DOCTORAL THESIS

Solving convex programming with simple convex constraints

Hou, Liangshao

Date of Award:
2020

HONG KONG BAPTIST UNIVERSITY

Doctor of Philosophy

THESIS ACCEPTANCE

DATE: July 9, 2020

STUDENT'S NAME: HOU Liangshao

THESIS TITLE: Solving Convex Programming with Simple Convex Constraints

This is to certify that the above student's thesis has been examined by the following panel members and has received full approval for acceptance in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Chairman:	Prof Chu Xiaowen Professor, Department of Computer Science, HKBU (Designated by Dean of Faculty of Science)
Internal Members:	Prof Ling Leevan Professor, Department of Mathematics, HKBU (Designated by Head of Department of Mathematics)
	Dr Zhou Zirui Assistant Professor, Department of Mathematics, HKBU
External Examiners:	Prof SUN Jie Professor School of Electrical Engineering, Computing & Mathematical Sciences Curtin University
	Prof Dang Chuangyin Professor Department of Systems Engineering and Engineering Management City University of Hong Kong
Proxy:	Dr Tong Tiejun Associate Professor, Department of Mathematics, HKBU
In-attendance:	Prof Liao Lizhi Professor, Department of Mathematics, HKBU

Issued by Graduate School, HKBU

Solving Convex Programming with Simple Convex Constraints

HOU Liangshao

A thesis submitted in partial fulfilment of the requirements

for the degree of

Doctor of Philosophy

Principal Supervisor:

Prof. LIAO Li-Zhi (Hong Kong Baptist University)

July 2020

DECLARATION

I hereby declare that this thesis represents my own work which has been done after registration for the degree of PhD at Hong Kong Baptist University, and has not been previously included in a thesis or dissertation submitted to this or any other institution for a degree, diploma or other qualifications.

I have read the Universitys current research ethics guidelines, and accept responsibility for the conduct of the procedures in accordance with the Universitys Research Ethics Committee (REC). I have attempted to identify all the risks related to this research that may arise in conducting this research, obtained the relevant ethical and/or safety approval (where applicable), and acknowledged my obligations and the rights of the participants.

Signature: ____________

Date: July 2020

Abstract

The problems we studied in this thesis are linearly constrained convex programming (LCCP) and nonnegative matrix factorization (NMF). The resolutions of these two problems are all closely related to convex programming with simple convex constraints. The work can mainly be described in the following three parts.

Firstly, an interior point algorithm following a parameterized central path for linearly constrained convex programming is proposed. The convergence and polynomial-time complexity are proved under the assumption that the Hessian of the objective function is locally Lipschitz continuous. Also, an initialization strategy is proposed, and some numerical results are provided to show the efficiency of the proposed algorithm.

Secondly, the path following algorithm is promoted for general barrier functions. A class of barrier functions is proposed, and their corresponding paths are proved to be continuous and converge to optimal solutions. Applying the path following algorithm to these paths provide more flexibility to interior point methods. With some adjustments, the initialization method is equipped to validate implementation and convergence.

Thirdly, we study the convergence of hierarchical alternating least squares algorithm (HALS) and its fast form (Fast HALS) for nonnegative matrix factorization. The coordinate descend idea for these algorithms is restated. With a precise estimation of objective reduction, some limiting properties are illustrated. The accumulation points are proved to be stationary points, and some adjustments are proposed to improve the implementation and efficiency.

Keywords: Linearly constrained convex programming; Interior point method; Nonnegative matrix factorization; Hierarchical alternating least squares algorithm.

Acknowledgements

I want to express my sincere gratitude to my supervisor Prof. Liao Li-Zhi for his inspiring guidance, enthusiastic support, and valuable advice. His superb intuition, professional knowledge, and artisan spirit in academic benefit me a lot. I learn to be patient, to think profoundly, and to focus on details, all of which will make a significant difference to the rest of my life. Also, I would like to thank my co-supervisor, Prof. Michael K. Ng, for his kind help and support. I wish to show my gratitude to Prof. Sun Jie, who is my guider in academic during the oversea attachment. His passion for the unknown, high efficiency of working, and the broad knowledge impress me a lot. I will always remember his valuable advice and encouragement. Also, I would like to thank Dr. Qian Xun, Dr. Zhang Min, and Dr. Zhu Hong, for helpful discussions.

I want to show my gratitude to Hong Kong Baptist University, which provides me such a wonderful learning chance and excellent study environment. Also, thanks for the devotion of the secretaries, technicians, and all the staff of the Mathematics department.

Thanks to my parents, their love and support is my secret to overcoming difficulties.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this thesis, we focus on the studies of convex programming with simple constraints. The most commonly used simple constraints may be linear and nonnegative constraints. The problems with simple constraints are more fundamental than the ones with complicated constraints in applications. It is not only because simple constraints are easier to be solved but also because simple constraints can well reveal some inherent property. Moreover, many popular methods solve complicated problems through solving simple subproblems in every iteration, such as the alternating direction method of multipliers (ADMM), the progressive hedging algorithm (PHA), the coordinate descent method, and so on.

The first problem we considered is the linearly constrained convex programming. The interior point method, which has been famous for its excellent performance and polynomial complexity, is an essential tool for us to solve this problem. One class of interior point methods called the path following algorithm is chasing the central path to derive an optimal solution. We develop a new path following interior point method which is following a parameterized central path. The second problem we focused on is the nonnegative matrix factorization problem. Although the objective function is non-convex, it shares a bi-convex property. Many efficient methods are applied to find a local minimum by solving some convex subproblems iteratively. The subproblems are usually convex quadratic subproblems with nonnegative constraints. We give some analysis of the convergence of hierarchical alternating least squares (**HALS**) algorithm.

1.1 Interior Point Methods

1.1.1 Background of Interior Point Methods

Linear programming (LP) plays a vital role in mathematical modeling and is widely used, for examples, to optimize portfolio selection [64, 89], to schedule transportation and assignment problems [4, 81], and to maximize productivity [20, 49]. A linear objective function is optimized over the convex polyhedron, which is defined by the linear constraints. And an optimal solution of an LP always lies at a vertex of the polyhedron. However, the vertex number associated with the number of constraints and variables can be very large [86].

In the late-1940s, the simplex method was invented by George Dantzig [16] which was the only method available to solve LP for several decades. It travels along the edges of a polyhedron from one vertex to another to improve the objective value. The simplex method performs well since the number of iterations before reaching the optimal solution seems small on average. However, Klee and Minty [34] in 1972 presented an example in which the simplex method needs to travel an exponential number of vertices before the optimal solution is found. It tells that the complexity of the simplex method is not polynomial while we prefer the technique whose worst execution time is bounded by a polynomial related to the problem size [86].

The first polynomial algorithm for LP, the ellipsoid method, is proposed by Khachiyan in 1979 [31]. A series of ellipsoids is constructed, and each of them contains the optimal set with decreasing volumes. Therefore the centers of the ellipsoids lead to an optimal solution of the LP. The ellipsoid method has a significant impact on the development of the theory for LP. Although the ellipsoid method enjoys polynomial complexity, the simplex method performs with much better efficiency in practice.

In 1984, Karmarkar proposed a new polynomial interior point method for LP with high efficient performance in practice [30]. This method can be viewed as a refinement of the ellipsoid method. It inserts balls into a simplex polyhedron, while the ellipsoid method inserts ellipsoids into the feasible polyhedrons, by projective geometry [23]. A potential function is accompanied to guarantee the improvement of the current solutions. One of the most attractive features for interior point methods is that the iterations number grows much more slowly than the problem dimension grows [86]. For problems with different dimensions, the iteration numbers of interior point methods seem very stable.

Inspired by the work of Karmarkar, the optimization community shows great interests in interior point methods, and various interior point methods are developed. Gill et al., in 1986 [21], illustrates the equivalence of Karmarkas's projective method and the projected Newton barrier method, which brings the logarithmic barrier function into the sight of the community. Also, the potential function and the KKT condition become useful tools [39]. Not only the primal problem but together with the dual problem of LP are analyzed [35,36,52], called primal-dual methods. Interior point methods can be roughly categorized into the following classes: (a) projective scaling methods (e.g., see [30]); (b) affine scaling methods (e.g., see [19, 61, 76, 79]); (c) potential reduction methods (e.g., see [39, 56, 85]); (d) path following methods (e.g., see [36,52,58,59]); (d) predictor-corrector methods (e.g., see [50,55,77,87,88]).

Unlike the projective scaling methods, the affine scaling methods apply a different search direction and perform well in practice but with unknown complexity [61, 79]. The potential reduction methods decrease the potential function value in every iteration to make progress of the current solution [56, 85]. Most of the path following methods share polynomial complexity and trace the central path in every iteration since the central path will converge to an optimal solution [58, 59]. The predictor-corrector methods, in every iteration, first apply a predictor search direction towards the terminal point of the path and then use a corrector direction to adjust and keep the current solution in a neighborhood of the central path [87, 88]. These methods can also be classified as primal-only, dual-only, or primal-dual, depending on the use of the primal or dual problem. As well, the implementations of interior methods for efficient calculation and large scale problems are well studied [50, 51, 53].

As the use of the logarithmic barrier function is fully understood, applying nonlinear programming techniques to solve the nonlinear system constructed by the KKT condition is regarded as the principal reason for the success of interior point methods. [23]. The beautiful results for LP are quickly extended to more complex problems, such as the convex quadratic problem (CQP) [59], the linear complementary problem (LCP) [38], the nonlinear complementary problem (NCP) [37], and the linearly constrained convex problem (LCCP) [40, 90]. Moreover, Nesterov and Nemirovskii's [63] insightful analysis of the self-concordance property of the logarithmic barrier function extended the interior point method to conic optimization, in particular, semidefinite programming, and second-order cone programming.

1.1.2 Linearly Constrained Convex Programming

The linearly constrained convex programming (LCCP) problem is of the form

$$\begin{array}{rl} \min & f(x) \\ \text{s.t.} & Ax = b, \\ & x \geq 0, \end{array} \tag{P}$$

where $f(x)$ is a smooth convex function, A is an $m \times n$ matrix with full row rank, $b \in \mathbb{R}^m$. We assume that the optimal value for (P) is finite and attainable. The Wolfe dual [82] problem associated to (P) is

$$\begin{array}{rl} \max & f(x) - \nabla f(x)^T x + b^T y \\ \text{s.t.} & -\nabla f(x) + A^T y + s = 0, \\ & s \geq 0. \end{array} \tag{D}$$

The followings are some polynomial complexity results for path-following algorithms.

In 1989, Kojima, Mizuno, and Yoshise [36, 37] stated primal-dual path-following algorithms for LP and LCP with the worst complexity of $O(n^4L)$ and $O(n^3L)$ arithmetic operations respectively. In the same year, Monteiro and Adler [58, 59] proposed path following primal-dual algorithms for LP and CQP, which both require $O(n^3L)$ arithmetic operations. Kojima et al. and Monteiro et al. both applied the Newton direction, but differed in stepsize choosing. Todd and Ye [78], in 1990, described a path following projective algorithm for LP. In 1991, Kojima, Megiddo, and Noma [37] extended their previous work to NCP. Besides LP, CQP, and LCP, interior point methods for separable linearly constrained convex programming problem [60, 62, 68], whose objective function is of the form $\sum_{j=1}^{n} \Psi_j((x)_j)$, have been developed and achieve polynomial time complexity. $(x)_j$ denotes the j-th element of x.

The above work are mainly related to the central path. The central path for (P) which plays a vital role in path following algorithms is as follow

$$\Gamma = \{(x, y, s) \in \mathbb{R}^n_+ \times \mathbb{R}^m \times \mathbb{R}^n_+ : Ax = b, A^T y + s - \nabla f(x) = 0, XSe = \mu e, \mu > 0\}. \tag{1.1}$$

Where $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n | x > 0\}$. Γ is derived from the primal-dual complementarity relationship [52] with barrier function $-\sum_{i=1}^{n} \ln(x)_i$.

There is a lot of research related to LCCP. In 1991, Mehrotra and Sun [54] developed a polynomial-time interior point algorithm for convex constrained smooth convex programs. In the same year Kortanek, Potra and Ye [40] generalized a path following algorithm for LCCP with outer and inner iterations. However, the inner iterations to solve nonlinear equation systems are unpredictable. Then, this work was developed in 1992 by Zhu [90], who proposed the scaled Lipschitz condition and simplified the nonlinear systems in inner iterations to linear systems resulting in polynomial-time complexity. In the same year, Hertog, Roos, and Terlaky [18] stated a polynomial-time logarithmic barrier function method for smooth convex programming under the relative Lipschitz condition. In 1993, Kortanek and Zhu [41] presented a discussion of inner iteration complexity for their previous algorithm under the scaled Lipschitz condition. Also, they proposed a log-barrier based algorithm [42] with total iterations of $O(|\log(\epsilon)|\times(\text{number of inner-loop iterations}))$. In 1994, Monteiro [57] developed a primal-dual potential reduction method with global convergence for LCCP. In 1998, Andrei [1] studied a predictor-corrector method for LCCP with cubic convergence. Grossmann [26, 27], in 2000, developed a polynomial-time path-following algorithms for LCCP without nonnegative constraints, under the assumption that $f(x)$ is twice Lipschitz continuously differentiable. Sheu [71] also developed a barrier function approach with a polynomial number of iterations. Shi [73,74], in 2002, combined the potential reduction and the projected steepest descent method for LCCP.

Inspired by the research work of the path-following algorithm [58, 59] for interior method, Chen [10] proposed a non-interior continuation method for the linear complementarity problem, which newly forms LCP in a smooth equation way. With weaker assumptions, Kanzow [29] restated non-interior continuation methods for LCP. Benefit from the work of Kanzow and Chen et al., Burke and Xu [6] used a new notion of the neighborhood of the central path developing a non-interior path-following method for LCP applying smoothing method. The path it follows is

$$\overline{\Gamma} = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^n : Mx - y + q = 0, \psi_\mu((x)_i, (y)_i) = 0, \mu > 0, i = 1, 2, ..., n\}.$$

The smooth function $\psi_\mu(a, b) = a + b - \sqrt{a^2 + b^2 + 2\mu}$ used here is of the same role $Xy - \mu e = 0$ played in the central path. The limit point as μ goes to zero, is the solution of the LCP. When $\mu > 0, \psi_\mu(a, b) = 0$ if and only if $0 < a, 0 < b$, and $ab = \mu$. Therefore the nonnegativity of the limit point is assured without additional nonnegativity constraints. Denote

$$\Psi_\mu(x, y) = \begin{pmatrix} \psi_\mu((x)_1, (y)_1) \\ \vdots \\ \psi_\mu((x)_n, (y)_n) \end{pmatrix}, F_\mu = \begin{pmatrix} Mx - y + q \\ \Psi_\mu(x, y) \end{pmatrix}.$$

The solution of $F_\mu(x, y) = 0$ is a point on the path $\overline{\Gamma}$ corresponding to continuation parameter μ. For a $\beta > 0$, the following is a neighborhood of $\overline{\Gamma}$ [6]

$$\mathcal{N}_{\overline{\Gamma}}(\beta) = \{(x, y) : Mx - y + q = 0, \frac{\|\Psi_\mu(x, y)\|^2}{\mu} \leq \beta, \mu > 0\}.$$

The continuation method first reforms the LCP into a new smooth equation system $F_\mu(x, y)$ relating to $\overline{\Gamma}$. And then, the line search technique is applied to update x and y using Newton's method to better fit in $F_\mu(x, y) = 0$. After that, μ is decreased as much as possible under the premise that (x, y) still in $\mathcal{N}_\Gamma(\beta)$.

Based on the same smoothing technique, Xu and Burke [83] established an interior point algorithm for LCP following the central path. Also, Burke and Xu [7] developed the non-interior path following method [6] to the predictor-corrector framework. Burke and Xu [8] obtained complexity bounds of the non-interior path following algorithm for LCP when the underlying matrix is a P-matrix. There is a lot of work [9, 9, 11, 12, 69] relating to continuation method on complementarity problems.

1.2 Nonnegative Matrix Factorization

Nonnegative matrix factorization (NMF) has attracted much attention as a popular dimension reduction method, which can provide a low-rank approximation of matrices under nonnegative constraints. The low-rank and nonnegative constraints meet the needs of representing the data in many application areas, such as signals and images processing, and give a natural and usually sparse interpretation [43]. As an efficient technique, NMF is widely applied to data mining [67, 84], pattern recognition [5], sparse nonnegative representation [45], and so on. Also, nonnegative tensor factorization (NTF), as an extension of NMF, is emerging for high dimension models these days.

Given a matrix $Y \in \mathbb{R}^{M \times N}$, usually every element in Y is nonnegative, the goal of NMF is to find an approximation

$$Y \approx AB^T.$$

Where $A \in \mathbb{R}^{M \times J}$ and $B \in \mathbb{R}^{N \times J}$ are nonnegative. The column number of A and B is J and $J \ll \min\{M, N\}$ which reflects the dimension reduction or low-rank representation. Each column of Y is expressed by a linear combination of columns of A. Different measures can be used to value the approximation, such as Frobenius norm, Kullback-Leibler divergence [44], Bregman divergence [75]. In our work, we consider the NMF based on Frobenius norm of the following form.

$$\min_{A \geq 0, B \geq 0} \|Y - AB^T\|_F^2. \tag{1.2}$$

All the elements of A and B are restricted to be nonnegative in (1.2). Problem (1.2) is a non-convex problem for variables A and B, and finding the global minimum is NP-hard [80]. So that a local minimum becomes a common expectation. Also, this problem is a bi-convex problem, because if A or B is fixed, the objective function is convex for B or A. This property will help in constructing subproblems.

NMF is first introduced by Paatero and Tapper [66], where the Y matrix is positive, but the A and B are not required to be nonnegative. After that, NMF is popularized by Lee and Seung [43,44], who proposed the first well-known NMF algorithm, the multiplicative updating algorithm, which has been one of the most commonly used algorithms. However, later studies [13, 24, 46] show that the guarantee for convergence and the slow speed of convergence can be a problem. The multiplicative updating algorithm can also be regarded as a particular method of gradient descent algorithm [13, 47]. The gradient descent algorithm updates A and B with a specific step size towards a descent direction. And then, a nonnegative projection is applied to guarantee the nonnegativity. This kind of method is sensitive to the initial point with slow convergence speed. Since the difficulty in step size choosing and the nonnegative projection analysis, there is no convergence theory for it [2]. Another popular algorithm for NMF is alternating nonnegative least squares (ANLS) algorithm [32,65,66]. This algorithm makes the use of bi-convex property [25] of the objective function, updating A and B by solving the convex subproblems where B or A is fixed in every iteration. The theory of block coordinate descent framework guarantees ANLS's convergence to a stationary point [33]. Inspired by the framework of alternating updates, the hierarchical alternating least squares (**HALS**) algorithm [14,15] is proposed. Different from ANLS, this algorithm updates A and B column by column solving convex subproblems in every iteration. Moreover, the convex subproblems for each column of A and B are easy to be addressed with explicit solutions. The high-efficiency performance in numerical makes it one of the most popular algorithms these days.

However, the convergence analysis for **HALS** [22,33] is based on block coordinate descent framework, which does not exactly match the case of **HALS**. Moreover, a more efficient algorithm is developed, **Fast HALS** [14], based on **HALS**. Our work provides a precise exploration of the convergence for **HALS** and **Fast HALS**. Also, with full understanding of **HALS**, we give some adjustments to improve the performance.

1.3 Notation and Outline

1.3.1 Notation

The following notations are used throughout this thesis. We denote the i-th element of vector x as $(x)_i$ and vector of ones as e. $\text{diag}(x)$ represent a diagonal matrix whose diagonal elements come from x. The notation $[l, u]$ or (l, u) means the close or open interval whose lower and upper bounds are l and u, respectively. The notation $g(x) = O(\varphi(x))$ means that $|g(x)| \leq \bar{c}|\varphi(x)|$ for some positive constant $\bar{c}$ (usually for the function value goes to infinity or zero). $f(x) = \Omega(\varphi(x))$ means $\underline{c}|\varphi(x)| \leq |f(x)| \leq \bar{c}|\varphi(x)|$ for some positive constants $\underline{c}$ and $\bar{c}$. The operator $[\cdot]_+$ and $[\cdot]_-$ means $[x]_+ = \max\{0, x\}$ and $[x]_- = \max\{0, -x\}$, respectively.

1.3.2 Our Contribution and Outline

Based on the parameterized central path in [70], a novel path-following interior point algorithm is developed in this thesis for linearly constrained convex programming. Under the assumption that the Hessian function of the objective function is locally Lipschitz continuous on positive half-plane, the polynomial-time complexity is obtained for our new algorithm. Our preliminary numerical results have clearly shown the attractiveness of our new algorithm. The result achieved in this thesis will no doubt strengthen the power of interior point methods.

From the view of barrier functions, we promote the path following algorithm for general barrier function paths. We propose a class of barrier functions whose convergence and optimality is guaranteed. Apply the path following technique to these paths, give more possibility for alternative ways to approaching the optimal solution for linearly constrained convex programming.

The third part of our work is a detailed convergence analysis for hierarchical alternating least squares and its fast form. These two algorithms are famous for their high efficiency. However, there seems to be lack of precise convergence analysis. Throughout analyzing the decrease of the objective function, we know more about these algorithms, and propose some adjustments to improve the algorithms.

The rest of the thesis is organized as follows. In Chapter 2, we first generalized the central path to a parameterized central path which is converging to an optimal solution. Then the parameterized path following algorithm is developed with good implementation and polynomial convergence. After that, an initial method, together with some numerical results, are presented to show the efficiency.

