Adaptive Algorithms for Robust Phase Retrieval

A Preprint

Zhong Zheng[1], Necdet Serhat Aybat[2], Shiqian Ma[3], and Lingzhou Xue[1]

[1]Department of Statistics, Pennsylvania State University
[2]Department of Industrial and Manufacturing Engineering, Pennsylvania State University
[3]Department of Computational Applied Mathematics and Operations Research, Rice University

September, 2024

Abstract

This paper considers the robust phase retrieval, which can be cast as a nonsmooth and nonconvex optimization problem. We propose two first-order algorithms with adaptive step sizes: the subgradient algorithm (`AdaSubGrad`) and the inexact proximal linear algorithm (`AdaIPL`). Our contribution lies in the novel design of adaptive step sizes based on quantiles of the absolute residuals. Local linear convergences of both algorithms are analyzed under different regimes for the hyper-parameters. Numerical experiments on synthetic datasets and image recovery also demonstrate that our methods are competitive against the existing methods in the literature utilizing predetermined (possibly impractical) step sizes, such as the subgradient methods and the inexact proximal linear method.

Keywords

Adaptive Steps, Subgradient Method, Proximal Linear Algorithm, Robust Phase Retrieval.

1 Introduction

Phase retrieval (PR) aims to recover a signal from intensity-or magnitude-based measurements. It finds various applications in different fields, including X-ray crystallography [1], optics [2], diffraction and array imaging [3], astronomy [4], and microscopy [5]. Mathematically, PR tries to find the true signal vectors $x_\star$ or $-x_\star$ in $\mathbb{R}^n$ from a set of magnitude measurements:

$$b_i = \langle a_i, x_\star \rangle^2, \text{ for } i = 1, 2, \ldots, m, \tag{1}$$

where $a_i \in \mathbb{R}^n$ and $b_i \geq 0, i = 1, 2, \ldots, m$. Directly solving the equations (1) is an NP-hard problem [6], and algorithms based on different designs of objective functions have been well studied in the literature, including Wirtinger flow [7], truncated Wirtinger flow [8], truncated amplitude flow [9], reshaped Wirtinger flow [10], etc.

In this paper, we focus on the robust phase retrieval (RPR) problem [11], which considers the case where b_i might contain infrequent but arbitrary noise due to measurement errors, i.e.,

$$b_i = \begin{cases} \langle a_i, x_\star \rangle^2, & i \in \mathcal{I}_1, \\ \xi_i, & i \in \mathcal{I}_2, \end{cases} \tag{2}$$

in which $\mathcal{I}_1 \bigcup \mathcal{I}_2 = \{1, 2 \ldots, m\}$, $\mathcal{I}_1 \cap \mathcal{I}_2 = \emptyset$, and ξ_i denotes the noise that only exists in measurements in $\mathcal{I}_2$, and it follows an arbitrary distribution. We will call such noisy measurements as *corrupted measurements* (this is the name used by [11]). Our objective is to recover $x_\star$ or $-x_\star$ using $\{(a_i, b_i)\}_{i=1}^m$ without knowing either $\mathcal{I}_1$ or $\mathcal{I}_2$.

Definition 1.1. $x_\epsilon \in \mathbb{R}^n$ is an ϵ-optimal solution if $\Delta(x_\epsilon) \leq \epsilon$, where

$$\Delta(x) \triangleq \min\{\|x - x_\star\|_2, \|x + x_\star\|_2\}. \tag{3}$$

[11] proposed to formulate RPR as an optimization problem employing the nonsmooth ℓ_1-loss:

$$\min_{x\in\mathbb{R}^n} F(x) \triangleq \frac{1}{m}\sum_{i=1}^{m}\left|\langle a_i, x\rangle^2 - b_i\right| = h(c(x)), \tag{4}$$

where $h(z) \triangleq \frac{1}{m}\|z\|_1$ and $c(x) \triangleq |Ax|^2 - b$ is a smooth map in which $|\cdot|^2$ operates element-wise on $Ax = [\langle a_1, x\rangle, \ldots, \langle a_m, x\rangle]^\top$ and $b = [b_1, \ldots, b_m]^\top$. In the rest of the paper, $A \in \mathbb{R}^{m\times n}$ such that $A = [a_1, a_2, \ldots, a_m]^\top$ denotes the measurement matrix.

In [11], the authors have shown that F is weakly convex; moreover, under some statistical assumptions on (2), F satisfies some sharpness condition with high probability – here, sharpness implies that minimizers of $F(\cdot)$ coincide with the true signal vectors. In [11], it is shown that (4) possesses better recoverability than the median truncated Wirtinger flow algorithm [12] based on the ℓ_2-loss.

1.1 Existing Algorithms and Challenges

In the literature, two kinds of algorithms were proposed for solving (4): *(i)* proximal-linear (PL)-type algorithms, and *(ii)* subgradient-type algorithms. First, we review proximal-linear-type algorithms. For any given $z, y \in \mathbb{R}^n$ and $t > 0$, let

$$F(z; y) \triangleq h(c(y) + J_c(y)(z - y)), \quad F_t(z; y) \triangleq F(z; y) + \frac{1}{2t}\|z - y\|_2^2, \tag{5}$$

where $J_c(\cdot) \in \mathbb{R}^{m\times n}$ denotes the Jacobian of $c(\cdot)$ and can be written explicitly as $J_c(y) = 2\text{diag}(Ay)A$. In one typical iteration of a PL-type algorithm, one *inexactly* solves a subproblem of the form:

$$x^{k+1} \approx \arg\min_{x\in\mathbb{R}^n}\ F_{t_k}(x; x^k), \tag{6}$$

where $t_k > 0$ is the chosen step size, and "≈" means that x^{k+1} is an "inexact" solution to the subproblem in (6). Using the fixed step sizes $t_k = L^{-1}$ for all $k \in \mathbb{N}$, with $L \triangleq 2\|A\|_2^2/m$, [11] has proposed the Proximal Linear (`PL`) algorithm, and assuming that the L-strongly convex subproblem in (6) is solved *exactly* for all $k \geq 0$, the local quadratic convergence rate of `PL` method is shown in terms of iteration counter k, i.e., in the number of exact minimizations of the form $\arg\min_x F_{t_k}(x; x^k)$. However, the total complexity for the `PL` method is unknown due to unclear cost of solving the subproblems in the form of (6) to *exact* optimality – it is worth emphasizing that closed-form solutions to subproblems are not available and in practice one cannot compute them exactly using iterative methods; therefore, this method is not practical. On the other hand, for the numerical experiments in [11], each subproblem in the form of (6) was inexactly solved by the proximal operator graph splitting (`POGS`) method [13], terminated as suggested in [13], i.e., when the primal and dual residuals satisfy a predetermined threshold – that said, the convergence analysis was not provided in [11] for this strategy.

To get better control over the cost of solving (6), under the same fixed step sizes, [14] has proposed the Inexact Proximal Linear (`IPL`) algorithm that solves (6) inexactly using one of the following *inexact* termination conditions:

$$F_{t_k}(x^{k+1}; x^k) - \min_{x\in\mathbb{R}^n}\ F_{t_k}(x; x^k) \leq \begin{cases} \rho_l\left(F(x^k) - F_{t_k}(x^{k+1}; x^k)\right) & \textbf{(LAC-exact)} \\ \frac{\rho_h}{2t_k}\|x^{k+1} - x^k\|_2^2 & \textbf{(HAC-exact)} \end{cases} \tag{7}$$

where $\rho_l > 0$, $\rho_h \in (0, 1/4)$ are given positive constants. Here, LAC is short for the low accuracy condition, and HAC is short for the high accuracy condition. Since $\min_x F_{t_k}(x; x^k)$ is not known in practice, to verify these conditions one needs to work with sufficient conditions for (7) obtained by replacing $\min_{x\in\mathbb{R}^n} F_{t_k}(x; x^k)$ with the dual function values of (6). In [14], (6) is solved through applying the Accelerated Proximal Gradient algorithm (`APG`) given in [15] to the dual problem of (6). In [14], it is proven that `IPL` can compute an ϵ-optimal solution for (4) within $\mathcal{O}(1/\epsilon)$ inner iterations in total for inexactly solving a sequence of subproblem in the form of (6), establishing the total complexity for the double-loop algorithm `IPL`. Numerical experiments in [14] empirically show that `IPL` enjoys better numerical performance in terms of CPU time compared to `PL` method. However, the efficiency of `IPL` is still unsatisfying due to the sublinear convergence rate; that said, there is a room for further improvement for `IPL`, and this is one of the objectives of this paper, where we improve the total complexity using diminishing step size strategy which is also practical.

Next, we review the existing subgradient-type algorithms [16, 17] which can handle (4) while avoiding (inexact) subproblem solves. Polyak subgradient descent (`PSubGrad`) is investigated in [17]:

$$x^{k+1} = x^k - \left(F(x^k) - F(x_\star)\right) \xi_k / \|\xi_k\|_2^2, \quad \xi_k \in \partial F(x^k), \quad k \in \mathbb{N}.$$

Subgradient algorithms with geometrically decaying step sizes (`GSubGrad`) are proposed in [16]:

$$x^{k+1} = x^k - \lambda_k \xi_k / \|\xi_k\|_2, \quad \xi_k \in \partial F(x^k), \quad \lambda_k = \lambda_0 q^k, \quad k \in \mathbb{N}, \tag{8}$$

where $\lambda_0 > 0$ and $q \in (0, 1)$ are constant algorithm parameters. For both algorithms, local linear convergence has been shown. `PSubGrad` only works for noiseless robust phase retrieval (1) as it relies on the value of $F(x_\star)$, and knowing a lower bound is not sufficient for establishing convergence for `PSubGrad`. Moreover, as discussed in [17], using fixed step sizes for subgradient algorithms only leads to suboptimal solutions. On the other hand, for `GSubGrad`, there is still no practical guidance on properly choosing the hyper-parameters (λ_0 and q). More precisely, for any fixed $\gamma \in (0, 1)$, the results in [17] require setting $\lambda_0 = \gamma \lambda_s^2 / (L B_\xi)$ and $q = \sqrt{1 - (1 - \gamma)(\lambda_s / B_\xi)^2}$, explicitly depending on the *unknown* quantity λ_s (see Assumption 2.4), where $B_\xi \triangleq \sup\{\|\xi\|_2 \colon \ \xi \in \partial F(x), \ x \in \mathcal{T}_\gamma\}$ and $\mathcal{T}_\gamma = \{x \in \mathbb{R}^n : \Delta(x) \leq \gamma \lambda_s / L\}$. In summary, for (4) with measurements as in (2), no *practical* hyper-parameter choice is known for [16, 17] with theoretical guarantees, and under improper step size choices, these algorithms might not perform well or even possibly fail to converge [16, 17, 14].

Stochastic algorithms are also studied in [18, 19, 20, 21, 22] where the proximal-linear type and the subgradient-type methods are unified. In addition, [17] and [23] also analyze the nonconvex landscape. These type of methods and their analysis are beyond the scope of our paper.

1.2 Proposed Algorithms: `AdaSubGrad` and `AdaIPL`

In this paper, we propose to incorporate an adaptive step size strategy within subgradient and inexact-proximal linear algorithm frameworks for improving their convergence behavior both in theory and practice. Next, we give some definitions. Throughout we set $L \triangleq 2\|A\|_2^2 / m$. For a $\tilde{p} \in (0, 1)$ such that $m\tilde{p} \in \mathbb{N}_+$, let

$$r_i(\cdot) \triangleq |\langle a_i, \cdot \rangle^2 - b_i|, \quad \forall\, i \in [m]; \quad r^{\tilde{p}}(\cdot) \triangleq \text{ the } \tilde{p}\text{-th quantile of } \{r_i(\cdot)\}_{i=1}^m, \tag{9}$$

i.e., $r^{\tilde{p}}(x)$ denotes the $(m\tilde{p})$-th order statistics of absolute residuals $\{r_i(x)\}_{i=1}^m$ at any given x. Quantile operator is known for its robustness to outliers; hence, it helps us design a step size strategy that would be robust to the corrupted measurements for (4), i.e., when the true measurement vector is corrupted by a sparse noise vector with non-zero entries having arbitrarily large magnitudes.

1.2.1 `AdaSubGrad`

We propose a subgradient method with adaptive step sizes (`AdaSubGrad`). The step size α_k at iteration $k \in \mathbb{N}$ is chosen as follows:

$$\alpha_k = G\, r^{\tilde{p}}(x^k), \quad \forall\, k \in \mathbb{N}, \tag{10}$$

where $G > 0$ is an algorithm parameter. `AdaSubGrad` iterates are computed as

$$x^{k+1} = x^k - \alpha_k \xi^k / \|\xi^k\|_2^2, \quad \xi^k \in \partial F(x^k), \quad \forall\, k \in \mathbb{N}. \tag{11}$$

According to [24, Theorem 10.6, Corollary 10.9], the subdifferential of weakly convex function F has the form $\partial F(x) \triangleq [J_c(x)]^\top \partial h(c(x))$ for $x \in \mathbb{R}^n$, where $J_c(x) \in \mathbb{R}^{m \times n}$ denotes the Jacobian of c at x. One can calculate a subgradient $\xi_k \in \partial F(x^k)$ as $\xi^k = \frac{2}{m} \sum_{i=1}^m \langle a_i, x^k \rangle \operatorname{sign}\Big(\langle a_i, x^k \rangle^2 - b_i \Big) a_i$ for any $k \in \mathbb{N}$. Our proposed method `AdaSubGrad` is formally stated in Algorithm 1.

Algorithm 1 Subgradient Algorithm with Adaptive Step Sizes (`AdaSubGrad`)

Input: Initial point $x^0 \in \mathbb{R}^n$, parameter $G > 0$, percentile $\tilde{p} \in (0, 1)$ such that $m\tilde{p} \in \mathbb{N}_+$.
for $k = 0, 1, \ldots,$ **do**
    Update x^{k+1} using (11) with α_k given in (10).
end for

Note that in our choice of the step size α_k, we adopt the quantile design, i.e., $r^{\tilde{p}}(x^k)$; the main motivation behind our choice is that under a fairly reasonable data generating process discussed in Section 3, one can show that $\alpha_k = \Theta(F(x^k) - F(x_\star))$ for all $k \in \mathbb{N}$ with *high probability*. Therefore, `AdaSubGrad` will exhibit a similar convergence behavior with the Polyak subgradient algorithm. More precisely, we prove that for sufficiently small $G > 0$, `AdaSubGrad` enjoys a local linear convergence.

1.2.2 AdaIPL

Second, we introduce the inexact proximal linear algorithm with adaptive step sizes (AdaIPL). Given some positive constant $G > 0$, let

$$t_k \triangleq \min\{L^{-1}, G\, r^{\tilde{p}}(x^k)\}. \tag{12}$$

In AdaIPL, iterates are computed by *inexactly* solving (6) with t_k chosen as in (12) such that for all $k \in \mathbb{N}$ either **(LAC-exact)** or **(HAC-exact)** given in (7) holds. Since $\min_{x\in\mathbb{R}^n}\ F_{t_k}(x; x^k)$ appearing in (7) is not available in practice, in AdaIPL we replace (7) with a practical one as described next. Given $t_k \leq \frac{1}{L} = \frac{m}{2\|A\|_2^2}$, (6) is convex and can be equivalently written as

$$\min_{z\in\mathbb{R}^n} H_k(z) \triangleq \frac{1}{2t_k}\|z\|_2^2 + \|B_k z - d_k\|_1, \tag{13}$$

after the change of variables: $z \triangleq x - x^k$ and setting $B_k \triangleq \frac{2}{m}\text{diag}(Ax^k)A$ and $d_k \triangleq \frac{1}{m}\left(b - |Ax^k|^2\right)$. The problem in (13) has the following min-max and dual forms:

$$\min_{z\in\mathbb{R}^n} \max_{\lambda\in\mathbb{R}^m:\ \|\lambda\|_\infty \leq 1} H_k(z,\lambda) \triangleq \frac{1}{2t_k}\|z\|_2^2 + \lambda^\top(B_k z - d_k), \tag{14a}$$

$$\max_{\lambda\in\mathbb{R}^m:\ \|\lambda\|_\infty \leq 1} D_k(\lambda) \triangleq -\frac{t_k}{2}\left\|\lambda^\top B_k\right\|_2^2 - \lambda^\top d_k. \tag{14b}$$

Let $z^k(\lambda) \triangleq -t_k B_k^\top \lambda$ and $\lambda^k(z) \triangleq \text{sign}\left(B_k z - d_k\right)$. Using $z^k \triangleq x^{k+1} - x^k$, we can rewrite (7) as

$$H_k(z^k) - \min_{z\in\mathbb{R}^n} H_k(z) \leq \begin{cases} \rho_l\left(H_k(0) - H_k(z^k)\right), & \textbf{(LAC-exact)} \\ \frac{\rho_h}{2t_k}\|z^k\|_2^2, & \textbf{(HAC-exact)}, \end{cases} \tag{15}$$

where $\rho_l > 0$, $\rho_h \in (0, 1/4)$ are given positive constants. As $\min_{z\in\mathbb{R}^n} H_k(z)$ may not be easily available, we provide *sufficient* conditions for (**LAC-exact**) and (**HAC-exact**) conditions in (15) that can be checked in practice. Due to weak duality, we have

$$H_k(z^k) - \min_{z\in\mathbb{R}^n} H_k(z) \leq H_k(z^k) - D_k(\lambda), \quad \forall\, \lambda \in \mathbb{R}^m:\ \|\lambda\|_\infty \leq 1.$$

Consider a generic solver such that when initialized from an arbitrary (z_0^k, λ_0^k), it generates a primal-dual iterate sequence $\{(z_j^k, \lambda_j^k)\}_{j=0}^\infty \subset \mathbb{R}^n \times \mathbb{R}^m$ for the k-th subproblem satisfying $\sup_{j\in\mathbb{N}}\|\lambda_k^j\|_\infty \leq 1$, and after j_k iterations it computes a primal-dual pair $(z^k, \lambda^k) = (z_{j_k}^k, \lambda_{j_k}^k)$ such that

$$H_k(z^k) - D_k(\lambda^k) \leq \begin{cases} \rho_l\left(H_k(0) - H_k(z^k)\right), & \textbf{(LAC)} \\ \frac{\rho_h}{2t_k}\|z^k\|_2^2, & \textbf{(HAC)}. \end{cases} \tag{16}$$

Since (16) is a sufficient condition on (15), for AdaIPL we can can adopt the practical condition given in (16) instead of the condition in (15).

Lemma 1.2. (**LAC**) *and* (**HAC**) *imply* (**LAC-exact**) *and* (**HAC-exact**)*, respectively.*

Algorithm 2 Inexact Proximal Linear Algorithm with Adaptive Step Sizes (AdaIPL)

Input: Initial point $x^0 \in \mathbb{R}^n$, $\rho_l > 0$ or $\rho_h \in (0, 1/4)$, percentile $\tilde{p} \in (0, 1)$ s.t. $m\tilde{p} \in \mathbb{N}_+$.
Set Cond to either **(LAC)** or to **(HAC)** in (16)
for $k = 0, 1, \ldots,$ **do**
    Compute t_k as in (12)
    Compute x^{k+1} by inexactly solving (6) such that Cond holds
end for

A pseudocode for AdaIPL is given in Algorithm 2. It is only a prototype algorithm because the method for inexactly solving (6) is not fixed at this point. We will discuss this issue later in Section 6.2. For AdaIPL, the step size t_k uses the quantile design such that under the data generation process introduced in Section 3, $t_k = \Theta(\Delta(x^k))$ for all $k \geq 0$ with high probability, where $\Delta(\cdot)$ is defined in (3). Analysis in Section 6 will show that under adaptive step sizes, we can reach a better balance between main iteration complexity and subproblem iteration complexity compared to IPL with fixed step sizes proposed in [14]. We will show that for any choice of algorithm parameter $G > 0$, AdaIPL enjoys local linear convergence in terms of total iteration complexity associated with inexactly solving all the subproblems in the form of (6).

