# Partial-Quasi-Newton Methods: Efficient Algorithms for Minimax Optimization Problems with Unbalanced Dimensionality

Chengchang Liu
Department of Computer Science & Engineering
The Chinese University of Hong Kong
7liuchengchang@gmail.com

Shuxian Bi
School of Cyber Science and Technology
University of Science and Technology of China
stanbi@mail.ustc.edu.cn

Luo Luo*
School of Data Science
Fudan University
luoluo@fudan.edu.cn

John C.S. Lui
Department of Computer Science & Engineering
The Chinese University of Hong Kong
cslui@cse.cuhk.edu.hk

## ABSTRACT

This paper studies the strongly-convex-strongly-concave minimax optimization with unbalanced dimensionality. Such problems contain several popular applications in data science such as few shot learning and fairness-aware machine learning task. The design of conventional iterative algorithm for minimax optimization typically focuses on reducing the total number of oracle calls, which ignores the unbalanced computational cost for accessing the information from two different variables in minimax. We propose a novel second-order optimization algorithm, called Partial-Quasi-Newton (PQN) method, which takes the advantage of unbalanced structure in the problem to establish the Hessian estimate efficiently. We theoretically prove our PQN method converges to the saddle point faster than existing minimax optimization algorithms. The numerical experiments on real-world applications show the proposed PQN performs significantly better than the state-of-the-art methods.

## CCS CONCEPTS

• **Mathematics of computing → Mathematical optimization**.

## KEYWORDS

Minimax Optimization, Quasi-Newton, Few-shot Learning, Fairness

**ACM Reference Format:**
Chengchang Liu, Shuxian Bi, Luo Luo, and John C.S. Lui . 2022. Partial-Quasi-Newton Methods: Efficient Algorithms for Minimax Optimization Problems with Unbalanced Dimensionality. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA.* ACM, Washington, DC, USA, 11 pages. https://doi.org/10.1145/3534678.3539379

## 1 INTRODUCTION

We study the minimax optimization problem of the form

$$\min_{\mathbf{x}\in\mathbb{R}^{n_x}} \max_{\mathbf{y}\in\mathbb{R}^{n_y}} f(\mathbf{x},\mathbf{y}), \tag{1}$$

where $f(\mathbf{x},\mathbf{y})$ is smooth, strongly-convex in $\mathbf{x}$ and strongly-concave in $\mathbf{y}$. This formulation has received increasing attention recently because of it contains a lot of applications in machine learning and data mining [8, 14, 16, 25, 26, 39, 54, 55].

There are a great number of first-order algorithms for solving the general strongly-convex-strongly-concave (SCSC) minimax problem (1), including extragradient method [19, 29, 33, 48], optimistic gradient descent ascent [9, 37], proximal point method [23, 41, 49, 50] and dual extrapolation [31]. These algorithms achieve linear convergence rate and their extension to stochastic setting also be well studied [1, 7, 27, 28, 34, 47]. Second-order optimization algorithms usually have superior convergence behavior compared with the first-order methods. Huang et al. [17] proposed cubic regularized Newton (CRN) method [30, 31] for solving problem (1). CRN has quadratic local convergence but it requires constructing the Hessian matrix exactly at each iteration. Recently, Liu and Luo [24] proposed quasi-Newton methods, which enjoy explicit local superlinear convergence rate and their iterations avoid computing the exact Hessian.

The minimax formulation for plenty of machine learning applications has the characterization of *unbalanced dimensionality*, that is, we usually have $n_x \gg n_y$ (or $n_x \ll n_y$)[1]. We list some popular models that naturally lead to minimax optimization with unbalanced dimensionality.

- **AUC Maximization:** Area under ROC (AUC) is a metric which is important in few-shot learning and widely used for measuring the performance of binary classification for imbalanced data [16]. The idea of AUC maximization [8, 54] is to find a classifier on the imbalanced training set $\{\mathbf{a}_i, b_i\}_{i=1}^m$ where $\mathbf{a}_i \in \mathbb{R}^d$ and $b_i \in \{+1, -1\}$, whose minimax formulation can be written as

$$\min_{\mathbf{x}\in\mathbb{R}^{d+2}} \max_{y\in\mathbb{R}} f(\mathbf{x},y) \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m} f_i(\mathbf{x},y;\mathbf{a}_i,b_i) + \frac{\lambda}{2}\|\mathbf{x}\|^2, \tag{2}$$

where each $f_i(\mathbf{x},y;\mathbf{a}_i,b_i)$ is convex in $\mathbf{x}$, strongly-concave in $y$ and $\lambda > 0$ is the regularization parameter.

*The corresponding author

[1] Without loss of generality, we only consider $n_x \gg n_y$ in remainders of this paper.

*KDD '22, August 14–18, 2022, Washington, DC, USA*

ACM ISBN 978-1-4503-9385-0/22/08...$15.00
https://doi.org/10.1145/3534678.3539379

- **Fairness-Aware Machine Learning:** The issue of addressing fairness in AI systems is a hot topic for data science in recent years [39]. Adversarial learning [26] is a popular way for fairness-aware machine learning [55]. Consider that we have the training set $\{\mathbf{a}_i, b_i, c_i\}_{i=1}^m$ where $\mathbf{a}_i \in \mathbb{R}^d$ contains all input variables of $i$-th example, $b_i \in \mathbb{R}$ is the corresponding output and $c_i \in \mathbb{R}$ is the input variable which we want to protect and make it unbiased. Then we can formulate the following minimax optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^d} \max_{y\in\mathbb{R}} \frac{1}{m}\sum_{i=1}^{m}(l_1(b_i\mathbf{a}_i^\top\mathbf{x}) - \beta l_2(c_i y\mathbf{a}_i^\top\mathbf{x})) + \lambda\|\mathbf{x}\|^2 - \gamma y^2, \tag{3}$$

where both $l_1(\cdot)$ and $l_2(\cdot)$ are the convex loss functions; $\beta > 0$ is the trade-off parameter.

Typically, computing the (second-order) partial derivative with respect to $\mathbf{x}$ is much more expensive than computing the one with respect to $\mathbf{y}$ in above applications[2]. However, the designs of all existing first-order and second-order algorithms for SCSC minimax optimization do not consider such unbalanced characterization.

In this paper, we proposed a novel type of second-order optimization algorithms for SCSC minimax optimization with unbalanced dimensionality, called partial-quasi-Newton (PQN) methods. The algorithms construct the Hessian estimator by approximating the second-order partial derivative with respect to $\mathbf{x}$ via quasi-Newton-type update and computing the other second-order information exactly. The exact second-order information helps PQN converges to saddle point faster than Liu and Luo [24]'s quasi-Newton methods for SCSC minimax problems. We summarize the comparison of the convergence results for PQN and existing quasi-Newton methods in Table 1. Furthermore, the unbalanced dimensionality in minimax problem allows PQN methods to compute the required exact second-order information efficiently. We also provide the numerical experiments on the applications of AUC maximization and fairness-aware machine learning, which show the proposed PQN performs significantly better than the state-of-the-art algorithms.

*Paper Organization.* In Section 2, we introduce the notation and preliminaries that will be used in this paper. In Section 3, we survey the recent advances in quasi-Newton methods. In Section 4, we propose partial-quasi-Newton (PQN) methods for solving SCSC minimax optimization problems with unbalanced dimensionality. In Section 5, we provide numerical experiments on popular machine learning models to validate the effectiveness of our algorithms. Some detailed proofs are deferred to appendix.

## 2 NOTATION AND PRELIMINARIES

We use $\|\cdot\|$ to present spectral norm and Euclidean norm of matrix and vector respectively. We denote the standard basis for $\mathbb{R}^d$ by $\{\mathbf{e}_1, \dots, \mathbf{e}_d\}$ and let $\mathbf{I}_{n_x}$ and $\mathbf{I}_{n_y}$ be the corresponding identity matrix. The trace of a square matrix is denoted by $\mathrm{tr}(\cdot)$. Following the notation of problem (1), we let $\mathbf{z} = [\mathbf{x}; \mathbf{y}] \in \mathbb{R}^n$ where $n \overset{\text{def}}{=} n_x + n_y$ and use $\mathbf{z}^* = [\mathbf{x}^*; \mathbf{y}^*] \in \mathbb{R}^n$ to present the solution of the minimax problem. We denote the gradient and Hessian matrix of $f$ at $(\mathbf{x}, \mathbf{y})$ as $\mathbf{g}(\mathbf{z}) \in \mathbb{R}^n$ and $\mathbf{H}(\mathbf{z}) \in \mathbb{R}^{n\times n}$. Additionally, we use $\mathbf{H}_{\mathbf{xx}}(\mathbf{z})$, $\mathbf{H}_{\mathbf{xy}}(\mathbf{z})$, $\mathbf{H}_{\mathbf{yx}}(\mathbf{z})$ and $\mathbf{H}_{\mathbf{yy}}(\mathbf{z})$ to denote $\nabla^2_{\mathbf{xx}}f(\mathbf{x},\mathbf{y}) \in \mathbb{R}^{n_x\times n_x}$, $\nabla^2_{\mathbf{xy}}f(\mathbf{x},\mathbf{y}) \in \mathbb{R}^{n_x\times n_y}$, $\nabla^2_{\mathbf{yx}}f(\mathbf{x},\mathbf{y}) \in \mathbb{R}^{n_y\times n_x}$ and $\nabla^2_{\mathbf{yy}}f(\mathbf{x},\mathbf{y}) \in \mathbb{R}^{n_y\times n_y}$ respectively. We also define $\mathbf{P}(\mathbf{z}) \overset{\text{def}}{=} \mathbf{H}_{\mathbf{xx}}(\mathbf{z}) - \mathbf{H}_{\mathbf{xy}}(\mathbf{z})\mathbf{H}_{\mathbf{yy}}^{-1}(\mathbf{z})\mathbf{H}_{\mathbf{yx}}(\mathbf{z})$.

We suppose the minimax optimization problem (1) satisfies the following assumptions.

**Assumption 2.1.** The objective function $f(\mathbf{x}, \mathbf{y})$ is twice differentiable and it has $L$-Lipschitz continuous gradient and $L_2$-Lipschitz continuous Hessian, i.e., there exist constants $L > 0$ and $L_2 > 0$ such that

$$\left\|\mathbf{g}(\mathbf{z}) - \mathbf{g}(\mathbf{z}')\right\| \le L\left\|\mathbf{z} - \mathbf{z}'\right\| \tag{4}$$

and

$$\left\|\mathbf{H}(\mathbf{z}) - \mathbf{H}(\mathbf{z}')\right\| \le L_2\left\|\mathbf{z} - \mathbf{z}'\right\|. \tag{5}$$

for any $\mathbf{z} = [\mathbf{x}; \mathbf{y}]$, $\mathbf{z}' = [\mathbf{x}'; \mathbf{y}'] \in \mathbb{R}^n$.

**Assumption 2.2.** The objective function $f(\mathbf{x}, \mathbf{y})$ is twice differentiable, $\mu$-strongly-convex in $\mathbf{x}$ and $\mu$-strongly-concave in $\mathbf{y}$, i.e., there exists constant $\mu > 0$ such that

$$\nabla^2_{\mathbf{xx}}f(\mathbf{x},\mathbf{y}) \succeq \mu\mathbf{I} \qquad \text{and} \qquad \nabla^2_{\mathbf{yy}}f(\mathbf{x},\mathbf{y}) \preceq -\mu\mathbf{I}$$

for any $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_y}$.

The inequality (4) means the spectral norm of Hessian matrix $\mathbf{H}(\mathbf{z})$ can be upper bounded, that is, we have $\|\mathbf{H}(\mathbf{z})\| \le L$ for all $\mathbf{z} \in \mathbb{R}^n$. Additionally, the condition number of the objective function is defined as $\kappa \overset{\text{def}}{=} L/\mu$ and $\kappa_2 \overset{\text{def}}{=} L_2/\mu$.