In Chapter 3, we consider the path following algorithm from the view of barrier functions. A class of barrier functions is proposed, which is corresponding to a class of continuous paths converging to optimal solutions. The algorithm developed in Chapter 2 is applied to these paths, and the implementation and convergence are discussed. Also, the initialization method is attached.

In Chapter 4, we first introduce the idea of **HALS** and **Fast HALS** algorithms. Then, the convergence of **HALS** is precisely analyzed; together, some adjustments are proposed. Next, **Fast HALS** is explained similarly. Some modification is recommended to improve efficiency and a simple numerical comparison is provided.

Chapter 2

The Parameterized Path Following Algorithm

In this chapter, we give a detailed study of the parameterized path following algorithm (**PPFA**) for LCCP. The parameterized central path is a generalization of the classical central path, and it also leads to an optimal solution. Applying the path following technique to the parameterized central path derives the **PPFA**, which offers many possibilities for interior point methods. With the assumptions of the existence of an optimal solution and the locally Lipschitz continuous property of the objective function, we show that: (i) the Parameterized Path Following Algorithm is well defined; (ii) this algorithm is implementable with polynomial complexity; (iii) an initialization method for the starting point can be provided; (iv) numerical tests show that the proposed algorithm is of high-efficiency.

2.1 The Parameterized Path Following Algorithm

2.1.1 The Parameterized Central Path

First, we make some denotations as follows.

$$\mathbb{R}^n_+ = \{x \in \mathbb{R}^n | x > 0\}, \quad \mathcal{S} = \{x \in \mathbb{R}^n | Ax = b, x > 0\}, \quad \mathcal{P} = \mathbb{R}^n_+ \times \mathbb{R}^m \times \mathbb{R}^n_+,$$

$$\mathcal{T} = \{(x, y, s) \in \mathcal{S} \times \mathbb{R}^m \times \mathbb{R}^n | A^T y + s = \nabla f(x), s > 0\}.$$

The weighted primal-dual path-following ODE system proposed by Qian [70] is as the following

$$
\begin{cases}
-\nabla^2 f(x)\frac{dx}{dt} + A^T\frac{dy}{dt} + \frac{ds}{dt} = 0, \\
A\frac{dx}{dt} = 0, \\
\gamma_1 X^{\gamma_1 - 1} S^{\gamma_2}\frac{dx}{dt} + \gamma_2 X^{\gamma_1} S^{\gamma_2 - 1}\frac{ds}{dt} = (X^{\gamma_1} S^{\gamma_2} e - \sigma\mu w), \\
(x(t_0), y(t_0), s(t_0)) = (x^0, y^0, z^0) \in \mathcal{F}^0,
\end{cases}
\tag{2.1}
$$

where

$$
\begin{aligned}
&t_0 > 0, \sigma \in (0,1), \mu = \frac{e^T X^{\gamma_1} S^{\gamma_2} e}{n}, \gamma_1 > 0, \gamma_2 > 0, w \in \mathbb{R}^n_+, e^T w = n, \\
&x \in \mathbb{R}^n_+, X = \operatorname{diag}(x) \in \mathbb{R}^{n\times n}, s \in \mathbb{R}^n_+, S = \operatorname{diag}(s) \in \mathbb{R}^{n\times n}.
\end{aligned}
$$

Here w is a weight vector. The unique solution of (2.1) defines the weighted primal-dual path following continuous trajectory. The convergence of this trajectory is thoroughly analyzed by Qian [70].

Next, the path our algorithm follows is corresponding to a special solution of system (2.1). The central path mentioned in (1.1) can be expressed as follows.

$$
\Gamma = \{(x, y, s) \in \mathcal{T} \mid XSe = \mu e, \mu > 0\}.
$$

Γ is a continuous path corresponding to the barrier function $-\sum_{i=1}^{n} \ln{(x)_i}$. The path our algorithm followed has a more general form

$$
\Xi = \{(x, y, s) \in \mathcal{T} \mid X^{\gamma_1} S^{\gamma_2} e = \mu e, \mu > 0\},
$$

where $\gamma_1 > 0$ and $\gamma_2 > 0$. Compared with Γ, we call Ξ the parameterized central path. Making the substitution $\mu = e^{-(1-\sigma)t}$, where $\sigma \in (0,1)$, and regarding (x, y, s) as functions of t, it is easy to verify that Ξ is corresponding to the solution of (2.1) where $(w)_i = 1, i = 1, 2, ..., n$. One of our motivations is developing an efficient algorithm to find the limiting point of this system.

In the barrier function framework, Ξ is corresponding to the barrier term

$$\begin{cases} -\mu^{\frac{1}{\gamma_2}}(\frac{\gamma_2}{\gamma_2-\gamma_1})\sum_{i=1}^{n}(x)_i^{1-\frac{\gamma_1}{\gamma_2}}, & \text{if } \gamma_1 \neq \gamma_2; \\ -\mu^{\frac{1}{\gamma_2}}\sum_{i=1}^{n}\ln{(x)_i}, & \text{if } \gamma_1 = \gamma_2. \end{cases} \tag{2.2}$$

Here $\mu^{\frac{1}{\gamma_2}}$ can be regarded as the barrier penalty parameter, and we call μ the continuation parameter. The barrier term for central path Γ is a special case of the parameterized central path Ξ when $\gamma_1 = \gamma_2 = 1$.

We restate the limiting property of (2.1) in the following. Before illustrating the convergence result, the definition of the analytic center and an assumption is needed.

Definition 2.1. *The analytic center of a closed convex set $\Omega \subseteq \mathbb{R}^n$ corresponding to convex function $g(x)$ is the minimizer of the following problem*

$$\begin{array}{ll} \min & g(x) \\ \text{s.t.} & x \in \Omega \cap dom\ g(x), \end{array}$$

where $dom\ g(x) = \{x | g(x) < +\infty\}$.

Assumption 1. *The set $\mathcal{S}$ and the set $\mathcal{T}$ are nonempty.*

Assumption 1 is mainly for the existence of an optimal solution. We denote $\mathcal{S}_P$ and $\mathcal{S}_D$ as the optimal solution sets of problem (P) and (D) respectively.

Theorem 2.1 (Theorem 2.11 in [70])**.** *Assume $\mathcal{S}_P$ and $\mathcal{S}_D$ are nonempty, under the Assumption 1, let x^* and (y^*, s^*) be optimal solutions for problem (P) and (D), respectively, such that x^* and s^* have the maximal numbers of positive components among all optimal solutions. Let $(x(t), y(t), s(t))$ be the unique solution of the weighted primal-dual path-following system* (2.1)*. Then*

(1) for any $1 \leq i \leq n$,

$$(x(t))_i^{\gamma_1}(s(t))_i^{\gamma_2} - (w)_i\mu = \frac{(x(t_0))_i^{\gamma_1}(s(t_0))_i^{\gamma_2} - (w)_i\mu_0}{(w)_i\mu_0}e^{-\sigma(t-t_0)}(w)_i\mu.$$

(2) (a) *if* $\gamma_1 < \gamma_2$, *then* $x(t)$ *will converge to the analytic center of* $\mathcal{S}_P$ *corresponding to* $-\sum_{(x^*)_i>0}((w)_i)^{1/\gamma_2}(x)_i^{1-\frac{\gamma_1}{\gamma_2}}$;

(b) *if* $\gamma_1 = \gamma_2$, *and either there exists a pair of primal and dual optimal solution satisfying the strict complementarity or* $f(x)$ *is analytic, then* $x(t)$ *will converge to the analytic center of* $\mathcal{S}_P$ *corresponding to* $-\prod_{(x^*)_i>0}(x)_i^{(w)_i^{1/\gamma_1}}$.

(c) *if* $\gamma_1 > \gamma_2$ *and* $f(x)$ *is analytic, then* $x(t)$ *will converge to the analytic center of* $\mathcal{S}_P$ *corresponding to* $\sum_{(x^*)_i>0}(w)_i^{1/\gamma_2}(x)_i^{1-\frac{\gamma_1}{\gamma_2}}$

(3) (a) *If* $\gamma_1 < \gamma_2$, *then* $s(t)$ *will converge to the analytic center of* $\mathcal{S}_D$ *corresponding to* $\sum_{(s^*)_i>0}(w)_i^{1/\gamma_1}(s)_i^{1-\frac{\gamma_2}{\gamma_1}}$;

(b) *if* $\gamma_1 = \gamma_2$, *then* $s(t)$ *will converge to the analytic center of* $\mathcal{S}_D$ *corresponding to* $-\prod_{(s^*)_i>0}(s)_i^{(w)_i^{1/\gamma_1}}$

(c) *if* $\gamma_1 > \gamma_2$, *then* $s(t)$ *will converge to the analytic center of* $\mathcal{S}_D$ *corresponding to* $-\sum_{(s^*)_i>0}(w)_i^{1/\gamma_1}(s)_i^{1-\frac{\gamma_2}{\gamma_1}}$.

The limiting behaviors described in Theorem 2.1 strictly indicate the convergence and optimality of Ξ. Then we come to the following corollary.

Corollary 2.1. *Assume* $\mathcal{S}_P$ *and* $\mathcal{S}_D$ *are nonempty, under Assumption 1, the parameterized central path* $\omega(\mu) = (x(\mu), y(\mu), s(\mu)) \in \Xi$ *uniquely exists. The limiting behaviors when* μ *converges to* 0 *are as follows.*

(1) *If* $\gamma_1 < \gamma_2$, *then* $x(\mu)$ *will converge to an optimal solution of problem* (P);

(2) *if* $\gamma_1 = \gamma_2$, *and either there exists a pair of primal and dual optimal solutions satisfying the strict complementarity or* $f(x)$ *is analytic, then* $x(\mu)$ *will converge to an optimal solution of problem* (P);

(3) *if* $\gamma_1 > \gamma_2$ *and* $f(x)$ *is analytic, then* $x(\mu)$ *will converge to an optimal solution of problem* (P).

According to the above corollary, the case $\gamma_1 < \gamma_2$ seems more practicable with no need for analytical assumption on $f(x)$, so we present the results for this case in this chapter. The results for the other two cases are similar when $f(x)$ is analytic. The method we propose in this paper mainly enjoys the advantages of a more general path, simple, and valid assumptions, and efficient numerical performance.

2.1.2 The Path Following Algorithm

The parameterized path following algorithm (**PPFA**) for LCCP is presented as follow. Denote $u = (x^T, y^T, s^T)^T \in \mathcal{P}$. Then function $H_\mu(u)$ can be defined as

$$H_\mu(u) = \begin{pmatrix} A^T y + s - \nabla f(x) \\ Ax - b \\ X^{\gamma_1} S^{\gamma_2} e - \mu e \end{pmatrix}.$$

Therefore, the solution of $H_\mu(u) = 0$ characterizes the parameterized central path. Also, $H_0(u) = 0$ corresponds to the KKT condition of (P). The Jacobian matrix of $H_\mu(u)$ is

$$J(u) = \begin{pmatrix} -\nabla^2 f(x) & A^T & I \\ A & 0 & 0 \\ \gamma_1 X^{\gamma_1 - 1} S^{\gamma_2} & 0 & \gamma_2 X^{\gamma_1} S^{\gamma_2 - 1} \end{pmatrix}.$$

Newton's direction Δu at u with the continuation parameter μ is the solution of the following linear system

$$J(u)\Delta u = -H_\mu(u),$$

where $\Delta u = (\Delta x^T, \Delta y^T, \Delta s^T)^T \in \mathcal{R}^n \times \mathcal{R}^m \times \mathcal{R}^n$ is used in our path following algorithm. More specifically, we derive the following expressions of $\Delta x, \Delta y$, and Δs

$$\begin{cases} \Delta y &= (AD^{-1}A^T)^{-1}(AD^{-1}(\frac{1}{\gamma_2} X^{-\gamma_1} S^{1-\gamma_2} r_c - r_d) + r_p), \\ \Delta x &= D^{-1}(-A^T \Delta y + \frac{1}{\gamma_2} X^{-\gamma_1} S^{1-\gamma_2} r_c - r_d), \\ \Delta s &= -\frac{1}{\gamma_2} X^{-\gamma_1} S^{1-\gamma_2} r_c - \frac{\gamma_1}{\gamma_2} X^{-1} S \Delta x, \end{cases} \tag{2.3}$$

where $D = -\nabla^2 f(x) - \frac{\gamma_1}{\gamma_2} X^{-1} S,\ r_p = Ax - b, r_d = A^T y + s - \nabla f(x)$, and $r_c = X^{\gamma_1} S^{\gamma_2} e - \mu e$. Furthermore, when x is feasible for (P), $r_p = 0$, therefore (2.3) can be simplified and $A\Delta x = 0$. The following is our algorithm.

Parameterized Path Following Algorithm (PPFA)

Step 0 (Initialization)

Set $u_0 = (x_0^T, y_0^T, s_0^T)^T \in \mathcal{T},\ \mu_0 > 0,\ \theta \in (0, min\{1, \frac{min(s_0)_i}{2\mu_0}\}),\ p \in (0, \frac{1}{2}),\ \alpha,\ \beta \in (0, 1)$, tolerance ϵ, $k = 0$, and satisfy

$$Ax_0 = b, \quad \|H_{\mu_0}(u_0)\|_\infty \leq \theta\mu_0.$$

Step 1 (Computation of Newton's Direction)

If $\|H_{\mu_k}(u_k)\|_\infty = 0$, $u_{k+1} = u_k$, go to **Step 3**, else let Δu_k solve

$$J(u_k)\Delta u_k = -H_{\mu_k}(u_k). \tag{2.4}$$

Step 2 (Backtracking Line Search)

Calculate t_{max} in the form

$$t_{max} = \frac{1}{\max\{1, \max\limits_i -(\Delta x_k)_i/(x_k)_i, \max\limits_j -(\Delta s_k)_j/(s_k)_j\}}.$$

Let stepsize t_k be the maximum of the values $t_{max}, \alpha t_{max}, \alpha^2 t_{max}, ...$ such that

$$\|H_{\mu_k}(u_k + t_k\Delta u_k)\|_\infty \leq (1 - pt_k)\|H_{\mu_k}(u_k)\|_\infty,$$

and set $u_{k+1} = u_k + t_k\Delta u_k$.

Step 3 (Update the Continuation Parameter)

Let σ_k be the maximum of the values $1, \beta, \beta^2, ...$ such that

$$\|H_{(1-\sigma_k)\mu_k}(u_{k+1})\|_\infty \leq \theta(1 - \sigma_k)\mu_k, \tag{2.5}$$

and set $\mu_{k+1} = (1 - \sigma_k)\mu_k, k = k + 1$, and go to **Step 1** until $\mu_k \leq \epsilon$.

The following results are for the case $\gamma_1 < \gamma_2$. The same results for the case $\gamma_1 \geq \gamma_2$ can be also obtained when $f(x)$ is analytic.

2.2 Implementation and Convergence of the Path Following Algorithm

In this part, under the following assumption, we first show that every step of **PPFA** is well defined and then prove the convergence with polynomial complexity.

Assumption 2. $\nabla^2 f(x)$ *is locally Lipschitz continuous with constant* L *such that*

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_\infty \le L\|x-y\|_\infty, \forall x, y \in \mathcal{R}^n_+.$$

Assumption 2 will be used in the polynomial-time complexity analysis of the worst case. Note that we specify the norm as the infinity-norm since it will be applied directly to our algorithm.

2.2.1 Implementation

Denote the sequence generated by **PPFA** as $\{u_k\}$. This sequence will be proved to be bounded. The boundedness helps in verifying the well definedness of every step. The following theorem tells the boundedness of the sequence.

Theorem 2.2. *The iteration sequence* $\{u_k\}$ *is in a bounded closed set* U_ϵ *which depends on* $n,\ \theta,\ x_0,\ s_0,\ \mu_0,\ \gamma_1,\ \gamma_2,\ A,$ *and* f*. More specifically, there exists positive* $x_l = \Omega(\epsilon^{\frac{1}{\gamma_1}} n^{-(\frac{\gamma_2}{\gamma_1})^2})$, $x_u = \Omega(n^{\frac{\gamma_2}{\gamma_1}}), s_l = \Omega(\epsilon^{\frac{1}{\gamma_2}} n^{-1})$, *and* $s_u = \Omega(n^{\frac{\gamma_2}{\gamma_1}})$ *such that*

$$0 < x_l \le (x_k)_i \le x_u,\ 0 < s_l \le (s_k)_i \le s_u, \forall i = 1, .., n. \tag{2.6}$$

The set U_ϵ *is of the form*

$$U_\epsilon = \left\{ \begin{pmatrix} x \\ y \\ s \end{pmatrix} \in \mathcal{P} \mid \begin{cases} x_l \le (x)_i \le x_u, \quad i = 1, 2...n, \\ s_l \le (s)_i \le s_u, \quad i = 1, 2...n, \\ \|y\|_\infty \le \|(AA^T)^{-1}A\|_\infty(\|\nabla f(x) - s\|_\infty + \theta\mu_0). \end{cases} \right\}.$$

Proof. First, let us denote $r_k = A^T y_k + s_k - \nabla f(x_k)$. From the algorithm and (2.5), we know that

$$|(r_k)_i| \le \|r_k\|_\infty \le \|H_{\mu_k}(u_k)\|_\infty \le \theta\mu_k \le \theta\mu_0. \tag{2.7}$$

Then we can obtain lower and upper bounds of $(s_0 + r_k - r_0)_i$ which will be used later.

$$\begin{cases} (s_0 + r_k - r_0)_i \ge (s_0)_i - |(r_k)_i| - |(r_0)_i| \ge (s_0)_i - 2\theta\mu_0 > 0; \\ (s_0 + r_k - r_0)_i \le (s_0)_i + |(r_k)_i| + |(r_0)_i| \le (s_0)_i + 2\theta\mu_0. \end{cases} \tag{2.8}$$

The positivity of $(s_0)_i - 2\theta\mu_0$ is guaranteed by the initial setting of θ. Next, we deduce

$$\begin{aligned} & (x_k - x_0)^T (s_k - s_0) \\ =& (x_k - x_0)^T (r_k + \nabla f(x_k) - A^T y_k - r_0 - \nabla f(x_0) + A^T y_0) \\ =& -(x_k - x_0)^T A^T (y_k - y_0) + (x_k - x_0)^T (\nabla f(x_k) - \nabla f(x_0)) + (x_k - x_0)^T (r_k - r_0) \\ =& (x_k - x_0)^T (\nabla f(x_k) - \nabla f(x_0)) + (x_k - x_0)^T (r_k - r_0) \\ \ge& (x_k - x_0)^T (r_k - r_0). \end{aligned}$$

The third equality is from $Ax_k = b$ which is deduced from $Ax_0 = b$ and $A\Delta x = 0$. The last inequality is based on the convexity of $f(x)$. Then we have

$$x_0^T s_k + x_k^T (s_0 + r_k - r_0) \le x_0^T (s_0 + r_k - r_0) + x_k^T s_k. \tag{2.9}$$

Since $x_0 \in \mathcal{R}_+^n$ and the t_{max} in **PPFA** guarantees the nonnegativity of x_k and s_k, by combining (2.8) and (2.9), we can derive

$$\begin{cases} (x_k)_i & \le \frac{x_0^T (s_0 + 2\theta\mu_0 e) + x_k^T s_k}{(s_0)_i - 2\theta\mu_0}, \\ (s_k)_i & \le \frac{x_0^T (s_0 + 2\theta\mu_0 e) + x_k^T s_k}{(x_0)_i}. \end{cases} \tag{2.10}$$

If the upper bound of $x_k^T s_k$ is known, so is the upper bound of $(x_k)_i$. Now we discuss the upper bound of $x_k^T s_k$ in the following. By (2.7), we obtain

$$(x_k)_i^{\gamma_1} (s_k)_i^{\gamma_2} < (1 + \theta)\mu_k \le (1 + \theta)\mu_0. \tag{2.11}$$

Since $\gamma_1 < \gamma_2$, we derive

$$\begin{aligned}(x_k)_i(s_k)_i &= ((x_k)_i^{\gamma_1}(s_k)_i^{\gamma_2})^{\frac{1}{\gamma_2}}(x_k)_i^{1-\frac{\gamma_1}{\gamma_2}} \\ &\leq ((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{x_0^T(s_0+2\theta\mu_0 e)+x_k^T s_k}{(s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}} \\ &\leq ((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{x_0^T(s_0+2\theta\mu_0 e)+x_k^T s_k}{\min_i (s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}}.\end{aligned}$$

Then we obtain

$$\begin{aligned}x_k^T s_k &\leq n((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{x_0^T(s_0+2\theta\mu_0 e)+x_k^T s_k}{\min_i (s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}} \\ &\leq n((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{n\max_i(x_0)_i((s_0)_i+2\theta\mu_0)+x_k^T s_k}{\min_i (s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}}.\end{aligned}$$

Dividing both sides of the above inequality by $n^{\frac{\gamma_2}{\gamma_1}}$, we have

$$\begin{aligned}x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}} &\leq ((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{n^{1-\frac{\gamma_2}{\gamma_1}}\max_i(x_0)_i((s_0)_i+2\theta\mu_0)+x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}}{\min_i (s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}} \\ &\leq ((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}(\frac{\max_i(x_0)_i((s_0)_i+2\theta\mu_0)+x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}}{\min_i (s_0)_i - 2\theta\mu_0})^{1-\frac{\gamma_1}{\gamma_2}}. \qquad (2.12)\end{aligned}$$