Algorithm	Ideal Complexity	Step sizes	Tuning
PL[11]	Unknown	Fixed	N/A
IPL-LAC[14]	$\mathcal{O}(C_S^2\kappa_0^2\|x_\star\|_2/\epsilon)$	Fixed	N/A
IPL-HAC [14]	$\mathcal{O}(C_S^2\kappa_0^3\|x_\star\|_2/\epsilon)$	Fixed	N/A
PSubGrad[17]	$\mathcal{O}(\kappa_0^2\log\frac{1}{\epsilon})$	Adaptive	Hard
GSubGrad[16]	$\mathcal{O}(\kappa_0^2\log\frac{1}{\epsilon})$	Predetermined	Hard
AdaSubGrad **(this work)**	$\mathcal{O}(\kappa_0^2\log\frac{1}{\epsilon})$	Adaptive	Easy
AdaIPL-LAC **(this work)**	$\mathcal{O}(C_S\kappa_0\log\frac{1}{\epsilon})$	Adaptive	Easy
AdaIPL-HAC **(this work)**	$\mathcal{O}(C_S\kappa_0\log\frac{1}{\epsilon})$	Adaptive	Easy

Table 1: Comparison of algorithms for RPR. In the first column, for IPL and AdaIPL, "-LAC" and "-HAC" correspond to **(LAC)** and **(HAC)** conditions, respectively. The second column compares the total complexity under the ideal choices of hyper-parameters for finding an ϵ-optimal point when $\epsilon > 0$ is small enough. $\kappa_0 \geq 1$ denotes the condition number, $C_S \geq 1$ is a factor related to solving (6), and $\mathcal{O}(\cdot)$ only hides numerical constants. The third column characterizes the types of step sizes for each algorithm. The fourth column summarizes the level of difficulties for hyper-parameter tuning. "N/A" here means that PL and IPL that employ fixed step sizes do not need tuning –refer to the second bullet point in Section 1.2.3 for explanations of "Easy" and "Hard".

1.2.3 Summary of Contributions

We propose AdaSubGrad and AdaIPL with adaptive step sizes chosen based on the quantiles of absolute residuals. To the best of our knowledge, we are the first to use quantile-based adaptive step sizes for robust phase retrieval to design practical methods that do not require intensive hyper-parameter tuning; moreover, unlike [16], the convergence guarantees of our methods do not need the algorithm parameters to satisfy some conditions involving unknown problem constants. The proposed algorithms enjoy the following advantages over existing algorithms.

- Our algorithms enjoy the best theoretical convergence rates. For finding an ϵ-optimal solution, we define the total complexity of subgradient-type algorithms as the number of iterations and the total complexity for proximal-linear-type algorithms as the total iterations used for inexactly solving all the subproblems (6). Under the ideal situation in terms of hyper-parameters that will be explained in Section 6, Table 1 summarizes the total complexity of all candidate algorithms for finding an ϵ-optimal solution where ϵ is sufficiently small. Here, $\kappa_0 \geq 1$ is the condition number of RPR that will be explained in Section 3, $C_S = \sqrt{m}\max_{i\in[m]}\|a_i\|_2/\|A\|_2$ is a constant factor related to the complexity of solving (6), and we treat ρ_l, ρ_h in **(LAC)** and **(HAC)** in (16) as numerical constants. AdaSubGrad enjoys a local linear rate comparable to other subgradient algorithms, and AdaIPL enjoys a better local linear rate in terms of the condition number.
- Our algorithms enjoy a linear rate even under imperfect choices of hyper-parameters: AdaIPL shows local linear convergence for any $G > 0$, and AdaSubGrad enjoys local linear convergence when G is sufficiently small. In contrast, PSubGrad[17] relies on the value of $F(x_\star)$, and GSubGrad[16] only converges under their specific choice of parameters depending on unknown constants for (4). The difficulty of tuning hyper-parameters is summarized in Table 1.
- We conduct numerical tests comparing AdaSubGrad and AdaIPL against the other state-of-the-art methods for solving the RPR problem. Empirical results show that both AdaSubGrad and AdaIPL are robust to parameter selection and perform better than the others.

Notations. For any $m_0 \in \mathbb{N}_+$, we denote $[m_0] = \{1, 2 \ldots m_0\}$. $\mathbb{S}^{n-1} = \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$. $\mathbf{1}[\cdot]$ is the indicator function that takes logic statements as its argument, it returns 1 when its argument is true and 0 otherwise when its argument is false. For $x \in \mathbb{R}$, we let $\text{sign}(x) = \mathbf{1}[x > 0] - \mathbf{1}[x < 0]$. We also adopted the Landau notation, i.e., for $f, g : \mathbb{R}_+ \to \mathbb{R}_+$, we use $f = \mathcal{O}(g)$ and $f = \Omega(g)$ if there exists some $C^0 > 0$ and $n^0 \in \mathbb{R}_+$ such that $f(n) \leq C^0 g(n)$ and $f(n) \geq C^0 g(n)$, respectively, for all $n \geq n^0$; moreover, if $f = \mathcal{O}(g)$ and $f = \Omega(g)$, then we use $f = \Theta(g)$.

Organization. Section 2 introduces basic properties of $F(\cdot)$. Section 3 discusses the data generation process and key conditions for our adaptive step sizes, and in Section 4, we give the proof of Theorem 3.4 that provides us with a proper statistical foundation for our convergence analysis.

Section 5 establishes the convergence rate of AdaSubGrad, and Section 6 shows the convergence rate of AdaIPL. In Section 7, we provide the proofs of the results given in Section 6. Finally, after the numerical experiments in Section 8, we conclude the paper with a brief discussion in Section 9.

2 Basic Properties of $F(\cdot)$

In this section, we provide some basic results regarding the important properties of $F(\cdot)$. Throughout this section, $L \triangleq 2\|A\|_2^2/m$.

Lemma 2.1 (Lemma 6 in [14], local Lipschitz continuity). *For any* $r \geq 0$*,*

$$\sup_{x,y\in\mathbb{R}^n}\left\{\frac{|F(x)-F(y)|}{\|x-y\|_2}:\ \Delta(x)\leq r,\ \Delta(y)\leq r,\ x\neq y\right\}\leq L(\|x_\star\|_2+r).$$

Lemma 2.2 (Absolute deviation bound). *For all* $x \in \mathbb{R}^n$*, it holds that*

$$|F(x)-F(x_\star)|\leq \frac{L}{2}\|x-x_\star\|_2\|x+x_\star\|_2. \tag{17}$$

Proof. When $x \in \{x_\star, -x_\star\}$, the relationship holds; otherwise, for $u = (x-x_\star)/\|x-x_\star\|_2$ and $v = (x+x_\star)/\|x+x_\star\|_2$, we have $|F(x)-F(x_\star)|\leq \frac{1}{m}\sum_{i=1}^m|(a_i^\top x)^2-(a_i^\top x_\star)^2|$ which is equal to $\|x-x_\star\|_2\|x-x_\star\|_2\frac{1}{m}\sum_{i=1}^m|u^\top a_i a_i^\top v|$. Note that

$$\frac{1}{m}\sum_{i=1}^m|u^\top a_i a_i^\top v|\leq\left(u^\top\left(\frac{1}{2m}\sum_{i=1}^m a_i a_i^\top\right)u+v^\top\left(\frac{1}{2m}\sum_{i=1}^m a_i a_i^\top\right)v\right)\leq L/2.$$

This completes the proof. □

Lemma 2.3 (Weak Convexity [11]). *The inequality below holds for any* $x, y \in \mathbb{R}^n$*:*

$$|F(x)-F(x;y)|\leq\frac{1}{2t}\|x-y\|_2^2,\quad \forall\, t:\ 0<t\in(0,1/L]. \tag{18}$$

Next, similar to [11], [16], and [17], we make the following sharpness assumption.

Assumption 2.4 (Condition C1 in [11]). There exists $\lambda_s > 0$ such that

$$F(x)-F(x_\star)\geq\lambda_s\Delta(x),\quad \forall x\in\mathbb{R}^n, \tag{19}$$

where F is defined in (4) and $\Delta(\cdot)$ is defined in (3).

Under some data generation process, in [11, Proposition 4], it is proven that (19) holds with high probability when m/n is large enough and the proportion of corrupted measurements is small enough.

Lemma 2.5 (Lemma 7 in [14]). *Under Assumption 2.4, for any* $r \geq 0$*,*

$$\{x\in\mathbb{R}^n:\Delta(x)\leq E(r)\}\subseteq\{x\in\mathbb{R}^n:F(x)-F(x_\star)\leq r\}\subseteq\{x\in\mathbb{R}^n:\Delta(x)\leq r/\lambda_s\},$$

implying $E(r)\leq r/\lambda_s$*, where* $E(r)\triangleq\left(\sqrt{L^2\|x_\star\|_2^2+4rL}-L\|x_\star\|_2\right)/(2L)$.

Lemma 2.6. *Assumption 2.4 implies that* $\lambda_s \leq L\|x_\star\|_2/2$.

Proof. Consider $x \in \mathbb{R}^n$ such that $\|x-x_\star\|_2=\Delta(x)$. Based on Lemma 2.2 and Assumption 2.4, we have $\lambda_s\Delta(x)\leq F(x)-F(x_\star)\leq(L/2)\Delta(x)\|x+x_\star\|_2$, which implies that $\lambda_s\leq(L/2)\|x+x_\star\|_2$. Letting $x=\mathbf{0}$, we have $\lambda_s\leq L\|x_\star\|_2/2$. □

Some differentiability properties of $F(\cdot)$ Recall $F(\cdot)=h(c(\cdot))$, where $h:\mathbb{R}^m\to\mathbb{R}$ and $c:\mathbb{R}^n\to\mathbb{R}^m$ such that $h(\cdot)=\|\cdot\|_1/m$ and $c(\cdot)=|A\cdot|^2-b$; hence, $\partial F(x)=[J_c(x)]^\top\partial h(c(x))$ for $x\in\mathbb{R}^n$ as defined in (11). For any fixed $\bar{x}\in\mathbb{R}^n$, let $c_{\bar{x}}(x)\triangleq c(\bar{x})+[J_c(\bar{x})](x-\bar{x})$ for all $x\in\mathbb{R}^n$; hence, $c_{\bar{x}}(\cdot)$ is an affine function. Consider $F(\cdot;\bar{x})$ defined in (5). Since $F(\cdot;\bar{x})=h(c_{\bar{x}}(\cdot))$, the function $F(x;\bar{x})$ is convex in x, and $\partial F(x;\bar{x})\mid_{x=\bar{x}}$ can be written as $[J_c(\bar{x})]^\top\partial h(c_{\bar{x}}(\bar{x}))=[J_c(\bar{x})]^\top\partial h(c(\bar{x}))$. Thus, $\partial F(\bar{x})=\partial F(x;\bar{x})|_{x=\bar{x}}$ for any $\bar{x}\in\mathbb{R}^n$. Next, based on this observation, we provide a useful inequality for $F(\cdot)$.

For any $\bar{x}\in\mathbb{R}^n$ and $v\in\partial F(\bar{x})$, since we have $v\in F(x;\bar{x})|_{x=\bar{x}}$ and $F(\cdot;\bar{x})$ is convex, we have $F(x;\bar{x})-F(\bar{x})=F(x;\bar{x})-F(\bar{x};\bar{x})\geq\langle x-\bar{x},v\rangle$. Together with $F(x;\bar{x})\leq F(x)+L\|x-\bar{x}\|_2^2/2$, which follows from (18), we can conclude that

$$F(x)-F(\bar{x})-\langle x-\bar{x},v\rangle+L\|x-\bar{x}\|_2^2/2\geq 0,\quad \forall\, x,\bar{x}\in\mathbb{R}^n,\ \forall\, v\in\partial F(\bar{x}). \tag{20}$$

3 Data Generation Process and Key Conditions

In this section, we provide the statistical background and some key conditions for RPR. We first introduce the assumption for the data generation process of RPR.

Assumption 3.1. Suppose that the following conditions hold:

(a) Given non-negative integers m_1, m_2, let $a_1, a_2 \ldots a_{m_1+m_2}$ in $\mathbb{R}^n$ be random vectors following independent and identical distributions, represented by a generic random variable a, which follows a σ^2-subGaussian distribution with $\mathbf{0}$ as mean and Σ as covariance matrix. The true signal vector $x_\star \in \mathbb{R}^n \backslash \{\mathbf{0}\}$ satisfies (2) such that $\mathcal{I}_1 = [m_1]$, $\mathcal{I}_2 = [m_1 + m_2] \setminus [m_1]$, and ξ_i for $i \in \mathcal{I}_2$ is a non-negative random variable following an arbitrary distribution.

(b) Suppose $\kappa_{\text{st}} \triangleq \inf_{u,v \in \mathbb{S}^{n-1}} \mathbb{E}[|\langle a, u\rangle\langle a, v\rangle|]$ and $p_{\text{fail}} \triangleq \frac{m_2}{m_1+m_2} < \frac{1}{2}$ satisfy $\kappa_{\text{st}} - 2p_{\text{fail}}\|\Sigma\|_2 > 0$.

(c) Given $\tilde{p} \in (p_{\text{fail}},\ 1 - p_{\text{fail}})$, suppose that there exists $\kappa > 0$ such that

$$p_0 \triangleq \inf_{u,v \in \mathbb{S}^{n-1}} \mathbb{P}(\min\{|\langle a, u\rangle|,\ |\langle a, v\rangle|\} \geq \kappa) > \frac{1-\tilde{p}}{1-p_{\text{fail}}}.$$

Next, we briefly discuss this assumption. First, (a) in Assumption 3.1 is a combination of Model M2 and Assumption A4 in [11]. Let $m \triangleq m_1 + m_2$ denote the total number of measurements, i.e., $|\mathcal{I}_1| = m_1$, $|\mathcal{I}_2| = m_2$; moreover, $p_{\text{fail}} = \frac{m_2}{m_1+m_2}$ represents the proportion of corrupted measurements – when $m_2 = p_{\text{fail}} = 0$, this model reduces to noiseless phase retrieval problem. We observe $\{(a_i, b_i)\}_{i=1}^m$ without knowing which measurements are corrupted. We also guarantee that $\{a_i : i \in \mathcal{I}_1\}$ and $\{a_i : i \in \mathcal{I}_2\}$ are mutually independent and they follow a σ^2-subGaussian distribution. Below we argue that the aforementioned subGaussian assumption in (a) provides statistical stability guarantees for the recovery process. That said, before proceeding in this direction, it is essential to emphasize that $F(\cdot)$ is weakly convex without any statistical assumption.

Remark 3.2. [11, Lemma 3.1] shows that under Assumption 3.1(a), when m/n is large enough, the weak convexity parameter L is close to $2\|\Sigma\|_2$ with high probability.

Remark 3.3. According to [11, Proposition 4], under (a) and (b) in Assumption 3.1, the sharpness condition in (19) holds with high probability when $\kappa_{\text{st}} > 0, m/n$ is large and p_{fail} is sufficiently small, in which case, λ_s is close to $\|x_\star\|_2(\kappa_{\text{st}} - 2p_{\text{fail}}\|\Sigma\|_2)$. More importantly, (19) implies that $\{x_\star, -x_\star\} = \arg\min_{x\in\mathbb{R}^n} F(x)$.

Using weak convexity and sharpness properties, we can define a *condition number* $\kappa_0 \triangleq L\|x_\star\|_2/(2\lambda_s)$ for F defined in (4); under (19), Lemma 2.6 shows that $\kappa_0 \geq 1$.

Since $\kappa_{\text{st}} \leq \inf_{u\in\mathbb{S}^{n-1}} \mathbb{E}(a^\top u)^2$, we argue that $L\|x_\star\|_2/(2\lambda_s)$ is related to the condition number of Σ and p_{fail}. Indeed, from Remarks 3.2 and 3.3, for $\kappa_{\text{st}} > 0$, m/n large and p_{fail} sufficiently small,

$$\kappa_0 \triangleq L\|x_\star\|_2/(2\lambda_s) \approx \frac{1}{\kappa_{\text{st}}/\|\Sigma\|_2 - 2p_{\text{fail}}}. \tag{21}$$

Therefore, κ_0 represents how ill-conditioned the RPR problem is.

Assumption 3.1(c) is new to the RPR literature and is the key to guarantee that t_k and α_k are proportional to $\Theta(\Delta(x^k))$. Indeed, consider $\tilde{p} \in (p_{\text{fail}},\ 1 - p_{\text{fail}})$, which can always be set to $\tilde{p} = \frac{1}{2}$ as $\frac{1}{2}$ clearly belongs to this interval, and recall that we use $\tilde{p}$-th percentile of the residuals at x^k to define α_k and t_k as in (10) and (12), respectively, e.g., for $\tilde{p} = \frac{1}{2}$, both α_k and t_k are set based on the median value of $\{r_i(x^k)\}_{i=1}^m$. Statistically, it means that for any $\|u\|_2 = 1$, the distribution of $\langle a, u\rangle$ should not concentrate too close around $\mathbf{0} \in \mathbb{R}^n$.

Next, we provide an example that (a), (b), and (c) of Assumption 3.1 hold simultaneously. Let $\tilde{p} = \frac{1}{2}$ and $a \sim N(\mathbf{0}, I_n)$, which indicates that $\kappa_{\text{st}} = 2/\pi$ (see [11, Example 4]) and $\Sigma = \mathcal{I}_n$. Thus, when $0 < p_{\text{fail}} \leq 1/(2\pi)$, we have that $\kappa_{\text{st}} - 2p_{\text{fail}}\|\Sigma\|_2 \geq 1/\pi > 0$. Moreover, for $p_0(w) \triangleq \inf_{u,v\in\mathbb{S}^{n-1}} \mathbb{P}(\min\{|\langle a, u\rangle|,\ |\langle a, v\rangle|\} \geq w)$, it holds that $\lim_{w\to 0} p_0(w) = 1$; and, we also have $\frac{1-\tilde{p}}{1-p_{\text{fail}}} \leq 1/(2 - 1/\pi) < 1$. Therefore, we can conclude that $\kappa > 0$ satisfying Assumption 3.1(c) exists.

Next, we argue that $r^{\tilde{p}}(x) = \Theta(F(x) - F(x_\star))$ with high probability.

Theorem 3.4. *Suppose that Assumption 3.1 holds and $\tilde{p} \in (p_{\text{fail}}, 1 - p_{\text{fail}})$, $m\tilde{p} \in \mathbb{N}$. There exists positive constants $u_L, u_H, \rho_1, \rho_2, \rho_3$, depending on $\Sigma, \sigma, \kappa_{\text{st}}, \kappa, p_0, p_{\text{fail}}, \tilde{p}$, such that when $m \geq \rho_1 n$,*

$$\mathbb{P}\left(r^{\tilde{p}}(x) \in \Big[u_L(F(x) - F(x_\star)),\ u_H(F(x) - F(x_\star))\Big],\ \forall x \in \mathbb{R}^n\right) \geq 1 - \rho_2 \exp(-m\rho_3). \tag{22}$$

The detailed proof of Theorem 3.4 will be provided in Section 4.

Finally, using (19) and (22) implies the following conclusions for `AdaSubGrad` and `AdaIPL`.

Corollary 3.5. *Under Assumption 2.4, suppose the event in* (22) *hold.*

(a) When $0 < G < 2/u_H$, the `AdaSubGrad` *step size α_k in* (10) *satisfies that*

$$\alpha_k \in \Big[c_1(F(x^k) - F(x_\star)),\ c_2(F(x^k) - F(x_\star))\Big], \quad \forall k \in \mathbb{N},$$

where $c_1 \triangleq u_L G$, $c_2 \triangleq u_H G$ and they satisfy $0 < c_1 \le c_2 < 2$.