## 3 RELATED WORK

Before presenting our algorithms, we briefly survey the related work of quasi-Newton methods.

### 3.1 Quasi-Newton Methods for Minimization Problems

Quasi-Newton methods [2–5, 10, 18, 20, 22, 43, 44, 46, 53] are popular algorithms for convex optimization. They have superior local convergence than first-order methods and avoid accessing exact second-order information. The famous quasi-Newton methods including Davidon-Fletcher-Powell (DFP) method [10, 13], Broyden-Fletcher-Goldfarb-Shanno (BFGS) method [3, 4, 46] and symmetric rank 1 (SR1) method [2, 10], which approximate the Hessian matrix based on the Broyden family updating formula [2] that is defined as follows.

**Definition 3.1** ([32, Section 6.3])**.** Suppose two positive definite matrices $\hat{\mathbf{H}}, \hat{\mathbf{G}} \in \mathbb{R}^{\hat{n}\times\hat{n}}$ satisfy $\hat{\mathbf{H}} \preceq \hat{\mathbf{G}}$. For any $\mathbf{u} \in \mathbb{R}^{\hat{n}}$, if $\hat{\mathbf{G}}\mathbf{u} = \hat{\mathbf{H}}\mathbf{u}$, we define $\mathrm{Broyd}_\tau(\hat{\mathbf{G}}, \hat{\mathbf{H}}, \mathbf{u}) \overset{\text{def}}{=} \hat{\mathbf{H}}$. Otherwise, we define

$$\begin{aligned}\mathrm{Broyd}_\tau(\hat{\mathbf{G}}, \hat{\mathbf{H}}, \mathbf{u}) &\overset{\text{def}}{=} (1-\tau)\left[\hat{\mathbf{G}} - \frac{(\hat{\mathbf{G}}-\hat{\mathbf{H}})\mathbf{u}\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})}{\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})\mathbf{u}}\right] \\ &\quad + \tau\left[\hat{\mathbf{G}} - \frac{\hat{\mathbf{H}}\mathbf{u}\mathbf{u}^\top\hat{\mathbf{G}} + \hat{\mathbf{G}}\mathbf{u}\mathbf{u}^\top\hat{\mathbf{H}}}{\mathbf{u}^\top\hat{\mathbf{H}}\mathbf{u}} + \left(\frac{\mathbf{u}^\top\hat{\mathbf{G}}\mathbf{u}}{\mathbf{u}^\top\hat{\mathbf{H}}\mathbf{u}} + 1\right)\frac{\hat{\mathbf{H}}\mathbf{u}\mathbf{u}^\top\hat{\mathbf{H}}}{\mathbf{u}^\top\hat{\mathbf{H}}\mathbf{u}}\right].\end{aligned} \tag{6}$$

In this paper, we focus on the popular SR1 update by choosing parameter $\tau = 0$ for formula (6), leading to

$$\mathrm{SR1}(\hat{\mathbf{G}}, \hat{\mathbf{H}}, \mathbf{u}) \overset{\text{def}}{=} \hat{\mathbf{G}} - \frac{(\hat{\mathbf{G}}-\hat{\mathbf{H}})\mathbf{u}\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})}{\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})\mathbf{u}}. \tag{7}$$

[2] We present the detailed expression of $f_i$ for AUC maximization and $l_1$, $l_2$ for fairness-aware machine learning models in Section 5.

**Table 1: We compare the convergence results of proposed PQN methods with existing quasi-Newton methods [24] for solving SCSC minimax problem (1). The measure of Liu and Luo [24]'s algorithms is the gradient norm $\|\nabla f(\mathbf{z}_k)\|$ after $(k+k_0)$ iterations. The measure of proposed PQN methods is based on the weighted gradient norm $\langle \nabla_{\mathbf{x}} f(\mathbf{z}_k), \mathbf{P}_k^{-1}\nabla_{\mathbf{x}} f(\mathbf{z}_k)\rangle^{1/2} + (2/\sqrt{\mu})\|\nabla_{\mathbf{y}} f(\mathbf{z}_k)\|$ after $(k+k_0)$ iterations, where $\mathbf{P}_k \succ 0$. The upper bounds of random algorithms hold with probability at least $1-\delta$ for $\delta > 0$.**

| Algorithm | Upper Bound | $k_0$ |
|---|---|---|
| Random Broyden [24, Corollary 3.17.2] | $\left(1-\frac{1}{n\kappa^2+1}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa^2}\right)^{k_0}$ | $O\left(n\kappa^2\ln\left(\frac{n\kappa}{\delta}\right)\right)$ |
| Random SR1 [24, Corollary 3.17.2] | $\left(1-\frac{1}{n+1}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa^2}\right)^{k_0}$ | $O\left((n+\kappa^2)\ln\left(\frac{n\kappa}{\delta}\right)\right)$ |
| Random Broyden PQN [This paper, Corollary 4.7 (a)] | $\left(1-\frac{1}{n\kappa+1}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa}\right)^{k_0}$ | $O\left(n\kappa\ln\left(\frac{n\kappa}{\delta}\right)\right)$ |
| Greedy Broyden PQN [This paper, Corollary 4.7 (b)] | $\left(1-\frac{1}{n\kappa}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa}\right)^{k_0}$ | $O\left(n\kappa\ln\left(n\kappa\right)\right)$ |
| Random SR1 PQN [This paper, Corollary 4.7 (c)] | $\left(1-\frac{1}{n+1}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa}\right)^{k_0}$ | $O\left((n+\kappa)\ln\left(\frac{n\kappa}{\delta}\right)\right)$ |
| Greedy SR1 PQN [This paper, Corollary 4.7 (d)] | $\left(1-\frac{1}{n}\right)^{k(k-1)/2}\left(\frac{1}{2}\right)^k\left(1-\frac{1}{4\kappa}\right)^{k_0}$ | $O\left((n+\kappa)\ln\left(n\kappa\right)\right)$ |

For minimization strongly-convex function, classical quasi-Newton methods [5, 11, 38] construct the Hessian estimator based on the framework of Broyden family update and select direction $\mathbf{u}$ by secant condition. As a result, these algorithms achieve local superlinear convergence.

Recently, Rodomanov and Nesterov [42] proposed a greedy algorithm for selecting the direction $\mathbf{u}$ for Broyden family update (6):

$$\hat{\mathbf{u}}_{\hat{\mathbf{H}}}(\hat{\mathbf{G}}) \overset{\text{def}}{=} \underset{\mathbf{u}\in\{\mathbf{e}_1,\cdots,\mathbf{e}_{\hat{n}}\}}{\arg\max} \frac{\mathbf{u}^\top\hat{\mathbf{G}}\mathbf{u}}{\mathbf{u}^\top\hat{\mathbf{H}}\mathbf{u}}. \tag{8}$$

Later, Lin et al. [22] provided a specific method to chose direction for SR1 update as follows

$$\bar{\mathbf{u}}_{\hat{\mathbf{H}}}(\hat{\mathbf{G}}) \overset{\text{def}}{=} \underset{\mathbf{u}\in\{\mathbf{e}_1,\cdots,\mathbf{e}_{\hat{n}}\}}{\arg\max} \frac{\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})^2\mathbf{u}}{\mathbf{u}^\top(\hat{\mathbf{G}}-\hat{\mathbf{H}})\mathbf{u}}. \tag{9}$$

Lin et al. [22], Rodomanov and Nesterov [42] also studies the random algorithm for selection $\mathbf{u}$ as follows

$$\mathbf{u} \sim \mathcal{N}(\mathbf{0},\mathbf{I}) \qquad \text{or} \qquad \mathbf{u} \sim \mathrm{Unif}\left(\mathcal{S}^{\hat{n}-1}\right), \tag{10}$$

which can be implemented more efficiently.

Minimizing strongly-convex objective function by either greedy or random quasi-Newton methods enjoys two-period local convergence behaviors: the first one has a linear convergence rate and the second one enjoys explicit non-asymptotic superlinear convergence.

### 3.2 Quasi-Newton Methods for SCSC Minimax Optimization Problems

Different with convex minimization, the Hessian matrix for the objective function in SCSC minimax problem is indefinite. Hence, we cannot apply the Broyden family update to approximate the Hessian directly. To address this issue, Liu and Luo [24] characterized the second-order information by approximating the square of the Hessian. Specifically, they considered the following update rule at $k$-th iteration

$$\mathbf{z}_{k+1} = \mathbf{z}_k - \hat{\mathbf{G}}_k^{-1}\mathbf{H}(\mathbf{z}_k)\mathbf{g}(\mathbf{z}_k),$$

where $\hat{\mathbf{G}}_k$ is an estimator for the square of $\mathbf{H}(\mathbf{z}_k)$. We can verify that the matrix $(\mathbf{H}(\mathbf{z}_k))^2$ must be positive definite [24, Lemma 3.1]. Then applying Broyden family updating on $(\mathbf{H}(\mathbf{z}_k))^2$ and selecting direction $\mathbf{u}$ by greedy strategy (8) or random strategy (10) obtains the quasi-Newton methods for SCSC minimax optimization with two-period local convergence rates: the first one has linear convergence and the second one has explicit superlinear convergence.

Note that the convergence rates of Liu and Luo [24]'s algorithms for SCSC minimax problems depend on $O(1-1/\kappa^2)$ (see Table 1), while the convergence rates of quasi-Newton methods for convex optimization [21, 42] depends on $O(1-1/\kappa)$. The reason is the condition number for $(\mathbf{H}(\mathbf{z}_k))^2$ is the square of the condition number for $\mathbf{H}(\mathbf{z}_k)$. On the other hand, it is difficult to implement greedy algorithm (8) with $\hat{\mathbf{H}}_k = (\mathbf{H}(\mathbf{z}_k))^2$ in practice since computing the diagonal entries for the square of Hessian is so expensive. In fact, even for quadratic SCSC minimax problem, computing all diagonal entries for the square of Hessian requires $O(n^3)$ time complexity in general.

## 4 PARTIAL-QUASI-NEWTON METHODS

In this section, we propose Partial-Quasi-Newton (PQN) methods for solving minimax problem (1) which satisfies Assumption 2.1-2.2 and $n_x \gg n_y$, then we provide their convergence analysis.

**Algorithm 1** Inverse($\mathbf{G}_{\mathbf{xx}}^{-1}, \mathbf{H}_{\mathbf{xy}}, \mathbf{H}_{\mathbf{yy}}$)

```
Input: $\mathbf{G}_{\mathbf{xx}}^{-1}$, $\mathbf{H}_{\mathbf{xy}}$ and $\mathbf{H}_{\mathbf{yy}}$
    $\mathbf{C}_2 = \mathbf{H}_{\mathbf{yy}} - \mathbf{H}_{\mathbf{xy}}^\top \mathbf{G}_{\mathbf{xx}}^{-1} \mathbf{H}_{\mathbf{xy}}$
    $\mathbf{C}_1^{-1} = \mathbf{G}_{\mathbf{xx}}^{-1} + (\mathbf{G}_{\mathbf{xx}}^{-1}\mathbf{H}_{\mathbf{xy}})\, \mathbf{C}_2^{-1}\, (\mathbf{G}_{\mathbf{xx}}^{-1}\mathbf{H}_{\mathbf{xy}})^\top$
    $\mathbf{G}^{-1} = \begin{bmatrix} \mathbf{C}_1^{-1} & -\mathbf{C}_1^{-1}\mathbf{H}_{\mathbf{xy}}\mathbf{H}_{\mathbf{yy}}^{-1} \\ -\left(\mathbf{C}_1^{-1}\mathbf{H}_{\mathbf{xy}}\mathbf{H}_{\mathbf{yy}}^{-1}\right)^\top & \mathbf{C}_2^{-1} \end{bmatrix}$
Output: $\mathbf{G}^{-1}$
```

## 4.1 Algorithms

The key idea of designing PQN methods is taking the advantage of the dimensionality-unbalanced structure in the object function. Since the problem holds that $n_x \gg n_y$, we can access the exact second-order information related to $\mathbf{H}_{\mathbf{xy}}(\cdot)$ and $\mathbf{H}_{\mathbf{yy}}(\cdot)$ efficiently. Thus, it is only necessary to approximate the matrix $\mathbf{H}_{\mathbf{xx}}(\cdot)$ whose dimensionality is relatively large.

