min\left\{1, \frac{1}{2}\sqrt{\frac{19}{15\kappa}}\right\}$, $\eta = \frac{1}{4\tau L}$, $\theta = \frac{15}{19\eta}$, and $\alpha = \mu$. Then, for every $\varepsilon > 0$, Algorithm 1 finds x^k for which $\|x^k - x^\star\|^2 \leq \varepsilon$ using $\mathcal{O}\left(\sqrt{\kappa}\log(1/\varepsilon)\right)$ gradient computations and $\mathcal{O}\left(\sqrt{\kappa\chi(\mathbf{W})}\log(1/\varepsilon)\right)$ matrix multiplications with $\mathbf{K}$ or $\mathbf{K}^T$.*

The complexity result given by Corollary 1 is summarized in Table 1. Algorithm 1 is a BBFO algorithm because each step of each iteration is a BBFO update. Thus, the complexity of Algorithm 1 matches the lower bounds of Theorem 1, in terms of both gradient computations and matrix multiplications.

6 Derivation of the Algorithm and Proof of Theorem 2

In this section, we explain how we derive our main algorithm from the PAPC algorithm and prove Theorem 2 step by step. First, we derive the primal–dual optimality conditions associated to Problem (1).

6.1 Primal–Dual Optimality Conditions

First, note that $\arg\min_{\mathbf{K}x=b} F(x) = \arg\min F(x) + \iota_{\{b\}}(\mathbf{K}x)$. Define the strongly convex function $G : x \mapsto F(x) + \iota_{\{b\}}(\mathbf{K}x)$. Then, $0 \in \partial G(x^\star) = \nabla F(x^\star) + \mathbf{K}^T\partial\iota_{\{b\}}(\mathbf{K}x^\star)$ (Bauschke and Combettes, 2017, Theorem 16.47). This means that there exists $y^\star \in \partial\iota_{\{b\}}(\mathbf{K}x^\star)$ such that $0 = \nabla F(x^\star) + \mathbf{K}^T y^\star$. Besides, $\partial\iota_{\{b\}}(\mathbf{K}x^\star)$ is nonempty if and only if $\mathbf{K}x^\star = b$. Finally, the pair $(x^\star, y^\star)$ must satisfy

$$\begin{cases} 0 = \nabla F(x^\star) + \mathbf{K}^T y^\star, \\ 0 = -\mathbf{K}x^\star + b. \end{cases} \tag{3}$$

These equations are called primal–dual optimality conditions, and are also the first-order conditions associated to the Lagrangian function $\mathcal{L}(x,y) \coloneqq F(x) + \langle \mathbf{K}x - b, y\rangle$ associated to Problem (1). Moreover, $(x^\star, y^\star)$ is called an optimal primal–dual pair. If $(x^\star, y^\star)$ is an optimal primal–dual pair, then $(x^\star, y^\star + \bar{y})$, where $\bar{y} \in \ker(\mathbf{K}^T)$, is also an optimal primal–dual pair. Thus, in the sequel, we denote by $(x^\star, y^\star)$ the only optimal primal–dual pair such that $y^\star \in$

Algorithm 3 Intermediate algorithm

Parameters: $x^0 \in \mathsf{X}$, $y^0 = 0_{\mathsf{Y}}$, $\eta, \theta, \alpha > 0$, $\tau \in (0,1)$
Set $x_f^0 = x^0$
for $k = 0, 1, 2, \dots$ **do**
  $x_g^k \coloneqq \tau x^k + (1-\tau) x_f^k$
  $x^{k+\frac{1}{2}} \coloneqq (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + \mathbf{K}^T y^k))$
  $y^{k+1} \coloneqq y^k + \theta(\mathbf{K} x^{k+\frac{1}{2}} - b)$
  $x^{k+1} \coloneqq (1+\eta\alpha)^{-1}(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + \mathbf{K}^T y^{k+1}))$
  $x_f^{k+1} \coloneqq x_g^k + \frac{2\tau}{2-\tau}(x^{k+1} - x^k)$
end for

range($\mathbf{K}$); that is, such that

$$\begin{cases} 0 = \nabla F(x^\star) + \mathbf{K}^T y^\star, \quad y^\star \in \operatorname{range}(\mathbf{K}), \\ 0 = -\mathbf{K} x^\star + b. \end{cases} \tag{4}$$

We can note that the sequence of iterates (x^k, y^k) of the PAPC algorithm, shown in (2), converges linearly to $(x^\star, y^\star)$ (Salim et al., 2020, Theorem 8), as reported in Table 1.

6.2 Nesterov's Acceleration

The first step to derive Algorithm 1 is to propose a variant of the PAPC (2) using Nesterov's acceleration (Nesterov, 2004). Nesterov acceleration is now classical for proximal gradient descent but its extension to primal-dual settings remains an open area. This intermediate algorithm is Algorithm 3, shown above. Its convergence is stated in Proposition 1.

Proposition 1 (Algorithm 3)**.** *Consider* $\lambda_1 \geq \lambda_{\max}(\mathbf{W})$ *and* λ_2 *such that* $0 < \lambda_2 \leq \lambda_{\min}^+(\mathbf{W})$*. Denote* $\chi \coloneqq \frac{\lambda_1}{\lambda_2}$*.*

Set the parameters of Algorithm 3 as $\tau \coloneqq \min\left\{1, \frac{1}{2}\sqrt{\frac{\chi}{\kappa}}\right\}$, $\eta \coloneqq \frac{1}{4\tau L}$, $\theta \coloneqq \frac{1}{\eta\lambda_1}$, *and* $\alpha \coloneqq \mu$*. Then,*

$$\frac{1}{\eta}\left\|x^k - x^\star\right\|^2 + \frac{\eta\alpha}{\theta(1+\eta\alpha)}\left\|y^k - y^\star\right\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \leq \left(1 + \frac{1}{4}\min\left\{\frac{1}{\sqrt{\kappa\chi}}, \frac{1}{\chi}\right\}\right)^{-k} C, \tag{5}$$

where $C \coloneqq \frac{1}{\eta}\left\|x^0 - x^\star\right\|^2 + \frac{1}{\theta}\|y^0 - y^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^0, x^\star)$.

Proposition 1 states the linear convergence of the distance between the iterates and the primal–dual optimal point. In particular, if $\lambda_1 = \lambda_{\max}(\mathbf{W})$ and $\lambda_2 = \lambda_{\min}^+(\mathbf{W})$, then $\|x - x^\star\|^2 \leq \varepsilon$ after

$$\mathcal{O}\left(\left(\sqrt{\kappa\chi(\mathbf{W})} + \chi(\mathbf{W})\right)\log\left(\tfrac{1}{\varepsilon}\right)\right)$$

gradient computations and matrix multiplications. Besides, Proposition 1 states the linear convergence of the Bregman divergence of F. Using (4), one can check that the Bregman divergence of F is equal to the restricted primal–dual gap, in particular: $\mathrm{D}_F(x_f^k, x^\star) = \mathcal{L}(x_f^k, y^\star) - \mathcal{L}(x^\star, y^k)$.