(b) For any $k \in \mathbb{N}$, if $\Delta(x^k) \le \|x_\star\|_2$, then there exists $g_k \in [g_L, g_H]$ such that the `AdaIPL` *step size t_k in* (12) *satisfies $t_k = \min\{L^{-1}, g_k\Delta(x^k)\}$, where $g_L \triangleq G\lambda_s u_L$ and $g_H \triangleq 3GL\|x_\star\|_2 u_H/2$.*

Proof. (a) is a direct conclusion of (22). For (b), we only need to prove that for any $x \in \mathbb{R}^n$ with $\Delta(x) \le \|x_\star\|$, we have $(g_L/G)\Delta(x) \le r^{\tilde p}(x) \le (g_H/G)\Delta(x)$. The first inequality follows from (19) and (22). For the second one, (22) implies that $r^{\tilde p}(x) \le u_H(F(x) - F(x_\star))$. By Lemma 2.2, $|F(x) - F(x_\star)| \le \frac{L}{2}\|x - x_\star\|_2\|x + x_\star\|_2$. In addition, we have that $\frac{L}{2}\|x - x_\star\|_2\|x + x_\star\|_2 \le \frac{L}{2}\Delta(x)(\Delta(x) + 2\|x\|_\star) \le 3L\|x_\star\|_2\Delta(x)/2 = g_H/(Gu_H)$. Thus, the second inequality is also proved. □

Note that larger choices for G lead to larger c_1, c_2, g_L, g_H coefficients.

4 Proof of Theorem 3.4

Fix $x \in \mathbb{R}^n$. First, we recall an upper on $F(x) - F(x_\star)$. According to Lemma 2.2, one has $F(x) - F(x_\star) \le \frac{L}{2}\|x - x_\star\|_2\|x + x_\star\|_2$, in which $L = 2\|A\|_2^2/m$ can be equivalently written as $L = 2\left\|\frac{1}{m}\sum_{i=1}^m a_i a_i^\top\right\|_2$. Moreover, Lemma 4.1 stated below shows that L is close to $2\|\Sigma\|_2$ when m/n is large, where Σ is given in Assumption 3.1 and $\|\Sigma\|_2 = \sup_{u\in\mathbb{S}^{n-1}} \mathbb{E}\langle a, u\rangle^2$. Throughout the proof, we adopt the notation $\tilde\Delta(x) \triangleq \|x - x_\star\|_2\|x + x_\star\|_2$ for brevity.

Lemma 4.1. *(Lemma 3.1 in [11]) Let Assumption 3.1 hold. Then for all $t \ge 0$,*

$$\mathbb{P}\left(\left\|\frac{1}{m}\sum_{i=1}^m a_i a_i^\top - \Sigma\right\|_2 \ge 11\sigma^2 \max\left\{\sqrt{\frac{4n}{m} + t}, \frac{4n}{m} + t\right\}\right) \le \exp(-mt).$$

Next, we begin the proof. It contains two main components.

4.1 $r^{\tilde p}(x) = \Theta(\tilde\Delta(x))$ with high probability

Next, we review the definitions related to (10) and (12) regarding our choice of α_k and t_k. Let $r_i(x) = |\langle a_i, x\rangle^2 - b_i|$ for all $i = 1, 2\ldots, m$.

Definition 4.2. Given $x \in \mathbb{R}^n$, for any $i \in [m]$, let $r_{(i)}(x)$ denote the i-th order statistic based on $\{r_i(x)\}_{i=1}^m$, which indicates that $r_{(1)}(x) \le \ldots \le r_{(m)}(x)$. Similarly, for any $i \in [m_1]$, let $\tilde r_{(i)}(x)$ denote the $i-$th order statistic based on $\{r_i(x)\}_{i=1}^{m_1}$. Moreover, for any given percentile $\tilde p \in (0, 1)$ such that $m\tilde p \in \mathbb{N}$, we define $r^{\tilde p}(x) \triangleq r_{(m\tilde p)}(x)$.

Remark 4.3. Recall that $p_{\text{fail}} = (m - m_1)/m$ is the fraction of corrupted measurements, and by definition $mp_{\text{fail}} = m - m_1 \in \mathbb{Z}_+$. We assume that $p_{\text{fail}} < \frac{1}{2}$ and $\tilde p \in (p_{\text{fail}}, 1 - p_{\text{fail}})$. For the sake of simplicity of notation, we assume that $m\tilde p \in \mathbb{Z}_+$; hence, $m(\tilde p - p_{\text{fail}}) \in \mathbb{Z}$, and note that $m\tilde p < m(1 - p_{\text{fail}}) = m_1$. Thus, the order statistics $\tilde r_{(m(\tilde p - p_{\text{fail}}))}$, $\tilde r_{(m\tilde p)}$ and $r_{(m\tilde p)}$ are all well defined.

Next, we compare $r_{(m\tilde p)}(x)$ to the mean and quantiles of the noiseless samples.

Lemma 4.4. *Suppose that Assumption 3.1 holds. If $p_{\text{fail}} < \tilde p < 1 - p_{\text{fail}}$, then for any $x \in \mathbb{R}^n$, $\tilde r_{(m(\tilde p - p_{\text{fail}}))}(x) \le r_{(m\tilde p)}(x) \le \tilde r_{(m\tilde p)}(x)$; moreover,*

$$\tilde r_{(m\tilde p)}(x) \le \frac{1}{m(1 - p_{\text{fail}} - \tilde p)}\sum_{i=m\tilde p+1}^{m_1} \tilde r_{(i)}(x) \le \frac{1 - p_{\text{fail}}}{(1 - p_{\text{fail}} - \tilde p)}\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x).$$

Proof. We denote $A_1 \triangleq \{i \in [m] : r_i(x) \le r_{(m\tilde{p})}(x)\}$, $A_2 \triangleq \{i \in [m_1] : r_i(x) \le r_{(m\tilde{p})}(x)\}$, $A_3 \triangleq \{i \in [m] : r_i(x) \le \tilde{r}_{(m\tilde{p})}(x)\}$ and $A_4 \triangleq \{i \in [m_1] : r_i(x) \le \tilde{r}_{(m\tilde{p})}(x)\}$. From the definition of order statistics, we know that $|A_1| \ge m\tilde{p}$ – it might happen that $r_i(x) = r_j(x)$ for some $i, j \in [m]$ such that $i \ne j$; hence, it is possible that $|A_1| > m\tilde{p}$. Furthermore, note that $A_2 = A_1 \backslash ([m]\backslash[m_1])$. Thus, $|A_2| = |A_1 \backslash ([m]\backslash[m_1])| \ge |A_1| - (m - m_1) \ge m(\tilde{p} - p_{\text{fail}})$, which indicates the first inequality. We can find that $A_4 \subseteq A_3$, which indicates that $|A_3| \ge |A_4| \ge m\tilde{p}$. Thus, the second inequality holds. The third inequality holds because $\tilde{r}_{(i)}(x) \ge \tilde{r}_{(m\tilde{p})}(x)$ when $i \ge m\tilde{p}$ and we also have $m(1 - p_{\text{fail}} - \tilde{p}) = m_1 - m\tilde{p}$. The fourth one holds because $\sum_{i=1}^{m_1} r_i(x) \ge \sum_{i=m\tilde{p}+1}^{m_1} \tilde{r}_{(i)}(x)$ and $\frac{1-p_{\text{fail}}}{(1-p_{\text{fail}}-\tilde{p})}\frac{1}{m_1} = \frac{1}{m(1-p_{\text{fail}}-\tilde{p})}$. ☐

Lemma 4.4 implies that

$$\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \le r_{(m\tilde{p})}(x) \le \frac{1-p_{\text{fail}}}{(1-p_{\text{fail}}-\tilde{p})}\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x), \tag{23}$$

where both bounds are quantities that are only related to the noiseless measurements. Next, we argue that $\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x) \le \mathcal{O}(\tilde{\Delta}(x))$ for all $x \in \mathbb{R}^n$ with high probability.

Lemma 4.5. *Suppose that Assumption 3.1 holds for $\tilde{p} \in (0,1)$ such that $m\tilde{p} \in \mathbb{N}$ and $p_{\text{fail}} < \tilde{p} < 1 - p_{\text{fail}}$. Then whenever $m \ge 8n/(u_1(1-p_{\text{fail}}))$, we have*

$$\mathbb{P}\left(\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x) \le \frac{3}{2}\|\Sigma\|_2\tilde{\Delta}(x), \quad \forall\, x \in \mathbb{R}^n\right) \ge 1 - \exp\Big(-\frac{mu_1}{2}(1-p_{\text{fail}})\Big) \tag{24}$$

where $u_1 = \min\{1,\ \|\Sigma\|_2/(22\sigma^2)\}\|\Sigma\|_2/(22\sigma^2)$.

Proof. For $m_2 = 0$, i.e., $m = m_1$, we have $F(x_\star) = 0$ and Lemma 2.2 implies $\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x) = F(x) - F(x_\star) \le \|\frac{1}{m_1}\sum_{i=1}^{m_1} a_i a_i^\top\|_2\tilde{\Delta}(x),\ \forall\, x \in \mathbb{R}^n$. In this setting, letting $m_1 \ge 8n/u_1$ and $t = u_1/2$ in Lemma 4.1, we have that $\|\frac{1}{m_1}\sum_{i=1}^{m_1} a_i a_i^\top - \Sigma\|_2 \le \frac{1}{2}\|\Sigma\|_2$ holds with probability at least $1 - \exp(-m_1u_1/2)$. Thus, under this event, we have $\frac{1}{m_1}\sum_{i=1}^{m_1} r_i(x) \le \frac{3}{2}\|\Sigma\|_2\tilde{\Delta}(x)$ for all $x \in \mathbb{R}^n$. Next, for $m_2 > 0$, using $m_1 = (1-p_{\text{fail}})m$ within the above inequality completes the proof. ☐

Next, we argue that $\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \ge \Omega(\tilde{\Delta}(x))$ holds for all $x \in \mathbb{R}^n$ with high probability.

Lemma 4.6. *Suppose that Assumption 3.1 holds for $\tilde{p} \in (0,1)$ such that $m\tilde{p} \in \mathbb{N}$ and $p_{\text{fail}} < \tilde{p} < 1 - p_{\text{fail}}$. There exists $c_0 > 0$ such that for all $x \in \mathbb{R}^n$, $\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \ge \kappa^2\tilde{\Delta}(x)$ holds w.p. at least $1 - 2\exp\left(-\frac{m(1-p_{\text{fail}})\kappa^2(p_0-(1-\tilde{p})/(1-p_{\text{fail}}))^2}{32}\right)$ whenever $m \ge \frac{c_0^2 n}{\kappa^2(1-p_{\text{fail}})}\left(\frac{p_0-(1-\tilde{p})/(1-p_{\text{fail}})}{4}\right)^{-2}$.*

Proof. When $\Delta(x) = 0$, the conclusion obviously holds; otherwise, let $u = (x - x_\star)/\|x - x_\star\|_2$, $v = (x + x_\star)/\|x + x_\star\|_2$, and for all $i \in [m_1]$ we define

$$h_i(u,v) = \mathbf{1}[|\langle a_i, u\rangle| \ge \kappa]\mathbf{1}[|\langle a_i, v\rangle| \ge \kappa].$$

Thus, for all $i \in [m_1]$, we have

$$r_i(x) = |\langle a_i, u\rangle| \cdot |\langle a_i, v\rangle|\tilde{\Delta}(x) \ge \kappa^2 h_i(u,v)\tilde{\Delta}(x). \tag{25}$$

Next, we consider the event $\frac{1}{m_1}\sum_{i=1}^{m_1} h_i(u,v) > \frac{1-\tilde{p}}{1-p_{\text{fail}}}$. Knowing that $h_i(u,v)$ takes value in $\{0,1\}$, the event indicates that number of 0 in $\{h_i(u,v)\}_{i=1}^{m_1}$ is strictly less than $m_1 - m_1(1-\tilde{p})/(1-p_{\text{fail}}) = m(\tilde{p} - p_{\text{fail}})$. Together with (25), this event indicates that $\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \ge \kappa^2\tilde{\Delta}(x)$. To conclude,

$$\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \ge \kappa^2\tilde{\Delta}(x)\ \mathbf{1}\left[\frac{1}{m_1}\sum_{i=1}^{m_1} h_i(u,v) > \frac{1-\tilde{p}}{1-p_{\text{fail}}}\right]. \tag{26}$$

Based on (c) in Assumption 3.1, we have $\mathbb{E}h_i(u,v) \ge p_0 > \frac{1-\tilde{p}}{1-p_{\text{fail}}}$ for all $u, v \in \mathbb{S}^{n-1}$ and $i \in [m]$. By the last display mode equation in the proof of [11, Proposition 1], there exists a numerical constant $c_0 < \infty$ such that for any $t \ge 0$,

$$\mathbb{P}\left(\sup_{u,v\in\mathbb{S}^{n-1}} \kappa\left|\frac{1}{m_1}\sum_{i=1}^{m_1} h_i(u,v) - \mathbb{E}\left[h_i(u,v)\right]\right| \ge c_0\sqrt{\frac{n}{m_1}} + t\right) \le 2\exp\left(-\frac{m_1t^2}{2}\right).$$

Therefore, if $m \geq \frac{c_0^2 n}{\kappa^2(1-p_{\text{fail}})}\left(\frac{p_0-(1-\tilde{p})/(1-p_{\text{fail}})}{4}\right)^{-2}$ and $t = \kappa\frac{p_0-(1-\tilde{p})/(1-p_{\text{fail}})}{4}$, then w.p. at least $1 - 2\exp\left(-\frac{m(1-p_{\text{fail}})\kappa^2(p_0-(1-\tilde{p})/(1-p_{\text{fail}}))^2}{32}\right)$, for all $u, v \in \mathbb{S}^{n-1}$ we have

$$\frac{1}{m_1}\sum_{i=1}^{m_1} h_i(u,v) \geq \mathbb{E}\left[h_i(u,v)\right] - c_0\sqrt{\frac{n}{m_1\kappa^2}} - t/\kappa \geq \frac{p_0 + \frac{1-\tilde{p}}{1-p_{\text{fail}}}}{2} > \frac{1-\tilde{p}}{1-p_{\text{fail}}}.$$

Together with (26), this indicates that $\tilde{r}_{(m(\tilde{p}-p_{\text{fail}}))}(x) \geq \kappa^2\tilde{\Delta}(x)$ for all $x \in \mathbb{R}^n$. □

Combining the event in Lemma 4.5 and the event in Lemma 4.6, applying (23), we can claim that, when $m \geq u_2 n$ where $u_2 \triangleq \frac{8}{1-p_{\text{fail}}}\max\left\{\frac{1}{u_1}, \frac{2c_0^2}{\kappa^2}\left(p_0 - \frac{1-\tilde{p}}{1-p_{\text{fail}}}\right)^{-2}\right\}$, w.p. at least $1 - 3\exp(-u_3 m)$ where $u_3 \triangleq \frac{1-p_{\text{fail}}}{2}\min\left\{\frac{\kappa^2}{16}\left(p_0 - \frac{1-\tilde{p}}{1-p_{\text{fail}}}\right)^2,\ u_1\right\}$, the following event holds:

$$u_4\tilde{\Delta}(x) \leq r_{(m\tilde{p})}(x) \leq u_5\tilde{\Delta}(x), \quad \forall\, x \in \mathbb{R}^n, \tag{27}$$

where $u_4 \triangleq \kappa^2$ and $u_5 \triangleq \frac{3}{2}\|\Sigma\|_2\left(1 - \frac{\tilde{p}}{1-p_{\text{fail}}}\right)^{-1}$.

4.2 $F(x) - F(x_\star) = \Theta(\tilde{\Delta}(x))$ with high probability

First, we provide an upper bound for $F(x) - F(x_\star)$. Letting $m \geq 8n/u_1$ and $t = u_1/2$ in Lemma 4.1, we have that $\|\frac{1}{m}\sum_{i=1}^m a_i a_i^\top - \Sigma\|_2 \leq \frac{1}{2}\|\Sigma\|_2$ holds with probability at least $1 - \exp(-mu_1/2)$. Thus, under this event, together with Lemma 2.2, and noticing that $L = 2\left\|\frac{1}{m}\sum_{i=1}^m a_i a_i^\top\right\|_2$, we have $F(x) - F(x_\star) \leq \frac{3}{2}\|\Sigma\|_2\tilde{\Delta}(x)$.

Next, we discuss a lower bound for $F(x) - F(x_\star)$ provided by [11].

Lemma 4.7. *(Proposition 4 in [11]) Under Assumption 3.1, $\exists C, c_3 > 0$ such that*

$$F(x) - F(x_\star) \geq \left(\kappa_{\text{st}} - 2p_{\textit{fail}}\,\|\Sigma\|_2 - C\sigma^2\sqrt[3]{\frac{n}{m}} - C\sigma^2 t\right)\tilde{\Delta}(x),$$

holds for all $x \in \mathbb{R}^n$ with probability at least $1 - 2e^{-c_3 m} - 2e^{-mt^2}$ for any $t > 0$.

Thus, for $m \geq \left(\frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{4C\sigma^2}\right)^{-3} n$ and $t = \frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{4C\sigma^2}$, it holds with probability at least $1 - 4\exp\Big(-m\min\{c_3, (\frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{4C\sigma^2})^2\}\Big)$ that $F(x) - F(x_\star) \geq \frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{2}\tilde{\Delta}(x)$ for all $x \in \mathbb{R}^n$. Thus, for $m \geq u_6 n$ with $u_6 \triangleq \max\left\{\frac{8}{u_1}, \left(\frac{4C_1\sigma^2}{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}\right)^3\right\}$, one has

$$u_8\tilde{\Delta}(x) \leq F(x) - F(x_\star) \leq u_9\tilde{\Delta}(x), \quad \forall x \in \mathbb{R}^n, \tag{28}$$

w.p. at least $1 - 5\exp(-mu_7)$, where $u_7 \triangleq \min\left\{c_3, \left(\frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{4C_1\sigma^2}\right)^2, \frac{u_1}{2}\right\}$, $u_8 \triangleq \frac{\kappa_{\text{st}}-2p_{\text{fail}}\|\Sigma\|_2}{2}$ and $u_9 \triangleq \frac{3}{2}\|\Sigma\|_2$. Finally, we finish the proof of Theorem 3.4 by combining (27) and (28); indeed, whenever $m \geq \rho_1 n$ for $\rho_1 = \max\{u_2, u_6\}$, (22) holds for $u_L = u_4/u_9$ and $u_H = u_5/u_8$ with probability at least $1 - \rho_2\exp(-m\rho_3)$ for $\rho_2 = 8$ and $\rho_3 = \min\{u_3, u_7\}$.

5 Convergence Rate of `AdaSubGrad`

In this section, we establish the local linear convergence of `AdaSubGrad` (Algorithm 1).