Assumption 2.1 means the matrix $\mathbf{H}_{\mathbf{xx}}(\cdot)$ is always positive definite, which implies we can apply Broyden family updating to approximate it. Based on the observation above, we propose our PQN methods as shown in Algorithm 2. The Hessian estimator $\mathbf{G}_k \in \mathbb{R}^{n\times n}$ in the algorithm can be partitioned into

$$\mathbf{G}_k = \begin{bmatrix} \mathbf{G}_{\mathbf{xx},k} & \mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k) \\ \mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k)^\top & \mathbf{H}_{\mathbf{yy}}(\mathbf{z}_k) \end{bmatrix} \in \mathbb{R}^{n\times n},$$

where $\mathbf{G}_{\mathbf{xx},k} \in \mathbb{R}^{n_x\times n_x}$, $\mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k) \in \mathbb{R}^{n_x\times n_y}$ and $\mathbf{H}_{\mathbf{yy}}(\mathbf{z}_k) \in \mathbb{R}^{n_y\times n_y}$. Compared with approximating $(\mathbf{H}(\mathbf{z}_k))^2$ [24], directly constructing the estimator for $\mathbf{H}(\mathbf{z}_k)$ is easier to obtain higher accuracy since its condition number is smaller.

The unbalanced dimensionality in the minimax problem means the cost for computing $\mathbf{G}_k^{-1}$ mainly depends on finding the inverse of $\mathbf{G}_{\mathbf{xx},k}$, which can be finished in $O(n_x^2)$ flops like existing quasi-Newton methods [21, 24, 42]. Following Woodbury identity [40], we can obtain $\mathbf{G}_k^{-1}$ with $O(n^2)$ flops by the given inverse of $\mathbf{G}_{\mathbf{xx},k}$ which is shown in Algorithm 1.

*Remark* 4.1. The choice of parameter $\tau_k \in [0,1]$ in Algorithm 2 leads to different types of Broyden family update (6) for approximating $\mathbf{H}_{\mathbf{xx}}(\mathbf{z}_k)$ [32]. Our theoretical analysis will focus on the setting of $\tau_k = 0$, which corresponds to SR1 update.

*Remark* 4.2. Note that we can calculate

$$\begin{aligned}&\mathbf{P}(\mathbf{z}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k)\\ =&\mathbf{H}_{\mathbf{xx}}(\mathbf{z}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) - \mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k)\mathbf{H}_{\mathbf{yy}}^{-1}(\mathbf{z}_k)(\mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k)^\top(\mathbf{x}_{k+1} - \mathbf{x}_k))\end{aligned}$$

in $O(n^2)$ flops, which means $r_k$ can be calculated in $O(n^2)$ flops.

## 4.2 Convergence Analysis

We introduce some notations to simplify the presentation for the analysis of the proposed PQN methods (Algorithm 2). We denote $\mathbf{g}_k \overset{\text{def}}{=} \mathbf{g}(\mathbf{z}_k)$, $\mathbf{H}_k \overset{\text{def}}{=} \mathbf{H}(\mathbf{z}_k)$. We use $\mathbf{g}_{\mathbf{x},k}$, $\mathbf{g}_{\mathbf{y},k}$, $\mathbf{H}_{\mathbf{xx},k}$, $\mathbf{H}_{\mathbf{xy},k}$ and $\mathbf{H}_{\mathbf{yy},k}$ to present $\nabla_{\mathbf{x}} f(\mathbf{z}_k)$, $\nabla_{\mathbf{y}} f(\mathbf{z}_k)$, $\mathbf{H}_{\mathbf{xx}}(\mathbf{z}_k)$, $\mathbf{H}_{\mathbf{xy}}(\mathbf{z}_k)$ and $\mathbf{H}_{\mathbf{yy}}(\mathbf{z}_k)$ respectively. We use $\mathbf{P}_k$ and $\mathbf{C}_k$ to denote $\mathbf{H}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{H}_{\mathbf{xy},k}^\top$ and $\mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{H}_{\mathbf{xy},k}^\top$ respectively. We let $\sigma_{\max}(\cdot)$, $\lambda_{\max}(\cdot)$ and $\lambda_{\min}(\cdot)$ be the largest singular value, the largest eigenvalue and the smallest eigenvalue of the matrix respectively.

**Algorithm 2** Partial-Quasi-Newton (PQN)

```
Input: $\mathbf{G}_{\mathbf{xx},0} \succeq \mathbf{H}_{\mathbf{xx},0}$, $\mathbf{z}_0$, $\tau_k \in [0,1]$ and $M \ge 0$
$\mathbf{G}_0^{-1} = \mathrm{Inverse}\big(\mathbf{G}_{\mathbf{xx},0}^{-1}, \mathbf{H}_{\mathbf{xy},0}, \mathbf{H}_{\mathbf{yy},0}\big)$
for $k = 0, 1, \dots$
    $\mathbf{z}_{k+1} = \mathbf{z}_k - \mathbf{G}_k^{-1}\mathbf{g}(\mathbf{z}_k)$
    $r_k = \langle \mathbf{x}_{k+1} - \mathbf{x}_k, \mathbf{P}(\mathbf{z}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k)\rangle^{1/2} + \|\mathbf{y}_{k+1} - \mathbf{y}_k\|$
    $\tilde{\mathbf{G}}_{\mathbf{xx},k} = (1 + Mr_k)\mathbf{G}_{\mathbf{xx},k}$
    Choose $\mathbf{u}_k$ from
        • Option I (greedy method):
            $\mathbf{u}_k = \begin{cases} \hat{\mathbf{u}}_{\mathbf{H}_{\mathbf{xx}}(\mathbf{z}_{k+1})}(\tilde{\mathbf{G}}_{\mathbf{xx},k}) & \text{if } \tau_k \in (0,1] \\ \bar{\mathbf{u}}_{\mathbf{H}_{\mathbf{xx}}(\mathbf{z}_{k+1})}(\tilde{\mathbf{G}}_{\mathbf{xx},k}) & \text{if } \tau_k = 0 \end{cases}$
        • Option II (random method):
            $\mathbf{u}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \text{or} \quad \mathbf{u}_k \sim \mathrm{Unif}\left(\mathcal{S}^{n_x-1}\right)$
    $\mathbf{G}_{\mathbf{xx},k+1} = \mathrm{Broyd}_{\tau_k}\left(\tilde{\mathbf{G}}_{\mathbf{xx},k}, \mathbf{H}_{\mathbf{xx}}(\mathbf{z}_{k+1}), \mathbf{u}_k\right)$
    $\mathbf{G}_{k+1}^{-1} = \mathrm{Inverse}\big(\mathbf{G}_{\mathbf{xx},k+1}^{-1}, \mathbf{H}_{\mathbf{xy}}(\mathbf{z}_{k+1}), \mathbf{H}_{\mathbf{yy}}(\mathbf{z}_{k+1})\big)$
end for
```

Different from the analysis in the work of existing quasi-Newton methods [22, 24, 42], we design the weighted gradient norm as the measure for our convergence analysis below

$$\gamma_k \overset{\text{def}}{=} \langle \mathbf{g}_{\mathbf{x},k}, \mathbf{P}_k^{-1}\mathbf{g}_{\mathbf{x},k}\rangle^{1/2} + \frac{2}{\sqrt{\mu}} \cdot \|\mathbf{g}_{\mathbf{y},k}\|,$$

where we have $\mathbf{P}_k \succ \mathbf{H}_{\mathbf{xx},k} \succ \mathbf{0}$ due to the fact that $\mathbf{H}_{\mathbf{yy},k}^{-1} \prec \mathbf{0}$.

Now, we establish the relation between $\gamma_{k+1}$ and $\gamma_k$ by the iteration rule $\mathbf{z}_{k+1} = \mathbf{z}_k - \mathbf{G}_k^{-1}\mathbf{g}_k$.

**Lemma 4.3.** *Using Algorithm 2 and assuming that*

$$\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \eta_k \mathbf{H}_{\mathbf{xx},k}$$

*with $\eta_k \ge 1$, then for $\alpha \overset{\text{def}}{=} 54\kappa^2\kappa_2/\mu$, we have*

$$\gamma_{k+1} \le \left(1 - \frac{1}{\eta_k}\right)\gamma_k + \alpha\gamma_k^2 \text{ and } r_k \le \frac{3\gamma_k}{\sqrt{\mu}} \tag{11}$$

Proof. See Appendix A. □

Then we prove that if the norm of $\mathbf{G}_{\mathbf{xx},k}$ is bounded, the matrix $\mathbf{G}_{\mathbf{xx},k+1}$ obtained from the Broyden update

$$\mathbf{G}_{\mathbf{xx},k+1} = \mathrm{Broyd}_{\tau_k}(\tilde{\mathbf{G}}_{\mathbf{xx},k}, \mathbf{H}_{\mathbf{xx},k+1}, \mathbf{u}_k), \tag{12}$$

where $\tilde{\mathbf{G}}_{\mathbf{xx},k} = (1 + Mr_k)\mathbf{G}_{\mathbf{xx},k}$ can be bounded by $\mathbf{H}_{\mathbf{xx},k+1}$.

**Lemma 4.4.** *Using Algorithm 2 and assuming that $\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \eta_k\mathbf{H}_{\mathbf{xx},k}$ for some $\eta_k \ge 1$, $\tilde{\mathbf{G}}_{\mathbf{xx},k} = (1+Mr_k)\mathbf{G}_{\mathbf{xx},k}$ and $M = L_2/\mu^{3/2}$. Then we have $\tilde{\mathbf{G}}_{\mathbf{xx},k} \succeq \mathbf{H}_{\mathbf{xx},k}$ and*

$$\mathbf{H}_{\mathbf{xx},k+1} \preceq \mathbf{G}_{\mathbf{xx},k+1} \preceq (1 + Mr_k)^2\eta_k\mathbf{H}_{\mathbf{xx},k+1}. \tag{13}$$

Proof. We first prove that $\|\mathbf{H}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k+1}\|$ can be bounded by $\|\mathbf{z}_{k+1} - \mathbf{z}_k\|$. Note that the difference $\mathbf{H}_{k+1} - \mathbf{H}_k$ can be written as the form of block matrix as follows

$$\mathbf{H}_{k+1} - \mathbf{H}_k = \begin{bmatrix} \mathbf{H}_{\mathbf{xx},k+1} - \mathbf{H}_{\mathbf{xx},k} & \mathbf{H}_{\mathbf{xy},k+1} - \mathbf{H}_{\mathbf{xy},k} \\ (\mathbf{H}_{\mathbf{xy},k+1} - \mathbf{H}_{\mathbf{xy},k})^\top & \mathbf{H}_{\mathbf{yy},k+1} - \mathbf{H}_{\mathbf{yy},k} \end{bmatrix}.$$

According to Assumption 2.1, we have

$$\|\mathbf{H}_{k+1} - \mathbf{H}_k\| \le L_2 \|\mathbf{z}_{k+1} - \mathbf{z}_k\|.$$

Thus we can obtain

$$\left\|\mathbf{H}_{\mathbf{xx},k+1} - \mathbf{H}_{\mathbf{xx},k}\right\| \le L_2 \|\mathbf{z}_{k+1} - \mathbf{z}_k\| \tag{14}$$

$$\le \frac{L_2}{\sqrt{\mu}} \|\mathbf{P}_k^{1/2}(\mathbf{x}_{k+1} - \mathbf{x}_k)\| + \frac{L_2}{2} \|\mathbf{y}_{k+1} - \mathbf{y}_k\| \le \frac{L_2}{\sqrt{\mu}} r_k. \tag{15}$$