The proof of Proposition 1 is provided in the Supplementary Material. The main tool of the proof is the following representation of Algorithm 3.

We denote by $\mathbf{Q}$ the $(d+p)\times(d+p)$ matrix defined blockwise by

$$\mathbf{Q} \coloneqq \begin{bmatrix} \frac{1}{\eta}\mathbf{I}_{\mathsf{X}} & 0 \\ 0 & \frac{1}{\theta}\mathbf{I}_{\mathsf{Y}} - \frac{\eta}{1+\eta\alpha}\mathbf{K}\mathbf{K}^T \end{bmatrix}, \tag{6}$$

where $\mathbf{I}_{\mathsf{X}}$ (resp. $\mathbf{I}_{\mathsf{Y}}$) is the identity matrix over X (resp. Y).

Lemma 1. *The following equality holds:*

$$\mathbf{Q}\begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix} = \begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + \mathbf{K}^T y^{k+1}) \\ \mathbf{K} x^{k+1} - b \end{bmatrix}. \tag{7}$$

Lemma 1, proved in the Supplementary Material, enables to view Algorithm 3 as a variant of the Forward–Backward algorithm involving monotone operators, see (Bauschke and Combettes, 2017, Section 26.14) or (Condat et al., 2019) for more details. The Forward–Backward algorithm is a fixed-point algorithm. For instance, one can see in Equation (7) that a fixed point $(x^k, y^k) = (x^\star, y^\star)$ is a solution to (3). Hence, Algorithm 3 can be viewed as an accelerated primal–dual fixed-point algorithm.

6.3 Chebyshev's Acceleration

Our main Algorithm 1 is obtained as a particular instantiation of Algorithm 3. More precisely, we use a finite number of steps of the Chebyshev iteration (Flanders and Shortley, 1950; Golub and Loan, 1983; Auzinger, 2011; Gutknecht and Röllin, 2002) to precondition the linear system and accelerate the resolution of Problem (1). This idea was already applied in the particular setting of decentralized optimization (Scaman et al., 2017, 2018).

Consider a polynomial $\mathbf{P}$ such that, for every eigenvalue t of $\mathbf{W}$, $\mathbf{P}(t) \geq 0$ and $(\mathbf{P}(t) = 0 \Leftrightarrow t = 0)$. Since $\mathbf{K}x^\star = b$,

$$\begin{aligned} \mathbf{K}x = b &\Leftrightarrow \mathbf{K}(x - x^\star) = 0 \Leftrightarrow \mathbf{K}^T\mathbf{K}(x - x^\star) = 0 \\ &\Leftrightarrow \mathbf{W}(x - x^\star) = 0 \Leftrightarrow \mathbf{P}(\mathbf{W})(x - x^\star) = 0 \\ &\Leftrightarrow \sqrt{\mathbf{P}(\mathbf{W})}(x - x^\star) = 0 \\ &\Leftrightarrow \sqrt{\mathbf{P}(\mathbf{W})}x = \sqrt{\mathbf{P}(\mathbf{W})}x^\star. \end{aligned}$$

Therefore, the problem

$$\min_{x\in\mathsf{X}} F(x) \quad \text{s.t.} \quad \sqrt{\mathbf{P}(\mathbf{W})}x = \sqrt{\mathbf{P}(\mathbf{W})}x^\star, \tag{8}$$

is equivalent to Problem (1). Consequently, to solve Problem (1), one can apply Algorithm 3 by replacing $\mathbf{K}$ by $\sqrt{\mathbf{P}(\mathbf{W})}$ and b by $\sqrt{\mathbf{P}(\mathbf{W})}x^\star$; we will see below that $x^\star$ is not needed in the computations, only b is. Since $\sqrt{\mathbf{P}(\mathbf{W})}$ is symmetric, this leads to the following algorithm:

$$\left\lfloor \begin{aligned} x_g^k &\coloneqq \tau x^k + (1-\tau)x_f^k \\ x^{k+\frac{1}{2}} &\coloneqq (1+\eta\alpha)^{-1}\big(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k \\ &\qquad + \sqrt{\mathbf{P}(\mathbf{W})}y^k)\big) \\ y^{k+1} &\coloneqq y^k + \theta\big(\sqrt{\mathbf{P}(\mathbf{W})}x^{k+\frac{1}{2}} - \sqrt{\mathbf{P}(\mathbf{W})}x^\star\big) \\ x^{k+1} &\coloneqq (1+\eta\alpha)^{-1}\big(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k \\ &\qquad + \sqrt{\mathbf{P}(\mathbf{W})}y^{k+1})\big) \\ x_f^{k+1} &\coloneqq x_g^k + \tfrac{2\tau}{2-\tau}(x^{k+1} - x^k) \end{aligned} \right. . \tag{9}$$

After applying the change of variable $u^k \coloneqq \sqrt{\mathbf{P}(\mathbf{W})}y^k$, we get:

$$\left\lfloor \begin{aligned} x_g^k &\coloneqq \tau x^k + (1-\tau)x_f^k \\ x^{k+\frac{1}{2}} &\coloneqq (1+\eta\alpha)^{-1}\big(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + u^k)\big) \\ u^{k+1} &\coloneqq u^k + \theta\big(\mathbf{P}(\mathbf{W})x^{k+\frac{1}{2}} - \mathbf{P}(\mathbf{W})x^\star\big) \\ x^{k+1} &\coloneqq (1+\eta\alpha)^{-1}\big(x^k - \eta(\nabla F(x_g^k) - \alpha x_g^k + u^{k+1})\big) \\ x_f^{k+1} &\coloneqq x_g^k + \tfrac{2\tau}{2-\tau}(x^{k+1} - x^k). \end{aligned} \right. \tag{10}$$

To obtain Algorithm 1 and Theorem 2, we have to choose a suitable polynomial $\mathbf{P}$ and show how to compute $\mathbf{P}(\mathbf{W})x^{k+\frac{1}{2}} - \mathbf{P}(\mathbf{W})x^\star$ efficiently.

6.3.1 Choice of P

The goal is to make Problem (8) better conditioned than Problem (1). For this, we want $\mathbf{P}$ to cluster all the positive eigenvalues of $\mathbf{W}$ around the same value, say 1 (the scaling of $\mathbf{P}$ does not matter, since it is compensated by the stepsizes). To that aim, the best choice is to set $\mathbf{P}$ as 1 minus a Chebyshev polynomial of appropriate degree (Auzinger, 2011, Theorem 6.1). More precisely, let $\boldsymbol{T}_n$ be the Chebyshev polynomial of the first kind of degree $n \geq 0$, which is such that $\{\boldsymbol{T}_n(t) \ : \ t \in [-1,1]\} = [-1,1]$. Let $\lambda_1 \geq \lambda_{\max}(\mathbf{W})$ and $0 < \lambda_2 \leq \lambda_{\min}^+(\mathbf{W})$ be upper and lower bounds of the eigenvalues of $\mathbf{W}$. Set $\chi \coloneqq \lambda_1/\lambda_2 \geq \chi(\mathbf{W}) \geq 1$.