Theorem 5.1 (Convergence Rate of `AdaSubGrad`)**.** *Suppose that Assumption 2.4 and the probability event in* (22) *hold, and c_1, c_2 are constants in Corollary 3.5. If x^0 satisfies $\Delta(x^0) \leq \lambda_s(1 - \frac{c_2}{2})/L$, then for $c \triangleq \max\{|1-c_1|, |1-c_2|\}$,* `AdaSubGrad` *sequence $\{x^k\}_{k=0}^\infty$ satisfies that*

$$\Delta(x^k) \leq \left(\sqrt{1 - \frac{2\lambda_s^2(1-c^2)}{9L^2\|x_\star\|_2^2}}\right)^k \Delta(x^0) \triangleq R_k, \quad \forall k \in \mathbb{N}. \tag{29}$$

Proof. Define $e_k \triangleq \alpha_k/(F(x^k)-F(x_\star))$ for $k \geq 0$. From Corollary 3.5, $e^k \in [c_1, c_2] \subseteq (0,2)$ and $e^k(2-e^k) \geq 1-c^2$ for $k \in \mathbb{N}$. Note that $1 - \frac{2\lambda_s^2(1-c^2)}{9L^2\|x_\star\|_2^2} \in (0,1)$ from Lemma 2.6. Let $\hat{x} \in \{x_\star, -x_\star\}$ such that $\Delta(x^0) = \|x^0 - \hat{x}\|_2$. In the rest we establish (29) by showing $\|x^k - \hat{x}\|_2 \leq R_k$ for $k \geq 0$ using induction. Note that the base case of the induction for $k = 0$ trivially holds, i.e., $\|x^0 - \hat{x}\|_2 \leq R_0 = \Delta(x^0)$. Next, we assume that $\|x^k - \hat{x}\|_2 \leq R_k$ holds for some $k \in \mathbb{Z}_+$, and we show that it also holds for $k+1$. From the update rule in (11), we have

$$\begin{aligned} \left\|x^{k+1} - \hat{x}\right\|_2^2 &= \left\|x^k - \hat{x}\right\|_2^2 + 2\left\langle x^k - \hat{x}, x^{k+1} - x^k\right\rangle + \left\|x^{k+1} - x^k\right\|_2^2 \\ &= \left\|x^k - \hat{x}\right\|_2^2 + \frac{2e_k\left(F(x^k) - F(\hat{x})\right)}{\|\xi^k\|_2^2} \cdot \left\langle \xi^k, \hat{x} - x^k\right\rangle + e_k^2 \frac{\left(F(x^k) - F(\hat{x})\right)^2}{\|\xi^k\|_2^2}. \end{aligned} \tag{30}$$

(20) with $x = \hat{x}, \bar{x} = x^k, v = \xi^k$ implies $\left\langle \xi^k, \hat{x} - x^k\right\rangle \leq F(\hat{x}) - F(x^k) + \frac{L}{2}\left\|x^k - \hat{x}\right\|^2$. Applying it to (30), we can further find that

$$\|x^{k+1} - \hat{x}\|_2^2 \leq \|x^k - \hat{x}\|_2^2 + \frac{e_k\left(F(x^k) - F(\hat{x})\right)}{\|\xi^k\|_2^2}\Big(L\left\|x^k - \hat{x}\right\|_2^2 - (2 - e_k)\left(F(x^k) - F(\hat{x})\right)\Big). \tag{31}$$

Next, we discuss three quantities appearing in (31).

Term 1: $\|x^k - \hat{x}\|_2^2$ By the induction hypothesis, we have $\|x^k - \hat{x}\|_2 \leq R_k \leq \Delta(x^0)$. Thus, $\|x^k - \hat{x}\|_2 \leq \Delta(x^0) \leq \lambda_s(2-c_2)/(2L) \leq \lambda_s/L$, where the second inequality follows from the hypothesis and the last one from $c_2 > 0$. Moreover, the last inequality and Lemma 2.6 imply that $\|x^k - \hat{x}\|_2 \leq \|x_\star\|_2/2$. Thus, $\|x^k + \hat{x}\|_2 \geq 2\|\hat{x}\|_2 - \|x^k - \hat{x}\|_2 = 2\|x_\star\|_2 - \|x^k - \hat{x}\|_2 \geq 3\|x_\star\|_2/2 \geq \|x^k - \hat{x}\|_2$, which means that x^k is closer to $\hat{x}$ compared to $-\hat{x}$ so that $\Delta(x^k) = \|x^k - \hat{x}\|_2$. To conclude,

$$\Delta(x^k) = \|x^k - \hat{x}\|_2 \leq \Delta(x^0) \leq \|x_\star\|_2/2. \tag{32}$$

Term 2: $\|\xi^k\|$ For any $\epsilon > 0$ and arbitrary $\xi^k \in \partial F(x^k)$ such that $\xi^k \neq \mathbf{0}$, since x^k is an interior point of $S_\epsilon^k \triangleq \{x \in \mathbb{R}^n : \Delta(x) \leq \Delta(x^k) + \epsilon\}$, there exists $t_0 > 0$ such that $x^k + t\xi^k \in S_\epsilon^k$ for $t \in [0, t_0]$. In (20), setting $x = \tilde{x}^k \triangleq x^k + t\xi^k$ for some $t \in (0, t_0]$, $\bar{x} = x^k$ and $v = \xi^k$, we get $F(\tilde{x}^k) - F(x^k) + Lt^2\|\xi^k\|_2^2/2 \geq t\|\xi^k\|_2^2$. Thus, noticing that $\|\tilde{x}^k - x^k\|_2 = t\|\xi^k\|_2$, we can obtain $\|\xi^k\|_2 \leq \frac{F(\tilde{x}^k) - F(x^k)}{\|\tilde{x}^k - x^k\|_2} + Lt\|\xi^k\|_2/2$. Moreover, $\frac{F(\tilde{x}^k) - F(x^k)}{\|\tilde{x}^k - x^k\|_2} \leq \sup\left\{\frac{|F(x) - F(y)|}{\|x-y\|_2} : x, y \in S_\epsilon^k,\ x \neq y\right\}$ since $\tilde{x}^k, x^k \in S_\epsilon^k$. Therefore, letting $t \to 0$, we have that

$$\|\xi^k\|_2 \leq \sup_{x,y \in S_\epsilon^k,\ x \neq y} \frac{|F(x) - F(y)|}{\|x - y\|_2} \triangleq B_\epsilon.$$

By Lemma 2.1 and (32), we have that

$$\|\xi^k\|_2 \leq \lim_{\epsilon \to 0} B_\epsilon = \lim_{r \to \Delta(x^k)} L(\|x_\star\|_2 + r) = L(\|x_\star\|_2 + \Delta(x^k)) \leq \frac{3}{2}L\|x_\star\|_2. \tag{33}$$

Since (33) trivially holds when $\xi^k = \mathbf{0}$, we have $\|\xi^k\|_2 \leq \frac{3}{2}L\|x_\star\|_2$ for all $\xi_k \in \partial F(x^k)$.

Term 3: $L\left\|x^k - \hat{x}\right\|_2^2 - (2 - e_k)\left(F(x^k) - F(\hat{x})\right)$ Due to (32) and (19) and $e_k \in (0,2)$, we have $(2 - e_k)(F(x^k) - F(\hat{x})) \geq \lambda_s(2 - e_k)\|x^k - \hat{x}\|_2$. Thus,

$$L\left\|x^k - \hat{x}\right\|_2^2 - (2 - e_k)\left(F(x^k) - F(\hat{x})\right) \leq \|x^k - \hat{x}\|_2\Big(L\|x^k - \hat{x}\|_2 - \lambda_s(2 - e_k)\Big).$$

Recall that by the hypothesis, x^0 satisfies $\Delta(x^0) \leq \lambda_s(1 - \frac{c_2}{2})/L$; therefore, it follows from (32) and $e_k \in [c_1, c_2] \subset (0,2)$ that $L\|x^k - \hat{x}\|_2 \leq L\Delta(x^0) \leq \lambda_s(2 - c_2)/2 \leq \lambda_s(2 - e_k)/2$. Thus, we get $\|x^k - \hat{x}\|_2(L\|x^k - \hat{x}\|_2 - \lambda_s(2 - e_k)) \leq -\lambda_s(2 - e_k)\|x^k - \hat{x}\|_2/2$. To conclude, we have

$$L\left\|x^k - \hat{x}\right\|_2^2 - (2 - e_k)\left(F(x^k) - F(\hat{x})\right) \leq -\lambda_s(2 - e_k)\|x^k - \hat{x}\|_2/2. \tag{34}$$

Thus, using (33) and (34) within (31), we have

$$\|x^{k+1} - \hat{x}\|_2^2 \leq \|x^k - \hat{x}\|_2^2 - \frac{2\lambda_s e^k(2 - e^k)}{9L^2\|x_\star\|_2^2}(F(x^k) - F(\hat{x}))\|x^k - \hat{x}\|_2.$$

From (19), we have $F(x^k) - F(\hat{x}) \geq \lambda_s \|x^k - \hat{x}\|_2$, which combined with the above inequality implies

$$\|x^{k+1} - \hat{x}\|_2 \leq \sqrt{1 - \frac{2\lambda_s^2 e^k(2 - e^k)}{9L^2\|x_\star\|_2^2}} \|x^k - \hat{x}\|_2, \quad \forall k \in \mathbb{N}. \tag{35}$$

Using $e^k(2 - e^k) \geq 1 - c^2$ and the inductive hypothesis, i.e., $\|x^k - \hat{x}\|_2 \leq R_k$, within this inequality implies that $\|x^{k+1} - \hat{x}\|_2 \leq R_{k+1}$ completing the induction, which also implies (29) since $\Delta(x^k) \leq \|x^k - \hat{x}\|_2$ for all $k \geq 0$. $\square$

Remark 5.2. The initialization requirement $\Delta(x^0) \leq \lambda_s(1 - c_2/2)/L$ in Theorem 5.1 can be satisfied by [11, Algorithm 3] with high probability when m/n is large enough and p_{fail} is small enough. In fact, under some regularity assumptions, [11, Theorem 3] claims that for any $C_{\text{init}} > 0$, x^0 returned by Algorithm 3 in [11] satisfies $\Delta(x^0) \leq C_{\text{init}}$ when $p_{\text{fail}} < 1/4$ and $m/n \geq \Omega((\|x_\star\|_2/C_{\text{init}})^2)$.

Theorem 5.1 shows that Algorithm 1 takes $\mathcal{O}(\frac{\kappa_0^2}{1-c^2}\log\frac{1}{\epsilon})$ iterations to find an ϵ-optimal solution for any $\epsilon \in (0, \Delta(x^0)$. According to Corollary 3.5(a), this complexity result is guaranteed to hold with high probability when the parameter $G > 0$ appearing in the step-size choice in (10) is sufficiently small such that $G < 2/u_H = \frac{2}{3}(\kappa_{\text{st}}/\|\Sigma\|_2 - 2p_{\text{fail}})(1 - \frac{\tilde{p}}{1-p_{\text{fail}}}) \approx \frac{2}{3}\kappa_0(1 - \frac{\tilde{p}}{1-p_{\text{fail}}})$ when m/n large and p_{fail} sufficiently small –see (21). The factor $1 - c^2$ represents how well our adaptive step size α_k in (10) approximates $F(x^k) - F(x_\star)$, i.e., $1 - c^2$ measures how close `AdaSubGrad` update rule in (11) mimicks `PSubGrad`; indeed, in the ideal situation where $c = 0$, which might be impossible in practice, `PSubGrad` and `AdaSubGrad` coincide. When G is overly small such that $c_2 = u_H G \leq 1$, we have $c = 1 - c_1$ since $0 < c_1 \leq c_2 < 1$; hence, $1 - c^2 = c_1(2 - c_1) \geq c_1$ and the required iterations is $\mathcal{O}(\frac{\kappa_0^2}{c_1}\log\frac{1}{\epsilon})$. Since $c_1 = u_L G$, the complexity bound in terms of G and ϵ becomes $\mathcal{O}(\frac{1}{G}\log\frac{1}{\epsilon})$.

6 Convergence Rate of `AdaIPL`

In this section, we analyze the convergence properties of `AdaIPL` stated in Algorithm 2.

6.1 Convergence Behaviours for Outer Iterations

In this subsection, we analyze a prototype of `AdaIPL` where we do not impose t_k being selected based on (12); instead, we establish convergence of `AdaIPL` for $\{t_k\}$ satisfying a more general condition. More precisely, in the result below, we provide a one-step analysis of `AdaIPL` when $t_k \in (0, L^{-1}]$ and $\Delta(x^k)$ is small.

Lemma 6.1. *Under Assumption 2.4, if x^k satisfies $\Delta(x^k) \leq \lambda_s/(4L)$ and $t_k \in (0, L^{-1}]$ for some $k \geq 0$, then the following conclusions hold for Algorithm 2.*

*(a) When **(LAC)** in* (16) *holds with $\rho_l \geq 0$, we have that*

$$F(x^{k+1}) - F(x_\star) \leq \left(1 - \frac{5\min\{\lambda_s t_k/(2\Delta(x^k)), 1\}}{8(1+\rho_l)}\right)(F(x^k) - F(x_\star)).$$

*(b) When **(HAC)** in* (16) *holds with $\rho_h \in [0, 1/4)$, we have that*

$$F(x^{k+1}) - F(x_\star) \leq \left(1 - \frac{5\min\{\lambda_s t_k/(2\Delta(x^k)), 1\}}{8(1+\frac{2\rho_h}{1-4\rho_h})}\right)(F(x^k) - F(x_\star)).$$

Lemma 6.1 indicates that in `AdaIPL` with `Cond` stated in Algorithm 2 set to either (**LAC**) or (**HAC**) in (16), if $\Delta(x^0)$ is small enough, the number of main iterations required for computing an $\epsilon-$optimal point is $\mathcal{O}(\log(1/\epsilon))$. Specifically, consider the t_k selection rule in (12), when $G > 0$ is chosen sufficiently *small* so that $g_H \triangleq 3GL\|x_\star\|_2 u_H/2 \leq 2/\lambda_s$, according to part (b) of Corollary 3.5 we have $t_k = g_k\Delta(x^k) \leq g_H\Delta(x^k)$ for all $k \geq 0$ with high probability. Note that $\lambda_s t_k/(2\Delta(x^k)) = \lambda_s g_k/2 \leq \lambda_s g_H/2 \leq 1$; therefore, the contraction rate is smaller than $1 - \frac{5\lambda_s g_L}{16}(1+\rho_l)^{-1}$ and $1 - \frac{5\lambda_s g_L}{16}(1 + \frac{2\rho_h}{1-4\rho_h})^{-1}$ for **(LAC)** and **(HAC)**, respectively, where we used the fact that $g_k \geq g_L$ for all $k \geq 0$ with high probability (see part (b) of Corollary 3.5). Hence, the number of main iterations for both **(LAC)** and (**HAC**) is $\mathcal{O}(\frac{1}{\lambda_s g_L}\log\frac{1}{\epsilon})$; and in terms of ϵ and G, we get $\mathcal{O}(\frac{1}{G}\log(\frac{1}{\epsilon}))$. On the other hand, when $G > 0$ is chosen sufficiently *large* so that $g_L \triangleq G\lambda_s u_L > 2/\lambda_s$, the number of main iterations for both **(LAC)** and **(HAC)** will become $\mathcal{O}(\log\frac{1}{\epsilon})$ without G explicitly appearing in the denominator. Indeed, when $g_L > 2/\lambda_s$, the discussion above implies that $\lambda_s t_k/(2\Delta(x^k)) \geq 1$ for all $k \geq 0$; therefore, the contraction rate is smaller than $1 - \frac{5}{8}(1+\rho_l)^{-1}$ and $1 - \frac{5}{8}(1 + \frac{2\rho_h}{1-4\rho_h})^{-1}$ for **(LAC)** and **(HAC)**, respectively. This implies that the $\mathcal{O}(1)$ constant for the outer iteration complexity does not explicitly depend on t_k; hence, G does not appear in the bound. That said, the above complexity bound for **(HAC)** case is not tight. Indeed, for **(HAC)** whenever G is selected *large* so that $g_L \geq 2/\lambda_s$, we next establish in Lemma 6.2 that Algorithm 2 has a better complexity bound than $\mathcal{O}(\log(\frac{1}{\epsilon}))$ implied by Lemma 6.1(b). In the result below, we provide one-step analysis of `AdaIPL` when $t_k \in (2\Delta(x^k)/\lambda_s, L^{-1}]$ and $\Delta(x^k)$ is small – for this result, similar to the above discussion, we also do not assume t_k being selected based on (12).

Lemma 6.2. *Under Assumption 2.4, for any $k \in \mathbb{N}$, if **(HAC)** in (16) holds with $\rho_h \in [0, 1/4)$ and $\Delta(x^k) \leq \min\{\|x_\star\|_2/\sqrt{M_0}, \lambda_s/(4L)\}$, then*

$$\lambda_s \Delta(x^{k+1}) \leq (4\rho_h/t_k + 4L)\, \Delta^2(x^k), \quad \forall\, t_k \in [2\Delta(x^k)/\lambda_s,\ L^{-1}],$$

where $M_0 \triangleq \left(2 + \sqrt{2}\rho_h^{3/4}/(1-2\rho_h^{1/2})\right)/(1-\sqrt{2}\rho_h^{1/4})$.

Lemma 6.2 shows that when $\Delta(x^0)$ is small enough, `AdaIPL` with `Cond` set to **(HAC)** in (16) shows linear or super-linear convergence in terms of main iterations. Specifically, consider the t_k selection rule in (12), when $G > 0$ is chosen sufficiently *large* such that $g_L > 2/\lambda_s$, the number of main iterations is $\mathcal{O}\Big(\log(\frac{1}{\epsilon})/\log(g_L\lambda_s)\Big)$; thus, in terms of ϵ and G, we get $\mathcal{O}(\log(\frac{1}{\epsilon})/\log(G))$. In the extreme situation with $G = \infty$, we get $t_k = L^{-1}$ and choosing constant step size $1/L$ corresponds to the `IPL` algorithm in [14], of which convergence rate in terms of main iterations (k) is quadratic.

6.2 Subproblem Solvers and Computational Complexity

In this subsection, we introduce a class of algorithms for *inexactly* solving the `AdaIPL` subproblem in (6) and their computational complexity. We consider two alternative solvers that can guarantee the suboptimality $H_k(z_j^k) - D_k(\lambda_j^k) = \mathcal{O}(1/j^2)$ for all $k \geq 0$. The first one is [15, Algorithm 1] and throughout this paper, we refer to it as the Accelerated Proximal Gradient (`APG`) method. The second one is the Accelerated Primal-Dual (`APD`) algorithm introduced in [25, Algorithm 4] and [26, Algorithm 2.2]. In the following result, we state the complexity result for both methods when applied to the k-th `AdaIPL` subproblem in (13).

Theorem 6.3 (Corollary 1(b) in [15] and Theorem 2.2 in [26])**.** *For any $k \geq 0$, consider either `APG` or `APD` applied to* (13)*. Let $\{(z_j^k, \lambda_j^k)\}_{j=0}^\infty$ be the iterate sequence generated. Then, $\sup_{j\in\mathbb{N}}\|\lambda_j^k\|_\infty \leq 1$ and there exists a constant*[1] *$C_0 \geq 2$ such that*

$$H_k(z_j^k) - D_k(\lambda_j^k) \leq \frac{t_k C_0 m \|B_k\|_2^2}{(j+1)^2}, \quad \forall j \in \mathbb{N}_+. \tag{36}$$

To the best of our knowledge, for solving (13) when B_k matrix is arbitrary, $\mathcal{O}(1/j^2)$ is the best rate we can get on the duality gap of the primal-dual iterate sequence.

Next result shows that $\|B_k\|_2$ on the r.h.s. of (36) can be uniformly bounded.

Lemma 6.4 (Lemma 8 in [14])**.** *If $\sup_{k\in\mathbb{N}} \Delta(x^k) \leq r$, then*

$$\sup_{k\in\mathbb{N}} \|B_k\|_2 \leq B(r) \triangleq \frac{2}{m}\|A\|_2(\|x_\star\|_2 + r) \max_{i\in[m]} \|a_i\|_2.$$

In the rest, we discuss the complexity of the `AdaIPL` when (13) is solved inexactly by some algorithm that can guarantee (36) and terminated according to (16). For any given $\epsilon > 0$, let K_ϵ denote the number of main (outer) iterations required by `AdaIPL` to compute an ϵ-optimal solution, i.e., $K_\epsilon \triangleq \inf\{k \in \mathbb{N}_+ : \Delta(x^k) \leq \epsilon\}$. Given x^k for any $k \geq 0$, let $N_k \in \mathbb{N}_+$ denote the number of iterations required by the solver, i.e., inner iterations of `AdaIPL`, to inexactly solve the k-th subproblem –in order to compute x^{k+1} within the k-th outer iteration of `AdaIPL`. For **(LAC)** in (16), $N_k = \inf\{j \in \mathbb{N} : H_k(z_j^k) - D_k(\lambda_j^k) \leq \rho_l(H_k(\mathbf{0}) - H_k(z_j^k))\}$. For **(HAC)** in (16), $N_k = \inf\{j \in \mathbb{N} : H_k(z_j^k) - D_k(\lambda_j^k) \leq \rho_h\|z_j^k\|_2^2/(2t_k)\}$. Therefore, the overall complexity of `AdaIPL` for computing an ϵ-optimal solution is thus given by $N(\epsilon) \triangleq \sum_{k=0}^{K_\epsilon - 1} N_k$.