Connecting (14) to the strongly-convex assumption on $\mathbf{x}$ leads to

$$\mathbf{H}_{\mathbf{xx},k+1} - \mathbf{H}_{\mathbf{xx},k} \preceq \frac{L_2}{\sqrt{\mu}} r_k \mathbf{I} \preceq \frac{L_2}{\mu^{3/2}} r_k \mathbf{H}_{\mathbf{xx},k} = M r_k \mathbf{H}_{\mathbf{xx},k}$$

and

$$\mathbf{H}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k+1} \preceq \frac{L_2}{\sqrt{\mu}} r_k \mathbf{I} \preceq \frac{L_2}{\mu^{3/2}} r_k \mathbf{H}_{\mathbf{xx},k+1} = M r_k \mathbf{H}_{\mathbf{xx},k+1}$$

which are equivalent to

$$\frac{\mathbf{H}_{\mathbf{xx},k}}{1 + M r_k} \preceq \mathbf{H}_{\mathbf{xx},k+1} \preceq (1 + M r_k) \mathbf{H}_{\mathbf{xx},k}. \tag{16}$$

We assume that

$$\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \eta_k \mathbf{H}_{\mathbf{xx},k}. \tag{17}$$

Hence, we have

$$\mathbf{H}_{\mathbf{xx},k+1} \overset{(16)}{\preceq} (1 + M r_k) \mathbf{H}_{\mathbf{xx},k} \overset{(17)}{\preceq} (1 + M r_k) \mathbf{G}_{\mathbf{xx},k} = \tilde{\mathbf{G}}_k$$

and

$$\tilde{\mathbf{G}}_k = (1 + M r_k) \mathbf{G}_{\mathbf{xx},k} \overset{(17)}{\preceq} (1 + M r_k) \eta_k \mathbf{H}_{\mathbf{xx},k} \overset{(16)}{\preceq} (1 + M r_k)^2 \eta_k \mathbf{H}_{k+1}.$$

According to Lemma 2.2 in Rodomanov and Nesterov [42], we have

$$\mathbf{H}_{\mathbf{xx},k+1} \preceq \mathrm{Broyd}_\tau(\tilde{\mathbf{G}}_k, \mathbf{H}_{\mathbf{xx},k+1}, \mathbf{u}) = \mathbf{G}_{\mathbf{xx},k+1} \preceq (1 + M r_k)^2 \eta_k \mathbf{H}_{k+1}.$$

□

The linear convergence of PQN methods can be established by Lemma 4.3 and 4.4. The convergence rate matches the result of quasi-Newton method for convex optimization [22].

**Theorem 4.5.** *Using Algorithm 2 and assuming the initial point be sufficiently close to the saddle point such that $\gamma_0 \le \mu/(216\kappa^3\kappa_2)$, then for all $k \ge 0$, we have $\gamma_k \le (1 - 1/4\kappa)^k \gamma_0$.*

Proof. Let $\rho_i = \frac{3L_2}{\mu^2} \gamma_i$. The initial assumption means we have

$$\frac{M\kappa}{\sqrt{\mu}} \gamma_0 \le \frac{\ln 2}{24} \quad \text{and} \quad \alpha\gamma_0 \le \frac{1}{4\kappa}. \tag{18}$$

We use induction to prove the following statements

$$\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \exp\left(2 \sum_{i=0}^{k-1} \rho_i\right) \kappa \mathbf{H}_{\mathbf{xx},k} \preceq 2\kappa \mathbf{H}_{\mathbf{xx},k}, \tag{19}$$

$$\gamma_k \le \left(1 - \frac{1}{4\kappa}\right)^k \gamma_0, \tag{20}$$

$$\eta_k \overset{\text{def}}{=} \exp\left(\sum_{i=0}^{k-1} 2\rho_i\right) \kappa \le 2\kappa \tag{21}$$

hold for all $k \ge 0$.

For $k = 0$, the initialization $\mathbf{G}_{\mathbf{xx},0} = L\mathbf{I}$ leads to $\eta_0 = \kappa$ and $\mathbf{H}_{\mathbf{xx},0} \preceq \mathbf{G}_0 \preceq \kappa \mathbf{H}_{\mathbf{xx},0}$, which satisfy (19), (20) and (21). Suppose the statements (19), (20) and (21) hold for all $k' \le k$, then we prove these results for $k' = k + 1$. The induction assumption means $\eta_k \le 2\kappa$ and $\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \eta_k \mathbf{H}_{\mathbf{xx},k}$. According to inequality (11), we have

$$\gamma_{k+1} \le \left(1 - \frac{1}{2\kappa}\right) \gamma_k + \alpha \gamma_k^2 \overset{(20)}{\le} \left(1 - \frac{1}{2\kappa} + \alpha\gamma_0\right) \gamma_k$$

$$\le \left(1 - \frac{1}{4\kappa}\right) \gamma_k \overset{(20)}{\le} \left(1 - \frac{1}{4\kappa}\right)^k \gamma_0.$$

Recall that we have defined $\rho_i = \frac{3M}{\sqrt{\mu}} \gamma_i$. Based on the elementary inequality $\mathrm{e}^x \ge x + 1$ and Lemma 4.4 of Rodomanov and Nesterov [42], we have

$$\mathbf{H}_{\mathbf{xx},k+1} \preceq \mathbf{G}_{\mathbf{xx},k+1} \preceq (1 + M r_k)^2 \eta_k \mathbf{H}_{\mathbf{xx},k+1} \preceq \left(1 + \frac{3M\gamma_k}{\sqrt{\mu}}\right)^2 \eta_k \mathbf{H}_{\mathbf{xx},k+1}$$

$$= (1 + \rho_k)^2 \eta_k \mathbf{H}_{\mathbf{xx},k+1} \preceq \mathrm{e}^{2\rho_k} \eta_k \mathbf{H}_{\mathbf{xx},k+1} \overset{(19)}{\preceq} \exp\left(2 \sum_{i=0}^{k} \rho_i\right) \kappa \mathbf{H}_{\mathbf{xx},k+1},$$

where the term of $\sum_{i=0}^k \rho_i$ can be bounded by

$$\sum_{i=0}^{k} \rho_i \overset{(20)}{\le} \frac{3M}{\sqrt{\mu}} \gamma_0 \sum_{i=0}^{k} \left(1 - \frac{1}{4\kappa}\right)^{i-1} \le \frac{12M\kappa}{\sqrt{\mu}} \gamma_0 \overset{(18)}{\le} \frac{\ln 2}{2}. \tag{22}$$

Hence, we have $\mathbf{G}_{\mathbf{xx},k+1} \preceq \exp\left(2 \sum_{i=0}^k \rho_i\right) \kappa \mathbf{H}_{\mathbf{xx},k+1} \preceq 2\kappa \mathbf{H}_{\mathbf{xx},k+1}$ and $\eta_{k+1} = \exp\left(2 \sum_{i=0}^k \rho_i\right) \kappa \le 2\kappa$. Then we complete the proof by induction. □

Note that the result of Theorem 4.5 does not depend on the choice of the direction $\mathbf{u}_k$. However, selecting $\mathbf{u}_k$ by the update rule in Algorithm 2 leads to the superlinear local convergence rates. We present the formal statement in the following theorem.

**Theorem 4.6.** *Solving minimax optimization problem* (1) *under Assumption 2.1 and 2.2 by proposed PQN methods (Algorithm 2) with $M = L_2/\mu^{3/2}$ and $\mathbf{G}_{\mathbf{xx},0} = L\mathbf{I}_{n_\mathbf{x}}$, and the initial point is sufficiently close to the saddle point such that $\gamma_0 \le \frac{\ln 2}{24} \cdot \frac{\mu}{\kappa_2 \kappa (1 + \beta_0 n \kappa)}$ where $\beta_0 \overset{\text{def}}{=} \max\{2, \frac{18\kappa}{n}\}$, then we have the following results:*

*(a) If we choose $\tau \in (0, 1]$ then Algorithm 2 (Broyden PQN) holds that*

$$\mathbb{E}\left[\frac{\gamma_{k+1}}{\gamma_k}\right] \le \left(1 - \frac{1}{n\kappa}\right)^k 2n\kappa, \quad \textit{for all } k \ge 0. \tag{23}$$

*(b) If we choose $\tau = 0$ then Algorithm 2 (SR1 PQN) holds that*

$$\mathbb{E}\left[\frac{\gamma_{k+1}}{\gamma_k}\right] \le \left(1 - \frac{1}{n}\right)^k 2n\kappa^2, \quad \textit{for all } k \ge 0. \tag{24}$$

Proof. See Appendix B. □

The choices of $\tau_k$ and $\mathbf{u}_k$ lead to different versions of PQN methods. Combining the local linear convergence result in Theorem 4.5 and the local superlinear convergence results in Theorem 4.6, we obtain the two-period convergence results for these PQN algorithms.

**Corollary 4.7.** *Solving minimax problem under Assumption 2.1 and 2.2 by proposed PQN methods (Algorithm 2) with $M = L_2/\mu^{3/2}$ and $\mathbf{G}_{\mathbf{xx},0} = L\mathbf{I}_{n_\mathbf{x}}$ and the initial point is sufficiently close to the saddle point such that $\gamma_0 \leq \frac{\mu}{216\kappa^3\kappa_2}$, we have the following results:*

- (a) *If we choose $\tau \in (0, 1]$ and use the random method (10) to determine $\mathbf{u}_k$, then Algorithm 2 (random Broyden PQN) holds that*

$$\gamma_{k_0+k} \leq \left(1 - \frac{1}{n\kappa + 1}\right)^{k(k-1)/2} \left(\frac{1}{2}\right)^k \left(1 - \frac{1}{4\kappa}\right)^{k_0} \gamma_0$$

*for all $k \geq 0$ and $k_0 = \mathcal{O}\left(n\kappa \ln\left(\frac{n\kappa}{\delta}\right)\right)$ with probability $1 - \delta$, where $\delta \in (0, 1)$.*

- (b) *If we choose $\tau \in (0, 1]$ and use the greedy method (8) to determine $\mathbf{u}_k$, then Algorithm 2 (greedy Broyden PQN) holds that*

$$\gamma_{k_0+k} \leq \left(1 - \frac{1}{n\kappa}\right)^{k(k-1)/2} \left(\frac{1}{2}\right)^k \left(1 - \frac{1}{4\kappa}\right)^{k_0} \gamma_0$$

*for all $k \geq 0$ and $k_0 = \mathcal{O}\left(n\kappa \ln(n\kappa)\right)$.*

- (c) *If we choose $\tau = 0$ and use the random method (10) to determine $\mathbf{u}_k$, then Algorithm 2 (random SR1 PQN) holds that*

$$\gamma_{k_0+k} \leq \left(1 - \frac{1}{n + 1}\right)^{k(k-1)/2} \left(\frac{1}{2}\right)^k \left(1 - \frac{1}{4\kappa}\right)^{k_0} \gamma_0$$

*for all $k \geq 0$ and $k_0 = \mathcal{O}\left((\kappa + n) \ln\left(\frac{n\kappa}{\delta}\right)\right)$ with probability $1 - \delta$, where $\delta \in (0, 1)$.*

- (d) *If we choose $\tau = 0$ and use the greedy method (9) to determine $\mathbf{u}_k$, then Algorithm 2 (greedy SR1 PQN) holds that*

$$\gamma_{k_0+k} \leq \left(1 - \frac{1}{n}\right)^{k(k-1)/2} \left(\frac{1}{2}\right)^k \left(1 - \frac{1}{4\kappa}\right)^{k_0} \gamma_0$$

*for all $k \geq 0$ and $k_0 = \mathcal{O}\left((\kappa + n) \ln(n\kappa)\right)$.*

Proof. Theorem 4.5 and 4.6 have shown the local linear and superlinear convergence rate for two periods of the algorithms. We can prove this corollary by combining these two theorems. The statement of cases (a), (b) and (d) can be easily obtained by following the proof of existing quasi-Newton methods [22, 42] as follow:

- For case (a), Theorem 4.5 and 4.6 mean we can use Corollary 11 of Lin et al. [22] by replacing the term $(1 - 1/(2\kappa))$ by $(1 - 1/(4\kappa))$, which directly obtains the two-period convergence result for our random Broyden PQN method.
- For case (c) and (d), Theorem 4.5 and 4.6 mean we can use Corollary 21 of Lin et al. [22] by replacing the term $(1 - 1/(2\kappa))$ by $(1 - 1/(4\kappa))$, which directly obtains the two-period convergence result for our random and greedy SR1 PQN methods.