If $\lambda_1 = \lambda_2$, no preconditioning is necessary and we could just set $\mathbf{P}(\mathbf{W}) = \mathbf{W}$. So, let us assume that $\lambda_2 < \lambda_1$ (the derivations can be shown to be still valid if $\lambda_2 = \lambda_1$).

For every $n \geq 1$, we define the shifted Chebyshev polynomial $\widetilde{\boldsymbol{T}}_n$ as

$$\widetilde{\boldsymbol{T}}_n(t) = \frac{\boldsymbol{T}_n\big((\lambda_1+\lambda_2-2t)/(\lambda_1-\lambda_2)\big)}{\boldsymbol{T}_n\big((\lambda_1+\lambda_2)/(\lambda_1-\lambda_2)\big)}. \tag{11}$$

Then, for every $n \geq 1$, $\widetilde{\boldsymbol{T}}_n(0) = 1$, $\widetilde{\boldsymbol{T}}_n(t)$ decreases monotonically for $t \in [0, \lambda_2]$, and

$$\begin{aligned} \max_{t\in[\lambda_2,\lambda_1]} |\widetilde{\boldsymbol{T}}_n(t)| &= \frac{1}{\boldsymbol{T}_n\big((\lambda_1+\lambda_2)/(\lambda_1-\lambda_2)\big)} \\ &= \frac{2\zeta^n}{1+\zeta^{2n}} < 1, \text{ where } \zeta = \frac{\sqrt{\chi}-1}{\sqrt{\chi}+1} < 1, \end{aligned} \tag{12}$$

see (Auzinger, 2011, Corollary 6.1). Hence, if $N \geq \sqrt{\chi}$, then

$$\max_{t\in[\lambda_2,\lambda_1]} |\widetilde{\boldsymbol{T}}_N(t)| < 0.266 < \frac{4}{15}. \tag{13}$$

Indeed, $-1/\ln((t-1)/(t+1)) < t/2$ for every $t \geq 1$, therefore by setting $t = \sqrt{\chi}$ we obtain $N \geq \sqrt{\chi} \Rightarrow N > -2/\ln(\zeta) \Rightarrow \zeta^N < e^{-2} \Rightarrow 2\zeta^N/(1+\zeta^{2N}) < 0.266$.

Therefore, we set

$$\mathbf{P} \coloneqq 1 - \widetilde{\boldsymbol{T}}_N \tag{14}$$

for some $N \geq \sqrt{\chi}$. Then, we have

$$\begin{aligned} &\lambda_{\max}(\mathbf{P}(\mathbf{W})) \leq \max_{t\in[\lambda_2,\lambda_1]} \mathbf{P}(t) \leq 1 + \max_{t\in[\lambda_2,\lambda_1]} |\widetilde{\boldsymbol{T}}_N(t)| \leq \frac{19}{15}, \\ &\lambda_{\min}^+(\mathbf{P}(\mathbf{W})) \geq \min_{t\in[\lambda_2,\lambda_1]} \mathbf{P}(t) \geq 1 - \max_{t\in[\lambda_2,\lambda_1]} |\widetilde{\boldsymbol{T}}_N(t)| \geq \frac{11}{15}, \\ &\chi\big(\mathbf{P}(\mathbf{W})\big) \leq \frac{19}{11}. \end{aligned}$$

6.3.2 Efficient Computation of $\mathbf{P}(\mathbf{W})x - \mathbf{P}(\mathbf{W})x^\star$ without Knowing $x^\star$

We still have to show how to compute $\mathbf{P}(\mathbf{W})x - \mathbf{P}(\mathbf{W})x^\star$, for any $x \in \mathsf{X}$. Consider $N \geq 1$ and $\mathbf{P}$ defined in (14). Now, we can observe that Algorithm 1 is equivalent to the iterations (10), and there remains to prove that for every $x \in \mathsf{X}$,

$$\mathbf{P}(\mathbf{W})x - \mathbf{P}(\mathbf{W})x^\star = x - \text{Chebyshev}(x, \mathbf{K}, b, N, \lambda_1, \lambda_2). \tag{15}$$

The vector $z^N = \text{Chebyshev}(x, \mathbf{K}, b, N)$ is the N^{th} iterate of the classical Chebyshev iteration to solve the linear system $\mathbf{K}z = b$, or equivalently $\mathbf{W}z = \mathbf{K}^T b$, starting with some initial guess $z^0 = x$, using the recurrence relation of the Chebyshev polynomials $\widetilde{\boldsymbol{T}}_N$, see Algorithm 4 in Gutknecht and Röllin (2002)[2]. The rest of the proof is given in the Supplementary Material.

7 Experiments

We illustrate the performance of our Algorithm 1 in a compressed-sensing-type experiment: we want to estimate a sparse vector $x^\sharp \in \mathsf{X} = \mathbb{R}^d$, with $d = 1000$,

[2]Several recurrence relations can be used to compute $\widetilde{\boldsymbol{T}}_n$, and we chose Algorithm 4 in Gutknecht and Röllin (2002) because it is proved to be numerically stable.

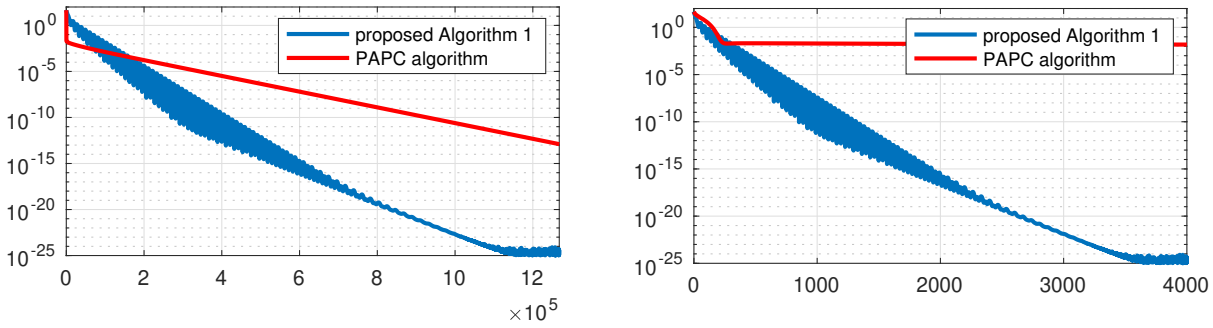

Figure 1: Error $\|x - x^\star\|^2$ with respect to the number of calls to $\mathbf{K}$ and $\mathbf{K}^T$ to obtain x (left) and to the number of calls to ∇F, equal to the number k of iterations, to obtain $x = x^k$ (right).