Next, we provide an upper bound for N_k; similar to Lemmas 6.1 and 6.2, we do not assume t_k being selected based on (12), instead we show the result for any $t_k \in (0, 1/L]$. First, in Lemma 6.5, we give a bound under **(LAC)** in (16).

[1] C_0 is dimension-free and does not depend on any problem or algorithm parameters.

Lemma 6.5 (N_k bound under **(LAC)**). *Suppose Assumption 2.4 holds and a subproblem solver satisfying and* (36) *for some constant $C_0 \geq 2$ is given. For any $k \in \mathbb{N}$, if $\Delta(x^k) \leq \lambda_s/(4L)$ and $0 < t_k \leq L^{-1}$, then for any $\rho_l > 0$,* ***(LAC)*** *in* (16) *holds within*

$$N_k \leq M_1 \sqrt{\frac{t_k}{\lambda_s \Delta(x^k) \min\{1, \lambda_s t_k/(2\Delta(x^k))\}}} \cdot \sqrt{\frac{1+\rho_l}{\rho_l}} \triangleq M_k^{\mathbf{LAC}},$$

inner iterations in which $M_1 \triangleq \sqrt{1.6 C_0 m B^2(\lambda_s/(4L))}$ and $B(\cdot)$ is defined in Lemma 6.4.

Second, in Lemma 6.6, we give another bound considering **(HAC)** in (16).

Lemma 6.6 (N_k bound under **(HAC)**). *Under the premise of Lemma 6.5, for any $\rho_h \in (0, 1/4)$,* ***(HAC)*** *in* (16) *holds within*

$$N_k \leq M_2 \frac{t_k}{\Delta(x^k) \min\{1, \lambda_s t_k/(2\Delta(x^k))\}} \sqrt{\frac{1+\rho_h}{\rho_h}} \triangleq M_k^{\mathbf{HAC}},$$

inner iterations in which $M_2 \triangleq \sqrt{16 C_0 m B^2(\lambda_s/(4L))}$.

These two results indicate that for `AdaIPL` with t_k selected as in (12), if $\Delta(x^0)$ is small enough, then the number of inner iterations needed to satisfy (16) is $\mathcal{O}(1)$. More precisely, it follows from the discussion below Lemma 6.1 that when $G > 0$ is chosen sufficiently *small* so that $g_H \triangleq 3GL\|x_\star\|_2 u_H/2 \leq 2/\lambda_s$, we would have $\lambda_s t_k/(2\Delta(x^k)) \leq \lambda_s g_H/2 \leq 1$ holding for all $k \geq 0$ with high probability. Thus, for both **(LAC)** and **(HAC)** in (16), we get $N_k{=}\mathcal{O}(\sqrt{m}B(\lambda_s/(4L))/\lambda_s)$. According to Lemma 2.6, $\lambda_s/(4L) \leq \|x_\star\|_2/8$; therefore, using the definition of $B(\cdot)$ in Lemma 6.4,

$$B\Big(\frac{\lambda_s}{4L}\Big) = \frac{2}{m}\|A\|_2 \max_{i\in[m]}\|a_i\|_2 \Big(\|x_\star\|_2 + \frac{\lambda_s}{4L}\Big) {=} \mathcal{O}(C_S\|x_\star\|_2\|A\|_2^2 m^{-3/2}), \tag{37}$$

where $C_S \triangleq \sqrt{m}\max_{i\in[m]}\|a_i\|_2/\|A\|_2$. Here, C_S is a factor related to the complexity of solving the subproblem in (13), i.e., equivalently (6) –see Theorem 6.3 and Lemma 6.4. Note $C_S \geq 1$ as $\max_{i\in[m]}\|a_i\|_2^2 \geq \frac{1}{m}\sum_{i=1}^m \|a_i\|_2^2 = \|A\|_F^2/m \geq \|A\|_2^2/m$. Thus, using $L \triangleq 2\|A\|_2^2/m$, we get

$$N_k {=} \mathcal{O}(C_S\|x_\star\|_2\|A\|_2^2/(m\lambda_s)) = \mathcal{O}(C_S \kappa_0),$$

in which $\kappa_0 \triangleq L\|x_\star\|_2/(2\lambda_s) \geq 1$ is the condition number for (4) introduced in Section 3 and Lemma 2.6. It is crucial to note that this upper bound holds for all sufficiently small G and does not increase as G chosen smaller.

On the other hand, when $G > 0$ is chosen sufficiently *large* so that $g_L \triangleq G\lambda_s u_L > 2/\lambda_s$, then according to part (b) of Corollary 3.5, with high probability we have $\lambda_s t_k/(2\Delta(x^k)) = \lambda_s g_k/2 \geq 1$ (since $g_k \geq g_L$) for all $k \geq 0$. Thus, Lemma 6.5 and Lemma 6.6 together with (37) imply that

$$N_k \leq \begin{cases} M_1 \sqrt{\frac{t_k}{\lambda_s \Delta(x^k)}} \cdot \sqrt{\frac{1+\rho_l}{\rho_l}} {=} \mathcal{O}(C_S \kappa_0 \sqrt{g_H \lambda_s}) & \text{for } \textbf{(LAC)}, \\ M_2 \frac{t_k}{\Delta(x^k)} \sqrt{\frac{1+\rho_h}{\rho_h}} {=} \mathcal{O}(C_S \kappa_0 g_H \lambda_s) & \text{for } \textbf{(HAC)}. \end{cases}$$

Since $g_H \triangleq 3GL\|x_\star\|_2 u_H/2$, in terms of the dependency on G, we have $N_k = \mathcal{O}(\sqrt{G})$ under **(LAC)** in (16), and $N_k = \mathcal{O}(G)$ under **(HAC)** in (16).

6.3 Overall Complexity

Finally, we are ready to provide the overall iteration complexity for `AdaIPL`. Indeed, in Theorem 6.7 we establish a bound on $N(\epsilon)$ for *any* given $G > 0$ when t_k is chosen according to (12).

Theorem 6.7. *Suppose Assumption 2.4 holds and a subproblem solver satisfying and* (36) *is given. Moreover, we assume that the event in* (22) *holds. If $\Delta(x^0) \leq \min\{E(\frac{\lambda_s^2}{4L}), E(\frac{\lambda_s}{g_H L})\}$, where $E(r) \triangleq \frac{1}{2L}\left(\sqrt{L^2\|x_\star\|_2^2 + 4rL} - L\|x_\star\|_2\right)$ defined in Lemma 2.5 and $g_H \geq g_L > 0$ are constants defined in Corollary 3.5, then the following conclusions hold with $C(\rho) \triangleq 1 - \frac{5\min\{\lambda_s g_L,\, 2\}}{16(1+\rho)}$.*

(a) When `Cond` in `AdaIPL`, stated in Algorithm 2, is set to ***(LAC)*** *in* (16) *for some $\rho_l > 0$, for any $\epsilon \in (0, \Delta(x^0))$, it holds that*

$$N(\epsilon) \leq M_1 \cdot \frac{\sqrt{\frac{g_H}{\lambda_s \min\{1, \lambda_s g_H/2\}}} \cdot \sqrt{\frac{1+\rho_l}{\rho_l}} \cdot \log\left(\frac{2\Delta(x^0)L\|x_\star\|_2}{\lambda_s C(\rho_l)} \cdot \frac{1}{\epsilon}\right)}{\log(1/C(\rho_l))}.$$

(b) When `Cond` *in* `AdaIPL`*, stated in Algorithm 2, is set to* ***(HAC)*** *in* (16) *for some* $\rho_h \in (0, 1/4)$*, for any* $\epsilon \in (0, \Delta(x^0))$*, it holds for* $\bar{\rho} \triangleq 2\rho_h/(1-4\rho_h)$ *that*

$$N(\epsilon) \le M_2 \cdot \frac{\frac{g_H}{\min\{1, \lambda_s g_H/2\}}\sqrt{\frac{1+\rho_h}{\rho_h}} \cdot \log\left(\frac{2\Delta(x^0)L\|x_\star\|_2}{\lambda_s C(\bar{\rho})} \cdot \frac{1}{\epsilon}\right)}{\log\left(1/C(\bar{\rho})\right)}.$$

Next, we discuss that whenever $\Delta(x^0)$ is sufficiently small and G in (12) is sufficiently large, using **(HAC)** in (16) leads to a better complexity bound when compared to that of Theorem 6.7(b).

Theorem 6.8. *Suppose Assumption 2.4 holds and a subproblem solver satisfying and* (36) *is given. Moreover, we assume that the probabilistic event in* (22) *holds. If constants defined in Corollary 3.5 satisfy* $g_H \ge g_L \ge 2/\lambda_s$*,* `Cond` *in* `AdaIPL`*, stated in Algorithm 2, is set to* ***(HAC)*** *in* (16) *for some* $\rho_h \in (0, 1/4)$*, and* $\Delta(x^0) \le \min\left\{\frac{\rho_h}{2Lg_L}, \|x_\star\|_2/\sqrt{M_0}, (g_H L)^{-1}\right\}$*, then for any* $\epsilon \in (0, \Delta(x^0))$*, it holds that*

$$N(\epsilon) \le M_2 \cdot \frac{g_H \cdot \sqrt{\frac{1+\rho_h}{\rho_h}} \cdot \log\left(\frac{\lambda_s \Delta(x^0) g_L}{6\rho_h} \cdot \frac{1}{\epsilon}\right)}{\log\left(\lambda_s g_L/(6\rho_h)\right)}$$

Remark 6.9. Initialization requirements on $\Delta(x^0)$ in both Theorem 6.7 and Theorem 6.8 can be satisfied by Algorithm 3 in [11] with high probability when m/n is large enough and p_{fail} is small enough. Please refer to Remark 5.2 for more explanations.

Next, we explain the results in Theorems 6.7 and 6.8. Indeed, when $\Delta(x^0)$ is sufficiently small, the total complexity for reaching an $\epsilon-$optimal point is $\mathcal{O}(\log\frac{1}{\epsilon})$ for any $G > 0$ and $\epsilon < \Delta(x^0)$. Specifically, when $G > 0$ is chosen sufficiently *small* so that $g_H \le 2/\lambda_s$, we have $\min\{1, \lambda_s g_H/2\} = \lambda_s g_H/2$ and $\min\{1, \lambda_s g_L/2\} = \lambda_s g_L/2$. Therefore, Theorem 6.7 implies that $N(\epsilon) = \mathcal{O}(\frac{C_S\kappa_0}{g_L\lambda_s}\log\frac{1}{\epsilon})$ for both **(LAC)** and **(HAC)** in (16). Moreover, since $g_L = G\lambda_s u_L$, the complexity bound is $\mathcal{O}\left(\frac{1}{G}\log(\frac{1}{\epsilon})\right)$ in terms of G and ϵ. On the other hand, when $G > 0$ is chosen sufficiently *large* so that $g_L > 2/\lambda_s$, we have $\min\{1, \lambda_s g_H/2\} = \min\{1, \lambda_s g_L/2\} = 1$. Thus, Theorem 6.7(a) implies that $N(\epsilon) = \mathcal{O}(C_S\kappa_0\sqrt{g_H\lambda_s}\log\frac{1}{\epsilon})$ for **(LAC)** in (16) and $N(\epsilon) = \mathcal{O}(\frac{C_S\kappa_0 g_H\lambda_s}{\log(g_L\lambda_s)}\log\frac{1}{\epsilon})$ for **(HAC)** in (16). Knowing that $g_L \triangleq G\lambda_s u_L$ and $g_H \triangleq 3GL\|x_\star\|_2 u_H/2$, the complexity depends on G and ϵ as $\mathcal{O}(\sqrt{G}\log\frac{1}{\epsilon})$ for **(LAC)** in (16) and as $\mathcal{O}(\frac{G}{\log G}\log\frac{1}{\epsilon})$ for **(HAC)** in (16). To conclude, if we were to set $t_k = 2\Delta(x^k)/\lambda_s$ for any $k \in \mathbb{N}$, `AdaIPL` would achieve a total complexity of $\mathcal{O}(C_S\kappa_0\log\frac{1}{\epsilon})$ under both **(LAC)** and **(HAC)**. That said, since setting $g_k = 2/\lambda_s$ for $k \in \mathbb{N}$ may be impractical, the corresponding bound can be interpreted as the ideal total complexity bound.

6.4 Comparison of `AdaSubGrad` and `AdaIPL` complexities

Here we compare the total complexity of `AdaSubGrad` and `AdaIPL` for reaching an $\epsilon-$optimal solution of (4) with the existing deterministic algorithms summarized in Section 1. The total complexity refers to total *inner iteration* numbers for `PL`, `IPL`, and `AdaIPL` and refers to iteration numbers for `PSubGrad`, `GSubGrad`, and `AdaSubGrad`. The comparison is fair as the computational complexity of either a subgradient-type iteration or an inner iteration for solving the subproblem in (6) using either `APG` or `APD` is dominated by matrix-vector multiplications involving matrices of size $m \times n$.

In Table 1 provided in the introduction we summarized the total complexities for all the methods under the ideal situations for each of them. The total complexity for `PL` is unknown because it requires solving (6) *exactly*, which is not practical. `IPL` is only guaranteed to achieve sublinear rate leading to $\mathcal{O}(1/\epsilon)$ complexity. In the ideal scenario, the convergence of `PSubGrad`, `GSubGrad`, and `AdaSubGrad` are all linear leasing to $\mathcal{O}(\kappa_0^2\log\frac{1}{\epsilon})$ complexity. Here, the ideal scenario means that $F(x_\star)$ is known for `PSubGrad`, λ_0 and q are set exactly as in [16, Theorem 5.1] for `GSubGrad` (which is not practical), and $c = 0$ in Theorem 5.1 for `AdaSubGrad`. The total complexity of `AdaIPL` is $\mathcal{O}(C_S\kappa_0\log\frac{1}{\epsilon})$ under the ideal situation that $g_L = g_H = 2/\lambda_s$ in Corollary 3.5(b). It is the best in terms of the condition number but contains an additional factor C_S that results from the iteration complexity associated with inexactly solving the subproblems in (6). Thus, `AdaIPL` is expected to show greater efficiency than `AdaSubGrad` when $\kappa_{\text{st}}/\|\Sigma\|_2 - 2p_{\text{fail}}$ is small (this quantity is discussed in Section 3 and is approximately equal to $1/\kappa_0$), i.e., when the condition number for (4) is *large*.

Next, aiming to compute an ϵ-optimal solution to (4) for a given sufficiently small $\epsilon > 0$, in Table 2, we discuss the robustness of `AdaIPL` and `AdaSubGrad` to the algorithm parameter G in terms of main (outer) iterations (K_ϵ), the largest number of inner iterations per subproblem solve ($\sup_{k\in\mathbb{N}} N_k$), and the total complexity (N_ϵ) –we consider the effects of overly large or overly small choices of the step size parameter $G > 0$ (see the stepsize rules in (10) and (12)). Here, we treat each AdaSubGrad iteration as an outer iteration requiring only one inner iteration for each outer iteration, i.e., $N(\epsilon) = K_\epsilon$. Table 2 highlights the advantages of AdaSubGrad and AdaIPL in hyper-parameter tuning over PSubGrad and GSubGrad: AdaSubGrad tolerates overly small G values and AdaIPL works with any $G > 0$ while PSubGrad and GSubGrad rely on some particular choice of hyper-parameter values to guarantee convergence.

Algorithm	G	K_ϵ	$\sup_{k\in\mathbb{N}} N_k$	$N(\epsilon)$
AdaSubGrad	Large ($c_1 \geq 2$)	diverge	diverge	diverge
AdaSubGrad	Small ($c_2 < 1$)	$\tilde{\mathcal{O}}\left(\kappa_0^2/c_1\right)$	1	$\tilde{\mathcal{O}}\left(\kappa_0^2/c_1\right)$
AdaIPL-LAC	Large ($g_L \geq \frac{2}{\lambda_s}$)	$\tilde{\mathcal{O}}(1)$	$\mathcal{O}(C_S\kappa_0\sqrt{g_H\lambda_s})$	$\tilde{\mathcal{O}}(C_S\kappa_0\sqrt{g_H\lambda_s})$
AdaIPL-HAC	Large ($g_L \geq \frac{2}{\lambda_s}$)	$\tilde{\mathcal{O}}(1/\log(g_L\lambda_s))$	$\mathcal{O}(C_S\kappa_0 g_H\lambda_s)$	$\tilde{\mathcal{O}}\left(\frac{C_S\kappa_0 g_H\lambda_s}{\log(g_L\lambda_s)}\right)$
AdaIPL-LAC	Small ($g_H < \frac{2}{\lambda_s}$)	$\tilde{\mathcal{O}}(1/(g_L\lambda_s))$	$\mathcal{O}(C_S\kappa_0)$	$\tilde{\mathcal{O}}(\frac{C_S\kappa_0}{g_L\lambda_s})$
AdaIPL-HAC	Small ($g_H < \frac{2}{\lambda_s}$)	$\tilde{\mathcal{O}}(1/(g_L\lambda_s))$	$\mathcal{O}(C_S\kappa_0)$	$\tilde{\mathcal{O}}(\frac{C_S\kappa_0}{g_L\lambda_s})$

Table 2: Robustness of *AdaSubGrad* and *AdaIPL* to parameter choice in terms of number of outer iterations (K_ϵ), the maximum number of inner iterations per subproblem solve ($\sup_{k\in\mathbb{N}} N_k$), and the total complexity ($N(\epsilon)$). "*-LAC*" and "*-HAC*" indicate whether (**LAC**) or (**HAC**) is used for AdaIPL. The $\tilde{\mathcal{O}}$ notation here hides $\log\frac{1}{\epsilon}$.

7 Proofs for Section 6

We first introduce some notation. For all $k \geq 0$, let

$$S_{t_k}(x^k) \triangleq \arg\min_{x\in\mathbb{R}^n} F_{t_k}(x; x^k), \quad \varepsilon_{t_k}(x; x^k) \triangleq F_{t_k}(x; x^k) - F_{t_k}(S_{t_k}(x^k); x^k),$$

where $F_t(\cdot;\cdot)$ is defined in (5). Here, the minimizer $S_{t_k}(x^k)$ is unique since $F_{t_k}(\cdot; x^k)$ is strongly convex with modulus $1/t_k$ for all $k \geq 0$. According to Lemma 1.2, **(LAC)** and **(HAC)** in (16) provide sufficient conditions for **(LAC-exact)** and **(HAC-exact)** in (15) to hold, respectively.

7.1 Proof of Lemma 6.1

Within this proof, without loss of generality, we assume that $\Delta(x^k) = \|x^k - x_\star\|_2$. We split the proof into four parts.