Inequality (2.12) can be reformulated as

$$\frac{x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}}{(\max_i(x_0)_i((s_0)_i+2\theta\mu_0)+x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}})^{1-\frac{\gamma_1}{\gamma_2}}} \leq \frac{((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}}{(\min_i (s_0)_i - 2\theta\mu_0)^{1-\frac{\gamma_1}{\gamma_2}}}.$$

The left-hand side can be regarded as a function of $x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}$, say $q(x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}})$, which is monotonically increasing in $(0, +\infty)$. Moreover,

$$\lim_{x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}} \to +\infty} q(x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}) = +\infty.$$

Since $q(x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}})$ is bounded above, thus $x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}$ cannot go to infinity. Therefore $x_k^T s_k n^{-\frac{\gamma_2}{\gamma_1}}$ is bounded. Again since function $q(x)$ is monotonically increasing and unbounded above, there must exist a unique solution $M > 0$ satisfying

$$\frac{M}{(\max_i(x_0)_i((s_0)_i+2\theta\mu_0)+M)^{1-\frac{\gamma_1}{\gamma_2}}} = \frac{((1+\theta)\mu_0)^{\frac{1}{\gamma_2}}}{(\min_i (s_0)_i - 2\theta\mu_0)^{1-\frac{\gamma_1}{\gamma_2}}}.$$

This M becomes an upper bound for $x_k^T s_k n^{-\gamma_2/\gamma_1}$, thus

$$x_k^T s_k \leq M n^{\frac{\gamma_2}{\gamma_1}}. \qquad (2.13)$$

The M is dependent on $x_0,\ s_0,\ \mu_0,\ \gamma_1,\ \gamma_2$, and θ. Substituting (2.13) into (2.10), we derive

$$\begin{cases} (x_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0)+Mn^{\gamma_2/\gamma_1}}{(s_0)_i-2\theta\mu_0}, \\ (s_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0)+Mn^{\gamma_2/\gamma_1}}{(x_0)_i}. \end{cases}$$

Together with the lower bound of $(x_k)_i^{\gamma_1}(s_k)_i^{\gamma_2}$ derived from (2.7) as

$$(x_k)_i^{\gamma_1}(s_k)_i^{\gamma_2} \geq (1-\theta)\mu_k \geq (1-\theta)\epsilon, \tag{2.14}$$

we derive

$$\begin{cases} (x_k)_i & \geq (\frac{(1-\theta)\epsilon(x_0)_i^{\gamma_2}}{(x_0^T(s_0+2\theta\mu_0)+Mn^{\gamma_2/\gamma_1})^{\gamma_2}})^{1/\gamma_1}, \\ (s_k)_i & \geq (\frac{(1-\theta)\epsilon((s_0)_i-2\theta\mu_0)^{\gamma_1}}{(x_0^T(s_0+2\theta\mu_0)+Mn^{\gamma_2/\gamma_1})^{\gamma_1}})^{1/\gamma_2}. \end{cases}$$

These imply $(x_k)_i > 0, (s_k)_i > 0$. From the above analysis, we know that $x_l,\ x_u,\ s_l$, and s_u exist, and (2.6) holds. From the definition of r_k at the beginning of the proof, we have

$$\begin{aligned} \|y_k\|_\infty &= \|(AA^T)^{-1}A(\nabla f(x_k) - s_k + r_k)\|_\infty \\ &\leq \|(AA^T)^{-1}A\|_\infty(\|\nabla f(x_k) - s_k\|_\infty + \|r_k\|_\infty) \\ &\leq \|(AA^T)^{-1}A\|_\infty(\|\nabla f(x_k) - s_k\|_\infty + \theta\mu_0), \end{aligned}$$

which is bounded. So that we have $u_k \in U_\epsilon$. □

With the help of the boundedness, under Assumption 2, we can prove the locally Lipschitz continuity of the $J(u)$ in U_ϵ.

Theorem 2.3. *Under Assumption 2 that $\nabla^2 f(x)$ is locally Lipschitz continuous for $x \in \mathbb{R}_+^n$ with Lipschitz constant L, $J(u)$ is locally Lipschitz continuous on U_ϵ with an $L_\epsilon > 0$ such that*

$$\|J(v) - J(w)\|_\infty \leq L_\epsilon\|v - w\|_\infty \quad \forall v, w \in U_\epsilon,$$

where L_ϵ is of the form

$$L_\epsilon = O(\epsilon^{-\frac{2}{\gamma_1}} n^{(\gamma_1+\gamma_2+\frac{2\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}). \tag{2.15}$$

Proof. In order to simplify the expressions, we denote the vectors $v, w \in U_\epsilon$ as $v = (x_v^T, y_v^T, s_v^T)^T, w = (x_w^T, y_w^T, s_w^T)^T$ and

$$\begin{aligned}
&X_v = \text{diag}(x_v),\ S_v = \text{diag}(s_v),\ X_w = \text{diag}(x_w),\ S_w = \text{diag}(s_w),\\
&G = -\nabla^2 f(x_v) + \nabla^2 f(x_w),\ E = \gamma_1 X_v^{\gamma_1 - 1} S_v^{\gamma_2} - \gamma_1 X_w^{\gamma_1 - 1} S_w^{\gamma_2},\\
&F = \gamma_2 X_v^{\gamma_1} S_v^{\gamma_2 - 1} - \gamma_2 X_w^{\gamma_1} S_w^{\gamma_2 - 1}.
\end{aligned}$$

Here E and F are diagonal matrices with positive diagonal elements and G is a symmetric matrix. Then we derive

$$\|J(v) - J(w)\|_\infty = \| \begin{pmatrix} G & 0 & 0 \\ 0 & 0 & 0 \\ E & 0 & F \end{pmatrix} \|_\infty \le \max\{\|G\|_\infty, \|E\|_\infty + \|F\|_\infty\}. \tag{2.16}$$

Under Assumption 2, the following holds

$$\|G\|_\infty \le L\|v - w\|_\infty. \tag{2.17}$$

Next, we consider the gradients of functions $h_1,\ h_2 : U_x \times U_s \to R$ defined by $h_1(x, s) = \gamma_1 x^{\gamma_1 - 1} s^{\gamma_2}$ and $h_2(x, s) = \gamma_2 x^{\gamma_1} s^{\gamma_2 - 1}$, where U_x and U_s denote the intervals $[x_l, x_u]$ and $[s_l, s_u]$, respectively, where x_l, x_u, s_l, and s_u are defined in Theorem 2.2. Here we consider the condition $\gamma_1 \neq 1$ and $\gamma_2 \neq 1$, and it is easy to verify that the results still hold when the equality holds.

$$\begin{aligned}
\|\nabla h_1(x, s)\|_\infty &= x^{\gamma_1} s^{\gamma_2} \| \begin{pmatrix} \gamma_1(\gamma_1 - 1)x^{-2} \\ \gamma_1\gamma_2 x^{-1} s^{-1} \end{pmatrix} \|_\infty \\
&\le x_u^{\gamma_1} s_u^{\gamma_2} \max\{\gamma_1|\gamma_1 - 1| x_l^{-2}, \gamma_1\gamma_2 x_l^{-1} s_l^{-1}\} \\
&= O(\epsilon^{-\frac{2}{\gamma_1}} n^{(\gamma_1 + \gamma_2 + 2\frac{\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}).
\end{aligned} \tag{2.18}$$

In the same way,

$$\begin{aligned}
\|\nabla h_2(x, s)\|_\infty &= x^{\gamma_1} s^{\gamma_2} \| \begin{pmatrix} \gamma_2\gamma_1 x^{-1} s^{-1} \\ \gamma_2(\gamma_2 - 1) s^{-2} \end{pmatrix} \|_\infty \\
&\le x_u^{\gamma_1} s_u^{\gamma_2} \max\{\gamma_1\gamma_2 x_l^{-1} s_l^{-1}, \gamma_2|\gamma_2 - 1| s_l^{-2}\} \\
&= O(\epsilon^{-\frac{2}{\gamma_1}} n^{(\gamma_1 + \gamma_2 + 2\frac{\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}).
\end{aligned} \tag{2.19}$$

Considering the power over ϵ and n in (2.18) and (2.19), we derive an upper bound $\widetilde{L}$ for $\|\nabla h_1(x,s)\|_\infty$ and $\|\nabla h_2(x,s)\|_\infty$ that

$$\|\nabla h_1(x,s)\|_\infty + \|\nabla h_2(x,s)\|_\infty \le \widetilde{L}, \quad \forall x \in U_x, \forall s \in U_s.$$

where

$$\widetilde{L} = O(\epsilon^{-\frac{2}{\gamma_1}} n^{(\gamma_1+\gamma_2+\frac{2\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}).$$

Considering the expressions of E and F, by the mean value theorem,

$$\|E\|_\infty + \|F\|_\infty = \sum_{j=1}^{2} \max_i |h_j((x_v)_i,(s_v)_i) - h_j((x_w)_i,(s_w)_i)| \le \widetilde{L}\|v-w\|_\infty. \quad (2.20)$$

Together with (2.16), (2.17) and (2.20), we have

$$\|J(v) - J(w)\|_\infty \le \|G\|_\infty + \|E\|_\infty + \|F\|_\infty \le L_\epsilon\|v-w\|, \quad \forall v,w \in U_\epsilon,$$

where $L_\epsilon = L + \widetilde{L}$. This proves that $J(v)$ is local Lipschitz continuous on U_ϵ and that L_ϵ exists in the expression of (2.15). □

The following theorem shows that **PPFA** is well defined and implementable.

Theorem 2.4. *Suppose at the k-th iteration, $u_k = (x_k^T, y_k^T, s_k^T)^T \in U_\epsilon$, satisfying*

$$Ax_k - b = 0, \|H_{\mu_k}(u_k)\|_\infty \le \theta\mu_k, \|H_{\mu_k}(u_k)\|_\infty \ne 0.$$

Then, the followings hold:

(a) The Jacobian matrix $J(u_k)$ is nonsingular. Hence, the Newton direction in Step 1 of **PPFA** *exists and is unique.*

(b) There exists a

$$\bar{t} = \frac{\|H_{\mu_k}(u_k)\|_\infty}{L_\epsilon\|\Delta u_k\|_\infty^2}$$

such that for every $t \in (0, \min\{1, \bar{t}\}]$,

$$\|H_{\mu_k}(u_k + t\Delta u_k)\|_\infty \le (1-pt)\|H_{\mu_k}(u_k)\|_\infty$$

and

$$(x_k)_i + t(\Delta x_k)_i > 0, (s_k)_i + t(\Delta s_k)_i > 0, \quad i = 1, 2, \ldots, n.$$

Furthermore, the backtracking process will generate a step size

$$t_k \begin{cases} = 1 & \text{if } \bar{t} \geq 1, \\ \geq \alpha \bar{t} & \text{if } \bar{t} < 1. \end{cases} \tag{2.21}$$

(c) There exists a $\sigma_k \in (0, 1)$ such that

$$\|H_{(1-\sigma_k)\mu_k}(u_{k+1})\|_\infty \leq \theta(1 - \sigma_k)\mu_k.$$

Proof. (a) Consider the determinant of $J(u), u \in U_\epsilon$. We make the following denotations

$$\widetilde{A} = \begin{pmatrix} -\nabla^2 f(x) & A^T \\ A & 0 \end{pmatrix}, \ \widetilde{B} = \begin{pmatrix} I \\ 0 \end{pmatrix}, \ \widetilde{C} = \begin{pmatrix} \gamma_1 X^{\gamma_1 - 1} S^{\gamma_2} & 0 \end{pmatrix}, \ \widetilde{D} = \gamma_2 X^{\gamma_1} S^{\gamma_2 - 1}.$$

Thus

$$J(u) = \begin{pmatrix} -\nabla^2 f(x) & A^T & I \\ A & 0 & 0 \\ \gamma_1 X^{\gamma_1 - 1} S^{\gamma_2} & 0 & \gamma_2 X^{\gamma_1} S^{\gamma_2 - 1} \end{pmatrix} = \begin{pmatrix} \widetilde{A} & \widetilde{B} \\ \widetilde{C} & \widetilde{D} \end{pmatrix}. \tag{2.22}$$

$D = -\nabla^2 f(x) - \frac{\gamma_1}{\gamma_2} X^{-1} S$ is defined in (2.3). Since $\widetilde{D}$ and D are nonsingular, we derive

$$\begin{aligned} |\det(J(u))| &= |\det(\begin{pmatrix} I & -\widetilde{B}\widetilde{D}^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} \widetilde{A} & \widetilde{B} \\ \widetilde{C} & \widetilde{D} \end{pmatrix})| \\ &= |\det(\widetilde{A} - \widetilde{B}\widetilde{D}^{-1}\widetilde{C})\det(\widetilde{D})| \\ &= |\det(\begin{pmatrix} D & A^T \\ A & 0 \end{pmatrix})\det(\widetilde{D})| \\ &= |\det(\begin{pmatrix} D & A^T \\ 0 & -AD^{-1}A^T \end{pmatrix})\det(\widetilde{D})| \\ &= |\det(D)\det(AD^{-1}A^T)\det(\widetilde{D})|. \end{aligned}$$

Since D is symmetric negative definite and A has full row rank, $|\det(J(u))| \neq 0$, which means $J(u)$ is nonsingular. Therefore $J(u_k)$ is nonsingular.

(b) $\|H_{\mu_k}(u_k + t\Delta u_k)\|$ can be expressed as follows

$$\begin{aligned}
&\|H_{\mu_k}(u_k + t\Delta u_k)\|_\infty \\
=&\|H_{\mu_k}(u_k) + \int_0^1 J(u_k + \xi t\Delta u_k)t\Delta u_k d\xi\|_\infty \\
=&\|H_{\mu_k}(u_k) + J(u_k)t\Delta u_k + \int_0^1 (J(u_k + \xi t\Delta u_k) - J(u_k))t\Delta u_k d\xi\|_\infty \\
\leq&|1-t|\|H_{\mu_k}(u_k)\|_\infty + \|\int_0^1 (J(u_k + \xi t\Delta u_k) - J(u_k))t\Delta u_k d\xi\|_\infty \\
\leq&|1-t|\|H_{\mu_k}(u_k)\|_\infty + t\int_0^1 L_\epsilon \|\xi t\Delta u_k\|_\infty \|\Delta u_k\|_\infty d\xi \\
=&|1-t|\|H_{\mu_k}(u_k)\|_\infty + \frac{1}{2}L_\epsilon t^2\|\Delta u_k\|_\infty^2,
\end{aligned} \tag{2.23}$$

where L_ϵ is defined in Theorem 2.3. (2.23) is a convex piecewise quadratic function and attains its minimal value at $\min\{1, \bar{t}\}$, where

$$\bar{t} = \frac{\|H_{\mu_k}(u_k)\|_\infty}{L_\epsilon\|\Delta u_k\|_\infty^2}. \tag{2.24}$$

For the case $\bar{t} < 1$, (2.23) attains its minimal value $(1 - \frac{1}{2}\bar{t})\|H_{\mu_k}(u_k)\|_\infty$ at $t = \bar{t}$. The minimal value is less equal than $(1 - p\bar{t})\|H_{\mu_k}(u_k)\|_\infty$. For the case $\bar{t} \geq 1$, (2.23) attains its minimal value $\frac{1}{2}L_\epsilon\|\Delta u_k\|_\infty^2$ at $t = 1$. $\frac{1}{2}L_\epsilon\|\Delta u_k\|_\infty^2 \leq (1-p)\|H_{\mu_k}(u_k)\|_\infty$ is directly derived from $\bar{t} \geq 1$. By the convexity of parabola, in respect to t, the following property holds

$$(1-t)\|H_{\mu_k}(u_k)\|_\infty + \frac{1}{2}L_\epsilon t^2\|\Delta u_k\|_\infty^2 \leq (1-pt)\|H_{\mu_k}(u_k)\|_\infty, \quad t \in (0, \min\{1, \bar{t}\}]. \tag{2.25}$$

Combining (2.23) and (2.25), we obtain

$$\|H_{\mu_k}(u_k + t\Delta u_k)\|_\infty \leq (1-pt)\|H_{\mu_k}(u_k)\|_\infty, \quad t \in (0, \min\{1, \bar{t}\}]. \tag{2.26}$$

If there exists a $\hat{t} \in (0, \min\{1, \bar{t}\}]$ makes $(x_k)_i + \hat{t}(\Delta x_k)_i \leq 0$, then there exists a $\widetilde{t} \in (0, \min\{1, \bar{t}\}]$ that makes $(x_k)_i + \widetilde{t}(\Delta x_k)_i = 0$, which results in $\|H_{\mu_k}(u_k +$

$\widetilde{t}\Delta u)\|_\infty \geq \mu_k$. This contradicts with (2.26). The analysis for s_k is the same, therefore

$$x_k + t\Delta x_k > 0, s_k + t\Delta s_k > 0, \quad \forall t \in (0, \min\{1, \bar{t}\}].$$

From the definition of t_{max} and the above analysis, we have $t_{max} \geq \min\{1, \bar{t}\}$. If $\bar{t} \geq 1$, $t_k = 1$ will be chosen in the backtracking step. If $\bar{t} < 1$, after finite number of backtracking steps, $t_k \geq \alpha\bar{t}$ will be chosen.

(c) From (2.26), we know the updated u_{k+1} satisfies

$$\frac{\|H_{\mu_k}(u_{k+1})\|_\infty}{\mu_k} < \frac{\|H_{\mu_k}(u_k)\|_\infty}{\mu_k} \leq \theta.$$

Viewing $\frac{\|H_{\mu_k}(u_{k+1})\|_\infty}{\mu_k}$ as a continuous function of μ_k, there must exist an interval $(0, z_k)$ such that for any $\sigma_k \in (0, z_k)$

$$\frac{\|H_{(1-\sigma_k)\mu_k}(u_{k+1})\|_\infty}{(1-\sigma_k)\mu_k} \leq \theta.$$

The value of σ_k will be generated by the finite backtracking steps in **Step 3**.

□

Theorem 2.4 ensures the implementation of **PPFA** when $\|H_{\mu_k}(u_k)\|_\infty \neq 0$. For the case of $\|H_{\mu_k}(u_k)\|_\infty = 0$, the **Step 2** will be skipped and the **Step 3** is obviously implementable based on the analysis in Theorem 2.4(c).

2.2.2 Convergence

In this part, we discuss the convergence property of **PPFA**. First, we derive an upper bound for $\|J(u)^{-1}\|_\infty$.

Theorem 2.5. *Let* $\{u_k\}$, $k = 0, 1, \ldots$ *be generated by* **PPFA**. *Then there exists a* $K_\epsilon > 0$ *such that*

$$\|J(u_k)^{-1}\|_\infty \leq K_\epsilon, \quad k = 0, 1, \ldots.$$

The K_ϵ is of the form

$$K_\epsilon = O(\epsilon^{-w_1} n^{w_2}), \tag{2.27}$$

where

$$\begin{cases} w_1 = \max\{1 + \dfrac{3}{\gamma_1} + \dfrac{4}{\gamma_2}, 2 + \dfrac{2}{\gamma_1} + \dfrac{2}{\gamma_2}\}, \\ w_2 = (7 + \dfrac{7\gamma_1}{\gamma_2} + \dfrac{3\gamma_2}{\gamma_1})\dfrac{\gamma_2}{\gamma_1}. \end{cases}$$

Proof. From Theorem 2.2, we have $u_k \in U_\epsilon,\ k = 0, 1, \ldots$. Following the denotations in (2.22) and 2.3, we have

$$\widetilde{A} = \begin{pmatrix} -\nabla^2 f(x) & A^T \\ A & 0 \end{pmatrix},\ \widetilde{B} = \begin{pmatrix} I \\ 0 \end{pmatrix},\ \widetilde{C} = \begin{pmatrix} \gamma_1 X^{\gamma_1 - 1} S^{\gamma_2} & 0 \end{pmatrix},\ \widetilde{D} = \gamma_2 X^{\gamma_1} S^{\gamma_2 - 1},$$

$D = -\nabla^2 f(x) - \dfrac{\gamma_1}{\gamma_2} X^{-1} S$. Let $\widetilde{E} = \begin{pmatrix} D & A^T \\ A & 0 \end{pmatrix} = \widetilde{A} - \widetilde{B}\widetilde{D}^{-1}\widetilde{C}$. Then

$$\begin{aligned} J(u)^{-1} &= \begin{pmatrix} \widetilde{E}^{-1} & 0 \\ 0 & \widetilde{D}^{-1} \end{pmatrix} \begin{pmatrix} I & 0 \\ -\widetilde{C}\widetilde{E}^{-1} & I \end{pmatrix} \begin{pmatrix} I & -\widetilde{B}\widetilde{D}^{-1} \\ 0 & I \end{pmatrix} \\ &= \begin{pmatrix} \widetilde{E}^{-1} & 0 \\ 0 & \widetilde{D}^{-1} \end{pmatrix} \begin{pmatrix} I & -\widetilde{B}\widetilde{D}^{-1} \\ -\widetilde{C}\widetilde{E}^{-1} & \widetilde{C}\widetilde{E}^{-1}\widetilde{B}\widetilde{D}^{-1} + I \end{pmatrix} \\ &= \begin{pmatrix} \widetilde{E}^{-1} & 0 \\ 0 & \widetilde{D}^{-1} \end{pmatrix} (\begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} + \begin{pmatrix} I \\ -\widetilde{C}\widetilde{E}^{-1} \end{pmatrix} \begin{pmatrix} I & -\widetilde{B}\widetilde{D}^{-1} \end{pmatrix}). \end{aligned}$$