having 50 randomly chosen nonzero elements (equal to 1) from $b = \mathbf{K}x^\sharp \in \mathsf{Y} = \mathbb{R}^p$, with $p = 250$, where $\mathbf{K}$ has random i.i.d. Gaussian elements and its nonzero singular values are modified so that they span the interval $[1/\sqrt{\chi}, 1]$ for some prescribed value of χ. Solving Problem (1) with F the ℓ_1 norm yields perfect reconstruction with $x^\star = x^\sharp$. Thus, without pretending in any way that this is the best way to solve this estimation problem, we solve Problem (1) with F a L-smooth and μ-strongly convex approximation of the ℓ_1 norm: we set $F : x = (x_i)_{i=1}^d \in \mathsf{X} \mapsto \sum_{i=1}^d f(x_i)$ with $f : t \in \mathbb{R} \mapsto \sqrt{t^2 + e^2} + (e/2)t^2$, for some $e > 0$, so that $L = 1/e + e$, $\mu = e$, $\kappa = L/\mu = 1 + 1/e^2$. So, given a prescribed value of κ, we set $e = \sqrt{1/(\kappa - 1)}$. The results are shown in Figure 1 for Algorithm 1 and the PAPC algorithm, for $\chi = 10^5$ and $\kappa = 10^4$; other values gave similar plots. The computation time is roughly the same as the number of calls to $\mathbf{K}$ and $\mathbf{K}^T$ here.

Both algorithms converge linearly, but Algorithm 1 has a much better rate, which corresponds visually to the slope of the curves in Figure 1. Algorithm 1 makes $N = 317$ calls to $\mathbf{K}$ and $\mathbf{K}^T$ corresponding to the Chebyshev 'inner loop' between two gradient evaluations. It needs less gradient calls than PAPC to achieve the same accuracy. The red curve in the right plot is the same as in the left plot, but stretched horizontally by a factor $N = 317$ (note the change in horizontal scale).

Acknowledgements

AS was supported by KAUST and by a Simons-Berkeley Research Fellowship.

References

Allen-Zhu, Z. (2017). Katyusha: The first direct acceleration of stochastic gradient methods. *The Journal of Machine Learning Research*, 18(1):8194–8244.

Arjevani, Y., Bruna, J., Can, B., Gürbüzbalaban, M., Jegelka, S., and Lin, H. (2020). IDEAL: Inexact DEcentralized Accelerated Augmented Lagrangian Method. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*.

Auzinger, W. (2011). Iterative solution of large linear systems. *Lecture notes*.

Bach, F., Jenatton, R., Mairal, J., and Obozinski, G. (2012). Optimization with sparsity-inducing penalties. *Found. Trends Mach. Learn.*, 4(1):1–106.

Bauschke, H. H., Burachik, R., Combettes, P. L., Elser, V., Luke, D. R., and Wolkowicz, H., editors (2010). *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer-Verlag, New York.

Bauschke, H. H. and Combettes, P. L. (2017). *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York, 2nd edition.

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.

Boţ, R. I., Csetnek, E. R., and Hendrich, C. (2014). Recent developments on primal–dual splitting methods with applications to convex minimization. In Pardalos, P. M. and Rassias, T. M., editors, *Mathematics Without Boundaries: Surveys in Interdisciplinary Research*, pages 57–99. Springer New York.

Bredies, K. and Sun, H. (2015). Preconditioned douglas–rachford splitting methods for convex-concave saddle-point problems. *SIAM Journal on Numerical Analysis*, 53(1):421–444.

Chambolle, A. and Dossal, C. (2015). On the convergence of the iterates of the "Fast Iterative Shrinkage/Thresholding Algorithm". *J. Optim. Theory Appl.*, 166:968–982.

Chambolle, A. and Pock, T. (2016). An introduction

to continuous optimization for imaging. *Acta Numerica*, 25:161–319.

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).

Combettes, P. L. and Pesquet, J.-C. (2012). Primal–dual splitting algorithm for solving inclusions with mixtures of composite, Lipschitzian, and parallel-sum type monotone operators. *Set-Val. Var. Anal.*, 20(2):307–330.

Condat, L. (2013). A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *J. Optim. Theory Appl.*, 158(2):460–479.

Condat, L., Kitahara, D., Contreras, A., and Hirabayashi, A. (2019). Proximal splitting algorithms for convex optimization: A tour of recent advances, with new twists. preprint arXiv:1912.00137.

Condat, L., Malinovsky, G., and Richtárik, P. (2022). Distributed proximal splitting algorithms with rates and acceleration. *Frontiers in Signal Processing*.

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

Dvinskikh, D. and Gasnikov, A. (2021). Decentralized and parallel primal and dual accelerated methods for stochastic convex programming problems. *Journal of Inverse and Ill-posed Problems*.

Flanders, D. A. and Shortley, G. (1950). Numerical determination of fundamental modes. *Journal of Applied Physics*, 21:1326–1332.

Glowinski, R., Osher, S. J., and Yin, W., editors (2016). *Splitting Methods in Communication, Imaging, Science, and Engineering*. Springer International Publishing.

Goldstein, T. and Zhang, X. (2016). Operator splitting methods in compressive sensing and sparse approximation. In Glowinski, R., Osher, S. J., and Yin, W., editors, *Splitting Methods in Communication, Imaging, Science, and Engineering*, pages 301–343, Cham. Springer International Publishing.

Golub, G. H. and Loan, C. F. V. (1983). *Matrix computations*. Johns Hopkins Univ. Press, Baltimore.

Gorbunov, E., Dvinskikh, D., and Gasnikov, A. (2019). Optimal decentralized distributed algorithms for stochastic convex optimization. *arXiv preprint arXiv:1911.07363*.

Gutknecht, M. H. and Röllin, S. (2002). The Chebyshev iteration revisited. *Parallel Computing*, 28:263–283.

Keriven, N., Bourrier, A., Gribonval, R., and Pérez, P. (2018). Sketching for large-scale learning of mixture models. *Information and Inference: a Journal of the IMA*, 7(3):447–508.

Komodakis, N. and Pesquet, J.-C. (2015). Playing with duality: An overview of recent primal–dual approaches for solving large-scale optimization problems. *IEEE Signal Process. Mag.*, 32(6):31–54.

Kovalev, D., Salim, A., and Richtárik, P. (2020). Optimal and practical algorithms for smooth and strongly convex decentralized optimization. In *Proc. of Conf. on Neural Information Processing Systems (NeurIPS)*.

Li, H., Fang, C., Yin, W., and Lin, Z. (2020a). Decentralized accelerated gradient methods with increasing penalty parameters. *IEEE Transactions on Signal Processing*, 68:4855–4870.

Li, H., Lin, Z., and Fang, Y. (2020b). Optimal accelerated variance reduced EXTRA and DIGING for strongly convex and smooth decentralized optimization. *arXiv preprint arXiv:2009.04373*.

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).

Mishchenko, K. and Richtárik, P. (2019). A stochastic decoupling method for minimizing the sum of smooth and non-smooth functions. preprint arXiv:1905.11535.

Nesterov, Y. (2004). *Introductory Lectures on Convex Optimization*. Kluwer Academic Publisher, Dordrecht, The Netherlands.