7.1.1 (6) is exactly solved and the step size belongs to $[2\Delta(x^k)/\lambda_s, L^{-1}]$

Let $\tilde{x}^{k+1} \triangleq S_{\tilde{t}_k}(x^k)$ for some arbitrary $\tilde{t}_k \in [2\Delta(x^k)/\lambda_s, L^{-1}]$ –the interval is not empty because we assume $\Delta(x^k) \leq \lambda_s/(4L)$ in Lemma 6.1; hence,

$$F(\tilde{x}^{k+1}) \leq F_{\tilde{t}_k}(\tilde{x}^{k+1}; x^k) \leq F(x_\star; x^k) + \frac{1}{2\tilde{t}_k}\|x_\star - x^k\|_2^2 \leq F(x_\star) + \left(\frac{1}{2\tilde{t}_k} + \frac{L}{2}\right)\|x_\star - x^k\|_2^2 \tag{38}$$

in which the first and the third inequalities hold because of (18) and the second one holds because $\tilde{x}^{k+1}$ is the minimizer of $F_{\tilde{t}_k}(\cdot; x^k)$. Due to the fact that $\Delta(x^k) = \|x^k - x_\star\|_2 \leq \lambda_s/(4L)$ and $\tilde{t}_k \geq 2\Delta(x^k)/\lambda_s$, we have

$$\left(\frac{1}{2\tilde{t}_k} + \frac{L}{2}\right)\|x_\star - x^k\|_2^2 \leq \frac{\lambda_s}{4}\Delta(x^k) + \frac{L}{2}\frac{\lambda_s}{4L}\Delta(x^k) = \frac{3}{8}\lambda_s\Delta(x^k);$$

hence, using (19), we get $\left(\frac{1}{2\tilde{t}_k} + \frac{L}{2}\right)\|x_\star - x^k\|_2^2 \leq 3(F(x^k) - F(x_\star))/8$. Applying it to (38), we have

$$F(x^k) - F(\tilde{x}^{k+1}) \geq F(x^k) - F_{\tilde{t}_k}(\tilde{x}^{k+1}; x^k) \geq \frac{5}{8}(F(x^k) - F(x_\star)). \tag{39}$$

7.1.2 (6) is exactly solved and the step size is less than $2\Delta(x^k)/\lambda_s$

Denote $\approx{x}^{k+1}$ $\triangleq S_{\approx{t}_k}(x^k)$ for some $\approx{t}_k \in (0, 2\Delta(x^k)/\lambda_s)$. Consider the result in Section 7.1.1 for $\tilde{t}_k = 2\Delta(x^k)/\lambda_s$ and let $\bar{x}^{k+1} \triangleq x^k + \frac{\approx{t}_k}{\tilde{t}_k}(\tilde{x}^{k+1} - x^k)$. Hence,

$$F_{\tilde{t}_k}(\bar{x}^{k+1}; x^k) \leq \frac{\approx{t}_k}{\tilde{t}_k}F_{\tilde{t}_k}(\tilde{x}^{k+1}; x^k) + \left(1 - \frac{\approx{t}_k}{\tilde{t}_k}\right)F(x^k) - \frac{(\frac{\approx{t}_k}{\tilde{t}_k})(1 - \frac{\approx{t}_k}{\tilde{t}_k})}{2\tilde{t}_k}\|x^k - \tilde{x}^{k+1}\|_2^2,$$

by strong convexity of $F_{\tilde{t}_k}(\cdot; x^k)$. Thus, we have

$$
\begin{aligned}
F(x^k) - F_{\tilde{\tilde{t}}_k}(\bar{x}^{k+1}; x^k) &= F(x^k) - F_{\tilde{t}_k}(\bar{x}^{k+1}; x^k) + \Big(\frac{1}{2\tilde{t}_k} - \frac{1}{2\tilde{\tilde{t}}_k}\Big)\|\bar{x}^{k+1} - x^k\|_2^2 \\
&\geq \frac{\tilde{\tilde{t}}_k}{\tilde{t}_k}(F(x^k) - F_{\tilde{t}_k}(\tilde{x}^{k+1}; x^k)) + \frac{(\frac{\tilde{\tilde{t}}_k}{\tilde{t}_k})(1 - \frac{\tilde{\tilde{t}}_k}{\tilde{t}_k})}{2\tilde{t}_k}\|x^k - \tilde{x}^{k+1}\|_2^2 + \Big(\frac{1}{2\tilde{t}_k} - \frac{1}{2\tilde{\tilde{t}}_k}\Big)\|\bar{x}^{k+1} - x^k\|_2^2 \\
&= \frac{\tilde{\tilde{t}}_k}{\tilde{t}_k}(F(x^k) - F_{\tilde{t}_k}(\tilde{x}^{k+1}; x^k)) \geq \frac{5\tilde{\tilde{t}}_k}{8\tilde{t}_k}(F(x^k) - F(x_\star)),
\end{aligned}
$$

where in the last inequality we used the last inequality in (39). In addition, since $\tilde{\tilde{x}}^{k+1}$ is the minimizer of $F_{\tilde{\tilde{t}}_k}(\cdot; x^k)$, we have $F_{\tilde{\tilde{t}}_k}(\tilde{\tilde{x}}^{k+1}; x^k) \leq F_{\tilde{\tilde{t}}_k}(\bar{x}^{k+1}; x^k)$ which implies $F(x^k) - F_{\tilde{\tilde{t}}_k}(\tilde{\tilde{x}}^{k+1}; x^k) \geq \frac{5\tilde{\tilde{t}}_k}{8\tilde{t}_k}(F(x^k) - F(x_\star))$; therefore,

$$
F(x^k) - F(\tilde{\tilde{x}}^{k+1}) \geq F(x^k) - F_{\tilde{\tilde{t}}_k}(\tilde{\tilde{x}}^{k+1}; x^k) \geq \frac{5\tilde{\tilde{t}}_k}{8\tilde{t}_k}(F(x^k) - F(x_\star)). \tag{40}
$$

Here, the first inequality is from (18). This finishes part 2.

7.1.3 Proof of part (a)

Combining (39) and (40), we can claim that, for any $t_k \in (0, L^{-1}]$, whenever $\Delta(x^k) \leq \lambda_s/(4L)$ holds, one also has

$$
F(x^k) - F_{t_k}(S_{t_k}(x^k); x^k) \geq \frac{5\min\{\lambda_s t_k/(2\Delta(x^k)), 1\}}{8}(F(x^k) - F(x_\star)). \tag{41}
$$

Furthermore, we have $F(x^k) - F(x^{k+1}) \geq F(x^k) - F_{t_k}(x^{k+1}; x^k) \geq \frac{1}{1+\rho_l}(F(x^k) - F_{t_k}(S_{t_k}(x^k); x^k))$, where the first inequality is due to (18), and the second one originates from **(LAC-exact)** in (7), which is implied by **(LAC)**. Finally, combining the last inequality with (41), we complete the proof.

7.1.4 Proof of part (b)

For part (b), we first establish some connections between **(LAC-exact)** and **(HAC-exact)** in (7). We know that

$$
\begin{aligned}
&F_{t_k}(x^k; x^k) - F_{t_k}(x^{k+1}; x^k) + \varepsilon_{t_k}(x^{k+1}; x^k) = F_{t_k}(x^k; x^k) - F_{t_k}(S_{t_k}(x^k); x^k) \\
&\geq \frac{1}{2t_k}\|x^k - S_{t_k}(x^k)\|_2^2 \geq \frac{1}{4t_k}\|x^k - x^{k+1}\|_2^2 - \frac{1}{2t_k}\|x^{k+1} - S_{t_k}(x^k)\|_2^2.
\end{aligned}
$$

where the first inequality is due to the strong convexity of $F_{t_k}(\cdot; x^k)$, and the second one is from Cauchy-Schwarz inequality. Using the fact that $\frac{1}{2t_k}\|x^{k+1} - S_{t_k}(x^k)\|_2^2 \leq \varepsilon_{t_k}(x^{k+1}; x^k)$ due to strong convexity of $F_{t_k}(\cdot; x^k)$, we further obtain

$$
F_{t_k}(x^k; x^k) - F_{t_k}(x^{k+1}; x^k) + 2\varepsilon_{t_k}(x^{k+1}; x^k) \geq \frac{1}{4t_k}\|x^k - x^{k+1}\|_2^2. \tag{42}
$$

Note that the inexact subproblem termination condition **(HAC)** implies **(HAC-exact)** in (7), which further implies that $\varepsilon_{t_k}(x^{k+1}; x^k) \leq \rho_h\|x^k - x^{k+1}\|_2^2/(2t_k)$; therefore, it follows from (42) that

$$
\frac{2\rho_h}{1 - 4\rho_h}\left(F_{t_k}(x^k; x^k) - F_{t_k}(x^{k+1}; x^k)\right) \geq \frac{\rho_h}{2t_k}\|x^k - x^{k+1}\|_2^2 \geq \varepsilon_{t_k}(x^{k+1}; x^k). \tag{43}
$$

This takes the same form as **(LAC-exact)** in (7) when $\rho_l = \frac{2\rho_h}{1-4\rho_h}$. Therefore, it directly follows from the arguments in the proof of part (a) that one can simply replace ρ_l in (a) with $\frac{2\rho_h}{1-4\rho_h}$.

7.2 Proof of Lemma 6.2

We first show an auxiliary result for one-step behavior of `AdaIPL`.

Lemma 7.1. *Given $\beta \in (0, 1]$ and arbitrary $t_k > 0$, for any* `AdaIPL` *iterate x^k,*

$$
F(x) - F(x_\star) + \frac{1-\beta}{2t_k}\|x - x_\star\|_2^2 \leq \left(\frac{1}{2t_k} + \frac{L}{2}\right)\|x^k - x_\star\|_2^2 + \beta^{-1}\varepsilon_{t_k}(x, x^k) + \left(\frac{L}{2} - \frac{1}{2t_k}\right)\|x - x^k\|_2^2, \tag{44}
$$

for all $x \in \mathbb{R}^n$. The above inequality is also valid if we replace $x_\star$ with $-x_\star$ in (44).

Proof. Given arbitrary $\beta \in (0,1]$ and any $t_k > 0$, then for any $x \in \mathbb{R}^n$,

$$
\begin{aligned}
-\frac{1}{2t_k}\|x_\star - S_{t_k}(x^k)\|_2^2 &\leq \frac{\beta-1}{2t_k}\|x - x_\star\|_2^2 + \frac{\beta^{-1}-1}{2t_k}\|x - S_{t_k}(x^k)\|_2^2 \\
&\leq \frac{\beta-1}{2t_k}\|x - x_\star\|_2^2 + (\beta^{-1}-1)\varepsilon_{t_k}(x;x^k),
\end{aligned}
\tag{45}
$$

in which the first inequality holds because of Cauchy–Schwarz inequality, and the second inequality holds since $F_{t_k}(\cdot;x^k)$ is $1/t_k-$strongly convex. We then have

$$
\begin{aligned}
F(x) - \varepsilon_{t_k}(x;x^k) &\leq F_{t_k}(S_{t_k}(x^k);x^k) + \left(\frac{L}{2} - \frac{1}{2t_k}\right)\|x - x^k\|_2^2 \\
&\leq F(x_\star;x^k) + \frac{1}{2t_k}\|x^k - x_\star\|_2^2 - \frac{1}{2t_k}\|x_\star - S_{t_k}(x^k)\|_2^2 + \frac{Lt_k-1}{2t_k}\|x - x^k\|_2^2 \\
&\leq F(x_\star) + \frac{Lt_k+1}{2t_k}\|x^k - x_\star\|_2^2 - \frac{1}{2t_k}\|x_\star - S_{t_k}(x^k)\|_2^2 + \frac{Lt_k-1}{2t_k}\|x - x^k\|_2^2,
\end{aligned}
$$

where the first and the last inequalities use (18), and the second inequality is from the strong convexity of $F_{t_k}(\cdot;x^k)$. Combining this inequality with (45) yields (44). It is easy to verify that the proof still holds when we replace $x_\star$ with $-x_\star$. □

Next, we provide another useful result to be used in the proof of Lemma 6.2.

Lemma 7.2. *Suppose Assumption 2.4 holds. For any $k \in \mathbb{N}$ with $t_k \in (0, 1/L]$, if* ***(HAC-exact)*** *in* (7) *holds with $\rho_h \in [0, 1/4)$ and $\Delta(x^k) \leq \|x_\star\|_2/\sqrt{M_0}$, then*

$$
\mathbf{1}\left[\Delta(x^k) = \|x^k - x_\star\|_2\right] = \mathbf{1}\left[\Delta(x^{k+1}) = \|x^{k+1} - x_\star\|_2\right],
\tag{46}
$$

where $M_0 \triangleq \left(2 + \sqrt{2}\rho_h^{3/4}/(1 - 2\rho_h^{1/2})\right)/(1 - \sqrt{2}\rho_h^{1/4})$ and $\mathbf{1}[\cdot]$ is the indicator function.

Proof. Without loss of generality, we assume that $\Delta(x^k) = \|x^k - x_\star\|_2$. We will split the proof into two cases.

CASE 1: $\rho_h \in (0, 1/4)$**.** Consider (44) with $\beta \in (0,1]$ and $x = x^{k+1}$. Since **(HAC-exact)** in (7) holds, we have $\beta^{-1}\varepsilon_{t_k}(x^{k+1}, x^k) \leq \beta^{-1}\rho_h\|x^k - x^{k+1}\|_2^2/(2t_k)$. Moreover, noticing $L/2 + 1/(2t_k) \leq 1/t_k$, $L/2 - 1/(2t_k) \leq 0$ and $F(x^{k+1}) - F(x_\star) \geq 0$, (44) implies that $\frac{1-\beta}{2t_k}\|x^{k+1} - x_\star\|_2^2 \leq \frac{1}{t_k}\|x^k - x_\star\|_2^2 + \frac{\beta^{-1}\rho_h}{2t_k}\|x^k - x^{k+1}\|_2^2$. Then, for any $u' \in (0,1)$, Cauchy-Schwarz inequality implies $(1-\beta)\|x^{k+1} - x_\star\|_2^2 \leq 2\|x^k - x_\star\|_2^2 + \beta^{-1}\rho_h\left(\frac{\|x^{k+1} - x_\star\|_2^2}{u'} + \frac{\|x^k - x_\star\|_2^2}{1-u'}\right)$, which further leads to

$$
\left(1 - \beta - \frac{\rho_h}{\beta u'}\right)\|x^{k+1} - x_\star\|_2^2 \leq \left(2 + \frac{\rho_h}{\beta(1-u')}\right)\|x^k - x_\star\|_2^2.
\tag{47}
$$

Hence, setting $u' = \sqrt{4\rho_h} \in (0,1)$ and $\beta = \sqrt{\rho_h/u'} = (\rho_h/4)^{1/4} \in (0,1)$, we have that $1 - \beta - \rho_h/(\beta u') = 1 - \sqrt{2}\rho_h^{1/4} \in (0,1)$ as $\rho_h \in (0, 1/4)$. Thus, (47) indicates that $\|x^{k+1} - x_\star\|_2^2 \leq \frac{2+\beta^{-1}\rho_h/(1-u')}{1-\beta-\rho_h/(\beta u')}\|x^k - x_\star\|_2^2 = M_0\|x^k - x_\star\|_2^2 \leq \|x_\star\|_2^2$. Thus, $\|x^{k+1} + x_\star\|_2 \geq 2\|x_\star\|_2 - \|x^{k+1} - x_\star\|_2 \geq \|x^{k+1} - x_\star\|_2$. This means that x^{k+1} is closer to $x_\star$ than $-x_\star$.

CASE 2: $\rho_h = 0$**.** Note $\rho_h = 0$ implies that $x^{k+1} = S_{t_k}(x^k)$ and $\varepsilon_{t_k}(x^{k+1}; x^k) = 0$. Thus, setting $x = x^{k+1}$ in (44) and using the facts that $L/2 + 1/(2t_k) \leq 1/t_k, L/2 - 1/(2t_k) \leq 0, F(x^{k+1}) - F(x_\star) \geq 0$, we have $\frac{1-\beta}{2t_k}\|x^{k+1} - x_\star\|_2^2 \leq \frac{1}{t_k}\|x^k - x_\star\|_2^2$ for any $\beta \in (0,1]$. Letting $\beta \to 0^+$, we have $\|x^{k+1} - x_\star\|_2^2 \leq 2\|x^k - x_\star\|_2^2 \leq \|x_\star\|_2^2$ since $M_0 = 2$. Thus, $\|x^{k+1} + x_\star\|_2 \geq 2\|x_\star\|_2 - \|x^{k+1} - x_\star\|_2 \geq \|x^{k+1} - x_\star\|_2$. This means that x^{k+1} is closer to $x_\star$ than $-x_\star$. □

Remark 7.3. Later, when we invoke Lemma 7.2 with $\rho_h = 0$, we use $M_0 = 2$.

Now we are ready to prove Lemma 6.2.

Proof of Lemma 6.2. Without loss of generality, we assume $\Delta(x^k) = \|x^k - x_\star\|_2$. By (46) in Lemma 7.2, we know that $\Delta(x^{k+1}) = \|x^{k+1} - x_\star\|_2$. Consider (44) with $x = x^{k+1}$ and $\beta = \frac{1}{2} \in (0,1]$. Since **(HAC-exact)** in (7) holds, we have $\beta^{-1}\varepsilon_{t_k}(x^{k+1};x^k) \le \beta^{-1}\rho_h\|x^k - x^{k+1}\|_2^2/(2t_k)$. Using this inequality within (44) together with $F(x^{k+1}) - F(x_\star) \ge \lambda_s\Delta(x^{k+1})$ due to (19), we get

$$\lambda_s\Delta(x^{k+1}) + \frac{1}{4t_k}\Delta^2(x^{k+1}) \le \left(\frac{1}{2t_k} + \frac{L}{2}\right)\Delta^2(x^k) + \overbrace{\left(\frac{L}{2} - \frac{1-2\rho_h}{2t_k}\right)\|x^k - x^{k+1}\|_2^2}^{\triangleq \Gamma_k}.$$

Note that $\Gamma_k = \frac{L}{2}\|(x^k - x_\star) - (x^{k+1} - x_\star)\|_2^2 - \frac{1-2\rho_h}{2t_k}\|(x^k - x_\star) - (x^{k+1} - x_\star)\|_2^2$. From (46), we also have that $\frac{L}{2}\|(x^k - x_\star) - (x^{k+1} - x_\star)\|_2^2 \le \frac{L}{2}(\Delta(x^k) + \Delta(x^{k+1}))^2$ and $-\frac{1-2\rho_h}{2t_k}\|(x^k - x_\star) - (x^{k+1} - x_\star)\|_2^2 \le -\frac{1-2\rho_h}{2t_k}(\Delta(x^k) - \Delta(x^{k+1}))^2$. Thus,

$$\left(\lambda_s - \left(L + \frac{1-2\rho_h}{t_k}\right)\Delta(x^k)\right)\Delta(x^{k+1}) + \left(\frac{3-4\rho_h}{4t_k} - \frac{L}{2}\right)\Delta^2(x^{k+1}) \le \left(\frac{\rho_h}{t_k} + L\right)\Delta^2(x^k). \tag{48}$$

From the hypothesis we have $\Delta(x^k) \le \lambda_s/(4L)$, $t_k \in [2\Delta(x^k)/\lambda_s,\ 1/L]$ and $\rho_h \in (0, 1/4)$; therefore, we get $\lambda_s - L\Delta(x^k) - \frac{1-2\rho_h}{t_k}\Delta(x^k) \ge \lambda_s - \lambda_s/4 - \Delta(x^k)/t_k \ge \lambda_s/4$ and $\frac{3-4\rho_h}{4t_k} - \frac{L}{2} = \left(\frac{1}{2t_k} - \frac{L}{2}\right) + \left(\frac{1-4\rho_h}{4t_k}\right) \ge 0$. Using these relations within (48), we obtain $\lambda_s\Delta(x^{k+1}) \le \left(\frac{4\rho_h}{t_k} + 4L\right)\Delta^2(x^k)$; hence, the proof is complete. ☐

7.3 Proofs of Lemma 6.5 and Lemma 6.6

We first give an auxiliary result to discuss the scenario where the subproblems in (6) are solved exactly.