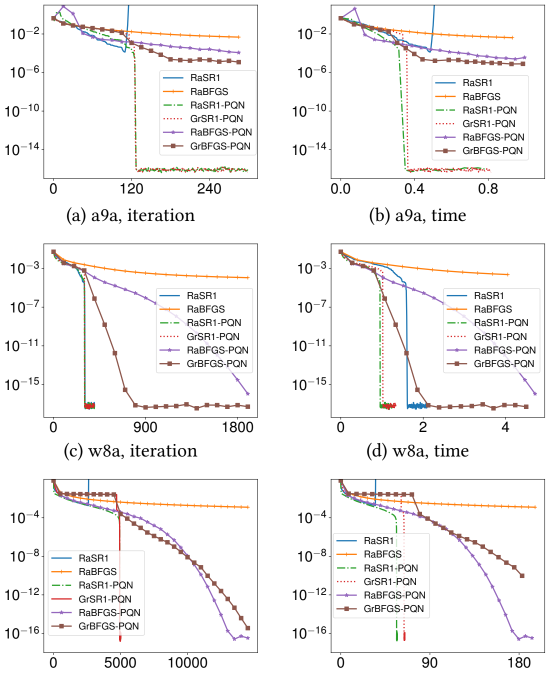
We provide the proof of case (b) as an example. We denote $k_1 \geq 0$ as the number of the first iteration which satisfies

$$\left(1 - \frac{1}{4\kappa}\right)^{k_1} \leq \frac{\kappa^2}{\beta_0 n\kappa + 1}.$$

Thus we have

$$\gamma_{k_1} \leq \left(1 - \frac{1}{4\kappa}\right)^{k_1} \gamma_0 \leq \frac{\ln 2}{24} \cdot \frac{\mu}{\kappa_2\kappa(1 + \beta_0 n\kappa)},$$

which means our algorithm satisfies the initial condition for the superlinear convergence of Theorem 4.6 after $k_1$ iterations where $k_1 = \mathcal{O}\left(\max\left\{\kappa, \kappa \ln\left(\frac{n}{\kappa}\right)\right\}\right)$.



(a) a9a, iteration

(b) a9a, time

(c) w8a, iteration

(d) w8a, time

(e) sido0, iteration

(f) sido0, time

**Figure 1: We demonstrate iteration numbers vs. $\|\mathrm{g}(\mathrm{z})\|_2$ and GPU time (second) vs. $\|\mathrm{g}(\mathrm{z})\|_2$ for AUC maximization on datasets "a9a" ($n = 126$, $m = 32561$), "w8a" ($n = 303$, $m = 45546$) and "sido" ($n = 4935$, $m = 12678$).**

We denote $k_2 \geq 0$ as the number of the first iteration satisfies

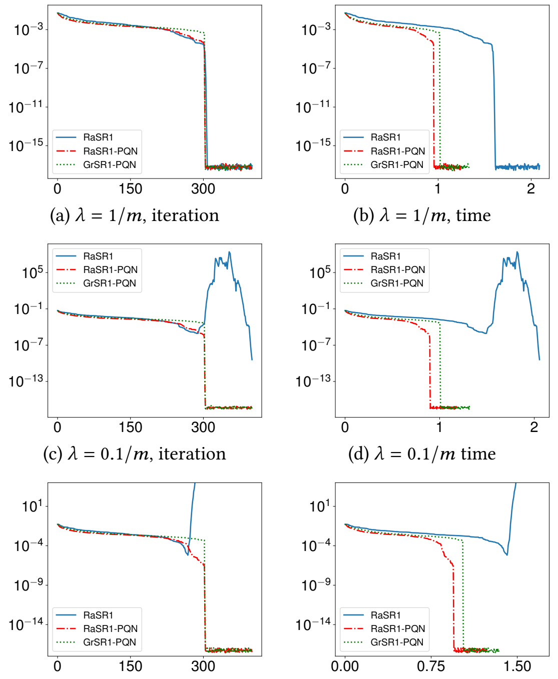
$$2n\kappa\left(1 - \frac{1}{n\kappa}\right)^{k_2} \leq \frac{1}{2},$$

where we have $k_2 = \mathcal{O}\left(n\kappa \ln(n\kappa)\right)$.

We obtain the result of (b) by setting $k_0 = k_1 + k_2 = \mathcal{O}\left(\kappa n \ln(n\kappa)\right)$. □

## 5 NUMERICAL EXPERIMENTS

In this section, we conduct our algorithms on popular machine learning applications and regularized nonlinear minimax optimization problem. We refer to random and greedy versions of proposed Algorithm 2 by choosing $\tau_k = 0$ as RaSR1-PQN and GrSR1-PQN. We also refer to random and greedy versions of Algorithm 2 by choosing $\tau_k = \mathbf{u}_k^\top \mathbf{H}_{\mathbf{xx},k+1}\mathbf{u}_k / (\mathbf{u}_k^\top \mathbf{G}_{\mathbf{xx},k}\mathbf{u}_k)$ as RaBFGS-PQN and GrBFGS-PQN respectively. We use the RaSR1 [24, Algorithm 7] and RaBFGSv1 methods [24, Algroithm 5] as baselines and refer them to RaSR1 and RaBFGS in our experiments respectively. We do not include the first-order method extragradient since the experiments of Liu and Luo [24] have already shown its performance is worse than their quasi-Newton methods.



(a) $\lambda = 1/m$, iteration

(b) $\lambda = 1/m$, time

(c) $\lambda = 0.1/m$, iteration

(d) $\lambda = 0.1/m$ time

(e) $\lambda = 0.01/m$, iteration

(f) $\lambda = 0.01/m$, time

**Figure 2: We demonstrate iteration numbers vs. $\|\mathbf{g}(\mathbf{z})\|_2$ and GPU time (second) vs. $\|\mathbf{g}(\mathbf{z})\|_2$ for AUC maximization on datasets "w8a" ($n = 303$, $m = 45546$) with different regularized parameter $\lambda$.**

Our experiments are conducted on a workstation with NVIDIA Tesla V100 GPU with 16GB memory. We use PyTorch 1.8.0 to run the code and the operating system is Ubuntu 20.04.2.

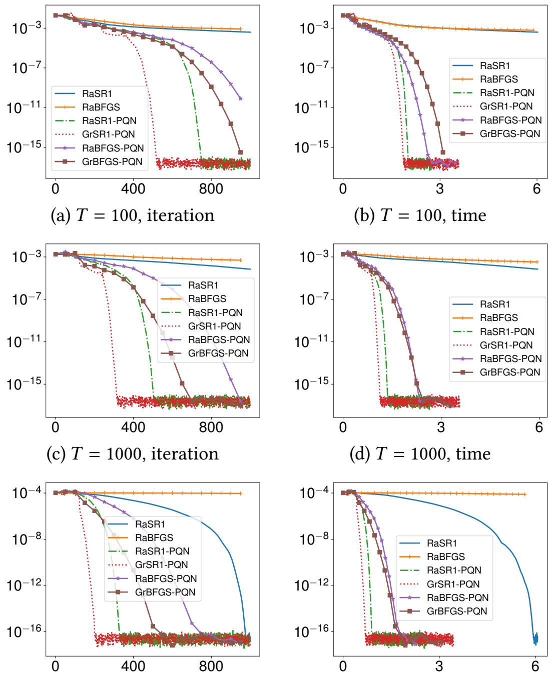
## 5.1 AUC Maximization

We first validate our PQN methods on AUC maximization. Each component $f_i$ in (2) is a quadratic function of the form

$$f_i(\mathbf{x}, y; \mathbf{a}_i, b_i) = (1-p)\big((\mathbf{w}^\top \mathbf{a}_i - u)^2 - 2(1+y)\mathbf{w}^\top \mathbf{a}_i\big)\mathbb{I}_{b_i=1} \\ - p(1-p)y^2 + p\big((\mathbf{w}^\top \mathbf{a}_i - v)^2 + 2(1+y)\mathbf{w}^\top \mathbf{a}_i\big)\mathbb{I}_{b_i=-1},$$

where $\mathbf{x} = [\mathbf{w}; u; v] \in \mathbb{R}^{d+2}$, $\mathbf{w} \in \mathbb{R}^d$, $u \in \mathbb{R}$, $v \in \mathbb{R}$, $p = m^+/m$ and $m^+$ is the number of positive instances. We have $n_x = d + 2$ and $n_y = 1$. Since the object function is quadratic which satisfies Assumption 2.1 with $L_2 = 0$, we set $M = L_2/\mu = 0$ for PQN methods by following the setting of Liu and Luo [24]. We tune the input $L$ from $\{1, 100, 1000\}$ for all algorithms (including the baselines).

We first let $\lambda = 1/m$ and evaluate all algorithms on three imbalanced binary classification datasets "a9a" ($p = 0.241$), "w8a" ($p = 0.029$) and "sido0" ($p = 0.0036$) where "sido0" comes from Causality Workbench [15] and the others can be downloaded from LIBSVM repository [6]. The results of iteration numbers against $\|\mathbf{g}(\mathbf{z})\|_2$ and GPU time against $\|\mathbf{g}(\mathbf{z})\|_2$ are presented in Figure 1, which show that our PQN algorithms performance significantly better than baseline algorithms. We observe that the baseline algorithm RaSR1 diverges for the dataset "a9a" and "sido0". This is because the SR1 method is not numerical stable [2, 10, 53] when the condition number is large. Recall that RaSR1 [24] address the square of Hessian, which leads to the approximate second-order information comes from more ill-conditioned matrices, while our PQN algorithm directly deals with the original Hessian.



(a) $T = 100$, iteration

(b) $T = 100$, time

(c) $T = 1000$, iteration

(d) $T = 1000$, time

(e) $T = 10000$, iteration

(f) $T = 10000$, time

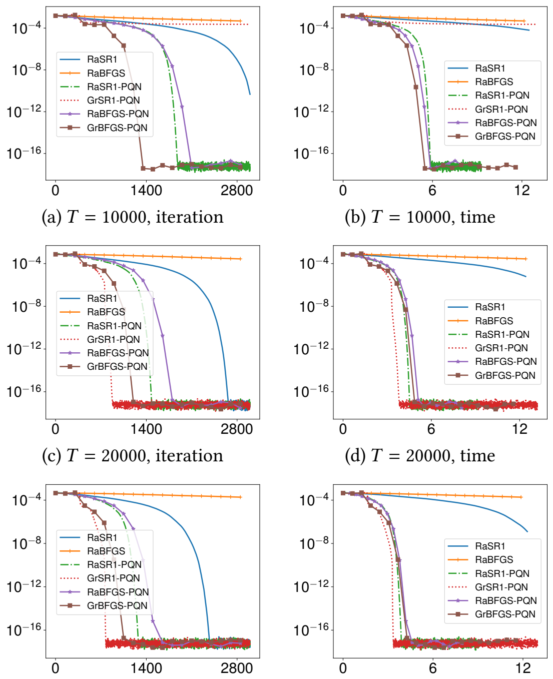
**Figure 3: We demonstrate iteration numbers vs. $\|\mathbf{g}(\mathbf{z})\|_2$ and GPU time (second) vs. $\|\mathbf{g}(\mathbf{z})\|_2$ for Fairness-aware machine learning on datasets "adult" ($n = 123$, $m = 32561$) with different rounds of extragradient iteration as warm-up.**

We also compare our RaSR1-PQN and GrSR1-PQN with RaSR1 under different settings of regularization parameter $\lambda$, which correspond to the SCSC minimax problem with different condition numbers. The results presented in Figure 2 show our algorithms are more stable than baseline RaSR1.