Peyré, G. and Cuturi, M. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends in Machine Learning*, 11(5–6):355–607.

Polson, N. G., Scott, J. G., and Willard, B. T. (2015). Proximal algorithms in statistics and machine learning. *Statist. Sci.*, 30(4):559–581.

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

vances 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 J. Optim.*, 25(2):944–966.

Sra, S., Nowozin, S., and Wright, S. J. (2011). *Optimization for Machine Learning*. The MIT Press.

Stathopoulos, G., Shukla, H., Szucs, A., Pu, Y., and Jones, C. N. (2016). Operator splitting methods in control. *Foundations and Trends in Systems and Control*, 3(3):249–362.

Vũ, B. C. (2013). A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Adv. Comput. Math.*, 38(3):667–681.

Yan, M. (2018). A new primal-dual algorithm for minimizing the sum of three functions with a linear operator. *J. Sci. Comput.*, 76(3):1698–1717.

Ye, H., Luo, L., Zhou, Z., and Zhang, T. (2020). Multi-consensus decentralized accelerated gradient descent. *arXiv preprint arXiv:2005.00797*.

Zargham, M., Ribeiro, A., Ozdaglar, A., and Jadbabaie, A. (2013). Accelerated dual descent for network flow optimization. *IEEE Trans. Automat. Contr.*, 59(4):905–920.

An Optimal Algorithm for Strongly Convex Minimization under Affine Constraints: Supplementary Material

8 Proof of Proposition 1

We denote by $\|\cdot\|_{\mathbf{Q}}$ (resp. $\langle\cdot,\cdot\rangle_{\mathbf{Q}}$) the norm (resp. inner product) induced by $\mathbf{Q}$, defined in (6). The norm $\|\cdot\|_{\mathbf{Q}}$ satisfies the following properties, stated as lemmas.

8.1 Preliminary Lemmas

Lemma 2. *If the parameters $\eta > 0$ and $\theta > 0$ satisfy*

$$\eta\theta\lambda_{\max}(\mathbf{W}) \leq 1, \tag{16}$$

and if $\alpha > 0$, then the symmetric matrix $\mathbf{Q}$ is positive definite and for every $x \in \mathsf{X}$, $y \in \mathsf{Y}$, the following inequality holds:

$$\frac{1}{\eta}\|x\|^2 \leq \frac{1}{\eta}\|x\|^2 + \frac{\eta\alpha}{\theta(1+\eta\alpha)}\|y\|^2 \leq \left\|\begin{bmatrix}x\\y\end{bmatrix}\right\|_{\mathbf{Q}}^2 \leq \frac{1}{\eta}\|x\|^2 + \frac{1}{\theta}\|y\|^2. \tag{17}$$

Proof. The nonzero eigenvalues of $\mathbf{W} = \mathbf{K}^T\mathbf{K}$ are the nonzero eigenvalues of $\mathbf{K}\mathbf{K}^T$, therefore $\lambda_{\max}(\mathbf{W}) = \lambda_{\max}(\mathbf{K}\mathbf{K}^T)$. Consequently, using (16),

$$\frac{\eta}{1+\eta\alpha}\|\mathbf{K}^T y\|^2 \leq \frac{\eta}{1+\eta\alpha}\lambda_{\max}(\mathbf{W})\|y\|^2 \leq \frac{\|y\|^2}{\theta(1+\eta\alpha)}.$$

Therefore, since $\alpha\eta > 0$,

$$\frac{1}{\eta}\|x\|^2 + \left(\frac{1}{\theta} - \frac{1}{\theta(1+\eta\alpha)}\right)\|y\|^2 \leq \left\|\begin{bmatrix}x\\y\end{bmatrix}\right\|_{\mathbf{Q}}^2 = \frac{1}{\eta}\|x\|^2 + \frac{1}{\theta}\|y\|^2 - \frac{\eta}{1+\eta\alpha}\|\mathbf{K}^T y\|^2,$$

which proves in particular that $\mathbf{Q}$ is positive definite. □

Besides, lines 5 to 7 of Algorithm 3 admit the following representation, which is at the core of the convergence proof.

Lemma 3. *The following equality holds:*

$$\mathbf{Q}\begin{bmatrix}x^{k+1}-x^k\\y^{k+1}-y^k\end{bmatrix} = \begin{bmatrix}\alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + \mathbf{K}^T y^{k+1})\\ \mathbf{K}x^{k+1} - b\end{bmatrix}. \tag{18}$$

Proof. Using the definition of $\mathbf{Q}$, we have

$$\mathbf{Q}\begin{bmatrix}x^{k+1}-x^k\\y^{k+1}-y^k\end{bmatrix} = \begin{bmatrix}\frac{1}{\eta}(x^{k+1}-x^k)\\ \frac{1}{\theta}(y^{k+1}-y^k) - \frac{\eta}{1+\eta\alpha}\mathbf{K}\mathbf{K}^T(y^{k+1}-y^k)\end{bmatrix}.$$

From line 7 of Algorithm 3 it follows that

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

and from line 6 of Algorithm 3

$$y^{k+1}-y^k=\theta(\mathbf{K}x^{k+\frac{1}{2}}-b).$$

Hence,

$$\mathbf{Q}\begin{bmatrix}x^{k+1}-x^k\\ y^{k+1}-y^k\end{bmatrix}=\begin{bmatrix}\alpha(x_g^k-x^{k+1})-(\nabla F(x_g^k)+\mathbf{K}^T y^{k+1})\\ (\mathbf{K}x^{k+\frac{1}{2}}-b)-\frac{\eta}{1+\eta\alpha}\mathbf{K}\mathbf{K}^T(y^{k+1}-y^k)\end{bmatrix}.$$

From lines 5 and 7 of Algorithm 3,

$$x^{k+1}-x^{k+\frac{1}{2}}=\frac{-\eta}{1+\eta\alpha}\mathbf{K}^T(y^{k+1}-y^k), \tag{19}$$

therefore,

$$(\mathbf{K}x^{k+\frac{1}{2}}-b)-\frac{\eta}{1+\eta\alpha}\mathbf{K}\mathbf{K}^T(y^{k+1}-y^k)=\mathbf{K}\left(x^{k+\frac{1}{2}}-\frac{\eta}{1+\eta\alpha}\mathbf{K}^T(y^{k+1}-y^k)\right)-b=\mathbf{K}x^{k+1}-b.$$

Finally,

$$\mathbf{Q}\begin{bmatrix}x^{k+1}-x^k\\ y^{k+1}-y^k\end{bmatrix}=\begin{bmatrix}\alpha(x_g^k-x^{k+1})-(\nabla F(x_g^k)+\mathbf{K}^T y^{k+1})\\ \mathbf{K}x^{k+1}-b\end{bmatrix}.$$

□

We now start the proof of Proposition 1.