Therefore

$$\begin{aligned} &\|J(u)^{-1}\|_\infty \\ \le &\| \begin{pmatrix} \widetilde{E}^{-1} & 0 \\ 0 & \widetilde{D}^{-1} \end{pmatrix} \|_\infty (\| \begin{pmatrix} 0 & 0 \\ 0 & I \end{pmatrix} \|_\infty + \| \begin{pmatrix} I \\ -\widetilde{C}\widetilde{E}^{-1} \end{pmatrix} \|_\infty \| \begin{pmatrix} I & -\widetilde{B}\widetilde{D}^{-1} \end{pmatrix} \|_\infty) \\ \le &(\|\widetilde{E}^{-1}\|_\infty + \|\widetilde{D}^{-1}\|_\infty)(1 + (1 + \| - \widetilde{C}\widetilde{E}^{-1}\|_\infty)(1 + \| - \widetilde{B}\widetilde{D}^{-1}\|_\infty)) \\ \le &(\|\widetilde{E}^{-1}\|_\infty + \|\widetilde{D}^{-1}\|_\infty)(1 + (1 + \|\widetilde{C}\|_\infty \|\widetilde{E}^{-1}\|_\infty)(1 + \|\widetilde{B}\|_\infty \|\widetilde{D}^{-1}\|_\infty)). \end{aligned} \tag{2.28}$$

In order to estimate an upper bound for $\|J(u)^{-1}\|_\infty$, we only need to approximate the upper bounds of $\|\widetilde{B}\|_\infty, \|\widetilde{C}\|_\infty, \|\widetilde{D}^{-1}\|_\infty$, and $\|\widetilde{E}^{-1}\|_\infty$. For $\|\widetilde{B}\|_\infty, \|\widetilde{C}\|_\infty$, and $\|\widetilde{D}^{-1}\|_\infty$, we have

$$\|\widetilde{B}\|_\infty = \| \begin{pmatrix} I \\ 0 \end{pmatrix} \|_\infty = 1, \tag{2.29}$$

$$\|\widetilde{C}\|_\infty = \max_i \gamma_1 \frac{(x)_i^{\gamma_1}(s)_i^{\gamma_2}}{(x)_i} \leq \gamma_1 \frac{(1+\theta)\mu_0}{x_l} = O(\epsilon^{-\frac{1}{\gamma_1}} n^{(\frac{\gamma_2}{\gamma_1})^2}), \tag{2.30}$$

$$\|\widetilde{D}^{-1}\|_\infty = \max_i \frac{(s)_i}{\gamma_2 (x)_i^{\gamma_1}(s)_i^{\gamma_2}} \leq \frac{1}{\gamma_2} \frac{s_u}{(1-\theta)\epsilon} = O(\epsilon^{-1} n^{\frac{\gamma_2}{\gamma_1}}), \tag{2.31}$$

where the two inequalities comes from (2.11) and (2.14). Next, we estimate an upper bound for $\|\widetilde{E}^{-1}\|_\infty$. $\widetilde{E}$ is a saddle system and has the following decomposition

$$\widetilde{E} = \begin{pmatrix} D & A^T \\ A & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ AD^{-1} & I \end{pmatrix} \begin{pmatrix} D & 0 \\ 0 & -AD^{-1}A^T \end{pmatrix} \begin{pmatrix} I & D^{-1}A^T \\ 0 & I \end{pmatrix}.$$

Then

$$\begin{aligned} &\|\widetilde{E}^{-1}\|_\infty \\ \leq &\| \begin{pmatrix} I & -D^{-1}A^T \\ 0 & I \end{pmatrix} \|_\infty \| \begin{pmatrix} D^{-1} & 0 \\ 0 & (-AD^{-1}A^T)^{-1} \end{pmatrix} \|_\infty \| \begin{pmatrix} I & 0 \\ -AD^{-1} & I \end{pmatrix} \|_\infty \\ \leq &(1 + \|D^{-1}\|_\infty \|A^T\|_\infty) \max\{\|D^{-1}\|_\infty, \|(-AD^{-1}A^T)^{-1}\|_\infty\} (1 + \|A\|_\infty \|D^{-1}\|_\infty). \end{aligned} \tag{2.32}$$

Then, the rest is to approximate $\|D^{-1}\|_\infty$ and $\|(-AD^{-1}A^T)^{-1}\|_\infty$. By the expression of D we obatin

$$\|D^{-1}\|_2 = \frac{1}{\lambda_{min}(-D)} \leq \frac{1}{\lambda_{min}(\frac{\gamma_1}{\gamma_2} X^{-1} S)}.$$

Note that $\frac{1}{\sqrt{n}} \|W\|_\infty \leq \|W\|_2 \leq \|W\|_\infty$ for any symmetric matrix W. So we obtain

$$\|D^{-1}\|_\infty \leq \sqrt{n} \|D^{-1}\|_2 \leq \frac{\sqrt{n}}{\lambda_{min}(\frac{\gamma_1}{\gamma_2} X^{-1} S)} \leq \frac{\sqrt{n}\gamma_2 x_u}{\gamma_1 s_l} = O(\epsilon^{-\frac{1}{\gamma_2}} n^{\frac{3}{2} + \frac{\gamma_2}{\gamma_1}}). \tag{2.33}$$

Before obtaining an upper bound of $\|(-AD^{-1}A^T)^{-1}\|$, we first estimate

$$\begin{aligned}
\lambda_{min}(A(-D)^{-1}A^T) &= \min_{\|x\|=1} x^T A(-D)^{-1}A^T x \\
&= \min_{\|x\|=1} \frac{x^T A(-D)^{-1}A^T x}{x^T AA^T x} x^T AA^T x \\
&\geq (\min_{\|x\|=1} \frac{x^T A(-D)^{-1}A^T x}{x^T AA^T x})(\min_{\|x\|=1} x^T AA^T x) \\
&= (\min_{\|x\|=1} \frac{x^T A(-D)^{-1}A^T x}{x^T AA^T x})\lambda_{min}(AA^T) \\
&\geq (\min_{y=A^T x} \frac{y^T(-D)^{-1}y}{y^T y})\lambda_{min}(AA^T) \\
&= \lambda_{min}((-D)^{-1})\lambda_{min}(AA^T) \\
&= \frac{\lambda_{min}(AA^T)}{\lambda_{max}(-D)}.
\end{aligned}$$

Therefore we obtain

$$\begin{aligned}
\|(-AD^{-1}A^T)^{-1}\|_2 &= \frac{1}{\lambda_{min}(A(-D)^{-1}A^T)} \\
&\leq \frac{\lambda_{max}(-D)}{\lambda_{min}(AA^T)} \\
&\leq \lambda_{min}^{-1}(AA^T)(\lambda_{max}(\nabla^2 f(x)) + \lambda_{max}(\frac{\gamma_1}{\gamma_2}X^{-1}S)) \\
&\leq \lambda_{min}^{-1}(AA^T)(\|\nabla^2 f(x)\|_2 + \frac{\gamma_1}{\gamma_2}\frac{s_u}{x_l}). \qquad (2.34)
\end{aligned}$$

By the assumption that $\nabla^2 f(x)$ is Lipschitz continuous, we know for all $x \in U_x$,

$$\|\nabla^2 f(x)\|_2 \leq \|\nabla^2 f(x)\|_\infty \leq \|\nabla^2 f(x_0)\|_\infty + L\|x - x_0\|_\infty \leq \|\nabla^2 f(x_0)\|_\infty + 2Lx_u, \qquad (2.35)$$

where x_0 is the initial value of **PPFA**. Combining (2.34) and (2.35), we have

$$\begin{aligned}
\|(-AD^{-1}A^T)^{-1}\|_\infty &\leq \sqrt{n}\|(-AD^{-1}A^T)^{-1}\|_2 \\
&\leq \sqrt{n}\lambda_{min}^{-1}(AA^T)(\|\nabla^2 f(x_0)\|_\infty + 2Lx_u + \frac{\gamma_1}{\gamma_2}\frac{s_u}{x_l}) \\
&= O(\epsilon^{-\frac{1}{\gamma_1}} n^{\frac{1}{2}+\frac{\gamma_2}{\gamma_1}+(\frac{\gamma_2}{\gamma_1})^2}). \qquad (2.36)
\end{aligned}$$

Comparing (2.33) and (2.36), we obtain

$$\max\{\|D^{-1}\|_\infty, \|(-AD^{-1}A^T)^{-1}\|_\infty\} = O(\epsilon^{-\frac{1}{\gamma_1}} n^{\frac{1}{2}+\frac{\gamma_2}{\gamma_1}+(\frac{\gamma_2}{\gamma_1})^2}). \qquad (2.37)$$

Substituting (2.33) and (2.37) into (2.32) and considering the powers of ϵ and n, we know there exists an $E_u > 0$ such that

$$\|\widetilde{E}^{-1}\|_\infty \leq E_u, \tag{2.38}$$

where

$$E_u = O(\epsilon^{-\frac{1}{\gamma_1}-\frac{2}{\gamma_2}} n^{(3+\frac{7\gamma_1}{2\gamma_2}+\frac{\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}).$$

Substituting (2.29), (2.30), (2.31), and (2.38) into (2.28), we have

$$\begin{aligned}\|J(u)^{-1}\|_\infty &\leq (E_u + \|\widetilde{D}^{-1}\|_\infty)(1 + (1 + \|\widetilde{C}\|_\infty E_u)(1 + \|\widetilde{D}^{-1}\|_\infty)) \\ &= O(\|\widetilde{C}\|_\infty \|\widetilde{D}^{-1}\|_\infty E_u^2) + O(\|\widetilde{C}\|_\infty \|\widetilde{D}^{-1}\|_\infty^2 E_u).\end{aligned}$$

From the upper bounds of E_u, $\|\widetilde{C}\|_\infty$ and $\|\widetilde{D}^{-1}\|_\infty$, we obtain

$$\begin{aligned}O(\|\widetilde{C}\|_\infty \|\widetilde{D}^{-1}\|_\infty E_u^2) &= O(\epsilon^{-1-\frac{3}{\gamma_1}-\frac{4}{\gamma_2}} n^{(7+\frac{7\gamma_1}{\gamma_2}+\frac{3\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}), \\ O(\|\widetilde{C}\|_\infty \|\widetilde{D}^{-1}\|_\infty^2 E_u) &= O(\epsilon^{-2-\frac{2}{\gamma_1}-\frac{2}{\gamma_2}} n^{(5+\frac{7\gamma_1}{2\gamma_2}+\frac{2\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}}).\end{aligned}$$

So that there exists an upper bound K_ϵ for $\|J(u)^{-1}\|_\infty$, expressed by (2.27). □

Next, we address the convergence of **PPFA**.

Theorem 2.6. *Let (u_k, μ_k) be the sequence generated by* **PPFA**. *Then we have the following results.*

(a) For all $k \geq 1$

$$\begin{cases} Ax_k - b = 0, \\ \|H_{\mu_k}(u_k)\|_\infty \leq \theta\mu_k, \\ (1-\sigma_{k-1})\cdots(1-\sigma_0)\mu_0 = \mu_k. \end{cases}$$

(b) For all $k \geq 1$

$$\sigma_k \geq \frac{\beta\theta p\eta}{\theta+1},$$

where

$$\eta = min\{1, \frac{\alpha}{L_\epsilon K_\epsilon^2 \theta\mu_0}\}.$$

(c) *After at most* $\lceil \frac{ln\epsilon - ln\mu_0}{ln(1-\frac{\beta\theta p\eta}{\theta+1})} \rceil$ *iterations, the tolerance* ϵ *is satisfied. To be more specific, the iteration complexity is as follows*

$$O(ln(\frac{\mu_0}{\epsilon})\epsilon^{-\overline{w}_1} n^{\overline{w}_2}), \tag{2.39}$$

where

$$\begin{cases} \overline{w}_1 = \max\{2 + \frac{8}{\gamma_1} + \frac{8}{\gamma_2}, 4 + \frac{6}{\gamma_1} + \frac{4}{\gamma_2}\}, \\ \overline{w}_2 = (14 + \gamma_1 + \gamma_2 + \frac{14\gamma_1}{\gamma_2} + \frac{8\gamma_2}{\gamma_1})\frac{\gamma_2}{\gamma_1}. \end{cases} \tag{2.40}$$

Proof. (a) Following the steps of **PPFA**, these results can be easily checked.

(b) When $\|H_{\mu_k}(u_k)\|_\infty \neq 0$. We first consider the value of t_k. From (2.4) and Theorem 2.5, we have

$$\|\Delta u_k\|_\infty \leq \|J(u_k)^{-1}\|_\infty \|H_{\mu_k}(u_k)\|_\infty \leq K_\epsilon \|H_{\mu_k}(u_k)\|_\infty.$$

Substituting the above result into equation (2.24), from (a) we can derive

$$\bar{t} = \frac{\|H_{\mu_k}(u_k)\|_\infty}{L_\epsilon \|\Delta u_k\|_\infty^2} \geq \frac{1}{L_\epsilon K_\epsilon^2 \|H_{\mu_k}(u_k)\|_\infty} \geq \frac{1}{L_\epsilon K_\epsilon^2 \theta \mu_k} \geq \frac{1}{L_\epsilon K_\epsilon^2 \theta \mu_0}.$$

Based on (2.21), we obtain

$$t_k \geq min\{1, \alpha\bar{t}\} \geq min\{1, \frac{\alpha}{L_\epsilon K_\epsilon^2 \theta \mu_0}\} = \eta.$$

Next we show that the inequality (2.4) holds for every $\sigma_k \in (0, \frac{\theta p\eta}{\theta+1})$, which means $\sigma_k \geq \frac{\beta\theta p\eta}{\theta+1}$ can be obtained after applying the backtracking.

For $\sigma_k \in (0, \frac{\theta p\eta}{\theta+1})$, we have

$$\begin{aligned} \|H_{(1-\sigma_k)\mu_k}(u_{k+1})\|_\infty &\leq \|H_{\mu_k}(u_{k+1})\|_\infty + \sigma_k\mu_k \\ &\leq (1 - pt_k)\|H_{\mu_k}(u_k)\|_\infty + \sigma_k\mu_k \\ &\leq (1 - p\eta)\theta\mu_k + \frac{\theta p\eta}{\theta+1}\mu_k \\ &= (1 - \frac{\theta p\eta}{\theta+1})\theta\mu_k \\ &\leq \theta(1 - \sigma_k)\mu_k. \end{aligned}$$

The second inequality is based on (2.26). Therefore inequality (2.4) always holds for every $\sigma_k \in (0, \frac{p\eta\theta}{\theta+1})$ after the implementation of **Step 2**. By the backtracking, $\sigma_k \geq \frac{\beta p\eta\theta}{\theta+1}$ can be derived.

When $\|H_{\mu_k}(u_k)\| = 0$, **Step 2** in **PPFA** is skipped. For $\sigma_k \in (0, \frac{\theta}{\theta+1})$, we have

$$\|H_{(1-\sigma_k)\mu_k}(u_{k+1})\|_\infty = \sigma_k\mu_k \leq (1 - \frac{\theta}{\theta+1})\theta\mu_k \leq \theta(1-\sigma_k)\mu_k.$$

So that $\sigma_k \geq \frac{\beta\theta}{\theta+1}$ can be derived by the backtracking. Combining the above analysis for $\|H_{\mu_k}(u_k)\|_\infty \neq 0$ and $\|H_{\mu_k}(u_k)\|_\infty = 0$, the lower bound of σ_k can be derived, which guarantees the convergence of **PPFA**.

(c) From the analysis of (a) and (b), it follows that

$$\mu_k = (1-\sigma_{k-1})\cdots(1-\sigma_0)\mu_0 \leq (1 - \frac{\beta\theta p\eta}{\theta+1})^k\mu_0.$$

So that for $\hat{k} \geq \lceil \frac{ln\epsilon - ln\mu_0}{ln(1-\frac{\beta\theta p\eta}{\theta+1})} \rceil$, $\mu_{\hat{k}} \leq \epsilon$ holds. The iteration complexity is presented as follows.

When n is large and ϵ is small, we have the following estimation

$$\frac{ln\epsilon - ln\mu_0}{ln(1 - \frac{\beta p\eta\theta}{\theta+1})} \approx \frac{ln\epsilon - ln\mu_0}{-\frac{\beta\theta p\eta}{\theta+1}} = \frac{\mu_0(\theta+1)ln(\frac{\mu_0}{\epsilon})}{\alpha\beta p}L_\epsilon K_\epsilon^2. \tag{2.41}$$

From the expressions of L_ϵ and K_ϵ in (2.15) and (2.27), respectively, we have

$$L_\epsilon K_\epsilon^2 = O(\epsilon^{-\overline{w}_1} n^{\overline{w}_2}), \tag{2.42}$$

where $\overline{w}_1$ and $\overline{w}_2$ are defined in (2.40). From (2.41) and (2.42), we can derive the iteration complexity expression (2.39).

□

2.3 Initialization for Implementation

As the implementation and convergence of **PPFA** are discussed, the next thing we need to consider is how to find a proper starting point. An augmented problem is developed, whose initial point lying on the parameterized central path Ξ is easy to be obtained. A sufficient condition revealing the relationship of any optimal solutions between the original problem and the augmented problem is provided. Based on this finding, the optimal solution of the original problem can be constructed from the optimal solution of the augmented problem. Therefore, it would be sufficient to solve this augmented problem, which has a warm start. Our work is mainly due to the initialization part of [36, 52, 58, 59, 71].

2.3.1 An Augmented Problem

Before introducing our augmented problem, we first define some quantities. Let λ and τ be some positive constants, and K_b and K_c be defined as follows

$$K_b = \tau\lambda(n+1) - \lambda\nabla f(\lambda e)^T e, \quad K_c = \tau\lambda^{\frac{\gamma_1}{\gamma_2}}.$$

Then our augmented (convex) problem of (P) is stated as follows

$$\begin{array}{ll} \min & f(x) + K_c x_{n+1} \\ \text{s.t.} & Ax + (b - \lambda Ae)x_{n+1} = b, \\ & (\tau e - \nabla f(\lambda e))^T x + \tau x_{n+2} = K_b, \\ & x \geq 0, x_{n+1} \geq 0, x_{n+2} \geq 0. \end{array} \tag{P'}$$

The dual problem of (P') is as follows

$$\begin{array}{ll} \max & f(x) - \nabla f(x)^T x + b^T y + K_b y_{m+1}, \\ \text{s.t.} & A^T y + (\tau e - \nabla f(\lambda e))y_{m+1} + s = \nabla f(x), \\ & (b - \lambda Ae)^T y + s_{n+1} = K_c, \\ & \tau y_{m+1} + s_{n+2} = 0, \\ & s \geq 0, s_{n+1} \geq 0, s_{n+2} \geq 0. \end{array} \tag{D'}$$

A pair of the feasible solutions for (P') and (D') can be given by

$$x_0 = \begin{pmatrix} \lambda e \\ 1 \\ \lambda \end{pmatrix}, \; y_0 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \; s_0 = \begin{pmatrix} \tau e \\ \tau\lambda^{\frac{\gamma_1}{\gamma_2}} \\ \tau \end{pmatrix}.$$

Here $(x_0)_i^{\gamma_1}(s_0)_i^{\gamma_2} = \lambda^{\gamma_1}\tau^{\gamma_2}$, which means that the triplet (x_0, y_0, s_0) is on the parameterized central path Ξ with $\mu_0 = \lambda^{\gamma_1}\tau^{\gamma_2}$.

The following proposition will be used to reveal the relationship of the augmented problem and the original problem.

Proposition 2.1. *If $\overline{x}$ and $(\overline{x}, \overline{y}, \overline{s})$ are optimal solutions for problem (P) and (D), then $\overline{x}^T\overline{s} = 0$, the optimal value of (P) and (D) are the same. Conversely, if $\overline{x}$ is feasible for (P) such that $(\overline{x}, \overline{y}, \overline{s})$ is feasible for (D), and $\overline{x}^T\overline{s} = 0$, then $\overline{x}$ and $(\overline{x}, \overline{y}, \overline{s})$ are optimal solutions for problem (P) and (D).*

Proof. The similar complementary slackness condition is presented in [59] Proposition 2.3 for quadratic programming problems. We give a proof for our case here.

For the first part of the proposition, if $\overline{x}$ is an optimal solution for problem (P), then there exists $(\overline{x}, y, s)$ satisfying the KKT condition

$$\begin{cases} A\overline{x} = b \\ -\nabla f(\overline{x}) + A^T y + s = 0 \\ \overline{x}^T s = 0, \overline{x} \geq 0, s \geq 0. \end{cases}$$

So that $(\overline{x}, y, s)$ is a feasible solution for (D) with object function value $f(\overline{x})$. Since $(\overline{x}, \overline{y}, \overline{s})$ is optimal for (D), we know

$$f(\overline{x}) - \nabla f(\overline{x})^T\overline{x} + b^T\overline{y} = f(\overline{x}) - \overline{x}^T\overline{s} \geq f(\overline{x}).$$

The nonnegativity of $\overline{x}$ and $\overline{s}$ tells that $\overline{x}^T\overline{s} = 0$ and the optimal value of (P) and (D) are the same.

For the second part of the proposition, $\overline{x}$ and $(\overline{x}, \overline{y}, \overline{s})$ are feasible for (P) and (D) respectively with $\overline{x}^T\overline{s} = 0$, which means $(\overline{x}, \overline{y}, \overline{s})$ is a solution for the KKT condition of (P), so that $\overline{x}$ is an optimal solution for (P) with optimal value $f(\overline{x})$. By the weak dual property of Wolfe dual [82], we know that, for any feasible solution (v, y, s) for (D), the following holds

$$f(v) - \nabla f(v)^T v + b^T y \leq f(\overline{x}).$$

Moreover, $(\overline{x}, \overline{y}, \overline{s})$ is feasible for (D) and

$$f(\overline{x}) - \nabla f(\overline{x})^T \overline{x} + b^T \overline{y} = f(\overline{x}) - \overline{x}^T \overline{s} = f(\overline{x}).$$

So that $(\overline{x}, \overline{y}, \overline{s})$ is a optimal solution for (D). □

Next, a sufficient condition is provided to show the relationship of optimal solutions between (P') and (P).