## 5.2 Fairness-Aware Machine Learning

Then we validate PQN-methods on the fairness-aware machine learning model defined in (3). We have $n_x = d$ and $n_y = 1$. Our experiments focus on fairness-aware binary classification such that both $l_1(\cdot)$ and $l_2(\cdot)$ are logit functions: $\text{logit}(x) = \ln(1 + \exp(-x))$. We use the fairness-aware datasets "adults" and "law school" which can be found in Quy et al.'s survey [39]. We convert all features of the original datasets into binary by following the prepossessing of previous work [6, 24, 36]. We set $\lambda = \gamma = 0.0001$ and $\beta = 0.5$.

We tune $M$ and $L$ from $\{1, 10, 100\}$ for all algorithms. Since the object function is nonlinear, we use the first-order algorithm extragradient [19, 48] as warm-up to achieve a good initial point,



(a) $T$ = 10000, iteration (b) $T$ = 10000, time

(c) $T$ = 20000, iteration (d) $T$ = 20000, time

(e) $T$ = 30000, iteration (f) $T$ = 30000, time

**Figure 4: We demonstrate iteration numbers vs. $\|g(z)\|_2$ and GPU time (second) vs. $\|g(z)\|_2$ for fairness-aware machine learning on datasets "law school" ($n$ = 380, $m$ = 20427) with different rounds of extragradient iteration as warm-up.**

which is a popular strategy in the numerical experiments for quasi-Newton methods [21, 24]. We run all algorithms with $T$ rounds of extragradient iteration as warm-up, where $T$ is selected from $\{100, 1000, 10000\}$ for "adult" dataset and $\{10000, 20000, 30000\}$ for "law school" dataset. The results presented in Figure 3 and 4 show that our PQN methods outperform the baselines under all of the warm-up settings, which implies our algorithms are more robust to the choice of initial point.

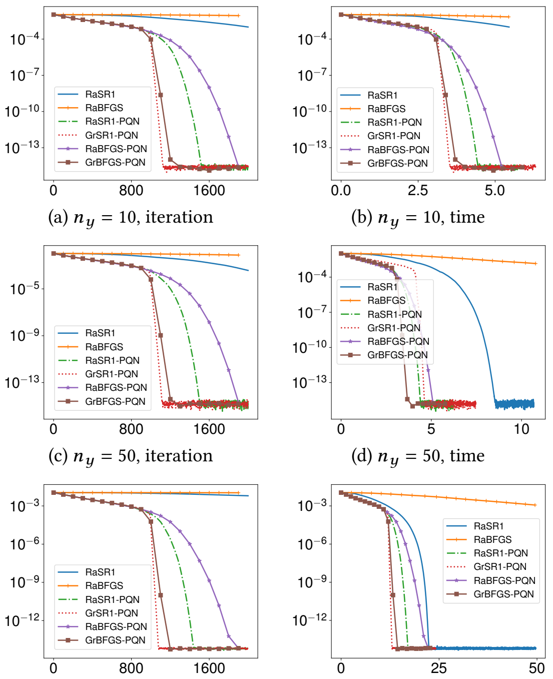
## 5.3 Regularized Nonlinear Minimax Problem

This section studies how the difference between dimensionalities of two variables in the problem affect the performance of algorithms. We consider the following minimax problem [17]

$$\min_{\mathbf{x}\in\mathbb{R}^{n_x}} \max_{y\in\mathbb{R}^{n_y}} f(\mathbf{x},\mathbf{y}) = \frac{1}{m_1}\sum_{i=1}^{m_1} \ln\left(1+\exp\left(-\mathbf{a}_i^\top \mathbf{x}\right)\right) + \frac{\lambda}{2}\|\mathbf{x}\|^2 + \mathbf{x}^\top \mathbf{A}\mathbf{y} - \frac{1}{m_2}\sum_{j=1}^{m_2} \ln\left(1+\exp\left(-\mathbf{b}_j^\top \mathbf{y}\right)\right) - \frac{\gamma}{2}\|\mathbf{y}\|^2,$$

where $\mathbf{a}_i \in \mathbb{R}^{n_x}$, $\mathbf{b}_j \in \mathbb{R}^{n_y}$ and $\mathbf{A} \in \mathbb{R}^{n_x \times n_y}$ are generated randomly by the Pytorch function `torch.randn()` with corresponding dimensions and $\lambda, \gamma > 0$ are the regularized parameters.

We set $\lambda = \gamma = 0.01$ and fix $n_x = 1000$ for our experiments. The inputs $M$ and $L$ are tuned from $\{1, 10, 100, 1000, 10000\}$ for all algorithms. We test all algorithms under different settings of the dimensionality of $\mathbf{y}$, that is, we select $n_y$ from $\{10, 50, 100\}$. We present the experimental results in Figure 5, which show our PQN methods always perform better than baselines in all cases.



(a) $n_y$ = 10, iteration (b) $n_y$ = 10, time

(c) $n_y$ = 50, iteration (d) $n_y$ = 50, time

(e) $n_y$ = 100, iteration (f) $n_y$ = 100, time

**Figure 5: We demonstrate iteration numbers vs. $\|g(z)\|_2$ and GPU time (second) vs. $\|g(z)\|_2$ for regularized nonlinear minimax problem with different settings for $n_y$.**

## 6 CONCLUSION

In this work, we have proposed the partial-quasi-Newton (PQN) methods for solving the SCSC minimax optimization problems with unbalanced dimensionality. The algorithms only approximate the positive-definite block in the Hessian matrix and compute the other second-order information exactly. The unbalanced structure in the problem allows we can efficiently update the Hessian estimator with the iteration. We prove our PQN methods enjoy better convergence rates than existing quasi-Newton methods for SCSC minimax optimization. The empirical studies on popular machine learning applications and synthetic minimax problems show PQN method perform significantly better than the state-of-the-art algorithms.

The framework of PQN methods is not only limited to quasi-Newton-type algorithms, we can also incorporate the idea of PQN into other classes of inexact Newton methods [12, 35, 45, 45, 51, 52].

## 7 ACKNOWLEDGEMENTS

We would like to thank Yong Lin for discussing the experiments. Luo Luo is supported by Shanghai Sailing Program (22YF1402900). John C.S. Lui is supported in part by the GRF 14200321.

# REFERENCES

[1] Ahmet Alacaoglu and Yura Malitsky. Stochastic variance reduction for variational inequality methods. *arXiv preprint arXiv:2102.08352*, 2021.

[2] Charles G. Broyden. Quasi-Newton methods and their application to function minimisation. *Mathematics of Computation*, 21(99):368–381, 1967.

[3] Charles G. Broyden. The convergence of a class of double-rank minimization algorithms 1. general considerations. *IMA Journal of Applied Mathematics*, 6(1): 76–90, 1970.

[4] Charles G. Broyden. The convergence of a class of double-rank minimization algorithms: 2. the new algorithm. *IMA journal of applied mathematics*, 6(3): 222–231, 1970.

[5] Charles G. Broyden, J. E. Dennis, and Jorge J. Moré. On the local and superlinear convergence of quasi-Newton methods. *IMA Journal of Applied Mathematics*, 12 (3):223–245, 1973.

[6] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software and datasets available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[7] Tatjana Chavdarova, Gauthier Gidel, Franccois Fleuret, and Simon Lacoste-Julien. Reducing noise in GAN training with variance reduced extragradient. In *NeurIPS*, 2019.

[8] Corinna Cortes and Mehryar Mohri. Auc optimization vs. error rate minimization. In *NIPS*, 2003.

[9] Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *ICLR*, 2018.

[10] William C. Davidon. Variable metric method for minimization. *SIAM Journal on Optimization*, 1(1):1–17, 1991.

[11] J. E. Dennis, Jr., and Jorge J. Moré. A characterization of superlinear convergence and its application to quasi-Newton methods. *Mathematics of Computation*, 28 (126):549–560, 1974.

[12] Murat A Erdogdu and Andrea Montanari. Convergence rates of sub-sampled newton methods. *NIPS*, 2015.

[13] Roger Fletcher and Micheal JD Powell. A rapidly convergent descent method for minimization. *The Computer Journal*, 6:163–168, 1963.

[14] Zhishuai Guo, Mingrui Liu, Zhuoning Yuan, Li Shen, Wei Liu, and Tianbao Yang. Communication-efficient distributed stochastic AUC maximization with deep neural networks. In *ICML*, 2020.

[15] Isabelle Guyon, Constantin Aliferis, Greg Cooper, André Elisseeff, Jean-Philippe Pellet, Peter Spirtes, and Alexander Statnikov. Design and analysis of the causation and prediction challenge. In *Causation and Prediction Challenge*, pages 1–33. PMLR, 2008. Dataset available at http://www.causality.inf.ethz.ch/data/SIDO.html.

[16] James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982.

[17] Kevin Huang, Junyu Zhang, and Shuzhong Zhang. Cubic regularized Newton method for saddle point models: a global and local convergence analysis. *arXiv preprint arXiv:2008.09919*, 2020.

[18] Qiujiang Jin and Aryan Mokhtari. Non-asymptotic superlinear convergence of standard quasi-Newton methods. *arXiv preprint arXiv:2003.13607*, 2020.

[19] G. M. Korpelevich. An extragradient method for finding saddle points and for other problems. *Matecon*, 12:747–756, 1976.

[20] Ching-pei Lee, Cong Han Lim, and Stephen J. Wright. A distributed quasi-newton algorithm for empirical risk minimization with nonsmooth regularization. In *SIGKDD*, 2018.

[21] Dachao Lin, Haishan Ye, and Zhihua Zhang. Explicit superlinear convergence rates of Broyden's methods in nonlinear equations. *arXiv preprint arXiv:2109.01974*, 2021.

[22] Dachao Lin, Haishan Ye, and Zhihua Zhang. Explicit convergence rates of greedy and random quasi-Newton methods. *arXiv preprint arXiv:2104.08764*, 2021.

[23] Tianyi Lin, Chi Jin, and Michael I. Jordan. Near-optimal algorithms for minimax optimization. In *COLT*, 2020.

[24] Chengchang Liu and Luo Luo. Quasi-Newton methods for saddle point problems and beyond. *arXiv preprint arXiv:2111.02708*, 2021.

[25] Mingrui Liu, Zhuoning Yuan, Yiming Ying, and Tianbao Yang. Stochastic AUC maximization with deep neural networks. *arXiv preprint arXiv:1908.10831*, 2019.

[26] Daniel Lowd and Christopher Meek. Adversarial learning. In *SIGKDD*, 2005.

[27] Luo Luo, Cheng Chen, Yujun Li, Guangzeng Xie, and Zhihua Zhang. A stochastic proximal point algorithm for saddle-point problems. *arXiv preprint arXiv:1909.06946*, 2019.

[28] Luo Luo, Guangzeng Xie, Tong Zhang, and Zhihua Zhang. Near optimal stochastic algorithms for finite-sum unbalanced convex-concave minimax optimization. *arXiv preprint arXiv:2106.01761*, 2021.

[29] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. In *AISTATA*, 2020.

[30] Yurii Nesterov. Accelerating the Cubic regularization of newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

[31] Yurii Nesterov and Laura Scrimali. Solving strongly monotone variational and quasi-variational inequalities. *Discrete and Continuous Dynamical Systems*, 31(4): 1383–1396, 2007.

[32] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

[33] Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *arXiv preprint:1808.02901*, 2018.

[34] Balamurugan Palaniappan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *NIPS*, 2016.

[35] Mert Pilanci and Martin J. Wainwright. Newton sketch: A near linear-time optimization algorithm with linear-quadratic convergence. *SIAM Journal on Optimization*, 27(1):205–245, 2017.

[36] John C. Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods-Support Vector Learning*, 1998.