Lemma 4. *Suppose that α satisfies $0\le\alpha\le\mu$. Then the following inequality holds:*

$$-\frac{1}{2\eta}\|x^{k+1}-x^k\|^2\le-\frac{\eta}{4}\|\mathbf{K}^T y^{k+1}-\mathbf{K}^T y^\star\|^2+\eta\alpha^2\|x^{k+1}-x^\star\|^2+2\eta L\mathrm{D}_F(x_g^k,x^\star). \tag{20}$$

Proof. From line 7 of Algorithm 3 and the optimality condition $\nabla F(x^\star)+\mathbf{K}^T y^\star=0$, it follows that

$$\begin{aligned}\|x^{k+1}-x^k\|^2&=\|\eta(\mathbf{K}^T y^{k+1}-\mathbf{K}^T y^\star)+\eta(\nabla F(x_g^k)-\nabla F(x^\star)-\alpha(x_g^k-x^\star))+\eta\alpha(x^{k+1}-x^\star)\|^2\\&\ge\frac{\eta^2}{2}\|\mathbf{K}^T y^{k+1}-\mathbf{K}^T y^\star\|^2-2\eta^2\alpha^2\|x^{k+1}-x^\star\|^2\\&\quad-2\eta^2\|\nabla F(x_g^k)-\nabla F(x^\star)-\alpha(x_g^k-x^\star)\|^2,\end{aligned}$$

where we used $\|a+b+c\|^2\ge0.5\|a\|^2-2\|b\|^2-2\|c\|^2$. Let $\bar F(x)\coloneqq F(x)-\frac{\alpha}{2}\|x\|^2$. The function $\bar F$ is a convex and $(L-\alpha)$-smooth function, therefore $\|\nabla\bar F(x)-\nabla\bar F(x')\|^2\le2(L-\alpha)\mathrm{D}_{\bar F}(x,x')$. Therefore, we can lower bound the last term and get

$$\begin{aligned}\|x^{k+1}-x^k\|^2&\ge\frac{\eta^2}{2}\|\mathbf{K}^T y^{k+1}-\mathbf{K}^T y^\star\|^2-2\eta^2\alpha^2\|x^{k+1}-x^\star\|^2-4\eta^2(L-\alpha)\mathrm{D}_{\bar F}(x_g^k,x^\star)\\&\ge\frac{\eta^2}{2}\|\mathbf{K}^T y^{k+1}-\mathbf{K}^T y^\star\|^2-2\eta^2\alpha^2\|x^{k+1}-x^\star\|^2-4\eta^2L\mathrm{D}_F(x_g^k,x^\star).\end{aligned}$$

Rearranging and dividing by 2η concludes the proof. □

Our last lemma states the linear convergence of a Lyapunov function to zero.

Lemma 5. *Consider $\lambda_1\ge\lambda_{\max}(\mathbf{W})$ and $\lambda_2\le\lambda_{\min}^+(\mathbf{W})$.*

Let parameter η be defined by

$$\eta=\frac{1}{4\tau L}. \tag{21}$$

Let us set the parameter θ as

$$\theta = \frac{1}{\eta\lambda_1}. \tag{22}$$

Let us set the parameter α as

$$\alpha = \mu. \tag{23}$$

Let us set the parameter τ as

$$\tau = \min\left\{1, \frac{1}{2}\sqrt{\frac{\mu}{L}\frac{\lambda_1}{\lambda_2}}\right\}. \tag{24}$$

Let Ψ^k be the following Lyapunov function:

$$\Psi^k = \left\|\begin{bmatrix} x^k - x^\star \\ y^k - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star), \tag{25}$$

Then the following inequality holds:

$$\Psi^{k+1} \leq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_2}{\lambda_1}}, \frac{\lambda_2}{\lambda_1}\right\}\right)^{-1}\Psi^k. \tag{26}$$

Proof.

$$\left\|\begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 = \left\|\begin{bmatrix} x^k - x^\star \\ y^k - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \left\|\begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix}\right\|_{\mathbf{Q}}^2 + 2\left\langle\begin{bmatrix} x^{k+1} - x^k \\ y^{k+1} - y^k \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\rangle_{\mathbf{Q}}.$$

Note that stepsize η defined by (21) and stepsize θ defined by (22) satisfy (16), hence inequality (17) holds. Using (17) and (18) we get

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^k - x^\star \\ y^k - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 \\
&\quad + 2\left\langle\begin{bmatrix} \alpha(x_g^k - x^{k+1}) - (\nabla F(x_g^k) + \mathbf{K}^T y^{k+1}) \\ \mathbf{K}x^{k+1} - b \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\rangle \\
&= \left\|\begin{bmatrix} x^k - x^\star \\ y^k - y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{1}{\eta}\|x^{k+1} - x^k\|^2 + 2\alpha\langle x_g^k - x^{k+1}, x^{k+1} - x^\star\rangle \\
&\quad - 2\left\langle\begin{bmatrix} \nabla F(x_g^k) + \mathbf{K}^T y^{k+1} \\ -\mathbf{K}x^{k+1} + b \end{bmatrix} - \begin{bmatrix} \nabla F(x^\star) + \mathbf{K}^T y^\star \\ -\mathbf{K}x^\star + b \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\rangle,
\end{aligned}$$

using the primal–dual optimality conditions (4). Rewrite

$$\begin{bmatrix} \nabla F(x_g^k) + \mathbf{K}^T y^{k+1} \\ -\mathbf{K}x^{k+1} + b \end{bmatrix} - \begin{bmatrix} \nabla F(x^\star) + \mathbf{K}^T y^\star \\ -\mathbf{K}x^\star + b \end{bmatrix} = \begin{bmatrix} \nabla F(x_g^k) - \nabla F(x^\star) \\ b - b \end{bmatrix} + \begin{bmatrix} 0 & \mathbf{K}^T \\ -\mathbf{K} & 0 \end{bmatrix}\begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}.$$

Using $\langle Az, z\rangle = 0$ for any skew-symmetric matrix A, we obtain

$$\begin{aligned}
-2\left\langle\begin{bmatrix} \nabla F(x_g^k) + \mathbf{K}^T y^{k+1} \\ -\mathbf{K}x^{k+1} + b \end{bmatrix} - \begin{bmatrix} \nabla F(x^\star) + \mathbf{K}^T y^\star \\ -\mathbf{K}x^\star + b \end{bmatrix}, \begin{bmatrix} x^{k+1} - x^\star \\ y^{k+1} - y^\star \end{bmatrix}\right\rangle \\
= -2\langle \nabla F(x_g^k) - \nabla F(x^\star), x^{k+1} - x^\star\rangle.
\end{aligned}$$

Hence,

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

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

$$\begin{aligned}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 &\le \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{1}{\eta}\|x^{k+1}-x^k\|^2 - 2\alpha\|x^{k+1}-x^\star\|^2\\ &\quad + \alpha\|x_g^k - x^\star\|^2 + \alpha\|x^{k+1}-x^\star\|^2 - 2\langle \nabla F(x_g^k) - \nabla F(x^\star), x^{k+1}-x^\star\rangle\\ &= \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{1}{\eta}\|x^{k+1}-x^k\|^2 - \alpha\|x^{k+1}-x^\star\|^2 + \alpha\|x_g^k - x^\star\|^2\\ &\quad - 2\langle \nabla F(x_g^k) - \nabla F(x^\star), x^{k+1}-x^\star\rangle.\end{aligned}$$