Theorem 2.7. *Let x^* and (x^*, y^*, s^*) be optimal solutions of the original problems (P) and (D), respectively. Suppose that*

$$\begin{cases} K_b - (\tau e - \nabla f(\lambda e))^T x^* > 0, \\ K_c - (b - \lambda A e)^T y^* > 0. \end{cases} \tag{2.43}$$

Then, a feasible solution $(\hat{x}, \hat{x}_{n+1}, \hat{x}_{n+2})$ of (P') is an optimal solution if and only if $\hat{x}$ is an optimal solution of (P) and $\hat{x}_{n+1} = 0$.

Proof. First, let us define the following notations

$$\begin{cases} \widetilde{x} = (x^{*T}, x^*_{n+1}, x^*_{n+2})^T = (x^{*T}, 0, \frac{1}{\tau}(K_b - (\tau e - \nabla f(\lambda e))^T x^*))^T, \\ \widetilde{y} = (y^{*T}, y^*_{m+1})^T = (y^{*T}, 0)^T, \\ \widetilde{s} = (s^{*T}, s^*_{n+1}, s^*_{n+2})^T = (s^{*T}, K_c - (b - \lambda A e)^T y^*, 0)^T. \end{cases} \tag{2.44}$$

We now show $\widetilde{x}$ and $(\widetilde{x}, \widetilde{y}, \widetilde{s})$ are optimal solutions for (P') and (D'), respectively. From (2.43), we can easily verify that $\widetilde{x}$ and $(\widetilde{x}, \widetilde{y}, \widetilde{s})$ are feasible for (P') and (D'), respectively. In addition we have $\widetilde{x}^T \widetilde{s} = 0$, by Proposition 2.1, $\widetilde{x}$ and $(\widetilde{x}, \widetilde{y}, \widetilde{s})$ are optimal solutions for (P') and (D'), respectively.

Next, we prove the only if part. Let $(\hat{x}, \hat{x}_{n+1}, \hat{x}_{n+2})$ be any optimal solution of (P'), and the corresponding complementary variable in (D') be $(\hat{s}, \hat{s}_{n+1}, \hat{s}_{n+2})$. We first show $\hat{x}_{n+1} = 0$ and then show that $\hat{x}$ is an optimal solution of (P).

If $\hat{x}_{n+1} > 0$, then

$$
\begin{aligned}
f(x^*) + 0 \cdot K_c &= f(x^*) - \nabla f(x^*)^T x^* + b^T y^* \\
&= f(x^*) - \nabla f(x^*)^T x^* + {y^*}^T (A\hat{x} + (b - \lambda Ae)\hat{x}_{n+1}) \\
&= f(x^*) - \nabla f(x^*)^T x^* + (\nabla f(x^*) - s^*)^T \hat{x} + {y^*}^T (b - \lambda Ae)\hat{x}_{n+1} \\
&= f(x^*) + \nabla f(x^*)(\hat{x} - x^*) - {s^*}^T \hat{x} + {y^*}^T (b - \lambda Ae)\hat{x}_{n+1} \\
&< f(x^*) + \nabla f(x^*)(\hat{x} - x^*) - {s^*}^T \hat{x} + K_c \hat{x}_{n+1} \\
&\leq f(\hat{x}) + K_c \hat{x}_{n+1} - {s^*}^T \hat{x} \\
&\leq f(\hat{x}) + K_c \hat{x}_{n+1}.
\end{aligned}
$$

The first equality comes from the same optimal value of (P) and (D). The second and third equalities are based on the constraints of (P') and (D), respectively. The strict inequality is due to (2.43) and $\hat{x}_{n+1} > 0$. The last two inequalities are based on some convex function property and nonnegativity of the feasible solutions, respectively.

$f(x^*) + 0 \cdot K_c < f(\hat{x}) + K_c \hat{x}_{n+1}$ contradicts to that $(\hat{x}, \hat{x}_{n+1}, \hat{x}_{n+2})$ is an optimal solution for (P'). Thus $\hat{x}_{n+1} = 0$. Therefore we can obtain $f(x^*) = f(\hat{x})$ and $A\hat{x} = b$, which indicates that $\hat{x}$ is an optimal solution of (P).

Now, we prove the if part. Since $(\hat{x}, \hat{x}_{n+1}, \hat{x}_{n+2})$ is feasible for (P'), we have $\hat{x}_{n+2} = \frac{1}{\tau}(K_b - (\tau e - \nabla f(\lambda e))^T \hat{x}$. This solution is of the form $(\hat{x}, 0, \frac{1}{\tau}(K_b - (\tau e - \nabla f(\lambda e))^T \hat{x})$, which is the same as $\widetilde{x}$ in (2.44). So from the above analysis, $(\hat{x}, \hat{x}_{n+1}, \hat{x}_{n+2})$ is an optimal solution of (P'). $\square$

In practice, x^* and y^* are unknown, (2.43) can be expressed as

$$
\begin{cases}
\tau((n+1)\lambda - e^T x^*) + \nabla f(\lambda e)^T (x^* - \lambda e) > 0, \\
\tau \lambda^{\frac{\gamma_1}{\gamma_2}} - (b - \lambda Ae)^T y^* > 0.
\end{cases}
\tag{2.45}
$$

In the selection of τ and λ such that (2.45) holds, we can consider the following procedure: (a) λ is chosen first to satisfy $(n+1)\lambda - e^T x^* > 0$; and (b) τ can be chosen large enough to ensure that both inequalities are satisfied. In real implementation, a pair of τ and λ can be selected first. Then the augmented problem is solved. If (2.43) is satisfied, stop; otherwise repeat with a larger pair of τ and λ.

2.3.2 Implementation Procedure

Following Theorem 2.7, if τ and λ satisfy (2.43), Assumption 1 and Assumption 2 surely hold for the augmented problem. Therefore, an optimal solution of the augmented problem will lead to an optimal solution of the original problem.

To implement **PPFA**, an augmented problem of $n + 2$ variables together with its initial solution can be derived. Then, an optimal solution for the augmented problem can be obtained by **PPFA**. The first n components of the derived optimal solution form an optimal solution for the original problem. In summary, the implementation procedure can be outlined by the following 3 steps.

(i) Construct an augmented problem and the corresponding initial solution.

(ii) Solve the augmented problem by **PPFA**.

(iii) Derive an optimal solution from an optimal solution of the augmented problem.

2.4 Numerical Tests

In this part, we first explain some conditions related to the interior point path following method and introduce some objective functions we used for the tests. Then, a comparison of **PPFA** and **fmincon** in MATLAB is presented. After that, we do some tests to show more characteristics of **PPFA**.

2.4.1 Different Conditions and Functions for Tests

There are some famous conditions related to interior point method: (a) the self-concordance condition (SC) [17, 63]; (b) the relative Lipschitz continuous condition

(RLC) [28]; (c) the scaled Lipschitz condition (SLC) [90]; and (d) the condition proposed by Menteiro and Alder [60].

The first two conditions are mainly for convex programming with nonlinear constraints, and the third condition is for linearly constrained convex programming (LCCP). The condition used by Monteiro et al. [60] is for separable convex problems with linear constraints, which is a special case of the scaled Lipschitz condition [90]. All of these conditions characterize a certain degree of smoothness for the problems.

The scaled Lipschitz condition is one of the main conditions proposed for LCCP [90] to achieve a polynomial complexity path following algorithm. It was proposed by Zhu [90] in 1992. The definition is as follows.

Definition 2.2 ([90]). *Given* $0 < \beta < 1$, *if there exists* $M > 0$ *and for any* $x > 0, \Delta x$ *satisfying* $\|X^{-1}\Delta x\| \leq \beta$, *we have*

$$\|X[\nabla f(x + \Delta x) - \nabla f(x) - \nabla^2 f(x)\Delta x]\| \leq M\Delta x^T \nabla^2 f(x)\Delta x,$$

then we say that $f(x)$ *satisfies the scaled Lipschitz condition (SLC).*

Under this condition, Zhu [90] improved the work of Kortanek, Potra, and Ye [40]. Besides linear and convex quadratic functions, the entropy functions of the following form satisfy the scaled Lipschitz condition.

$$f(x) = \sum_{i=1}^{n} (x)_i \log((x)_i/(u)_i), (u)_i > 0, \forall i.$$

The required condition for **PPFA** is that $\nabla^2 f(x)$ is local Lipschitz continuous mentioned in Assumption 2. This condition makes some complementary to the scaled Lipschitz condition or the self-concordance condition.

The followings are some examples that satisfy our assumption but do not satisfy the scaled Lipschitz condition.

Example 2.1 (This function comes from [42]).

$$f(x) = \sum_i (x)_i \ln((x)_i) - (\sum_i (x)_i) \ln(\sum_i (x)_i).$$

This function is proved in [42] that does not satisfy the scaled Lipschitz continuous. However, the convexity of it is not explained. The Hessian of it is as follow

$$H_f(x) = \text{diag}(\frac{1}{(x)_i}) - \frac{1}{\sum_i (x)_i} ee^T.$$

Then, we show that $H_f(x)$ is positive semidefinite. For any vector y, we know

$$\begin{aligned}
y^T H_f(x) y &= \sum_i \frac{1}{(x)_i}(y)_i^2 - \frac{1}{\sum_i (x)_i}(\sum_i (y)_i)^2 \\
&= \frac{1}{\sum_i (x)_i}(\sum_i \frac{\sum_i (x)_i}{(x)_i}(y)_i^2 - (\sum_i (y)_i)^2) \\
&= \frac{1}{\sum_i (x)_i}(\sum_{i,j} \frac{(x)_j}{(x)_i}(y)_i^2 - (\sum_i (y)_i)^2) \\
&= \frac{1}{\sum_i (x)_i}(\frac{1}{2}\sum_{i,j}(\frac{(x)_j}{(x)_i}(y)_i^2 + \frac{(x)_i}{(x)_j}(y)_j^2) - (\sum_i (y)_i)^2) \\
&\geq \frac{1}{\sum_i (x)_i}(\sum_{i,j}(y)_i(y)_j - (\sum_i (y)_i)^2) \\
&= 0.
\end{aligned}$$

So that this function is convex. To make this example locally Lipschitz continuous on the positive half space, we make a shift of x resulting

$$\overline{f}(x) = \sum_{i=1}^{n}((x)_i+(a)_i)(\ln((x)_i+(a)_i)+\ln((c)_i)) - (\sum_{i=1}^{n}((x)_i+(a)_i))\ln(\sum_{i=1}^{n}((x)_i+(a)_i)).$$

Where $(a)_i > 0, (c)_i > 1$. Next, we show $\nabla^2 \overline{f}(x)$ is locally Lipschitz continuous.

$$\begin{aligned}
&\|H_{\overline{f}}(x) - H_{\overline{f}}(y)\|_F \\
=&\|\text{diag}(\frac{1}{(x)_i + (a)_i} - \frac{1}{(y)_i + (a)_i}) + (\frac{1}{e^T y + e^T a} - \frac{1}{e^T x + e^T a})ee^T\|_F \\
\leq&\|\text{diag}(\frac{1}{(x)_i + (a)_i} - \frac{1}{(y)_i + (a)_i})\|_F + \|(\frac{1}{e^T y + e^T a} - \frac{1}{e^T x + e^T a})ee^T\|_F \\
=&\sqrt{\sum \frac{((y)_i - (x)_i)^2}{((x)_i + (a)_i)^2((y)_i + (a)_i)^2}} + n|(\frac{1}{e^T y + e^T a} - \frac{1}{e^T x + e^T a})| \\
\leq&\frac{\|y - x\|}{\min((x)_i + (a)_i)((y)_i + (a)_i)} + \frac{n|e^T(y - x)|}{(e^T x + e^T a)(e^T y + e^T a)} \\
\leq&\frac{1}{\min(a)_i^2}\|x - y\| + \frac{n^{3/2}}{(e^T a)^2}\|x - y\| \\
\leq&(\frac{1}{\min(a)_i^2} + \frac{n^{3/2}}{(e^T a)^2})\|x - y\|.
\end{aligned}$$

When $(a)_i = 0.5, (c)_i = 2, n = 2$, the function is of the form

$$f(x) = \sum_{i=1}^{2} (x_i + 0.5)(\ln(x_i + 0.5) + \ln 2) - (x_1 + x_2 + 1) \ln(x_1 + x_2 + 1).$$

Example 2.2.

$$f(x) = \frac{1}{4}(x-1)^2 - \frac{1}{8} \cos 2(x-1).$$

The derivatives of different orders of this function are as

$$f^{'}(x) = \frac{1}{2}(x-1) + \frac{1}{4} \sin 2(x-1),\ f^{''}(x) = \frac{1}{2} + \frac{1}{2} \cos 2(x-1),\ f^{'''}(x) = -\sin 2(x-1).$$

Since $f^{'''}(x)$ is bounded on $(0, +\infty)$, $f^{''}(x)$ is locally Lipschitz continuous on $(0, +\infty)$. $f^{''}(1 + \frac{\pi}{2}) = 0$ indicates it does not satisfy the scaled Lipschitz condition. $f^{'}(x) = 0$ is attainable at $x = 1$. To create large scale problems of this kind, we construct separable functions of the form

$$f(x) = c^T x + \sum_{i=1}^{n} \frac{1}{4}((x)_i - 1)^2 - \cos(2((x)_i - 1)).$$

This kind of function is used in our numerical test as the objective function. The followings are other examples that are locally Lipschits continuous on the positive space and do not obey the scaled Lipschitz continuous.

Example 2.3.

$$f(x) = \frac{1}{2}(x+b)^2 - 2a(x+b)\ln(x+b) + 2a(x+b) - a^2 \ln(x+b), \text{with } a > b > 1.$$

Example 2.4.

$$f(x) = \frac{1}{6}(x+b)^3 - a(x+b)^2 + a^2(x+b)\ln(x+b) - a^2(x+b), \text{with } a > b > 0.$$

Example 2.5.

$$f(x) = \frac{1}{4}e^{-2x} - e^{-x} + 0.125x^2 - 2x.$$

2.4.2 Comparison with *fmincon* in Matlab

In this part, **PPFA** is compared with a Matlab function *fmincon* for two classes of randomly generated test problems. All our numerical experiments were conducted in Matlab (2019a) platform on a computer with an Intel Core i5-6500 CPU @3.20GH and 32GB memory.

Two Classes of Test Problems

The objective functions in the following two classes of test problems are convex, obeying locally Lipchitz assumption and not satisfying the scaled Lipschitz condition.

Problem 1:

$$\begin{aligned} \min_x \quad & f(x) = c^T x + \textstyle\sum_{i=1}^n \frac{1}{4}((x)_i - 1)^2 - \frac{1}{8}\cos(2((x)_i - 1)) \\ s.t. \quad & Ax = b, \ x \geq 0. \end{aligned} \tag{2.46}$$

Problem 2: The second test problem is from [42].

$$\begin{aligned} \min_x \quad & f(x) = c^T x + \textstyle\sum_{i=1}^n ((x)_i + 0.5)(\ln((x)_i + 0.5) + \ln(2)) \\ & -(\textstyle\sum_{i=1}^n ((x)_i + 0.5)) \ln(\textstyle\sum_{i=1}^n ((x)_i + 0.5) \\ s.t. \quad & Ax = b, \ x \geq 0. \end{aligned} \tag{2.47}$$

For above problems, we set (i) $A = randn(m, n)$; (ii) an optimal solution $x^* = abs(randn(n, 1))$ with $p\%$ of x^* components being 0; (iii) the dual variable (s^*, y^*) being randomly generated and satisfying complementarity with x^*, in addition, the strict complementarity is not needed; only half components of s^* corresponding to zero components of x^* are set to 0; (iv) $b = Ax^*$; and (v) c is obtained from the KKT condition.

fmincon and PPFA Setting

Our algorithm **PPFA** is compared with the Matlab function *fmincon*. In our test, both the gradient and Hessian are used in *fmincon*.

The main setting of *fmincon* is as follows.

Table 2.1: Setting of *fmincon*

Algorithm	Interior point
Max Function Evaluations	$300,000$
Specify Objective Gradient	TRUE
Hessian Function	Hessian of the tested problem
Max Iteration	$1,000$
Optimality Tolerance (OptTol)	Controlled from $1.0e-9$ to $1.0e-11$

For **PPFA**, the augmented problem is solved after fixing λ and τ which satisfy (2.45). In our numerical tests, since the optimal solution is randomly generated, we simply set τ and λ 10 times of the least requirement, i.e.,

$$\lambda = 10 \times \frac{e^T x^*}{n+1}, \quad \tau = 10 \times |\max\{-\frac{\nabla f(\lambda e)(x^* - \lambda e)}{(n+1)\lambda e^T x^*}, \frac{(b - \lambda A e)^T y^*}{\lambda^{\frac{\gamma_1}{\gamma_2}}}\}|.$$

The derived initial solution for the augmented problem has the property that $r_0 = A^T y_0 + s_0 - \nabla f(x_0) = 0$, which relaxes the θ setting in **PPFA** as $\theta \in (0, min\{1, \frac{min(s_0)_i}{\mu_0}\})$. As illustrated in **PPFA**, the stopping criterion is $\mu_k \leq \epsilon$, the smaller ϵ would get a better solution.

Numerical Results

The number of linear equality constraints is set to $m = 0.4n$, 30% of x^* components are set to 0, $\gamma_1 = \gamma_2 = 0.5$. In the following tables, we report the achieved optimal value f^*, the relative error (RelErr) $\frac{f(x_k) - f^*}{1 + |f^*|}$, CPU time in seconds, iteration number (Iter), and constraint error (ConsErr) $\|Ax - b\|_\infty$. Each number reported represents the average of 5 runs.

Table 2.2: Results for Problem 1 with various OptTol and ϵ values

	fmincon: OptTol=1.0e-9				**PPFA**: ϵ=1.0e-4			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	37.75	17.2	4.75e-09	2.10e-13	9.22	32.8	1.37e-08	2.19e-09
5,000	194.74	17.8	1.30e-08	4.63e-13	47.72	33.8	1.46e-08	3.67e-09
7,500	528.80	18.8	2.46e-09	2.10e-12	138.61	36.0	1.34e-08	4.29e-09
10,000	1063.00	18.6	3.75e-09	2.86e-12	289.49	35.8	1.04e-08	1.47e-09
12,500	1900.95	18.4	2.02e-08	3.30e-12	804.83	53.4	1.27e-08	1.21e-10
15,000	3368.12	20.0	5.52e-09	4.45e-12	846.65	36.2	9.81e-09	2.74e-09
17,000	*	*	*	*	1203.58	36.8	3.08e-08	5.05e-09
	fmincon: OptTol=1.0e-10				**PPFA**: ϵ=1.0e-5			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	44.32	20.0	1.22e-10	1.85e-13	10.18	36.0	5.28E-11	3.69E-11
5,000	215.04	19.6	2.36e-09	4.21e-13	53.46	37.6	1.22e-10	3.17e-11
7,500	559.18	19.8	6.60e-10	1.98e-12	152.48	39.6	3.22e-10	1.07e-10
10,000	1091.53	19.2	1.26e-09	3.07e-12	313.04	38.8	8.54e-10	3.39e-11
12,500	2001.59	19.6	1.62e-09	3.26e-12	841.23	56.4	6.67e-11	1.83e-11
15,000	3412.31	20.6	9.87e-09	4.21e-12	924.60	39.8	2.19e-10	5.89e-11
17,000	*	*	*	*	1299.96	39.8	2.26e-10	6.47e-11
	fmincon: OptTol=1.0e-11				**PPFA**: ϵ=1.0e-6			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	44.83	20.6	5.34e-11	1.98e-13	11.35	40.0	3.38e-13	1.23e-12
5,000	235.26	21.4	8.23e-11	4.38e-13	59.49	41.6	7.63e-13	3.02e-12
7,500	616.11	21.8	3.70e-11	1.87e-12	167.65	43.4	5.30e-12	1.19e-11
10,000	1280.16	22.2	1.83e-11	2.48e-12	346.28	42.8	5.59e-12	1.36e-11
12,500	2305.72	22.2	5.81e-11	3.46e-12	901.05	60.2	6.04e-13	1.80e-11
15,000	3663.21	21.8	5.65e-10	4.29e-12	1022.69	43.8	1.68e-12	2.39e-11
17,000	*	*	*	*	1421.96	43.6	5.88e-12	2.96e-11

* "Out of memory" error was reported by Matlab.

From the results in Table 2.4.2, it can be observed that: (i) The CPU times of **PPFA** is about 1/4 to 1/3 of CPU times of *fmincon* while maintaining the same order of accuracy (RelErr). (ii) The iteration numbers of **PPFA** are about twice as much as the ones of *fmincon*, but the cost per iteration for **PPFA** is about 1/8 to 1/6 of the ones for *fmincon*. (iii) The iteration numbers for both **PPFA** and *fmincon* are relatively insensitive to the problem size n. This is one of the attractive features for interior point methods. (iv) The values of $\|Ax - b\|$ (ConsErr) for *fmincon* are relatively smaller than the ones for **PPFA**. (v) The largest solvable problem size for *fmincon* is $n = 15,000$, while the largest solvable problem size for **PPFA** is $n = 37,000$. The results for problem 2 are summarized as follows.