[37] Leonid Denisovich Popov. A modification of the arrow-hurwicz method for search of saddle points. *Mathematical notes of the Academy of Sciences of the USSR*, 28(5):845–848, 1980.

[38] M. J. D. Powell. On the convergence of the variable metric algorithm. *IMA Journal of Applied Mathematics*, 7(1):21–36, 1971.

[39] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, and Eirini Ntoutsi. A survey on datasets for fairness-aware machine learning, 2021.

[40] Kurt S Riedel. A sherman–morrison–woodbury identity for rank augmenting matrices with application to centering. *SIAM Journal on Matrix Analysis and Applications*, 13(2):659–662, 1992.

[41] R. Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976.

[42] Anton Rodomanov and Yurii Nesterov. Greedy quasi-Newton methods with explicit superlinear convergence. *SIAM Journal on Optimization*, 31(1):785–811, 2021.

[43] Anton Rodomanov and Yurii Nesterov. New results on superlinear convergence of classical quasi-Newton methods. *Journal of optimization theory and applications*, 188(3):744–769, 2021.

[44] Anton Rodomanov and Yurii Nesterov. Rates of superlinear convergence for classical quasi-Newton methods. *Mathematical Programming*, pages 1–32, 2021.

[45] Farbod Roosta-Khorasani and Michael W Mahoney. Sub-sampled Newton methods. *Mathematical Programming*, 174(1):293–326, 2019.

[46] David F. Shanno. Conditioning of quasi-Newton methods for function minimization. *Mathematics of computation*, 24(111):647–656, 1970.

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

[48] Paul Tseng. On linear convergence of iterative methods for the variational inequality problem. *Journal of Computational and Applied Mathematics*, 60(1-2): 237–252, 1995.

[49] Yuanhao Wang and Jian Li. Improved algorithms for convex-concave minimax optimization. In *NeurIPS*, 2020.

[50] Guangzeng Xie, Luo Luo, Yijiang Lian, and Zhihua Zhang. Lower complexity bounds for finite-sum convex-concave minimax optimization problems. In *ICML*, 2020.

[51] Peng Xu, Jiyan Yang, Fred Roosta, Christopher Ré, and Michael W. Mahoney. Sub-sampled Newton methods with non-uniform sampling. *NIPS*, 2016.

[52] Haishan Ye, Luo Luo, and Zhihua Zhang. Approximate Newton methods and their local convergence. In *ICML*, 2017.

[53] Haishan Ye, Dachao Lin, Zhihua Zhang, and Xiangyu Chang. Explicit superlinear convergence rates of the SR1 algorithm. *arXiv preprint arXiv:2105.07162*, 2021.

[54] Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online AUC maximization. *NIPS*, 2016.

[55] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *AIES*, 2018.

# A THE PROOF OF LEMMA 4.3

PROOF. We use $\mathbf{J}_k \overset{\text{def}}{=} \begin{bmatrix} \mathbf{P}_k^{1/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_y} \end{bmatrix}$ in the following analysis.

We first give the bounds for some matrices which will be useful in the analysis. We have $\mathbf{G}_{\mathbf{xx},k} \succeq \mu\mathbf{I}$ and $\mathbf{H}_{\mathbf{yy},k} \preceq -\mu\mathbf{I}$, then $\mathbf{G}_k^2 \succeq \mu^2\mathbf{I}$, which implies $\lambda_{\min}(\mathbf{G}_k^2) \geq \mu^2$ and $\|\mathbf{G}_k^{-1}\| = \lambda_{\max}((\mathbf{G}_k^{-1})^2) \leq 1/\mu^2$. Hence, we have $\|\mathbf{G}_k^{-1}\| = \sqrt{\lambda_{\max}((\mathbf{G}_k^{-1})^2)} \leq 1/\mu$.

Recall that we've defined $\mathbf{C}_k = \mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{H}_{\mathbf{xy},k}^\top$ and according to the assumption that $\mathbf{H}_{\mathbf{xx},k} \preceq \mathbf{G}_{\mathbf{xx},k} \preceq \eta_k\mathbf{H}_{\mathbf{xx},k}$, we have $\mathbf{P}_k \preceq \mathbf{C}_k \preceq \eta_k\mathbf{P}_k$ and combining with Lemma B.1 of Liu and Luo [24], we have

$$\left\|\mathbf{I} - \mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\right\| \leq 1 - \frac{1}{\eta_k} \text{ and } \left\|\mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\right\| \leq 1. \tag{25}$$

If we further define the matrices $\mathbf{Q}_k \overset{\text{def}}{=} \mathbf{H}_{\mathbf{yy},k} - \mathbf{H}_{\mathbf{yx},k}\mathbf{H}_{\mathbf{xx},k}^{-1}\mathbf{H}_{\mathbf{xy},k}$ and $\mathbf{B}_k \overset{\text{def}}{=} \mathbf{H}_{\mathbf{yy},k} - \mathbf{H}_{\mathbf{yx},k}\mathbf{G}_{\mathbf{xx},k}^{-1}\mathbf{H}_{\mathbf{xy},k}$, we can easily obtain that $\|\mathbf{Q}_k^{-1}\| \leq 1/\mu$ and $\|\mathbf{B}_k^{-1}\| \leq 1/\mu$. According to the Woodbury identity, we have

$$\mathbf{Q}_k^{-1} = \mathbf{H}_{\mathbf{yy},k}^{-1} + \mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{H}_{\mathbf{yx},k}\mathbf{P}_k^{-1}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1},$$

thus we have

$$\begin{aligned} \|\mathbf{P}_k^{-1/2}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\| &= \sqrt{\lambda_{\max}(\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{H}_{\mathbf{xy},k}^\top\mathbf{P}_k^{-1}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1})} \\ =&\sqrt{\|\mathbf{Q}_k^{-1} - \mathbf{H}_{\mathbf{yy},k}^{-1}\|} \leq \sqrt{\|\mathbf{Q}_k^{-1}\| + \|\mathbf{H}_{\mathbf{yy},k}^{-1}\|} \leq \frac{2}{\sqrt{\mu}}. \end{aligned} \tag{26}$$

We can obtain that $\mu\mathbf{I} \preceq \mathbf{P}_k \preceq (2\kappa/\mu)\mathbf{I}$ and $\mathbf{P}(\mathbf{z})$ is $3\kappa^2L_2$-lipschitz countinous which means

$$\frac{\mathbf{P}_k}{1 + 3\kappa^2\kappa_2 r_k/\sqrt{\mu}} \preceq \mathbf{P}_{k+1} \preceq \left(1 + \frac{3\kappa^2\kappa_2 r_k}{\sqrt{\mu}}\right)\mathbf{P}_k. \tag{27}$$

We rewrite $\nabla f(\mathbf{z}_{k+1})$ according to the iteration (12) as

$$\begin{aligned} \nabla f(\mathbf{z}_{k+1}) = &\underbrace{\nabla f(\mathbf{z}_k) + \mathbf{H}_k(-\mathbf{G}_k^{-1}\nabla f(\mathbf{z}_k))}_{\mathbf{a}_k} \\ &+ \underbrace{\int_0^1 \left(\nabla^2 f(\mathbf{z}_k + s(\mathbf{z}_{k+1} - \mathbf{z}_k)) - \nabla^2 f(\mathbf{z}_k)\right)(\mathbf{z}_{k+1} - \mathbf{z}_k)\,\mathrm{d}s}_{\mathbf{b}_k}. \end{aligned}$$

The term of $\mathbf{a}_k$ can be written as

$$\mathbf{a}_k = (\mathbf{G}_k - \mathbf{H}_k)\mathbf{G}_k^{-1}\mathbf{g}_k = \begin{bmatrix} \mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{G}_k^{-1}\mathbf{g}_k.$$

We define $\mathbf{a}_k \overset{\text{def}}{=} [\mathbf{a}_{\mathbf{x},k}^\top; \mathbf{a}_{\mathbf{y},k}^\top]^\top$, and we have

$$\begin{bmatrix} \mathbf{a}_{\mathbf{x},k} \\ \mathbf{a}_{\mathbf{y},k} \end{bmatrix} = \begin{bmatrix} (\mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k})\mathbf{C}_k^{-1}\mathbf{g}_{\mathbf{x},k} - (\mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k})\mathbf{C}_k^{-1}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}\mathbf{g}_{\mathbf{y},k} \\ \mathbf{0} \end{bmatrix}.$$

We bound the term of $\mathbf{P}_{\mathbf{x},k}^{-1/2}\mathbf{a}_{\mathbf{x},k}$ by

$$\begin{aligned} \|\mathbf{P}_k^{-1/2}\mathbf{a}_{\mathbf{x},k}\| &\leq \|\mathbf{I} - \mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\|\|\mathbf{P}_k^{-1/2}\mathbf{g}_{\mathbf{x},k}\| \\ &+ \|\mathbf{I} - \mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\|\|\mathbf{P}_k^{-1/2}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{g}_{\mathbf{y},k}\| \\ &\overset{(25),\,(26)}{\leq} \left(1 - \frac{1}{\eta_k}\right)\|\mathbf{P}_k^{-1/2}\mathbf{g}_{\mathbf{x},k}\| + \left(1 - \frac{1}{\eta_k}\right)\frac{2}{\sqrt{\mu}}\|\mathbf{g}_{\mathbf{y},k}\|. \end{aligned} \tag{28}$$

The term of $\mathbf{b}_k \overset{\text{def}}{=} [\mathbf{b}'_{\mathbf{x},k}; \mathbf{b}'_{\mathbf{y},k}]^\top$ can be bounded by Lipschitz-continuous of $\mathbf{H}(\mathbf{z})$ that is

$$\begin{aligned} \|\mathbf{b}_k\| &\leq \int_0^1 \left\|\nabla^2 f(\mathbf{z}_k + s(\mathbf{z}_{k+1} - \mathbf{z}_k)) - \nabla^2 f(\mathbf{z}_k)\right\| \|(\mathbf{z}_{k+1} - \mathbf{z}_k)\|\,\mathrm{d}s \\ &\overset{(4)}{\leq} \frac{L_2}{2}\|\mathbf{z}_{k+1} - \mathbf{z}_k\|^2 \leq \frac{L_2}{2\mu}r_k^2. \end{aligned} \tag{29}$$

We further denote $\mathbf{h}_k = \begin{bmatrix} \mathbf{h}_{\mathbf{x},k} \\ \mathbf{h}_{\mathbf{y},k} \end{bmatrix} \overset{\text{def}}{=} \mathbf{J}_k^{-1}\mathbf{g}_k = \begin{bmatrix} \mathbf{P}_k^{-1/2}\mathbf{g}_{\mathbf{x},k} \\ \mathbf{g}_{\mathbf{y},k} \end{bmatrix}$. Then, we can bound $\mathbf{h}_{\mathbf{x},k+1}$ and $\mathbf{h}_{\mathbf{y},k+1}$ as follows:

$$\|\mathbf{h}_{\mathbf{y},k+1}\| \leq \|\mathbf{b}_{\mathbf{y},k}\| \leq \|\mathbf{b}_k\| \leq \frac{L_2}{2\mu}r_k^2$$

and

$$\begin{aligned} &\|\mathbf{h}_{\mathbf{x},k+1}\| \leq \|\mathbf{P}_{k+1}^{-1/2}\mathbf{a}_{\mathbf{x},k}\| + \|\mathbf{P}_{k+1}^{-1/2}\mathbf{b}_{\mathbf{x},k}\| \\ &\overset{(27)}{\leq} \sqrt{1 + \frac{3\kappa^2\kappa_2}{\sqrt{\mu}}r_k}\|\mathbf{P}_k^{-1/2}\mathbf{a}_{\mathbf{x},k}\| + \|\mathbf{P}_{k+1}^{-1/2}\|\|\mathbf{b}_{\mathbf{x},k}\| \\ &\overset{(28),\,(29)}{\leq} \sqrt{1 + \frac{3\kappa^2\kappa_2}{\sqrt{\mu}}r_k}\left(1 - \frac{1}{\eta_k}\right)\left(\|\mathbf{h}_{\mathbf{x},k}\| + \frac{2}{\sqrt{\mu}}\|\mathbf{h}_{\mathbf{y},k}\|\right) + \frac{L_2}{2\sqrt{\mu}\mu}r_k^2. \end{aligned}$$