Using line 4 of Algorithm 3, we have $x^k - x^\star = (x_g^k - x^\star) + \frac{1-\tau}{\tau}(x_g^k - x_f^k)$ and using line 8, $x^{k+1} - x^k = \frac{2-\tau}{2\tau}(x_f^{k+1} - x_g^k)$. Therefore, decomposing $x^{k+1} - x^\star = (x^{k+1} - x^k) + (x^k - x^\star)$,

$$\begin{aligned}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 &\le \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \alpha\|x_g^k - x^\star\|^2 - \frac{1}{2\eta}\|x^{k+1}-x^k\|^2\\ &\quad - \frac{2-\tau}{\tau}\left(\langle \nabla F(x_g^k) - \nabla F(x^\star), x_f^{k+1} - x_g^k\rangle + \frac{1}{2\eta}\frac{(2-\tau)}{4\tau}\|x_f^{k+1} - x_g^k\|^2\right)\\ &\quad - 2\langle \nabla F(x_g^k) - \nabla F(x^\star), x_g^k - x^\star\rangle + \frac{2(1-\tau)}{\tau}\langle \nabla F(x_g^k) - \nabla F(x^\star), x_f^k - x_g^k\rangle.\end{aligned}$$

Since η defined by (21) satisfies $\eta \le \frac{2-\tau}{4\tau L}$, we get

$$\begin{aligned}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 &\le \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \alpha\|x_g^k - x^\star\|^2 - \frac{1}{2\eta}\|x^{k+1}-x^k\|^2\\ &\quad - \frac{2-\tau}{\tau}\left(\langle \nabla F(x_g^k) - \nabla F(x^\star), x_f^{k+1} - x_g^k\rangle + \frac{L}{2}\|x_f^{k+1} - x_g^k\|^2\right)\\ &\quad - 2\langle \nabla F(x_g^k) - \nabla F(x^\star), x_g^k - x^\star\rangle\\ &\quad + \frac{2(1-\tau)}{\tau}\langle \nabla F(x_g^k) - \nabla F(x^\star), x_f^k - x_g^k\rangle.\end{aligned}$$

Using μ-strong convexity and L-smoothness of F we get

$$\begin{aligned}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 &\le \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \alpha\|x_g^k - x^\star\|^2 - \frac{1}{2\eta}\|x^{k+1}-x^k\|^2\\ &\quad - \frac{2-\tau}{\tau}\left(\mathrm{D}_F(x_f^{k+1}, x^\star) - \mathrm{D}_F(x_g^k, x^\star)\right)\\ &\quad + \frac{2(1-\tau)}{\tau}\left(\mathrm{D}_F(x_f^k, x^\star) - \mathrm{D}_F(x_g^k, x^\star)\right)\\ &\quad - 2\left(\mathrm{D}_F(x_g^k, x^\star) + \frac{\mu}{2}\|x_g^k - x^\star\|^2\right)\\ &= \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star)\\ &\quad - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star)\\ &\quad + (\alpha-\mu)\|x_g^k - x^\star\|^2 - \frac{1}{2\eta}\|x^{k+1}-x^k\|^2 - \mathrm{D}_F(x_g^k, x^\star).\end{aligned}$$

Now, we define $\delta = \min\left\{1, \frac{1}{2\eta L}\right\}$. Since α defined by (23) satisfies conditions of Lemma 4, we can use (20) and

get

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \\
&\quad - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star) \\
&\quad + (\alpha-\mu)\|x_g^k - x^\star\|^2 - \frac{\delta}{2\eta}\|x^{k+1}-x^k\|^2 - \mathrm{D}_F(x_g^k, x^\star) \\
&\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \\
&\quad - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star) - \frac{\eta\delta}{4}\|\mathbf{K}^T y^{k+1} - \mathbf{K}^T y^\star\|^2 + \eta\alpha^2\delta\|x^{k+1}-x^\star\|^2 \\
&\quad + 2\eta L\delta \mathrm{D}_f(x_g^k, x^\star) + (\alpha-\mu)\|x_g^k - x^\star\|^2 - \mathrm{D}_F(x_g^k, x^\star) \\
&\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \alpha\|x^{k+1}-x^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \\
&\quad - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star) - \frac{\eta\delta}{4}\|\mathbf{K}^T y^{k+1} - \mathbf{K}^T y^\star\|^2 + \frac{\alpha^2}{2L}\|x^{k+1}-x^\star\|^2 \\
&\quad + (\alpha-\mu)\|x_g^k - x^\star\|^2 \\
&= \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \left(\alpha - \frac{\alpha^2}{2L}\right)\|x^{k+1}-x^\star\|^2 - \frac{\eta\delta}{4}\|\mathbf{K}^T y^{k+1} - \mathbf{K}^T y^\star\|^2 \\
&\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star) + (\alpha-\mu)\|x_g^k - x^\star\|^2.
\end{aligned}$$

Using the parameter $\alpha = \mu$ defined in (23) and using $\mu \leq L$, we get

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{\mu}{2}\|x^{k+1}-x^\star\|^2 - \frac{\eta\delta}{4}\|\mathbf{K}^T y^{k+1} - \mathbf{K}^T y^\star\|^2 \\
&\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star).
\end{aligned}$$

For every $y \in \mathrm{range}(\mathbf{K})$, $\lambda_2\|y\|^2 \leq \lambda_{\min}^+(\mathbf{W})\|y\|^2 \leq \|\mathbf{K}^T y\|^2$. Using line 6 of Algorithm 3, one can check by induction that $y^k \in \mathrm{range}(\mathbf{K})$ for every $k \geq 0$. Moreover, using (4), $y^\star \in \mathrm{range}(\mathbf{K})$. Therefore,

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \frac{\mu}{2}\|x^{k+1}-x^\star\|^2 - \frac{\eta\delta\lambda_2}{4}\|y^{k+1} - y^\star\|^2 \\
&\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star).
\end{aligned}$$

Using (17) we get

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \min\left\{\frac{\eta\mu}{2}, \frac{\eta\theta\delta\lambda_2}{4}\right\}\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 \\
&\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star).
\end{aligned}$$

Using the parameter θ defined in (22) and the definition of δ, we get

$$\begin{aligned}
\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix} x^{k}-x^\star \\ y^{k}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 - \min\left\{\frac{\eta\mu}{2}, \frac{\lambda_2}{4\lambda_1}, \frac{\lambda_2}{8\eta L\lambda_1}\right\}\left\|\begin{bmatrix} x^{k+1}-x^\star \\ y^{k+1}-y^\star \end{bmatrix}\right\|_{\mathbf{Q}}^2 \\
&\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star).
\end{aligned}$$