Table 2.3: Results for Problem 2 with various OptTol and ϵ values

	fmincon: OptTol=1.0e-9				**PPFA**: ϵ=1.0e-4			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	93.82	15.2	1.3eE-09	2.12e-13	8.67	28.6	1.17e-09	2.76e-09
5,000	608.36	16.0	6.01e-10	4.35e-13	45.94	30.4	6.50e-10	2.22e-09
7,500	1961.47	16.2	4.99e-10	1.96e-12	217.38	47.8	7.37e-10	5.16e-09
10,000	4452.88	17.0	3.73e-10	2.37e-12	269.00	32.0	5.35e-10	1.27e-09
12,500	*	*	*	*	458.05	33.8	9.14e-10	4.41e-10
	fmincon: OptTol=1.0e-10				**PPFA**: ϵ=1.0e-5			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	108.07	17.4	9.15e-11	2.29e-13	9.83	32.2	1.21e-11	2.85e-11
5,000	659.87	17.4	1.06e-10	4.50e-13	51.96	34.2	6.19e-12	2.34e-11
7,500	2197.25	18.0	9.18e-11	1.98e-12	231.92	51.4	9.31e-12	6.01e-11
10,000	4838.22	18.6	8.60e-11	2.54e-12	295.20	35.2	1.23e-11	3.76e-11
12,500	*	*	*	*	504.49	37.2	1.28e-11	1.77e-11
	fmincon: OptTol=1.0e-11				**PPFA**: ϵ=1.0e-6			
n	CPU	Iter	RelErr	ConsErr	CPU	Iter	RelErr	ConsErr
2,500	119.00	19.0	6.69e-12	2.08e-13	11.11	36.0	1.04e-13	1.46E-12
5,000	726.90	19.2	6.61e-12	4.49e-13	57.38	37.4	1.03e-13	3.05e-12
7,500	2355.87	19.2	7.73e-12	1.90e-12	249.90	55.6	6.85e-14	1.21e-11
10,000	5231.99	20.2	5.68e-12	2.64e-12	327.38	39.0	9.24e-14	1.43e-11
12,500	*	*	*	*	558.56	41	1.22e-13	1.81e-11

* "Out of memory" error was reported by Matlab.

Problem 2 is non-separable and thus is a little bit more difficult. Similar numerical performance was observed for the two algorithms, yet the CPU ratio for the two algorithms tends to be larger. The largest solvable problem size for *fmincon* is $n = 10,000$, while the largest solvable problem size for **PPFA** is $n = 32,000$.

2.4.3 Characteristics of the Parameterized Path Following Algorithm

More characteristics of **PPFA** are tested and illustrated in this part. Problem 1 (2.46) and Problem 2 (2.47) are randomly generated for these tests. Except for special statements, all the other settings are the same with section 2.4.2 for every test. We explore (i) different choice of γ_1 and γ_2; (ii) two different stop criteria; (iii) comparison of ∞-norm and 2-norm cases; (iv) sensitive analysis of τ and λ.

Different Choice of γ_1 and γ_2

γ_1 and γ_2 should be neither too large nor too small. The different γ_1 and γ_2 indicate different search directions, which will lead to different efficiency of **PPFA**. From the worst-case complexity expression, (2.39) and (2.40), we know that γ_1 and γ_2 should not be too small, or the worst-case complexity will be too large. By the μ_k reduction expression in Theorem 2.6 that

$$\mu_k = (1-\sigma_{k-1})\cdots(1-\sigma_0)\mu_0, \sigma_i \geq \frac{\beta\theta p\eta}{\theta+1}, i = 0, 1, \cdots, k-1,$$

we know that θ should not be too small or μ_k will decrease slowly. It is reasonable since small θ will make a narrow neighborhood of the path. The θ is decided by

$$\theta \in (0, \min\{1, \frac{\min_i(s_0)_i}{\mu_0}\}) = (0, \min\{1, \frac{\min_i(\lambda^{-\gamma_1}, \lambda^{\gamma_1/\gamma_2-\gamma_1})}{\tau^{\gamma_2-1}}\}). \tag{2.48}$$

The equality holds due to the formulas of s_0 and μ_0 in the augmented problem. The chosen τ and λ are large enough to guarantee (2.45) so that larger γ_1 and γ_2 will make the interval for θ smaller. Therefore, γ_1 and γ_2 should not be too large.

Moreover, for the case $\gamma_1 = \gamma_2 = \gamma$ and $\lambda > 1$, (2.48) is simplified as

$$\theta \in (0, \min\{1, \frac{\tau}{(\lambda\tau)^\gamma}\}).$$

To larger θ, we should better make $\frac{\tau}{(\lambda\tau)^\gamma}$ larger or equal to 1. We may set $\gamma \leq \frac{\ln\tau}{\ln\tau+\ln\lambda}$. Set $n = 1000$, γ_1 and γ_2 are ranging from 0.2 to 1.3, the time cost (average of 5 times) to achieve relative error 1.0e-5 for different γ_1 and γ_2 are presented.

Table 2.4: Time Cost for Different γ_1 and γ_2 (Problem 1)

$\gamma_1 \backslash \gamma_2$	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3
0.3	0.91	0.97	1.07	1.12	1.23	1.31	2.77	10.64	19.55	25.62	31.90
0.4	0.97	0.92	0.97	1.01	1.07	1.15	3.45	6.85	10.35	13.85	17.77
0.5	1.08	0.95	0.97	0.97	1.11	1.47	2.84	4.60	6.60	8.97	11.99
0.6	1.20	1.01	0.97	0.96	1.02	1.66	2.72	3.86	5.40	7.19	9.73
0.7	1.32	1.09	0.98	1.01	1.30	1.97	2.73	3.67	5.12	6.99	9.60
0.8	1.53	1.25	1.09	1.07	1.81	2.34	2.96	3.96	5.21	7.10	9.83
0.9	1.73	1.34	1.22	1.67	2.37	2.87	3.41	4.45	6.08	8.31	11.56
1	2.00	1.51	1.33	2.57	3.06	3.45	4.24	5.08	6.77	9.26	13.27
1.1	2.27	1.70	2.52	3.65	3.92	4.23	5.07	6.46	8.65	11.65	16.15
1.2	2.60	1.90	4.47	4.94	4.94	5.41	6.08	7.61	10.02	13.89	20.54
1.3	2.95	4.01	6.38	6.30	6.10	6.58	7.78	9.77	12.91	17.36	23.80

Table 2.5: Time Cost for Different γ_1 and γ_2 (Problem 2)

$\gamma_1 \backslash \gamma_2$	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1	1.1	1.2	1.3
0.3	0.81	0.84	0.95	1.05	1.14	2.43	11.22	28.79	51.02	65.04	76.04
0.4	0.93	0.82	0.83	0.93	1.01	3.22	8.71	14.59	20.70	27.50	34.30
0.5	1.01	0.94	0.91	0.90	1.47	2.99	5.08	7.63	10.72	14.40	18.79
0.6	1.20	1.02	0.94	1.15	1.80	2.70	3.82	5.34	7.41	9.73	12.67
0.7	1.36	1.14	1.13	1.71	2.21	2.76	3.59	4.56	6.06	8.08	10.40
0.8	1.57	1.32	2.20	2.64	2.85	3.15	3.74	4.66	5.66	7.24	9.21
0.9	1.73	2.56	3.59	3.45	3.49	3.70	4.22	5.04	6.28	7.93	9.94
1	3.87	5.34	4.95	4.51	4.35	4.62	4.99	5.66	6.86	8.40	10.51
1.1	7.20	8.02	6.58	5.90	5.46	5.56	6.13	7.07	8.44	10.29	12.76
1.2	14.38	10.64	8.45	7.38	6.90	6.73	7.20	8.19	9.67	11.72	14.78
1.3	20.77	13.68	10.43	8.93	8.33	8.45	9.15	10.33	12.28	14.84	18.18

The average time cost for the case $\gamma_1 = \gamma_2 = \frac{\ln(\tau)}{\ln(\tau)+\ln(\lambda)}$ for Problem 1 and Problem 2 are 0.9996s and 0.8994s respectively.

From above results we conclude: (i) the case $\gamma_1 = \gamma_2$ results in better efficiency; (ii) $\gamma_1 = \gamma_2 = \frac{\ln \tau}{\ln \tau + \ln \lambda}$ is a solid choice with a relative small time cost. (iii) The time cost is high when γ_1 and γ_2 differ from each other greatly.

Two Different Stop Criteria

PPFA will solve the augmented problem of the original problem. Two signs will indicate the optimal solution is found. One is that μ_k is close to 0, and the other is that $(x_k)_{n+1}$ is close to 0. Therefore we propose two possible stop criteria:

(1) Stop criterion 1: $\mu_k \leq \epsilon$;

(2) Stop criterion 2: $(x_k)_{n+1} \leq \epsilon$.

The relative errors achieved along with different tolerance ϵ under these two stop criteria are presented as follows. Problem 1 and Problem 2 with different scale n are tested. γ_1 and γ_2 are set to be 0.5 for the tests.

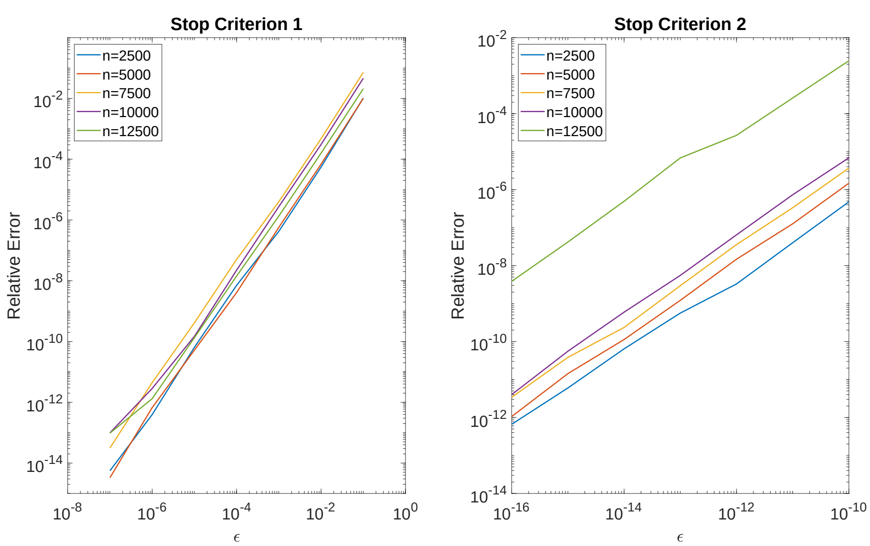

Figure 2.1: Relative Error Achieved for Different Stop Criteria (Problem 1)

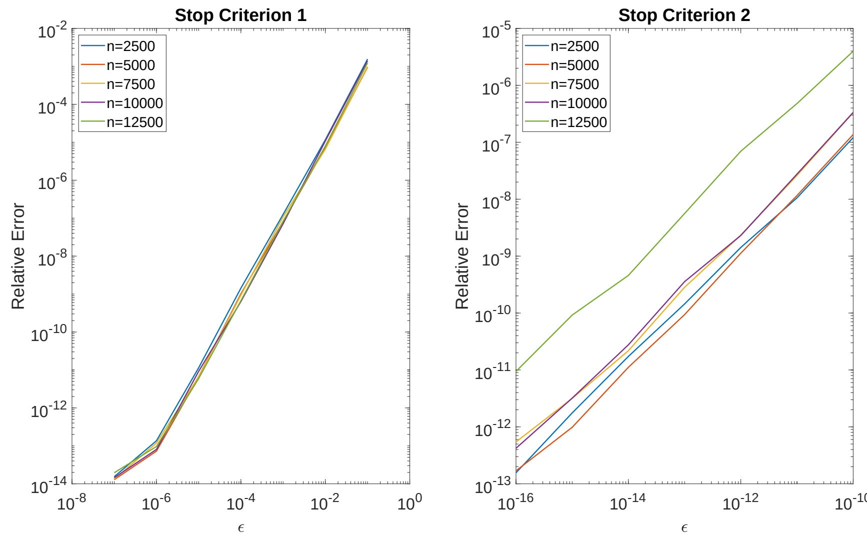

Figure 2.2: Relative Error Achieved for Different Stop Criteria (Problem 2)

From the above figures, we can conclude that, to achieve the same relative error, the tolerance needed for stop criterion 2 is much smaller than stop criterion 1. For example, to ensure the relative error under 1.0e-12, ϵ=1.0e-16 should be applied for stop criterion 2, while ϵ=1.0e-6 is enough to stop criterion 1. So that, stop criterion 1 may be better in numerical. With $\gamma_1 = \gamma_2 = 0.5$, μ^2 is the duality gap, which is the sum of the primal error and the dual error. Which means that, to achieve relative error less than ϵ, we may need to set stop criterion $\mu_k \leq \sqrt{\epsilon}$.

Next, under stop criterion 1, the decrease of μ and the relative error along with iterations are presented.

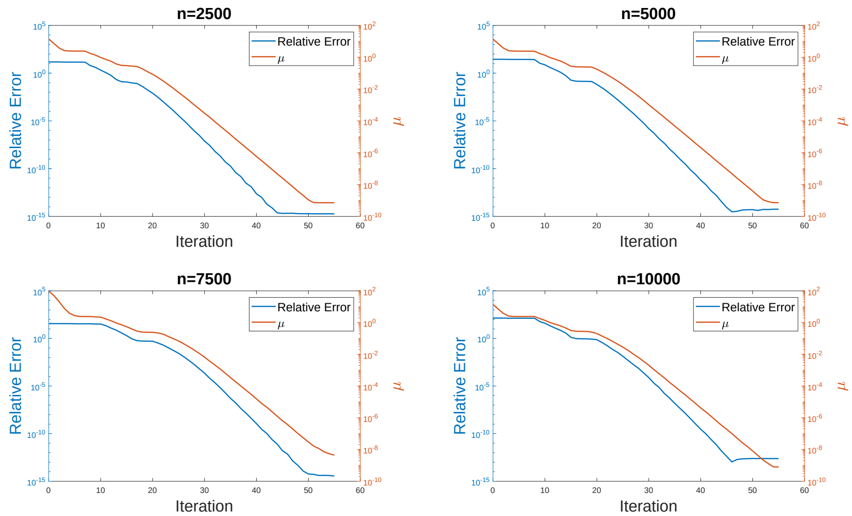

Figure 2.3: The Relative Error and μ along with Iterations (Problem 1)

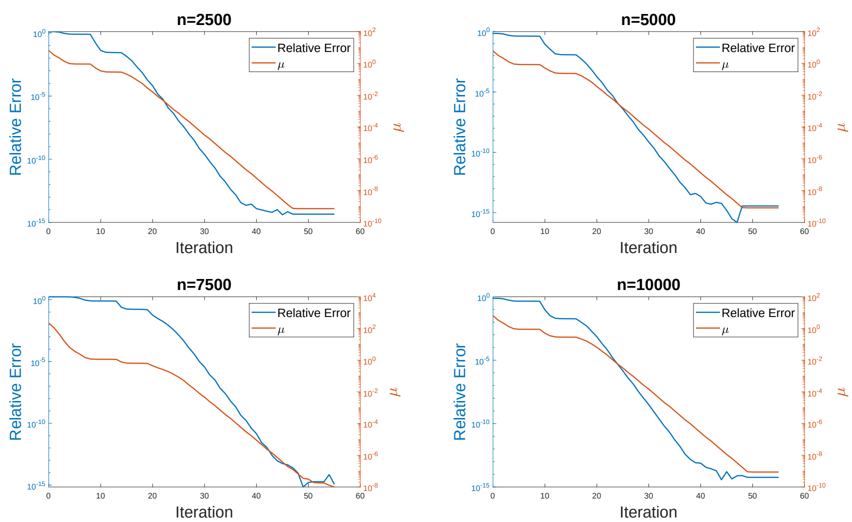

Figure 2.4: The Relative Error and μ along with Iterations (Problem 2)

These figures show that

(i) The relative error and μ will decrease along with the iterations. The middle part of every curve is likely linear so that the decreasing speed is more likely exponential since the y-axis is a log-axis.

(ii) The reliable precisions for relative error and μ are approximately less equal than 1.0e-14 and 1.0e-9, respectively. Numerical instability may happen when μ is too close to zero.

Comparison of ∞-Norm and 2-Norm Applied

The ∞-norm is used in **PPFA**, and it is the least required norm to guarantee the convergence. The convergence result will still work if we replace the ∞-norm with 2-norm in our algorithm. Becasue, if we restrict $\|H_\mu(u)\|_2 \leq \theta\mu$ in every iteration, surely $\|H_\mu(u)\|_\infty \leq \theta\mu$ will still hold. The ∞-norm provides a broader neighborhood of the path, which will improve efficiency.

For a given tolerance ϵ, the worst-case iteration numbers k_∞ and k_2 for ∞-norm and 2-norm cases are as follows

$$k_\infty = \lceil \frac{\ln\epsilon - \ln\mu_0}{\ln(1-\frac{\beta p\overline{\eta}\theta}{\theta+1})} \rceil \approx \frac{\mu_0(\theta+1)\ln\frac{\mu_0}{\epsilon}}{\alpha\beta p}\overline{L}_\epsilon\overline{K}_\epsilon^2,$$

$$k_2 = \lceil \frac{\ln\epsilon - \ln\mu_0}{\ln(1-\frac{\beta p\hat{\eta}\theta}{\theta+\sqrt{n}})} \rceil \approx \frac{\mu_0(\theta+\sqrt{n})\ln\frac{\mu_0}{\epsilon}}{\alpha\beta p}\hat{L}_\epsilon\hat{K}_\epsilon^2.$$

Where $\overline{\eta}, \overline{L}_\epsilon$, and $\overline{K}_\epsilon$ are the realizations of η, L_ϵ, and K_ϵ in (2.41) for the ∞-norm case. The same for $\hat{\eta}, \hat{L}_\epsilon$, and $\hat{K}_\epsilon$ to the 2-norm case. These approximations hold when ϵ is small and n is large. Then, we can derive the following approximation

$$\frac{k_2}{k_\infty} \approx \frac{\theta+\sqrt{n}}{\theta+1}\frac{\hat{L}_\epsilon\hat{K}_\epsilon^2}{\overline{L}_\epsilon\overline{K}_\epsilon^2}$$

The above formula tells that the quotient k_2/k_∞ is approximately a linear function for $\frac{\theta+\sqrt{n}}{\theta+1}$. Since k_2 and k_∞ are the theoretical results for the worst-case, we calculate the iteration numbers needed to satisfy $\mu_k \leq \epsilon$ instead.

Problem 1 and problem 2 are both tested, the problem scale n is ranging from 500 to 5000 with step size 500, the ϵ is set to be $1.0e-8$. The y-axis stands for the quotient k_2/k_∞ and the x-axis stands for $\frac{\theta+\sqrt{n}}{\theta+1}$. The result is showed below.

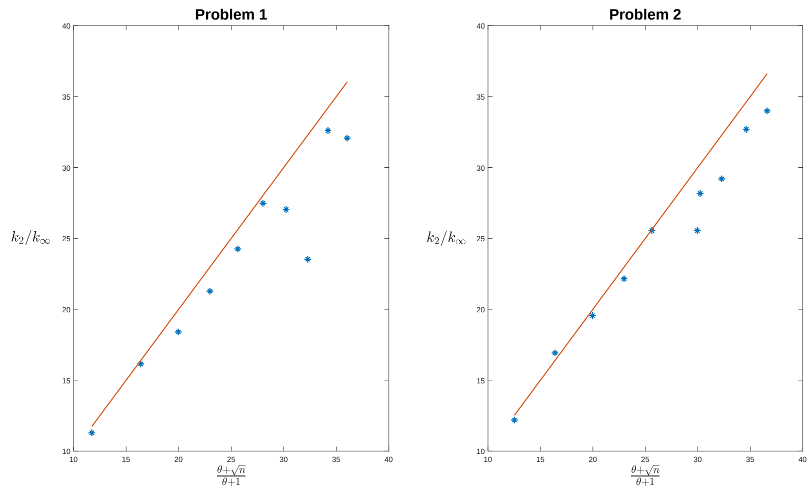

Figure 2.5: k_2/k_∞ along with $\frac{\theta+\sqrt{n}}{\theta+1}$

The above figure tells that k_2/k_∞ may have a linear relationship with $\frac{\sqrt{n}+\theta}{1+\theta}$. And, the 2-norm case needs much more iterations than ∞-norm case to achieve the same relative error. The quotient is increasing as the increasing of n. Moreover, the time cost of every iteration for 2-norm case is more than the ∞-norm case so that ∞-norm case is more efficient than the 2-norm case.

Sensitive Analysis for τ and λ

In our above numerical tests, the chosen τ and γ are set in the following way

$$\begin{cases} \lambda = k \times \dfrac{e^T x^*}{n+1}, \\ \tau = k \times |\max\{-\dfrac{\nabla f(\lambda e)(x^* - \lambda e)}{(n+1)\lambda e^T x^*}, \dfrac{(b - \lambda A e)^T y^*}{\lambda^{\frac{\gamma_1}{\gamma_2}}}\}|. \end{cases}$$

Where $k = 10$. The choice of λ and τ is related to the unknown solution. In previous test, we set it to be 10 times of the least required. Here, we make the sensitive analysis of λ and τ by ranging k from 100 to 1000 and see how it will affect the efficiency. The problem scale n is set to be 1000, the stop criterion is $\mu_k \leq 1.0e - 8$. The average execution time of five tests is showed below.