Thus, we have

$$\begin{aligned} &\|\mathbf{h}_{\mathbf{x},k+1}\| + (2/\sqrt{\mu})\|\mathbf{h}_{\mathbf{y},k+1}\| \\ &\leq \sqrt{1 + 3\kappa^2\kappa_2 r_k/\sqrt{\mu}}\left(1 - \frac{1}{\eta_k}\right)\left(\|\mathbf{h}_{\mathbf{x},k}\| + \frac{2}{\sqrt{\mu}}\|\mathbf{h}_{\mathbf{y},k}\|\right) + \frac{L_2}{\sqrt{\mu}\mu}r_k^2. \end{aligned}$$

Finally, we bound $r_k$ by $\gamma_k$ as follows

$$\begin{aligned} \mathbf{J}_k(\mathbf{z}_{k+1} - \mathbf{z}_k) &= \begin{bmatrix} \mathbf{P}_k^{1/2}(\mathbf{x}_{k+1} - \mathbf{x}_k) \\ \mathbf{y}_{k+1} - \mathbf{y}_k \end{bmatrix} = \mathbf{J}_k\mathbf{G}_k^{-1}\mathbf{J}_k\mathbf{h}_k \\ &= \begin{bmatrix} \mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\mathbf{h}_{\mathbf{x},k} - \mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\mathbf{h}_{\mathbf{y},k} \\ -(\mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\mathbf{P}_k^{-1/2}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1})^\top\mathbf{h}_{\mathbf{x},k} + (\mathbf{B}_k)^{-1}\mathbf{h}_{\mathbf{y},k} \end{bmatrix}. \end{aligned}$$

Thus we have

$$\|\mathbf{P}_k^{1/2}(\mathbf{x}_{k+1} - \mathbf{x}_k)\| \overset{(25),\,(26)}{\leq} \|\mathbf{h}_{\mathbf{x},k}\| + \frac{2}{\sqrt{\mu}}\|\mathbf{h}_{\mathbf{y},k}\|$$

and

$$\begin{aligned} \|\mathbf{y}_{k+1} - \mathbf{y}_k\| &\leq \|\mathbf{P}_k^{1/2}\mathbf{C}_k^{-1}\mathbf{P}_k^{1/2}\|\|\mathbf{P}_k^{-1/2}\mathbf{H}_{\mathbf{xy},k}\mathbf{H}_{\mathbf{yy},k}^{-1}\|\|\mathbf{h}_{\mathbf{x},k}\| \\ &+ \|\mathbf{B}_k^{-1}\|\|\mathbf{h}_{\mathbf{y},k}\| \overset{(25),\,(26)}{\leq} \frac{2}{\sqrt{\mu}}\|\mathbf{h}_{\mathbf{x},k}\| + \frac{1}{\mu}\|\mathbf{h}_{\mathbf{y},k}\|. \end{aligned}$$

Thus, we can bound the term $r_k$

$$r_k \leq \|\mathbf{P}_k^{1/2}(\mathbf{x}_{k+1} - \mathbf{x}_k)\| + \|\mathbf{y}_{k+1} - \mathbf{y}_k\| \leq \frac{3}{\sqrt{\mu}}\gamma_k. \tag{30}$$

In the end, we obtain the relation between $\gamma_{k+1}$ and $\gamma_k$

$$\begin{aligned} &\gamma_{k+1} \leq \left(1 - \frac{1}{\eta_k}\right)\gamma_k + \frac{3\kappa^2\kappa_2}{\sqrt{\mu}}r_k\gamma_k + \frac{2L_2}{\mu^{3/2}}r_k^2 \\ &\overset{(30)}{\leq} \left(1 - \frac{1}{\eta_k}\right)\gamma_k + \frac{27\kappa^2\kappa_2}{\mu}\gamma_k^2 + \frac{18L_2}{\mu^{5/2}}\gamma_k^2 \leq \left(1 - \frac{1}{\eta_k}\right)\gamma_k + \frac{54\kappa^2\kappa_2}{\mu}\gamma_k^2. \end{aligned}$$

□

## B THE PROOF OF THEOREM 4.6

The proof is standard which can be modified from Liu and Luo [24]. We only provide proof sketch of this theorem.

We first define the following constants for analysis

$$\alpha_0 \overset{\text{def}}{=} \max\{2n, 18\kappa^2\} \qquad \text{and} \qquad \beta_0 \overset{\text{def}}{=} \max\left\{2, \frac{18\kappa}{n}\right\}.$$

We have $\frac{24}{\ln 2}(1+\beta_0 n\kappa) \geq 24 \times 18\kappa^2 \geq 216\kappa^2$, so the initial condition

$$\gamma_0 \leq \frac{\ln 2}{24} \cdot \frac{\mu}{\kappa_2\kappa(1+\beta_0 n\kappa)} \tag{31}$$

satisfies $\gamma_0 \leq \frac{\mu}{216\kappa^3\kappa_2}$, which is the initial condition for Theorem 4.5. Recall that we define $\rho_i = \frac{3L_2}{\mu^2}\gamma_i$ then it enjoys that

$$\rho_k \leq \left(1 - \frac{1}{4\kappa}\right)^k \rho_0. \tag{32}$$

We also have

$$\beta_0\kappa n = \max\{2n\kappa, 18\kappa^2\} \geq \max\{2n, 18\kappa^2\} = \alpha_0,$$

which implies that if $\gamma_0$ satisfies (31),then it also satisfies

$$\gamma_0 \leq \frac{\ln 2}{24} \cdot \frac{\mu}{\kappa_2\kappa(1+\alpha_0)} \tag{33}$$

Proof. We define the random sequence $\{\eta_k\}, \{\sigma_k\}$ as follows

$$\eta_k \overset{\text{def}}{=} \frac{\operatorname{tr}(\mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k})}{\operatorname{tr}(\mathbf{H}_{\mathbf{xx},k})}, \tag{34}$$

$$\sigma_k \overset{\text{def}}{=} \operatorname{tr}\left((\mathbf{G}_{\mathbf{xx},k} - \mathbf{H}_{\mathbf{xx},k})\mathbf{H}_{\mathbf{xx},k}^{-1}\right). \tag{35}$$

We have

$$\mathbf{H}_k \preceq \mathbf{G}_k \preceq (1+\sigma_k)\mathbf{H}_k. \tag{36}$$

From Lemma 4.3, we have

$$\gamma_{k+1} \leq \sigma_k\gamma_k + \alpha\gamma_k^2 \quad \text{and} \quad r_k \leq \frac{3\gamma_k}{\sqrt{\mu}}. \tag{37}$$

**Broyden Case ($0 < \tau_k \leq 1$):** One can obtain the following results by Lin et al. [22]

$$\mathbb{E}_{\mathbf{u}_k}[\sigma_{k+1}] \leq \left(1 - \frac{1}{n\kappa}\right)(1+Mr_k)^2\left(\sigma_k + \frac{2nMr_k}{1+Mr_k}\right). \tag{38}$$

We set $\theta_k \overset{\text{def}}{=} \sigma_k + \alpha_0\rho_k$ and use induction to show that

$$\mathbb{E}[\theta_k] \leq \left(1 - \frac{1}{n\kappa}\right)^k 2n\kappa. \tag{39}$$

In the case of $k = 0$, we have

$$\begin{aligned}\theta_0 &= \langle \mathbf{H}_{\mathbf{xx},0}^{-1}, \mathbf{G}_{\mathbf{xx},0}\rangle - n_{\mathbf{x}} + 2n\rho_0 \\ &\leq \langle \mathbf{H}_{\mathbf{xx},0}^{-1}, \kappa\mathbf{H}_{\mathbf{xx},0}\rangle - n_{\mathbf{x}} + \alpha_0\rho_0 = n\kappa + \alpha_0\rho_0 - n_{\mathbf{x}} \overset{(33)}{\leq} n\kappa.\end{aligned} \tag{40}$$

Thus for $k = 0$, inequality (39) is satisfied.

Suppose inequality (39) holds for $0 \leq k' \leq k$. For $k+1$, using the inequality $\mathrm{e}^x \geq 1 + x$ and recall that $\rho_k = \frac{3M}{\sqrt{\mu}}\gamma_k$, we have

$$\begin{aligned}\rho_{k+1} &\leq \sigma_k\rho_k + \frac{54\kappa^2}{3}\rho_k^2 \leq \rho_k(\sigma_k + 18\kappa^2\rho_k) \leq \rho_k\theta_k \\ &\leq \left(1 - \frac{1}{n\kappa}\right) 2\theta_k\rho_k \exp(2\rho_k)\end{aligned} \tag{41}$$

and

$$\begin{aligned}\mathbb{E}[\sigma_{k+1}] &\overset{(38)}{\leq} \left(1 - \frac{1}{n\kappa}\right)\mathbb{E}\left[(1+\rho_k)^2\left(\sigma_k + \frac{2n\rho_k}{1+\rho_k}\right)\right] \\ &\leq \left(1 - \frac{1}{n\kappa}\right)\mathbb{E}\left[(1+\rho_k)^2(\sigma_k + \alpha_0\rho_k)\right] \\ &= \left(1 - \frac{1}{n\kappa}\right)\mathbb{E}\left[\theta_k \exp(2\rho_k)\right].\end{aligned} \tag{42}$$

Thus we obtain by reduction that

$$\mathbb{E}[\theta_{k+1}] \leq \left(1 - \frac{1}{n\kappa}\right)\exp\left(2(1+\alpha_0)\left(1 - \frac{1}{4\kappa}\right)^k \rho_0\right)\mathbb{E}[\theta_k].$$

Therefore, we have

$$\mathbb{E}[\theta_{k+1}] \leq \left(1 - \frac{1}{n}\right)^{k+1} 2n\kappa, \tag{43}$$

which proves (39). Hence, for any $k \geq 0$, we have

$$\mathbb{E}[\sigma_k] \leq \mathbb{E}[\theta_k] \leq \left(1 - \frac{1}{n}\right)^k 2n\kappa,$$

which implies

$$\mathbb{E}\left[\frac{\gamma_{k+1}}{\gamma_k}\right] = \mathbb{E}\left[\frac{\rho_{k+1}}{\rho_k}\right] \overset{(41)}{\leq} \mathbb{E}[\theta_k] \leq \left(1 - \frac{1}{n\kappa}\right)^k 2n\kappa. \tag{44}$$

**SR1 Case ($\tau_k = 0$):** The proof of the SR1 case is almost the same as the Broyden case.

One can obtain the following results from Lin et al. [22]

$$\mathbb{E}_{\mathbf{u}_k}[\eta_{k+1}] \leq \left(1 - \frac{1}{n}\right)(1+Mr_k)^2(\eta_k + 2Mr_k).$$

It also holds that

$$\rho_{k+1} \leq \left(1 - \frac{1}{n}\right)(1+\rho_k)^2 2n\kappa\rho_k(\eta_k + \beta_0\rho_k). \tag{45}$$

Setting $\theta_k \overset{\text{def}}{=} \eta_k + \beta_0\rho_k$, same as Broyden case, one can use induction to show that

$$\mathbb{E}[\theta_k] \leq \left(1 - \frac{1}{n}\right)^k 2\kappa. \tag{46}$$

We can obtain that

$$\mathbb{E}\left[\frac{\gamma_{k+1}}{\gamma_k}\right] = \mathbb{E}\left[\frac{\rho_{k+1}}{\rho_k}\right] \leq \mathbb{E}[n\kappa\theta_k] \leq \left(1 - \frac{1}{n}\right)^k 2n\kappa^2, \tag{47}$$

which is equivalent to the result of (24). □