Plugging the parameter η defined in (21), we get

$$\begin{aligned}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 &\leq \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \min\left\{\frac{\mu}{8\tau L}, \frac{\lambda_2}{4\lambda_1}, \frac{\tau\lambda_2}{2\lambda_1}\right\}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2\\ &\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \frac{2-\tau}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star)\\ &\leq \left\|\begin{bmatrix}x^{k}-x^\star\\ y^{k}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2 - \min\left\{\frac{\mu}{8\tau L}, \frac{\lambda_2}{4\lambda_1}, \frac{\tau\lambda_2}{2\lambda_1}\right\}\left\|\begin{bmatrix}x^{k+1}-x^\star\\ y^{k+1}-y^\star\end{bmatrix}\right\|_{\mathbf{Q}}^2\\ &\quad + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) - \left(1+\frac{\tau}{2}\right)\frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^{k+1}, x^\star).\end{aligned}$$

After rearranging the terms and using the definition of Ψ^k in (25), we get

$$\Psi^k \geq \left(1 + \min\left\{\frac{\tau}{2}, \frac{\mu}{8\tau L}, \frac{\lambda_2}{4\lambda_1}, \frac{\tau\lambda_2}{2\lambda_1}\right\}\right)\Psi^{k+1}.$$

Plugging the parameter τ defined in (24), we get

$$\Psi^k \geq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_2}{\lambda_1}}, \frac{\lambda_2}{\lambda_1}\right\}\right)\Psi^{k+1}.$$

□

8.2 End of the Proof of Proposition 1

The conditions of Lemma 5 are satisfied, hence the following inequality holds for every $k \geq 0$:

$$\Psi^{k+1} \leq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_2}{\lambda_1}}, \frac{\lambda_2}{\lambda_1}\right\}\right)^{-1}\Psi^{k}.$$

After telescoping we get

$$\Psi^{k} \leq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_2}{\lambda_1}}, \frac{\lambda_2}{\lambda_1}\right\}\right)^{-k}\Psi^{0}.$$

Inequality (17) implies $\Psi^0 \leq C$, where $C \coloneqq \frac{1}{\eta}\left\|x^0 - x^\star\right\|^2 + \frac{1}{\theta}\|y^0 - y^\star\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^0, x^\star)$. Hence, we obtain

$$\Psi^{k} \leq \left(1 + \frac{1}{4}\min\left\{\sqrt{\frac{\mu}{L}\frac{\lambda_2}{\lambda_1}}, \frac{\lambda_2}{\lambda_1}\right\}\right)^{-k} C. \tag{27}$$

It remains to lower bound Ψ^k using (17) one more time:

$$\frac{1}{\eta}\left\|x^k - x^\star\right\|^2 + \frac{\eta\alpha}{\theta(1+\eta\alpha)}\left\|y^k - y^\star\right\|^2 + \frac{2(1-\tau)}{\tau}\mathrm{D}_F(x_f^k, x^\star) \leq \Psi^k.$$

Combining with (27) gives the result. □

9 Proof of Theorem 2

9.1 Proof of Equation (15)

The vector $z^N = \mathrm{Chebyshev}(x, \mathbf{K}, b, N)$ is the N^{th} iterate of the Chebyshev iteration, which amounts to applying the Chebyshev polynomials $\widetilde{\boldsymbol{T}}_N$ to the residual $\mathbf{K}^T(\mathbf{K}z^0 - b)$, to make it converge to zero. So, z^N satisfies

$$\mathbf{K}^T(\mathbf{K}z^N - b) = \widetilde{\boldsymbol{T}}_N(\mathbf{W})\big(\mathbf{K}^T(\mathbf{K}z^0 - b)\big), \tag{28}$$

so that $\|\mathbf{K}z^N - b\|$ converges linearly to zero when $N \to +\infty$.

Since $\widetilde{\boldsymbol{T}}_N(0) = 1$, there exists a polynomial $\widetilde{\boldsymbol{R}}_N$ such that $\widetilde{\boldsymbol{T}}_N(X) = 1 + X\widetilde{\boldsymbol{R}}_N(X)$. Therefore,

$$\mathbf{K}^T(\mathbf{K}z^n - b) = \big(\mathbf{K}^T(\mathbf{K}z^0 - b)\big) + \mathbf{W}\widetilde{\boldsymbol{R}}_N(\mathbf{W})\big(\mathbf{K}^T(\mathbf{K}z^0 - b)\big);$$

that is,

$$\mathbf{W}z^N = \mathbf{W}\left(z^0 + \widetilde{\boldsymbol{R}}_N(\mathbf{W})\big(\mathbf{K}^T(\mathbf{K}z^0 - b)\big)\right).$$

One can check by induction that $z^N \in z^0 + \mathrm{range}(\mathbf{W})$. Using $\mathbf{I}_{\mathsf{X}} + \mathbf{W}\widetilde{\boldsymbol{R}}_N(\mathbf{W}) = \widetilde{\boldsymbol{T}}_N(\mathbf{W})$,

$$\begin{aligned} z^n &= z^0 + \widetilde{\boldsymbol{R}}_N(\mathbf{W})\big(\mathbf{K}^T(\mathbf{K}z^0 - b)\big) \\ &= z^0 + \mathbf{W}\widetilde{\boldsymbol{R}}_N(\mathbf{W})z^0 - \widetilde{\boldsymbol{R}}_N(\mathbf{W})\mathbf{K}^T b \\ &= \widetilde{\boldsymbol{T}}_N(\mathbf{W})z^0 - \widetilde{\boldsymbol{R}}_N(\mathbf{W})\mathbf{K}^T b \\ &= \widetilde{\boldsymbol{T}}_N(\mathbf{W})z^0 - \widetilde{\boldsymbol{R}}_N(\mathbf{W})\mathbf{W}x^\star \\ &= \widetilde{\boldsymbol{T}}_N(\mathbf{W})z^0 - \widetilde{\boldsymbol{T}}_N(\mathbf{W})x^\star + x^\star. \end{aligned}$$

Finally, for every $z^0 \in \mathsf{X}$,

$$\mathbf{P}(\mathbf{W})z^0 - \mathbf{P}(\mathbf{W})x^\star = z^0 - \widetilde{\boldsymbol{T}}_N(\mathbf{W})z^0 - x^\star + \widetilde{\boldsymbol{T}}_N(\mathbf{W})x^\star = z^0 - z^N.$$

9.2 End of proof of Theorem 2

In Sections 6.3.2 and 9.1, we proved that Algorithm 3 applied to the equivalent Problem (8) is equivalent to our main Algorithm 1. Therefore, we can prove our main Theorem 2 by applying Proposition 1 to Problem (8). Indeed, the proof of Theorem 2 is a direct application of Proposition 1 to Problem (8), using that $N \geq \sqrt{\chi}$ implies $\lambda_{\max}(\mathbf{P}(\mathbf{W})) \leq 19/15, \lambda^+_{\min}(\mathbf{P}(\mathbf{W})) \geq 11/15$ and $\chi(\mathbf{P}(\mathbf{W})) \leq 19/11$, see Section 6.3.1. □