Table 2.6: Time Cost for Different τ and λ

k	100	200	300	400	500	600	700	800	900	1000
problem 1	2.01	2.43	3.15	3.88	4.57	5.51	5.56	6.21	6.57	7.01
problem 2	1.38	1.39	1.44	1.45	1.44	1.47	1.51	1.49	1.49	1.53

From the above, when k is enlarged from 100 to 1000, the execution time for Problem 2 is more stable. Although the execution time for Problem 1 is increasing, the increasing speed is much less than k. We therefore conclude that the change of λ and τ will not make a great effect to the efficiency.

Chapter 3

Path Following Algorithm for General Barrier Functions

In this chapter, the parameterized path following algorithm introduced in Chapter 2 is extended for more general cases. The problem we considered in this chapter is LCCP. Different barrier functions will produce different paths. We propose a class of barrier functions whose corresponding paths will lead to optimal solutions. The **PPFA** is extended for this kind of barrier functions. Also, implementation and convergence are studied in this chapter. Assumption 1 and Assumption 2 are needed for these results.

This extension provides wider class of barrier functions to be used for **PPFA**. This class of barrier function is large part of the Legendre type functions proposed by López [48] in 2010. A good thing for this class of barrier functions is that the sum of every two barrier functions is still in this class, and usually the sum of barrier functions will perform better than a single barrier function numerically. We can test and find good barrier functions more flexibly.

3.1 Paths for a Class of Barrier Function

The parameterized central path is corresponding to the barrier term (2.2). Using different barrier functions, we can derive different paths. We introduce a class of barrier functions in the following. The barrier functions $\varphi(x)$ we introduced are proper, extended real-valued, C^2, and strictly convex. They satisfy the following barrier conditions

(i) $\varphi'(x) < 0$ for $x \in (0, +\infty)$;

(iii) $\lim_{x \to +\infty} \varphi'(x) = 0$ and $\lim_{x \to 0^+} \varphi'(x) = -\infty$;

(iii) there exists a $\gamma < 1$ that $\varphi'(x) = O(x^{\gamma-1})$ as $x \to +\infty$.

This kind of function is similar to the Legendre function introduced by López [48]. For example, the function $\varphi_1(x) = -\ln(x)$ and $\varphi_2(x) = -x^{-1}$ satisfy these requirements. Also, the finite sum of these functions still satisfies the requirements.

We define the primal barrier problem $(LCCP_\mu)$ as follows

$$\begin{aligned} \min \quad & f(x) + \mu\Phi(x) \\ \text{s.t.} \quad & Ax = b, \\ & x \geq 0. \end{aligned} \tag{P_μ}$$

Where $\Phi(x) = \sum_{i=1}^{n} \varphi((x)_i)$. The Wolfe dual for P_μ is as follows

$$\begin{aligned} \max \quad & f(x) + \mu\Phi(x) - (\nabla f(x) + \mu\nabla\Phi(x))^T x + b^T y \\ \text{s.t.} \quad & -\nabla f(x) - \mu\nabla\Phi(x) + A^T y + s = 0, \\ & s \geq 0. \end{aligned} \tag{D_μ}$$

The optimal solution $x(\mu)$ for (P_μ) can be regarded as a path with parameter μ. Next, we discuss the convergence and optimality of these paths.

Theorem 3.1. *Under Assumption 1 and the property $\lim_{x \to +\infty} \varphi'(x) = 0$, the unique solution $x(\mu)$ for (P_μ) exists for all $\mu > 0$.*

Proof. The proof is inspired by [58] for linear programming. For a given $\mu > 0$, (P_μ) is either unbounded or has a unique optimal solution. We mainly prove that the unbounded case will not hold under our assumption.

We first prove that under Assumption 1, the set of optimal solutions of (P) is bounded. Denote x a feasible solution for (P) and (v, y, s) an interior feasible solution for (D), based on Wolfe dual in [82], we obtain

$$\begin{aligned}
f(x) &\geq f(v) + \nabla f(v)^T(x - v) \\
&= f(v) + \nabla f(v)^T x - \nabla f(v)^T v \\
&= f(v) + (A^T y + s)^T x - \nabla f(v)^T v \\
&= f(v) - \nabla f(v)^T v + b^T y + s^T x \\
&\geq f(v) - \nabla f(v)^T v + b^T y \\
&> -\infty.
\end{aligned}$$

The first inequality is due to convexity of $f(x)$. These equalities are based on the feasibility property of (P) and (D). The last two inequalities are base on the positivity of x and s and nonempty of $\mathcal{T}$, respectively. So that the optimal value of (P) is bounded. The existence of the interior feasible points for (D) means $s > 0$, which results in the optimal solution x^* for (P) should be bounded, or $s^T x^*$ will go to $+\infty$.

Next, we prove that the unbounded case for (P_μ) will not hold. If the feasible region of (P) is bounded, surely (P_μ) is bounded. If (P_μ) is unbounded, the infinity objective value can only achieved at the infinity, then there exists a feasible direction $d \geq 0$ that

$$\lim_{t \to +\infty} f(x^* + td) + \mu \Phi(x^* + td) = -\infty, \tag{3.1}$$

and $x^* + td$ is feasible for all $t \geq 0$. Next, we present a contradiction. There exist large enough $0 < t_0 \leq t_1$ that

$$0 < f(x^* + t_0 d) - f(x^*) \leq t_1 \nabla f(x^* + t_1 d)^T d. \tag{3.2}$$

The first inequality holds because the set of optimal solutions of (P) is bounded. The second inequality holds due to the monotonicity of the gradient of the convex function. Moreover, $\lim\limits_{x \to +\infty} \varphi'(x) = 0$ leads to

$$\lim_{s \to +\infty} \nabla \Phi(x^* + td)^T d = 0. \tag{3.3}$$

Combining (3.2) and (3.3), we know that there exist large enough t_1 and t_2 that $0 < t_1 < t_2$ making

$$\nabla f(x^* + t_1 d)^T (t_2 - t_1)d + \mu \nabla \Phi(x^* + t_1 d)(t_2 - t_1)d > 0.$$

For this pair t_1 and t_2, we can derive

$$\begin{aligned} & f(x^* + t_2 d) + \mu\Phi(x^* + t_2 d) - (f(x^* + t_1 d) + \mu\Phi(x^* + t_1 d)) \\ \geq & \nabla f(x^* + t_1 d)^T (t_2 - t_1)d + \mu \nabla \Phi(x^* + t_1 d)(t_2 - t_1)d \\ > & 0 \end{aligned}$$

which is contradicting to (3.1). So that (P_μ) is bounded. Since (P_μ) is strict convex, the unique solution $x(\mu)$ exists. The above proof holds for all $\mu > 0$. □

Theorem 3.2. *The path $\{x(\mu) : \mu \in (0, \mu_0]\}$ is bounded, and the upper bound is dependent on μ_0 and $\varphi(x)$.*

Proof. The KKT condition of (P_μ) is as follows

$$\begin{cases} \nabla f(x) + \mu \nabla \Phi(x) - A^T y - z = 0, \\ Ax = b, \\ x^T z = 0, \\ x \geq 0, z \geq 0. \end{cases}$$

The unique existence of $x(\mu)$ is showed in Theorem 3.1. The barrier functions guarantee $x(\mu) > 0$. The above system is simplified as

$$\begin{cases} \nabla f(x) + \mu \nabla \Phi(x) - A^T y = 0, \\ Ax = b, \\ x \geq 0. \end{cases} \tag{3.4}$$

Denote $(x(\mu), y(\mu))$ a solution of the above system. Simply use x and x_0 to represent $x(\mu)$ and $x(\mu_0)$ for the following deduction. The same for y and y_0. We can derive the followings

$$
\begin{aligned}
&(x-x_0)^T(-\mu\nabla\Phi(x)+\mu_0\nabla\Phi(x_0))\\
=&(x-x_0)^T(\nabla f(x)-A^Ty-\nabla f(x_0)+A^Ty_0)\\
=&(x-x_0)^TA^T(y_0-y)+(x-x_0)^T(\nabla f(x)-\nabla f(x_0))\\
=&(x-x_0)^T(\nabla f(x)-\nabla f(x_0))\\
\geq&0.
\end{aligned}
$$

The fist equality is due to system (3.4). The third equality is because of the constraint $Ax = b$. The last inequality is due to the convexity of $f(x)$. Expanding the first formula, we can know that

$$
-\mu x^T\nabla\Phi(x)+\mu_0x^T\nabla\Phi(x_0)+\mu x_0^T\nabla\Phi(x)-\mu_0x^T\nabla\Phi(x_0)\geq 0. \tag{3.5}
$$

By the property of $\Phi(x)$, we know that $-\mu x^T\nabla\Phi(x)$ and $-\mu_0x^T\nabla\Phi(x_0)$ are positive, and $\mu_0x^T\nabla\Phi(x_0)$ and $\mu x_0^T\nabla\Phi(x)$ are negative. Denote the largest element of x as $\hat{x}$, we know if $\hat{x}$ goes to $+\infty$, the followings hold

$$
\begin{aligned}
-\mu x^T\nabla\Phi(x)&=O(\hat{x}^\gamma),\\
\mu_0x^T\nabla\Phi(x_0)&=\Omega(\hat{x}).
\end{aligned}
$$

Since $\gamma<1$, the increasing speed of $-\mu x^T\nabla\Phi(x)$ will be slower than the decreasing speed of $\mu_0x^T\nabla\Phi(x_0)$. The right side of (3.5) will go to $-\infty$ as $\hat{x}$ goes to $+\infty$, which is contradicting to (3.5). So that x is bounded above, the upper bound is dependent on (3.5), or μ_0 and $\varphi(x)$. □

Theorem 3.3. *The path $\{x(\mu):\mu\in(0,\mu_0]\}$ is continuous.*

Proof. A proof for linear programming case is proposed in [72] with the assumption that the feasible region is bounded. We make some modification and generalize it for LCCP case here. Assume $x(\mu)$ is not continuous at $\overline{\mu}>0$, then at least one side (left side or right side) continuity can not hold at $\overline{\mu}$. Suppose the right side continuity does not hold, which means there exists an $\varepsilon > 0$ such that for all $\Delta\mu > 0$,

$$\|x(\overline{\mu} + \Delta\mu) - x(\overline{\mu})\| > \varepsilon.$$

Since $x(\mu)$ is the unique optimal solution for (P_μ), and $f(x) + \overline{\mu}\Phi(x)$ is a continuous function, we obtain that there exists a $\tau > 0$ such that

$$f(x(\overline{\mu})) + \overline{\mu}\Phi(x(\overline{\mu})) + \tau < f(x(\overline{\mu} + \Delta\mu)) + \overline{\mu}\Phi(x(\overline{\mu} + \Delta\mu)), \forall \Delta\mu > 0. \tag{3.6}$$

Because $x(\mu)$ is bounded in $(0, \mu_0]$ and $\Phi(x)$ is continuous, $|\Phi(x(\mu))|$ is bounded. Therefore there exists a $\Delta\overline{\mu} > 0$ that

$$0 < |\Delta\overline{\mu}||\Phi(x(\overline{\mu} + \Delta\overline{\mu}))| < \frac{\tau}{2} \quad \text{and} \quad 0 < |\Delta\overline{\mu}||\Phi(x(\overline{\mu}))| < \frac{\tau}{2} \tag{3.7}$$

Then, we can derive

$$\begin{aligned} f(x(\overline{\mu})) + (\overline{\mu} + \Delta\overline{\mu})\Phi(x(\overline{\mu})) &< f(x(\overline{\mu})) + \overline{\mu}\Phi(x(\overline{\mu})) + \frac{\tau}{2} \\ &< f(x(\overline{\mu} + \Delta\overline{\mu})) + \overline{\mu}\Phi(x(\overline{\mu} + \Delta\overline{\mu})) - \frac{\tau}{2} \\ &< f(x(\overline{\mu} + \Delta\overline{\mu})) + (\overline{\mu} + \Delta\overline{\mu})\Phi(x(\overline{\mu} + \Delta\overline{\mu})). \end{aligned}$$

The first and third inequalities is based on (3.7). The second inequality is due to (3.6). The above result contradicts that $x(\overline{\mu} + \Delta\overline{\mu})$ is the optimal solution for (P_μ) when $\mu = \overline{\mu} + \Delta\overline{\mu}$, so that the right continuity of $x(\mu)$ at $\overline{\mu}$ is hold.

The same discussion can be conducted for the left continuity when $\Delta\mu < 0$. □

Lemma 3.1. *Denote $V(\mu) = f(x(\mu))$, then*

$$\lim_{\mu \to 0} V(\mu) = f^*,$$

where f^* is the optimal value of (P).

Proof. Let $\mu_1 > \mu_2 > 0$, by the definition of $x(\mu)$, we obtain

$$f(x(\mu_1)) + \mu_1\Phi(x(\mu_1)) < f(x(\mu_2)) + \mu_1\Phi(x(\mu_2)), \tag{3.8}$$

$$f(x(\mu_2)) + \mu_2\Phi(x(\mu_2)) < f(x(\mu_1)) + \mu_2\Phi(x(\mu_1)). \tag{3.9}$$

Adding up (3.8) and (3.9) and making simplification, we obtain

$$\Phi(x(\mu_1)) < \Phi(x(\mu_2)). \tag{3.10}$$

Compare (3.9) and (3.10), we derive that

$$f(x(\mu_1)) > f(x(\mu_2)),$$

which means $V(\mu)$ is strict decreasing as μ decreases. Moreover, $f(x(\mu)) \geq f^*$, so that there is a $\overline{f}$ such that

$$\lim_{\mu \to 0^+} f(x(\mu)) = \overline{f}.$$

Next, we prove $\overline{f} = f^*$. Assume that $\overline{f} > f^*$, then there exists a small enough $\varepsilon > 0$ that $f^* + 2\varepsilon < \overline{f}$. By (3.10) we know $\Phi(x(\mu))$ is monotonic increasing as μ goes to 0. We discuss the following two cases.

- Case 1: if $|\Phi(x(\mu))|$ is upper bounded by $M > 0$ as μ goes from μ_0 to 0.

  Let x^* be an optimal solution for (P), due to the continuity, there exists a Δx that $f(x^* + \Delta x) < f^* + \varepsilon$ and $x^* + \Delta x$ is feasible. We can carefully choose Δx makes $\Phi(x^* + \Delta x) \neq 0$. Let $\overline{\mu} = \min\{\frac{\varepsilon}{M}, \frac{\varepsilon}{|\Phi(x^*+\Delta x)|}, \mu_0\}$, for this $\overline{\mu}$, we obtain

  $$\overline{\mu}|\Phi(x(\overline{\mu}))| < \varepsilon, \overline{\mu}\Phi(x^* + \Delta x) < \varepsilon. \tag{3.11}$$

  By the optimality of $x(\overline{\mu})$ and the setting of Δx, the following holds.

  $$\begin{aligned} f(x(\overline{\mu})) + \overline{\mu}\Phi(x(\overline{\mu})) &\leq f(x^* + \Delta x) + \overline{\mu}\Phi(x^* + \Delta x) \\ &< f^* + \varepsilon + \varepsilon = \overline{f}. \end{aligned} \tag{3.12}$$

  Together with (3.11), we derive $f(x(\overline{\mu})) < \overline{f}$. It is a contradiction.

- Case 2: if $\Phi(x(\mu))$ is unbounded μ goes from μ_0 to 0..

  Choose a Δx in the same way with case 1. Since $\Phi(x(\mu))$ is increasing as μ goes to 0, there is $\mu_1 > 0$ makes $\Phi(x(\mu_1)) > 0$. Let $\mu_2 = \frac{\varepsilon}{|\Phi(x^*+\Delta x)|}, \overline{\mu} = \min\{\mu_1, \mu_2\}$, for this $\overline{\mu}$ we have

  $$\overline{\mu}\Phi(\overline{\mu}) > 0, \overline{\mu}\Phi(x^* + \Delta x) < \epsilon.$$

In that case, (3.12) still holds. So that $f(\overline{\mu}) < \overline{f}$, a contradiction. The above discussion leads to $\lim_{\mu \to 0^+} V(\mu) = f^*$.

The above contradiction completes the proof. □

Theorem 3.4. *$x(\mu)$ will converge to an optimal solution of (P) as μ tends to zero. Moreover, this optimal solution is the unique solution of the following problem*

$$\begin{aligned} \min \quad & \Phi_B(x) \\ \text{s.t.} \quad & Ax = b, \\ & f(x) \le f^*, \\ & (x)_i = 0, \forall i \in N, \\ & (x)_j > 0, \forall j \notin N. \end{aligned} \tag{$\widetilde{P}$}$$

Where $N = \{i : (x)_i = 0, \forall x \text{ that } f(x) = f^*\}, \Phi_B(x) = \Phi(x) - \sum_{i \in N} \varphi((x)_i)$.

Proof. The proof is an extension of Megiddo's work [52]. $x(\mu)$ and $V(\mu)$ are the optimal solution and value of (P_μ) respectively also means $x(\mu)$ is the unique solution of

$$\begin{aligned} \min \quad & f(x) + \mu\Phi(x) \\ \text{s.t.} \quad & Ax = b, \\ & f(x) \le V(\mu), \\ & x \ge 0. \end{aligned} \tag{P_1}$$

(P_1) is also equal to

$$\begin{aligned} \min \quad & \Phi(x) \\ \text{s.t.} \quad & Ax = b, \\ & f(x) \le V(\mu), \\ & x \ge 0. \end{aligned} \tag{P_2}$$

(P_2) is equal to

$$
\begin{aligned}
\min \quad & \Phi_B(x) \\
\text{s.t.} \quad & Ax = b, \\
& f(x) \le V(\mu), \\
& (x)_i = (x(\mu))_i \ge 0, i \in N, \\
& (x)_j > 0, j \notin N.
\end{aligned} \tag{P_3}
$$

$x(\mu)$ satisfies the KKT condition of (P_3) that

$$
\begin{cases}
\varphi'((x)_j) - A_j^T y + \lambda(\nabla f(x))_j = 0, j \notin N \\
Ax = b, \\
f(x) \le V(\mu), \\
(x)_i = (x(\mu))_i, i \in N, \\
(x)_j > 0, j \notin N, \\
\lambda \ge 0.
\end{cases}
$$

Where A_j^T is the j-th row of A^T. Since $x(\mu)$ is a bounded continuous path at $(0, \mu_0]$, there exists a $\widetilde{x}$ that $\lim_{\mu \to 0^+} x(\mu) = \widetilde{x}$. Assume there is a decreasing positive series $\{\mu_k\}$ that $\lim_{k\to\infty} \mu_k = 0$, then $\lim_{k\to\infty} x(\mu_k) = \widetilde{x}$. Firstly, $(\widetilde{x})_j > 0, \forall j \in N$, or the optimal value of (P_3) will go to $+\infty$ which will not happened. Then, by the continuity of this system, we know $\widetilde{x}$ satisfies

$$
\begin{cases}
\varphi'((\widetilde{x})_j) - A_j^T \widetilde{y} + \lambda(\nabla f(\widetilde{x}))_j = 0, j \notin N \\
A\widetilde{x} = b, \\
f(\widetilde{x}) \le f^*, \\
(\widetilde{x})_i = (\widetilde{x})_i \ge 0, i \in N, \\
(\widetilde{x})_j > 0, j \notin N, \\
\lambda \ge 0.
\end{cases} \tag{3.13}
$$

Where f^* comes from $\lim_{\mu_k \to 0} V(\mu_k) = f^*$ as mentioned in Lemma 1. Moreover, from

(3.13), $\widetilde{x}$ is an optimal solution of (P), so that $(\widetilde{x})_i = 0, \forall i \in N$. So that $\widetilde{x}$ satisfies

$$\begin{cases} \varphi'((\widetilde{x})_j) - A_j^T \widetilde{y} + \lambda(\nabla f(\widetilde{x}))_j = 0, j \notin N \\ A\widetilde{x} = b, \\ f(\widetilde{x}) \leq f^*, \\ (\widetilde{x})_i = 0, i \in N, \\ (\widetilde{x})_j > 0, j \notin N, \\ \lambda \geq 0. \end{cases}$$

Which is the KKT condition of $(\widetilde{P})$. So that the limiting point of $x(\mu)$ as μ goes to zero is the unique optimal solution of $(\widetilde{P})$. □

3.2 Path Following Algorithm for General Barrier Functions

In this part, the **PPFA** introduced in Chapter 2 is applied to the paths generated from different barrier functions.