Lemma 7.4. *Suppose that Assumption 2.4 holds. Let $k \in \mathbb{N}$.*

(a) If $\Delta(x^k) \le \lambda_s/(2L)$ and $x^{k+1} = S_{t_k}(x^k)$, then

$$\Delta(x^{k+1}) \le (2L/\lambda_s)\Delta^2(x^k), \quad \forall\, t_k \in [2\Delta(x^k)/\lambda_s, L^{-1}]. \tag{49}$$

(b) If $\Delta(x^k) \le \lambda_s/(4L)$, then for all $t_k \in (0, L^{-1}]$,

$$F_{t_k}(x^k;x^k) - F_{t_k}(S_{t_k}(x^k);x^k) \ge \frac{5}{8}\lambda_s \min\left\{1,\ \frac{\lambda_s t_k}{2\Delta(x^k)}\right\}\Delta(x^k), \tag{50a}$$

$$\|x^k - S_{t_k}(x^k)\|_2 \ge \frac{1}{2}\min\left\{1,\ \frac{\lambda_s t_k}{2\Delta(x^k)}\right\}\Delta(x^k). \tag{50b}$$

Proof. (**Part a**) Without loss of generality, we assume that $\Delta(x^k) = \|x^k - x_\star\|_2$. Since $\Delta(x^k) \le \lambda_s/(2L)$, Lemma 2.6 implies that $\Delta(x^k) \le \|x_\star\|_2/\sqrt{2}$. Moreover, as $x^{k+1} = S_{t_k}(x^k)$, invoking Lemma 7.2 with $\rho_h = 0$, for which case $M_0 = 2$ (hence, we have $\Delta(x^k) \le \|x_\star\|_2/\sqrt{M_0}$), (46) implies that $\mathbf{1}\left[\Delta(x^k) = \|x^k - x_\star\|_2\right] = \mathbf{1}\left[\Delta(x^{k+1}) = \|x^{k+1} - x_\star\|_2\right]$. Therefore, we have $\Delta(x^{k+1}) = \|x^{k+1} - x_\star\|_2$. Using $\Delta(x^k) = \|x^k - x_\star\|_2$, $\Delta(x^{k+1}) = \|x^{k+1} - x_\star\|_2$, and $\frac{L}{2} - \frac{1}{2t_k} \le 0$, we get

$$\left(\frac{L}{2} - \frac{1}{2t_k}\right)\|x^k - x^{k+1}\|_2^2 \le \left(\frac{L}{2} - \frac{1}{2t_k}\right)\left(\Delta(x^k) - \Delta(x^{k+1})\right)^2.$$

Next we use it within (44) for $x = x^{k+1}$. Note that $\varepsilon_{t_k}(x^{k+1};x^k) = 0$ since $x^{k+1} = S_{t_k}(x^k)$, and we also have $F(x^{k+1}) - F(x_\star) \ge \lambda_s\Delta(x^{k+1})$ due to (19); thus, $\lambda_s\Delta(x^{k+1}) + \frac{1-\beta}{2t_k}\Delta^2(x^{k+1}) \le (\frac{L}{2} + \frac{1}{2t_k})\Delta^2(x^k) + (\frac{L}{2} - \frac{1}{2t_k})(\Delta(x^k) - \Delta(x^{k+1}))^2$. Letting $\beta \to 0$, we further get

$$(t_k^{-1} - L/2)\Delta^2(x^{k+1}) + \left(\lambda_s - (t_k^{-1} - L)\Delta(x^k)\right)\Delta(x^{k+1}) - L\Delta^2(x^k) \le 0.$$

Note that $t_k^{-1} - L/2 \ge t_k^{-1}/2$ and $(t_k^{-1} - L)\Delta(x^k) \le \lambda_s/2$ since $L^{-1} \ge t_k \ge 2\Delta(x^k)/\lambda_s$; therefore, we have $t_k^{-1}\Delta^2(x^{k+1}) + \lambda_s\Delta(x^{k+1}) - 2L\Delta^2(x^k) \le 0$, which also implies that

$$\Delta(x^{k+1}) \le 4L\Delta^2(x^k)/\left(\sqrt{\lambda_s^2 + 8t_k^{-1}L\Delta^2(x^k)} + \lambda_s\right) \le 2L\Delta^2(x^k)/\lambda_s.$$

(**Part b**) Recall that (19) implies $F(x^k) - F(x_\star) \ge \lambda_s\Delta(x^k)$. Moreover, (41) holds since we assume $\Delta(x^k) \le \lambda_s/(4L)$. These two relations directly lead to (50a) since $F_{t_k}(x^k;x^k) = F(x^k)$. Next, we focus on the proof of (50b). Without loss of generality, we assume that $\|x^k - x_\star\|_2 = \Delta(x^k)$.

First, we consider the scenario with $t_k \in [2\Delta(x^k)/\lambda_s, L^{-1}]$. From Lemma 2.6, we get $\Delta(x^k) \leq \lambda_s/(4L) \leq \|x_\star\|_2/8$. Therefore, Lemma 7.2 with $\rho_h = 0$ and $M_0 = 2$ implies that $\Delta(S_{t_k}(x^k)) = \|S_{t_k}(x^k) - x_\star\|_2$ since $\Delta(x^k) \leq \|x_\star\|_2/\sqrt{M_0}$. In addition, since $\Delta(x^k) \leq \lambda_s/(4L)$, (49) implies $\Delta(x^{k+1}) \leq \frac{2L}{\lambda_s} \cdot \frac{\lambda_s}{4L}\Delta(x^k) = \Delta(x^k)/2$; hence,

$$\|x^k - S_{t_k}(x^k)\|_2 \geq \|x^k - x_\star\|_2 - \|S_{t_k}(x^k) - x_\star\|_2 \geq \Delta(x^k)/2. \tag{51}$$

Next, we consider $t_k \in (0, t')$ where $t' \triangleq 2\Delta(x^k)/\lambda_s$. By strong convexity of $F_{t'}(\cdot; x^k)$ and $F_{t_k}(\cdot; x^k)$ defined in (5), $F_{t'}(S_{t_k}(x^k); x^k) - F_{t'}(S_{t'}(x^k); x^k) \geq \frac{1}{2t'}\|S_{t_k}(x^k) - S_{t'}(x^k)\|_2^2$ and $F_{t_k}(S_{t'}(x^k); x^k) - F_{t_k}(S_{t_k}(x^k); x^k) \geq \frac{1}{2t_k}\|S_{t_k}(x^k) - S_{t'}(x^k)\|_2^2$; hence, for $u \triangleq x^k - S_{t'}(x^k)$ and $v \triangleq x^k - S_{t_k}(x^k)$,

$$F(S_{t_k}(x^k); x^k) - F(S_{t'}(x^k); x^k) + \frac{1}{2t'}(\|v\|_2^2 - \|u\|_2^2) \geq \frac{1}{2t'}\|u - v\|_2^2,$$

$$F(S_{t'}(x^k); x^k) - F(S_{t_k}(x^k); x^k) + \frac{1}{2t_k}(\|u\|_2^2 - \|v\|_2^2) \geq \frac{1}{2t_k}\|u - v\|_2^2.$$

Adding these two inequalities, we have

$$\left(\frac{1}{2t_k} - \frac{1}{2t'}\right)(\|u\|_2^2 - \|v\|_2^2) \geq \left(\frac{1}{2t_k} + \frac{1}{2t'}\right)\|u - v\|_2^2, \tag{52}$$

which further implies that $\|u\|_2 \geq \|v\|_2$. Next, we consider two cases: $\|u\|_2 = \|v\|_2$ (**case 1**), and $\|u\|_2 > \|v\|_2$ (**case 2**). First, we consider (**case 1**), i.e., $\|x^k - S_{t'}(x^k)\|_2 = \|u\|_2 = \|v\|_2 = \|x^k - S_{t_k}(x^k)\|_2$. Note that (51) implies that $\|x^k - S_{t'}(x^k)\|_2 \geq \Delta(x^k)/2$ since $t' = 2\Delta(x^k)/\lambda_s$; therefore,

$$\|x^k - S_{t_k}(x^k)\|_2 = \|x^k - S_{t'}(x^k)\|_2 \geq \Delta(x^k)/2, \tag{53}$$

implying that (50b) holds. Second, we consider (**case 2**). Since $\|u\|_2 > \|v\|_2$, (52) implies $(t' - t_k)(\|u\|_2^2 - \|v\|_2^2) \geq (t' + t_k)\|u - v\|_2^2 \geq (t' + t_k)(\|u\|_2 - \|v\|_2)^2$. Thus, $(t' - t_k)(\|u\|_2 + \|v\|_2) \geq (t' + t_k)(\|u\|_2 - \|v\|_2)$, which gives $\|v\|_2 \geq \frac{t_k}{t'}\|u\|_2$, i.e.,

$$\|x^k - S_{t_k}(x^k)\|_2 \geq \frac{1}{2}t_k\lambda_s\|x^k - S_{t'}(x^k)\|_2/\Delta(x^k), \tag{54}$$

where we used $t' = 2\Delta(x^k)/\lambda_s$. Similar to (**case 1**), we have $\|x^k - S_{t'}(x^k)\|_2 \geq \Delta(x^k)/2$ due to (51). Therefore, using this relation within (54), we get $\|x^k - S_{t_k}(x^k)\|_2 \geq \frac{1}{4}t_k\lambda_s$. Combining this bound with (51) and (53) completes the proof of (50b). □

Next, we provide the proofs for Lemmas 6.5 and 6.6.

Proof of Lemma 6.5. Given $k \geq 0$ such that $\Delta(x^k) \leq \lambda_s/(4L)$, Lemma 6.4 shows that $\|B_k\|_2 \leq B(\frac{\lambda_s}{4L})$. Let $j \in \mathbb{N}$ such that $j \geq \max\left\{0, \lceil M_k^{\mathbf{LAC}} \rceil - 2\right\}$. From (36),

$$H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \frac{C_0 t_k m\|B_k\|_2^2}{(j+2)^2} \leq \frac{\rho_l}{1+\rho_l}\frac{5}{8}\lambda_s \min\{1,\ \lambda_s t_k/(2\Delta(x^k))\}\Delta(x^k).$$

By (50a), we have $H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \frac{\rho_l}{1+\rho_l}(H_k(\mathbf{0}) - \min_{z\in\mathbb{R}^n} H_k(z))$. Thus,

$$H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \rho_l\Big(- H_k(z_{j+1}^k) + D_k(\lambda_{j+1}^k) + H_k(\mathbf{0}) - \min_{z\in\mathbb{R}^n} H_k(z)\Big),$$

which further implies that $H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \rho_l\Big(H_k(\mathbf{0}) - H_k(z_{j+1}^k)\Big)$, where we used weak duality, i.e., $D_k(\lambda_k^{j+1}) - \min_{z\in\mathbb{R}^n} H_k(z) \leq 0$. Therefore, **(LAC)** in (16) holds within N_k inner iterations and $N_k \leq \max\{1, M_k^{\mathbf{LAC}}\}$. Moreover, it can be shown that $M_k^{\mathbf{LAC}} \geq 1$ (we omit the details due to limited space); therefore, we can conclude that $\mathbb{N}_+ \ni N_k \leq M_k^{\mathbf{LAC}}$, which completes the proof. □

Proof of Lemma 6.6. Given $k \geq 0$ such that $\Delta(x^k) \leq \lambda_s/(4L)$, Lemma 6.4 shows that $\|B_k\|_2 \leq B(\frac{\lambda_s}{4L})$. Let $j \in \mathbb{N}$ such that $j \geq \max\left\{0, \lceil M_k^{\mathbf{HAC}} \rceil - 2\right\}$. From (36),

$$H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \frac{C_0 t_k m\|B_k\|_2^2}{(j+2)^2} \leq \frac{\rho_h}{1+\rho_h}\frac{1}{4t_k}\left(\frac{1}{2}\min\{1,\ \lambda_s t_k/(2\Delta(x^k))\}\ \Delta(x^k)\right)^2.$$

By (50b), we have

$$H_k(z_{j+1}^k) - D_k(\lambda_{j+1}^k) \leq \frac{\rho_h}{1+\rho_h}\frac{1}{4t_k}\|x^k - S_{t_k}(x^k)\|_2^2. \tag{55}$$

For $z_\star^k \triangleq \arg\min_{z\in\mathbb{R}^n} H_k(z)$, we have $\|z_\star^k\|_2 = \|x^k - S_{t_k}(x^k)\|_2$; moreover, using $\frac{1}{t_k}$-strong convexity of $H_k(\cdot)$ and weak duality together, we obtain $\frac{1}{2t_k}\|z_{j+1}^k - z_\star^k\|_2^2 \leq H_k(z_{j+1}^k) - \min_{z\in\mathbb{R}^n} H_k(z) \leq H_k(z_{j+1}^k) - D_k(\lambda_k^{j+1})$. Therefore, (55) implies that

$$\|z_{j+1}^k - z_\star^k\|_2^2 \leq \rho_h/(2(1+\rho_h))\,\|z_\star^k\|_2^2.$$

By Cauchy-Schwarz inequality, we further have

$$\frac{\rho_h}{2t_k}\|z_{j+1}^k\|_2^2 \geq \frac{\rho_h}{4t_k}\|z_\star^k\|_2^2 - \frac{\rho_h}{2t_k}\|z_{j+1}^k - z_\star^k\|_2^2 \geq \frac{\rho_h}{4t_k}\|z_\star^k\|_2^2 - \frac{\rho_h^2}{4t_k(1+\rho_h)}\|z_\star^k\|_2^2 = \frac{\rho_h}{4t_k(1+\rho_h)}\|z_\star^k\|_2^2.$$

Therefore, **(HAC)** in (16) holds within $N_k \leq \max\{1, M_k^{\mathbf{HAC}}\}$ inner iterations. Moreover, it can be shown that $M_k^{\mathbf{HAC}} \geq 1$ (we omit the details due to limited space); therefore, we can conclude $\mathbb{N}_+ \ni N_k \leq M_k^{\mathbf{HAC}}$, which completes the proof. □

Finally, we are ready to provide the proofs for Theorems 6.7 and 6.8.

7.4 Proof of Theorem 6.7

(**Part a**) By Lemma 1.2, **(LAC)** implies that **(LAC-exact)** in (7) also holds for all $k \in \mathbb{N}$. As $\Delta(x^0) > \epsilon$, we have $\mathbb{N}_+ \ni K_\epsilon \geq 1$. Next, we use induction to show the following relations hold simultaneously for $k \in \mathbb{N}$:

$$F(x^k) - F(x_\star) \leq (C(\rho_l))^k\,(F(x^0) - F(x_\star)), \tag{56a}$$

$$\Delta(x^k) \leq \min\{\lambda_s/(4L), (g_H L)^{-1}\}, \tag{56b}$$

$$\exists\, g_k \in [g_L, g_H] :\ t_k = g_k\Delta(x^k), \tag{56c}$$

where $C(\rho) = 1 - \frac{5\min\{\lambda_s g_L/2, 1\}}{8(1+\rho)}$ for $\rho > 0$. For $k = 0$, (56a) holds. Moreover, by Lemma 2.5, we have $F(x^0) - F(x_\star) \leq \min\{\lambda_s^2/(4L), \lambda_s(g_H L)^{-1}\}$ and $\Delta(x^0) \leq \min\{\lambda_s/(4L), (g_H L)^{-1}\}$. Therefore, (56b) holds for $k = 0$, which also implies that $g_0\Delta(x^0) \leq g_H\Delta(x^0) \leq L^{-1}$ since $g_0 \leq g_H$. Recall that according to Corollary 3.5(b), we have $t_0 = \min\{g_0\Delta(x^0), L^{-1}\}$ since $\Delta(x^0) \leq \|x_\star\|_2$ due to Lemma 2.6. Thus, (56c) holds for $k = 0$ as well. Next, we assume that (56) holds for some $k \geq 0$. Together with (56b), (56c) and $g_k \geq g_L$, Lemma 6.1(a) implies $F(x^{k+1}) - F(x_\star) \leq \left(1 - \frac{5\min\{\lambda_s g_L/2, 1\}}{8(1+\rho_l)}\right)(F(x^k) - F(x_\star))$; hence, using it with (56a) establishes (56a) for $k+1$. Since $F(x^{k+1}) - F(x_\star) \leq F(x^0) - F(x_\star) \leq \min\{\lambda_s^2/(4L), \lambda_s/(g_H L)\}$ holds, using (19) we get (56b) for $k+1$. Finally, Corollary 3.5(b) implies that (56c) holds for $k+1$ as well because $g_{k+1}\Delta(x^{k+1}) \leq g_H\Delta(x^{k+1}) \leq L^{-1}$ and $\Delta(x^{k+1}) \leq \|x_\star\|_2$ due to Lemma 2.6. This completes the induction.

By Lemma 6.5 together with (56b) and (56c), we have

$$\sup_{k\in\mathbb{N}} N_k \leq \sup_{k\in\mathbb{N}} M_1\sqrt{\frac{g_k(1+\rho_l)}{\lambda_s\rho_l\min\{1, \lambda_s g_k/2\}}} \leq M_1\sqrt{\frac{g_H(1+\rho_l)}{\lambda_s\rho_l\min\{1, \lambda_s g_H/2\}}}. \tag{57}$$

Without loss of generality, suppose $\Delta(x^0) = \|x^0 - x_\star\|_2$. Due to Lemma 2.6, $\Delta(x^0) \leq \lambda_s/(4L) \leq \|x_\star\|_2$; hence, invoking Lemma 2.1 with $r = \|x_\star\|_2$, we get $F(x^0) - F(x_\star) \leq 2\Delta(x^0)L\|x_\star\|_2$ since $\max\{\Delta(x^0),\ \Delta(x^*)\} \leq \|x^*\|_2$. Recall that $K_\epsilon = \inf\{k \in \mathbb{N}_+ :\ \Delta(x^k) \leq \epsilon\}$; therefore, from (19), $F(x^{K_\epsilon - 1}) - F(x_\star) \geq \lambda_s\Delta(x^{K_\epsilon-1}) \geq \lambda_s\epsilon$. Thus, by (56a), $(C(\rho_l))^{K_\epsilon - 1} \geq \lambda_s\epsilon/\big(2\Delta(x^0)L\|x_\star\|_2\big)$, implying

$$K_\epsilon \leq \log\Big(\frac{2\Delta(x^0)L\|x_\star\|_2}{\lambda_s\epsilon C(\rho_l)}\Big)/\log\Big(\frac{1}{C(\rho_l)}\Big). \tag{58}$$

Since $N(\epsilon) \leq K_\epsilon\sup_{k\in\mathbb{N}} N_k$, using (57), we get the desired result for part (a).

(**Part b**) By Lemma 1.2, **(HAC)** implies that **(HAC-exact)** in (7) also holds for all $k \in \mathbb{N}$. Note that (43) implies that **(LAC-exact)** in (7) holds for any $k \in \mathbb{N}$ for $\rho_l = \bar{\rho} \triangleq 2\rho_h/(1-4\rho_h)$. Thus, similar to the proof of (**Part a**), (56a), (56b) and (56c) with $\rho_l = \bar{\rho}$ hold for any $k \in \mathbb{N}$, which implies that (58) holds for $\rho_l = \bar{\rho}$. Moreover, from Lemma 6.6 together with (56b) and (56c), we get

$$\sup_{k\in\mathbb{N}} N_k \leq \sup_{k\in\mathbb{N}} \frac{M_2 g_k}{\min\{1, \lambda_s g_k/2\}}\sqrt{\frac{1+\rho_h}{\rho_h}} \leq \frac{M_2 g_H}{\min\{1, \lambda_s g_H/2\}}\sqrt{\frac{1+\rho_h}{\rho_h}}.$$

Since $N(\epsilon) \leq K_\epsilon\sup_{k\in\mathbb{N}} N_k$, we get the desired result for part (b).

7.5 Proof of Theorem 6.8

By Lemma 1.2, **(HAC)** implies that **(HAC-exact)** in (7) also holds for all $k \in \mathbb{N}$. We use induction to show the following relations hold simultaneously for $k \in \mathbb{N}$:

$$\Delta(x^k) \leq (6\rho_h/(\lambda_s g_L))^k \Delta(x^0), \tag{59a}$$

$$\Delta(x^k) \leq \min\{\rho_h/(2Lg_L),\ \|x_\star\|_2/\sqrt{M_0},\ 1/(g_H L)\}, \tag{59b}$$

$$\exists g_k \in [g_L, g_H]:\ t_k = g_k \Delta(x^k). \tag{59c}$$

For $k = 0$, (59a) trivially holds and (59b) is true due to hypothesis. Moreover, since $\rho_h \in (0, 1/4)$ and $g_L \geq 2/\lambda_s$, we have $\Delta(x^0) \leq \rho_h/(2Lg_L) \leq \lambda_s/(16L)$; hence, Lemma 2.6 implies that $\Delta(x^0) \leq \|x_\star\|_2$. Thus, by Corollary 3.5(b), we have $t_0 = \{g_0\Delta(x^0), L^{-1}\}$ for some $g_0 \in [g_L, g_H]$. In addition, (59b) implies that $g_0\Delta(x^0) \leq g_H\Delta(x^0) \leq L^{-1}$, which leads to (59c) for $k = 0$. Thus, the base case ($k = 0$) for induction holds. Next, we assume that (59) holds for some $k \in \mathbb{N}$, and we prove that it also holds for $k + 1$. The induction hypothesis on (59a) implies that $\Delta(x^k) \leq \Delta(x^0)$. Thus, $\Delta(x^k) \leq \min\{\lambda_s/(4L),\ \|x_\star\|_2/\sqrt{M_0},\ 1/(g_H L)\}$, and $t_k \in [2\Delta(x^k)/\lambda_2,\ L^{-1}]$ since $g_L \geq 2/\lambda_s$ and $\Delta(x^0) \leq \frac{1}{g_H L}$. Therefore, Lemma 6.2 together with (59c) implies that $\lambda_s\Delta(x^{k+1}) \leq (4\rho_h/g_k + 4L\Delta(x^k))\Delta(x^k)$. By $g_k \geq g_L$ and $\Delta(x^k) \leq \rho_h/(2Lg_L)$, we further have

$$\lambda_s\Delta(x^{k+1}) \leq (4\rho_h/g_L + 4L\rho_h/(2Lg_L))\Delta(x^k) = 6\rho_h\Delta(x^k)/g_L.$$

Together with (59a) for k, this proves (59a) for $k + 1$. Moreover, observing that the rate coefficient in (59a) satisfies $\frac{6\rho_h}{\lambda_s g_L} \leq \frac{3}{4}$ as $g_L \geq 2/\lambda_s$ and $\rho_h \in (0, 1/4)$, we also get $\Delta(x^{k+1}) \leq 3\Delta(x^k)/4$. Thus, (59b) for k immediately implies that (59b) also holds for $k + 1$. Finally, since $\Delta(x^{k+1}) \leq \Delta(x^0) \leq \lambda_s/(4L) \leq \|x_\star\|_2$, according to Corollary 3.5(b), we have $t_{k+1} = \min\{g_{k+1}\Delta(x^{k+1}), L^{-1}\}$ for some $g_{k+1} \in [g_L, g_H]$, and $\Delta(x^{k+1}) \leq (g_H L)^{-1}$ due to (59b) shows that (59c) holds for $k + 1$, establishing the induction. Since (59c) holds for all $k \in \mathbb{N}$, we have $2\Delta(x^k)/\lambda_s \leq g_L\Delta(x^k) \leq t_k \leq g_H\Delta(x^k)$ for $k \geq 0$ since $g_L \geq 2/\lambda_s$. Using these inequalities within the bound for N_k in Lemma 6.6, we get $\sup_{k\in\mathbb{N}} N_k \leq M_2 g_H \sqrt{\frac{1+\rho_h}{\rho_h}}$. As $\Delta(x^{K_\epsilon - 1}) > \epsilon$, (59a) implies $\left(\frac{6\rho_h}{\lambda_s g_L}\right)^{K_\epsilon - 1} \geq \frac{\epsilon}{\Delta(x^0)}$; thus, $K_\epsilon \leq \log\left(\frac{\lambda_s\Delta(x^0)g_L}{6\rho_h} \cdot \frac{1}{\epsilon}\right)/\log\left(\frac{\lambda_s g_L}{6\rho_h}\right)$. Since $N(\epsilon) \leq K_\epsilon \sup_{k\in\mathbb{N}} N_k$, we get the desired result.

8 Numerical Experiments

In this section, we conduct numerical experiments on the RPR problem in (4). We tested the following algorithms:

(i) `PL`: The original proximal linear algorithm proposed in [11] where each subproblem (6) is solved using `POGS` [13]. In our comparison, we use all the default parameters set by the authors in their code[2].

(ii) `GSubGrad`: The subgradient method with geometrically decaying step sizes was proposed in [16] for solving (4) –`GSubGrad` supdates are stated in (8). We picked the best performing contraction coefficient $q \in (0, 1)$ among all the values tested in [16], i.e., $q \in \{0.983, 0.989, 0.993, 0.996, 0.997\}$; on the other hand, [16] does not specify how to choose λ_0, and we set $\lambda_0 = 0.1\|x^0\|_2$.

(iii) `IPL-LAC`, `IPL-HAC`: `IPL` algorithm with fixed step sizes $t_k = 1/L$ and the subproblems in (6) are solved using `APG`. "`-LAC`" and "`-HAC`" indicate whether (**LAC**) or (**HAC**) is used.

(iv) `AdaIPL-LAC`, `AdaIPL-HAC`: These methods, stated in Algorithm 2, use adaptive step sizes given in (12). The subproblems in (6) are solved using `APG`, which performed better compared to `APD` on the problems we tested. "`-LAC`" and "`-HAC`" indicate whether (**LAC**) or (**HAC**) is used.

(v) `AdaSubGrad`: It is the subgradient method, stated in Algorithm 1, with adaptive step sizes given in (10) and (11).

We didn't include `PSubGrad` because according to [17], it only works for the noiseless situation. All the methods are initialized from the same point x^0 which is generated by [11, Algorithm 3].

8.1 Synthetic Data

We generate synthetic data as in [11] and [14]. Specifically, a_i's are drawn randomly from the normal distribution $\mathcal{N}(0, \text{diag}([s_1, s_2 \dots s_n]))$ where $s_i = 1 - 0.75\frac{i-1}{n-1}$ for $i \in [n]$. Throughout the experiments in this subsection, we set

[2] The code of [11] can be downloaded from `https://web.stanford.edu/~jduchi/projects/phase-retrieval-code.tgz`

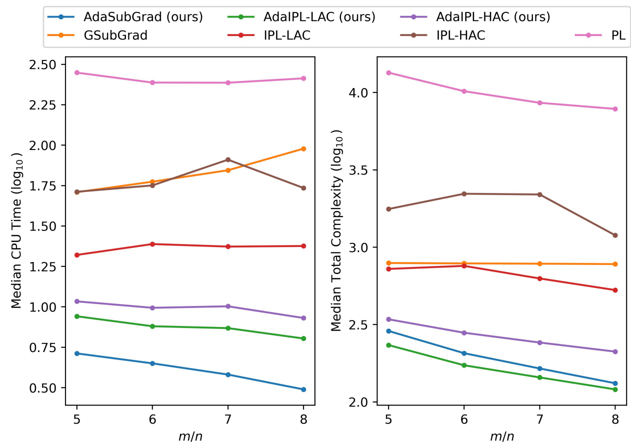

Figure 1: Comparisons on Synthetic Datasets.

$n = 1500$ and we choose m such that $m/n \in \{4, 5, 6, 7, 8\}$. The entries of $x_\star \in \mathbb{R}^n$ are drawn uniformly at random from $\{-1, 1\}$. For each value of m tested, the index set $\mathcal{I}_2$ for corrupted measurements is generated with random sampling $\lceil m\, p_{\text{fail}} \rceil$ elements from $\{1, 2, \ldots, m\}$ without replacement, where $p_{\text{fail}} = 0.1$. Corrupted measurements $b_i = \xi_i$ for $i \in \mathcal{I}_2$ are independently drawn from Cauchy distribution, i.e., $b_i = \xi_i = \tilde{M} \tan\left(\frac{\pi}{2} U_i\right)$ and $U_i \sim U(0, 1)$ for all $i \in \mathcal{I}_2$, where $\tilde{M}$ is the sample median of $\{(a_i^\top x_\star)^2\}_{i=1}^m$. For a given threshold $\epsilon > 0$, we call an algorithm successful if it returns some $x_\epsilon \in \mathbb{R}^n$ such that the relative error $\Delta(x_\epsilon)/\|x_\star\|_2 \leq \epsilon$.

All the algorithms are terminated after such x_ϵ is computed. For each $m/n \in \{4, 5, 6, 7, 8\}$, we randomly generate 10 instances according to the above procedure. For both `IPL` and `AdaIPL`, the parameters are set to $\rho_l = \rho_h = 0.24$ as in [14]; and we choose $G \in \{0.1, 0.5\}$ for `AdaSubGrad` and $G \in \{1000/n, 100/n, 10/n, 1/n\}$ for `AdaIPL`, and for both methods we set $\tilde{p} = 0.5$. In this set of experiments, the success rate is 1 for all the algorithms and we find that $q = 0.983$ for `GSubGrad`, $G = 100/n$ for `AdaIPL`, and $G = 0.5$ for `AdaSubGrad` consistently yield the best performance, respectively; thus, we make comparisons based on these parameter choices.

In Figure 1 we provide the experimental results where we set $\epsilon = 10^{-3}$ for `PL` and $\epsilon = 10^{-7}$ for all other algorithms. The left panel shows the median CPU time in seconds[3] for all replications. The right one shows the median of total iteration numbers, i.e., total inner iteration numbers for `PL`, `IPL`, and `AdaIPL` and total iteration numbers for `GSubGrad` and `AdaSubGrad`. This image shows that both `AdaSubGrad` and `AdaIPL` perform better than the other algorithms.

Next, Table 3 provides results for the setting in Figure 1 with $m/n = 8$ to show the performance under difference choices of G. In this table, median values for CPU time, total subproblem iterations, and main iterations are reported. XL, L, M, S correspond to $G = 1000/n, 100/n, 10/n, 1/n$ for `AdaIPL`, and $0.5, 0.1$ correspond to the values of G for `AdaSubGrad`. We can find that using large G for `AdaSubGrad` reduces iteration numbers. For `AdaIPL`, using a larger G leads to a larger ratio of total iterations to main iterations and a smaller main iteration number, which is lower bounded by the `IPL` counterparts. In addition, when G is very small, the ratio of total iterations to main iterations is less than 3, which means that very few iterations are needed to solve a subproblem.

8.2 Image Recovery

In this subsection, we conducted experiments on images as in [11]. In particular, consider an RGB image array $X_\star \in \mathbb{R}^{n_1 \times n_2 \times 3}$, we construct the signal as $x_\star = [\text{vec}(X_\star)^\top \mathbf{0}^\top]^\top \in \mathbb{R}^n$ where $n = \min\{2^s \mid s \in \mathbb{N},\ 2^s \geq 3n_1 n_2\}$ and $\mathbf{0} \in \mathbb{R}^{n - 3n_1 n_2}$ –here, $\text{vec}(\cdot)$ vectorize its argument. Let $H_n \in \frac{1}{\sqrt{n}}\{-1, 1\}^{n \times n}$ be the Hadamard matrix. We generate *diagonal* $S_j \in \mathbb{R}^{n \times n}$ for $j = 1, 2, \ldots, k$ such that all the diagonal entries are chosen uniformly at random from $\{-1, 1\}$. Next, for $m = 6n$, we set $A \triangleq \sqrt{n}[(H_n S_1)^\top (H_n S_2)^\top \ldots (H_n S_6)^\top]^\top \in \mathbb{R}^{m \times n}$ –for this setting, it can be shown that $L = 2$. The advantage of such a mapping is that it mimics the fast Fourier transform and calculating Ax requires $O(m \log n)$ work for any $x \in \mathbb{R}^n$. We conduct the test on an RNA image[4] of size $n = 2^{22}$. In the experiment we considered $p_{\text{fail}} = 0.1$, and we generate corrupted measurements as in the synthetic datasets. The

[3] The CPU time for initialization and approximately calculating L is negligible in this example.

[4] https://visualsonline.cancer.gov/details.cfm?imageid=11167

	CPU Time (sec)	Total Iterations	Main Iterations
IPL-LAC	23.78	528.5	11
AdaIPL-LAC-XL	6.86	144	11
AdaIPL-LAC-L	6.37	120.5	11.5
AdaIPL-LAC-M	8.93	162.5	16.5
AdaIPL-LAC-S	55.83	491	220.5
	CPU Time (sec)	Total Iterations	Main Iterations
IPL-HAC	54.34	1194	6
AdaIPL-HAC-XL	16.65	425	7
AdaIPL-HAC-L	8.53	211.5	7
AdaIPL-HAC-M	9.47	182	17
AdaIPL-HAC-S	55.41	475.5	220.5
	CPU Time (sec)	Total Iterations	Main Iterations
AdaSubGrad: 0.5	3.09	132	132
AdaSubGrad: 0.1	10.62	473	473

Table 3: Comparison of different G for simulated datasets: median values for 10 replicates are reported.

Method	**CPU Time**
PL	> 1440
GSubGrad	44.21 (26.79)
IPL-LAC	177.21 (136.30)
IPL-HAC	235.75 (136.23)
AdaIPL-LAC	16.44 (9.56)
AdaIPL-HAC	25.35 (15.06)
AdaSubGrad	4.12 (0.35)

Table 4: Comparison of CPU time (in minutes) for image recovery problem. Median (interquartile range) values are reported over 10 replications.

algorithm parameters are set to $G = 0.5$ for AdaSubGrad, $G = 1000/n$ for AdaIPL, and $q = 0.983$ for GSubGrad while keeping the other hyper-parameters the same as in Section 8.1.

All the algorithms are terminated whenever x_ϵ with a relative error at most $\epsilon = 10^{-7}$ is computed. The results are reported in Table 4, where we report the median and interquartile range of CPU times in *minutes* based on 10 replications. The results show that both AdaSubGrad and AdaIPL perform better than the other algorithms. The result for PL is not reported as it cannot reach the desired accuracy within the time limit of 1440 min.

9 Conclusion

In this paper, we propose two adaptive algorithms for solving the robust phase retrieval problem. Our contribution lies in designing new adaptive step size rules that are based on the quantiles of absolute residuals and are robust to sparse corruptions. Employing adaptive step sizes, both methods show local linear convergence and are robust to hyper-parameter selection. Numerical results demonstrate that both AdaSubGrad and AdaIPL perform significantly better than the existing state-of-the-art methods tested on both syntetic- and real-data RPR problems.

References

[1] J. Miao, P. Charalambous, J. Kirz, and D. Sayre, "Extending the methodology of x-ray crystallography to allow imaging of micrometre-sized non-crystalline specimens," *Nature*, vol. 400, no. 6742, pp. 342–344, 1999.

[2] R. P. Millane, "Phase retrieval in crystallography and optics," *Journal of the Optical Society of America A*, vol. 7, no. 3, pp. 394–411, 1990.

[3] A. Chai, M. Moscoso, and G. Papanicolaou, "Array imaging using intensity-only measurements," *Inverse Problems*, vol. 27, no. 1, p. 015005, 2010.

[4] C. Fienup and J. Dainty, "Phase retrieval and image reconstruction for astronomy," in *Image Recovery: Theory and Application*. Academic Press, 1987, pp. 231–275.

[5] J. Miao, T. Ishikawa, Q. Shen, and T. Earnest, "Extending x-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes," *Annual Review of Physical Chemistry*, vol. 59, pp. 387–410, 2008.

[6] M. Fickus, D. G. Mixon, A. A. Nelson, and Y. Wang, "Phase retrieval from very few measurements," *Linear Algebra and its Applications*, vol. 449, pp. 475–499, 2014.

[7] E. J. Candes, X. Li, and M. Soltanolkotabi, "Phase retrieval via wirtinger flow: Theory and algorithms," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 1985–2007, 2015.

[8] Y. Chen and E. J. Candès, "Solving random quadratic systems of equations is nearly as easy as solving linear systems," *Communications on Pure and Applied Mathematics*, vol. 70, no. 5, pp. 822–883, 2017.

[9] G. Wang, G. B. Giannakis, and Y. C. Eldar, "Solving systems of random quadratic equations via truncated amplitude flow," *IEEE Transactions on Information Theory*, vol. 64, no. 2, pp. 773–794, 2017.

[10] H. Zhang, Y. Zhou, Y. Liang, and Y. Chi, "A nonconvex approach for phase retrieval: Reshaped wirtinger flow and incremental algorithms," *Journal of Machine Learning Research*, vol. 18, pp. 1–35, 2017.

[11] J. C. Duchi and F. Ruan, "Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval," *Information and Inference: A Journal of the IMA*, vol. 8, no. 3, pp. 471–529, 2019.

[12] H. Zhang, Y. Chi, and Y. Liang, "Provable non-convex phase retrieval with outliers: Median truncated wirtinger flow," in *International Conference on Machine Learning*. PMLR, 2016, pp. 1022–1031.

[13] N. Parikh and S. Boyd, "Block splitting for distributed optimization," *Mathematical Programming Computation*, vol. 6, no. 1, pp. 77–102, 2014.

[14] Z. Zheng, S. Ma, and L. Xue, "A new inexact proximal linear algorithm with adaptive stopping criteria for robust phase retrieval," *IEEE Transactions on Signal Processing*, vol. 72, pp. 1081–1093, 2024.

[15] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," *submitted to SIAM Journal on Optimization*, 2008.

[16] D. Davis, D. Drusvyatskiy, K. J. MacPhee, and C. Paquette, "Subgradient methods for sharp weakly convex functions," *Journal of Optimization Theory and Applications*, vol. 179, no. 3, pp. 962–982, 2018.

[17] D. Davis, D. Drusvyatskiy, and C. Paquette, "The nonsmooth landscape of phase retrieval," *IMA Journal of Numerical Analysis*, vol. 40, no. 4, pp. 2652–2695, 2020.

[18] J. C. Duchi and F. Ruan, "Stochastic methods for composite and weakly convex optimization problems," *SIAM Journal on Optimization*, vol. 28, no. 4, pp. 3229–3259, 2018.

[19] D. Davis and D. Drusvyatskiy, "Stochastic subgradient method converges at the rate $o(k^{-1/4})$ on weakly convex functions," *arXiv preprint arXiv:1802.02988*, 2018.

[20] ——, "Stochastic model-based minimization of weakly convex functions," *SIAM Journal on Optimization*, vol. 29, no. 1, pp. 207–239, 2019.

[21] D. Davis and B. Grimmer, "Proximally guided stochastic subgradient method for nonsmooth, nonconvex problems," *SIAM Journal on Optimization*, vol. 29, no. 3, pp. 1908–1930, 2019.

[22] D. Davis, D. Drusvyatskiy, and V. Charisopoulos, "Stochastic algorithms with geometric step decay converge linearly on sharp functions," *Mathematical Programming*, pp. 1–46, 2023.

[23] Z. Zheng and L. Xue, "Smoothed robust phase retrieval," *arXiv preprint arXiv:2409.01570*, 2024.

[24] R. Rockafellar and R. J.-B. Wets, *Variational Analysis*. Heidelberg, Berlin, New York: Springer, 1998.

[25] A. Chambolle and T. Pock, "On the ergodic convergence rates of a first-order primal–dual algorithm," *Mathematical Programming*, vol. 159, no. 1, pp. 253–287, 2016.

[26] E. Y. Hamedani and N. S. Aybat, "A primal-dual algorithm with line search for general convex-concave saddle point problems," *SIAM Journal on Optimization*, vol. 31, no. 2, pp. 1299–1329, 2021.