3.2.1 Construction of Path Following Algorithm

From above analysis, we know the solution $x(\mu)$ for (P_μ) is positive and satisfies the KKT condition (3.4). Denote $s = -\mu\nabla\Phi(x)$ and $d\Phi(x) = \text{diag}(\varphi'((x)_i))$, together with that $\varphi(x)' < 0, \forall x \in (0, +\infty)$, the KKT condition (3.4) is transferred to the form

$$\begin{cases} A^T y + s - \nabla f(x) = 0, \\ Ax = b, \\ -d\Phi(x)^{-1} s = \mu e, \\ x > 0, s > 0. \end{cases} \tag{3.14}$$

Denote $u = (x^T, y^T, s^T)^T \in \mathcal{R}^n_+ \times R^m \times \mathcal{R}^n_+$, define the function $H_\mu(u)$

$$H_\mu(u) = \begin{pmatrix} A^T y + s - \nabla f(x) \\ Ax - b \\ -d\Phi(x)^{-1} s - \mu e \end{pmatrix}.$$

So that $H_\mu(u) = 0$ characterize the path we are following. Also, $H_0(u) = 0$ is corresponding to the KKT condition of (P). The Jacobian matrix of $H_\mu(u)$ is

$$J(u) = \begin{pmatrix} -\nabla^2 f(x) & A^T & I \\ A & 0 & 0 \\ d\Phi(x)^{-2}\nabla^2\Phi(x)S & 0 & -d\Phi(x)^{-1} \end{pmatrix}.$$

Newton's direction Δu at u with continuation parameter μ is the solution of the linear system

$$J(u)\Delta u = -H_\mu(u).$$

$\Delta u = (\Delta x^T, \Delta y^T, \Delta s^T)^T \in \mathcal{R}^n \times R^m \times \mathcal{R}^n$ is used in our path following algorithm. More specifically, we derive the following expression of $\Delta x, \Delta y$ and Δs

$$\begin{cases} \Delta y &= (AD^{-1}A^T)^{-1}(AD^{-1}(-d\Phi(x)r_c - r_d) + r_p), \\ \Delta x &= D^{-1}(-A^T\Delta y - d\Phi(x)r_c - r_d), \\ \Delta s &= d\Phi(x)r_c + d\Phi(x)^{-1}\nabla^2\Phi(x)S\Delta x. \end{cases}$$

Where $D = d\Phi(x)^{-1}\nabla^2\Phi(x)S - \nabla^2 f(x)$, $r_d = A^T y + s - \nabla f(x), r_p = Ax - b, r_c = -d\Phi(x)^{-1}s - \mu e$. Substitute the new $H_\mu(u)$ and $J(u)$ defined here to the **PPFA** framework introduced in Chapter 2, **PPFA** still works with convergence.

3.2.2 Implementation of the Path Following Algorithm

Denote the sequence generated form **PPFA** as $\{u_k\}$. The following theorem illustrates the boundedness of $\{u_k\}$.

Theorem 3.5. *The iteration sequence* $\{u_k\}$ *generated by PPFA is bounded. More specifically, there exists positive* x_l, x_u, s_l *and* s_u *which depends on* $n, \theta, x_0, s_0, \mu_0$, $\varphi(x), A$ *and* f *such that*

$$0 < x_l \leq (x_k)_i \leq x_u, 0 < s_l \leq (s_k)_i \leq s_u, \forall i = 1, .., n. \tag{3.15}$$

Moreover, y_k *is bounded.*

Proof. Denote $r_k = A^T y_k + s_k - \nabla f(x_k)$. Following the same proof of Theorem 2.2, (2.10) still holds. We restate it as follows

$$\begin{cases} (x_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0 e)+x_k^T s_k}{(s_0)_i - 2\theta\mu_0} \\ (s_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0 e)+x_k^T s_k}{(x_0)_i} \end{cases}. \tag{3.16}$$

If the upper bound of $x_k^T s_k$ is known, so is the upper bound of $(x_k)_i$. We discuss the upper bound of $x_k^T s_k$ in the following. By (2.7) we obtain

$$\frac{(s_k)_i}{|\varphi'((x_k)_i)|} < (1+\theta)\mu_k \leq (1+\theta)\mu_0.$$

We derive

$$\begin{aligned} (x_k)_i(s_k)_i &= \frac{(s_k)_i}{|\varphi'((x_k)_i)|}|\varphi'((x_k)_i)|(x_k)_i \\ &\leq (1+\theta)\mu_0|(x_k)_i\varphi'((x_k)_i)| \\ &= O((x_k)_i^\gamma) \\ &= O((x_k^T s_k)^\gamma) \end{aligned}$$

The second equlity is because $\varphi'(x) = O(x^{\gamma-1})$. The last equality is due to (3.16). Then we obtain

$$x_k^T s_k = nO((x_k^T s_k)^\gamma) \Rightarrow x_k^T s_k n^{-\frac{1}{1-\gamma}} = O((x_k^T s_k n^{-\frac{1}{1-\gamma}})^\gamma). \tag{3.17}$$

Since $\gamma < 1$, $x_k^T s_k n^{-\frac{1}{1-\gamma}}$ will not go to infinity, or (3.17) will not hold. So that there is a $M > 0$ satisfying

$$x_k^T s_k \leq M n^{\frac{1}{1-\gamma}}. \tag{3.18}$$

M is dependent on $x_0, s_0, \mu_0, \gamma, \theta$. Substitute (3.18) into (3.16), we derive

$$\begin{cases} (x_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0)+Mn^{\frac{1}{1-\gamma}}}{(s_0)_i-2\theta\mu_0} = x_u, \\ (s_k)_i & \leq \frac{x_0^T(s_0+2\theta\mu_0)+Mn^{\frac{1}{1-\gamma}}}{(x_0)_i} = s_u. \end{cases}$$

Together with the lower bound of $\frac{(s_k)_i}{|\varphi'((x_k)_i)|}$ derived from (2.7) as

$$\frac{(s_k)_i}{|\varphi'((x_k)_i)|} \geq (1-\theta)\mu_k \geq (1-\theta)\epsilon.$$

Since $\varphi(x)$ is strict convex and $\varphi'(x) < 0$ in $(0, +\infty)$, we derive

$$\begin{cases} (x_k)_i & \geq (\varphi')^{-1}(-\frac{s_u}{(1-\theta)\epsilon}) = x_l, \\ (s_k)_i & \geq -(1-\theta)\epsilon\varphi(x_u) = s_l, \end{cases}$$

which surely imply $(x_k)_i > 0, (s_k)_i > 0$. From the above analysis, we know that x_u, x_l, s_l and s_u are existed, and (3.15) holds. Conduct y_k in the form

$$\begin{aligned} \|y_k\|_\infty &= \|(AA^T)^{-1}A(\nabla f(x_k) - s_k + r_k)\|_\infty \\ &\leq \|(AA^T)^{-1}A\|_\infty(\|\nabla f(x_k) - s_k\|_\infty + \|r_k\|_\infty) \\ &\leq \|(AA^T)^{-1}A\|_\infty(\|\nabla f(x_k) - s_k\|_\infty + \theta\mu_0), \end{aligned}$$

which is bounded. □

Theorem 3.5 is a little bit different with Theorem 2.2 since explicit expressions of the upper bound and the lower bound are hard to derive for different barrier functions.

Theorem 3.6. *Under the Assumption 2 that $\nabla^2 f(x)$ is locally Lipschitz continuous for $x > 0$ with Lipschitz constant L, $J(u)$ is locally Lipschitz continuous on $\{u_k\}$ with an $L_\epsilon > 0$ such that*

$$\|J(v) - J(w)\|_\infty \leq L_\epsilon\|v - w\|_\infty \quad \forall v, w \in \{u_k\}.$$

Proof. In order to simplify the expressions, we denote the vectors $v, w \in \{u_k\}$ as $v = (x_v^T, y_v^T, s_v^T)^T, w = (x_w^T, y_w^T, s_w^T)^T$ and

$$\begin{aligned}
X_v &= \mathrm{diag}(x_v), S_v = \mathrm{diag}(s_v), X_w = \mathrm{diag}(x_w), S_w = \mathrm{diag}(s_w),\\
G &= -\nabla^2 f(x_v) + \nabla^2 f(x_w),\\
E &= d\Phi(x_v)^{-2}\nabla^2\Phi(x_w)S_v - d\Phi(x_w)^{-2}\nabla^2\Phi(x_w)S_w,\\
F &= -d\Phi(x_v)^{-1} + d\Phi(x_w)^{-1}.
\end{aligned}$$

Here E and F are diagonal matrices with positive diagonal elements and G is a symmetric matrix. Then we derive

$$\|J(v) - J(w)\|_\infty = \| \begin{pmatrix} G & 0 & 0 \\ 0 & 0 & 0 \\ E & 0 & F \end{pmatrix} \|_\infty \le \max\{\|G\|_\infty, \|E\|_\infty + \|F\|_\infty\}. \tag{3.19}$$

Under the assumption $\nabla^2 f(x)$ is Lipschitz continuous, the following holds

$$\|G\|_\infty \le L\|v - w\|_\infty. \tag{3.20}$$

Next, we consider the gradients of function h_1 and h_2 given by $h_1(x, s) = \frac{\varphi''(x)s}{(\varphi'(x))^2}$ and $h_2(x) = -\frac{1}{\varphi'(x)}$. Denote $U_x = [x_l, x_u]$ and $U_s = [s_l, s_u]$. Since $h_1(x, s)$ is continuous differentiable on $U_x \times U_s$ and $h_2(x)$ is continuous differentiable on U_x, $\|\nabla h_1(x, s)\|_\infty$ and $\|\nabla h_2(x)\|_\infty$ are bounded. There is an upper bound $\widetilde{L}$ for $\|\nabla h_1(x, s)\|_\infty$ and $\|\nabla h_2(x, s)\|_\infty$ that

$$\|\nabla h_1(x, s)\|_\infty + \|\nabla h_2(x)\|_\infty \le \widetilde{L}, \quad \forall x \in U_x, \forall s \in U_s.$$

Considering the expression of E and F, by the mean value theorem,

$$\|E\|_\infty + \|F\|_\infty \le \widetilde{L}\|u_v - u_w\|_\infty. \tag{3.21}$$

Together with (3.19), (3.20) and (3.21), we derive $L_\epsilon = L + \widetilde{L}$ such that

$$\|J(v) - J(w)\|_\infty \le \|G\|_\infty + \|E\|_\infty + \|F\|_\infty \le L_\epsilon\|v - w\|, \quad \forall v, w \in \{u_k\},$$

which proves that $J(v)$ is local Lipschits continuous on $\{u_k\}$. □

With the L_ϵ defined above, Theorem 2.4 holds. Which tells that **PPFA** is well established for this class of barrier functions. The proof is the same except that $\widetilde{C}, \widetilde{D}$, and D appeared in the proof that should be replaced by the following denotations.

$$\begin{aligned}
\widetilde{C} &= \begin{pmatrix} d\Phi(x)^{-2}\nabla^2\Phi(x)S & 0 \end{pmatrix}, \\
\widetilde{D} &= -d\Phi(x)^{-1}, \\
D &= -\nabla^2 f(x) + d\Phi(x)^{-1}\nabla^2\Phi(x)S.
\end{aligned}$$

3.2.3 Convergence of the Path Following Algorithm

In this part, we discuss the convergence property of the **PPFA**. First, we derive the upper bound of $\|J(u)^{-1}\|_\infty$ on the iteration sequence $\{u_k\}$, then apply it to obtain the convergence property.

Theorem 3.7. *There exists a $K_\epsilon > 0$ such that*

$$\|J(u)^{-1}\|_\infty \leq K_\epsilon, \quad \forall u \in \{u_k\}.$$

$\|J(u)^{-1}\|_\infty$ is a continuous function for u, and u_k is in a bounded close set, so that K_ϵ is existed. Next, we illustrate the convergence property.

Theorem 3.8. *Let (u_k, μ_k) be the sequence generated by the* **PPFA**. *Then*

(a) for all $k \geq 1$

$$\begin{cases} Ax_k - b = 0, \\ \|H_{\mu_k}(u_k)\|_\infty \leq \theta\mu_k, \\ (1-\sigma_{k-1})\cdots(1-\sigma_0)\mu_0 = \mu_k. \end{cases}$$

(b) for all $k \geq 1$

$$\sigma_k \geq \frac{\beta\theta p\eta}{\theta+1},$$

where

$$\eta = min\{1, \frac{\alpha}{L_\epsilon K_\epsilon^2\theta\mu_0}\}.$$

(c) after at most $\lceil \frac{ln\epsilon - ln\mu_0}{ln(1-\frac{\beta\theta p\eta}{\theta+1})} \rceil$ *iterations, the tolerance* ϵ *is satisfied.*

Theorem 3.8 is the same with Theorem 2.6, except that Theorem 3.8(c) does not give a polynomial complexity expression for the iteration number. Use the L_ϵ and K_ϵ defined in this chapter, the proof of Theorem 2.6 can be directly applied for Theorem 3.8.

3.3 Initialization of the Path Following Algorithm

The process of **PPFA** requires an initial point, which should be close to the path and satisfy the initial setting of **PPFA**. Using the same technique illustrated in Chapter 2.3.1, we construct an augmented problem here. Our work is mainly due to the initialization part of [36, 52, 58, 59, 71].

The augmented problems for different barrier functions are of the same form as that presented in 2.3.1, the problem (P') and (D'). However, the parameter settings are different for different barrier functions. For the barrier function $\varphi(x)$, K_b and K_c appeared in (P') and (D') should be changed to the following forms.

$$K_b = (n+1)\lambda\tau - \lambda\nabla f(\lambda e)^T e, \quad K_c = \frac{\varphi'(1)\tau}{\varphi'(\lambda)},$$

where λ and τ are given positive constants. In this case, a good initial point for (P') and (D') is as follows.

$$x_0 = \begin{pmatrix} \lambda e \\ 1 \\ \lambda \end{pmatrix}, y_0 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, s_0 = \begin{pmatrix} \tau e \\ \frac{\varphi'(1)\tau}{\varphi'(\lambda)} \\ \tau \end{pmatrix}.$$

The corresponded μ_0 is $-\frac{\tau}{\varphi'(\lambda)}$ which can be calculated from the definition of $H_\mu(u)$. For this augmented problem, Theorem 2.7 still holds with the same proof. Theorem 2.7 guarantees the feasibility of generating the optimal solution of (P) from the solution of the augmented problem, under the condition of (2.43).

In practice, x^* and y^* are unknown, (2.43) can be expressed

$$\begin{cases} \tau((n+1)\lambda - e^T x^*) + \nabla f(\lambda e)^T (x^* - \lambda e) > 0, \\ \hfill -\dfrac{\tau}{\varphi'(\lambda)} - (b - \lambda A e)^T y^* > 0. \end{cases} \tag{3.22}$$

In the selection of τ and λ such that (3.22) holds, we can consider the following procedure: (a) λ is chosen first to satisfy $(n+1)\lambda - e^T x^* > 0$; and (b) τ can be chosen large enough to ensure that both inequalities are satisfied. In real implementation, a pair of τ and λ can be selected first. Then the augmented problem is solved. If (3.22) is satisfied, stop; otherwise repeat with a larger pair of τ and λ.

The implementation procedure of the path following algorithm is the same with it stated in Chapter 2.3.2.

Chapter 4

Nonnegative Matrix Factorization

The nonnegative matrix factorization (NMF) (1.2) plays an essential role in many application fields. Hierarchical alternating least squares algorithm (**HALS**) and Fast hierarchical alternating least square algorithm (**Fast HALS**) are two efficient methods for NMF. However, as far as we know, there is no precise analysis for the convergence of these two algorithms. In this chapter, we study the convergence of **HALS** and **Fast HALS**.

4.1 Hierarchical Alternating Least Squares Algorithm

In this part, we will introduce the idea and the derivation of **HALS** and **Fast HALS**.

Denoting the columns of A as $[a_1, a_2, \cdots, a_J]$ and B as $[b_1, b_2, \cdots, b_J]$, $f(A, B) = \|Y - AB^T\|^2$ has the following equivalent expressions

$$f(A, B) = \|Y - \sum_{i=1}^{J} a_i b_i^T\|^2 = \|Y^j - a_j b_j^T\|^2 = a_j^T a_j b_j^T b_j - 2a_j^T Y^j b_j + \|Y^j\|^2, \quad (4.1)$$

where $Y^j = Y - \sum_{i \neq j} a_i b_i^T$. $f(A, B)$ is a convex quadratic function of a_j or b_j while the other columns of A and B are fixed. Fixing all the other columns except a_j, the optimal value for a_j under nonnegative constraint is

$$\frac{[Y^j b_j]_+}{b_j^T b_j} = \arg\min_{a_j \geq 0} \|Y^j - a_j b_j^T\|^2.$$

In the same way, fixing all the other columns except b_j, the optimal solution for b_j

under nonnegative constraint is

$$\frac{[(Y^j)^T a_j]_+}{a_j^T a_j} = \arg\min_{b_j \geq 0} \|Y^j - a_j b_j^T\|^2.$$

Since all the columns of A and B can be regarded as different blocks, it is an natural idea to apply the coordinate descend method to update A and B column by column based on the above explicit optimal solution. Moreover, to enhence the stability, an normalization for a_j or b_j can be applied. With the following updating order,

$$b_1 \to a_1 \to b_2 \to a_2 \to \cdots \to b_J \to a_J,$$

the **HALS** algorithm is presented below

HALS (Algorithm 1 in [15])

Initialize nonnegative matrix A and B using ALS [15];
Normalize the vectors a_j to unit ℓ_2-norm length;
$E = Y - AB^T$;
repeat
  for $j = 1$ to J **do**
    $Y^j = E + a_j b_j^T$;
    $b_j = [(Y^j)^T a_j]_+$;
    $a_j = [Y^j b_j]_+$;
    $a_j = a_j / \|a_j\|_2$;
    $E = Y^j - a_j b_j^T$;
  end for
until convergence criterion is reached

The initial A and B can be derived by alternating least square (**ALS**) method which is illustrated in [15]. Since $\|a_j\| = 1$, the updated b_j is an nonnegative optimal solution for $\min_{x \geq 0} \|Y^j - a_j x^T\|^2$. However, the normalization process makes the updated a_j is not an exactly nonnegative optimal solution for $\min_{x \geq 0} \|Y^j - x b_j^T\|^2$.

Moreover, the update of b_j can expressed as follows

$$\begin{aligned}\frac{[(Y^j)^T a_j]_+}{a_j^T a_j} &= \frac{[(Y - AB^T + a_j b_j^T)^T a_j]_+}{a_j^T a_j} \\ &= \frac{[Y^T a_j - BA^T a_j + b_j a_j^T a_j]_+}{a_j^T a_j} \\ &= [b_j + Y^T a_j - BA^T a_j]_+.\end{aligned}$$

In a similar way, line 8 in **HALS** for updating a_j can be simplified as

$$[Y^j b_j]_+ = [a_j b_j^T b_j + Y b_j - AB^T b_j]_+$$

The above $Y^T a_j, BA^T a_j, Y b_j$, and $AB^T b_j$ are the j-th column of $Y^T A, BA^T A, YB$, and $AB^T B$, respectively. One way to save the cost is calculating these matrices first, and then using certain parts of them for updating. For better use of these matrices, the updating order for columns is changed to be

$$b_1 \to b_2 \to \cdots \to b_J \to a_1 \to a_2 \cdots \to a_J.$$

The **Fast HALS** algorithm is as follows

Fast HALS (Algorithm 2 in [15])

Initialize nonnegative matrix A and B using ALS [15];
Normalize the vectors a_j to unit ℓ_2-norm length;
repeat
  $W = Y^T A$;
  $V = A^T A$;
  for $j = 1$ to J **do**
    $b_j = [b_j + w_j - Bv_j]_+$;
  end for
  $P = YB$;
  $Q = B^T B$;
  for $j = 1$ to J **do**
    $a_j = [a_j q_{jj} + p_j - Aq_j]_+$;
    $a_j = a_j / \|a_j\|_2$;
  end for
until convergence criterion is reached

The calculation amount for **Fast HALS** is greatly decreased while the convergence performance is similar with **HALS**.

4.2 Convergence for Hierarchical Alternating Least Squares Algorithm

Denote the iteration sequence of **HALS** as $\{(A_k, B_k)\}$. We review the updating process of **HALS** in the following way. At the k-th iteration, (A_k, B_k) is updated to (A_{k+1}, B_{k+1}) that the i_k-th columns of A_k and B_k, denoted as a_{i_k} and b_{i_k}, are updated to $\hat{a}_{i_k}$ and $\hat{b}_{i_k}$ in the following expressions.

$$\begin{cases} b_{i_k} \to \hat{b}_{i_k} = [(Y^{i_k})^T a_{i_k}]_+, \\ a_{i_k} \to \hat{a}_{i_k} = \frac{[Y^{i_k}\hat{b}_{i_k}]_+}{\|[Y^{i_k}\hat{b}_{i_k}]_+\|}. \end{cases} \tag{4.2}$$

i_k is ranging from 1 to J for different iteration k. For this denotation, we know $\hat{a}_{i_k} = a_{i_{k+J}}$. However, **HALS** will come across a problem if $[Y^j\hat{b}_{i_k}]_+ = 0$ occurs since the normalization to derive $\hat{a}_{i_k}$ can not proceed.

Proposition 4.1. *The case* $[Y^{i_k}\hat{b}_{i_k}]_+ = 0$ *occurs if and only if* $\hat{b}_{i_k} = 0$.

Proof. Surely, $\hat{b}_{i_k} = 0$ will lead to $[Y^{i_k}\hat{b}_{i_k}]_+ = 0$. Next we prove that if $\hat{b}_{i_k} \neq 0$, then $[Y^j\hat{b}_{i_k}]_+ \neq 0$. Suppose there is a $\hat{b}_{i_k} \neq 0$ making $[Y^j\hat{b}_{i_k}]_+ = 0$. Denote

$$g(x) = \|Y^{i_k} - x\hat{b}_{i_k}^T\|^2,$$

then we know

$$0 = \frac{[Y^{i_k}\hat{b}_{i_k}]_+}{\hat{b}_{i_k}^T\hat{b}_{i_k}} = \arg\min_{x\geq 0} g(x), \text{ and } \min_{x\geq 0} g(x) = \|Y^{i_k}\|^2.$$

However,

$$\begin{aligned} g(a_{i_k}) &= \|Y^{i_k} - a_{i_k}\hat{b}_{i_k}^T\|^2 \\ &= a_{i_k}^T a_{i_k}\hat{b}_{i_k}^T\hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k}\hat{b}_{i_k} + \|Y^{i_k}\|^2 \\ &= -\|\hat{b}_{i_k}\|^2 + \|Y^{i_k}\|^2 \\ &< \|Y^{i_k}\|_F^2. \end{aligned}$$

The third equality is derived by $\|a_{i_k}\| = 1$ and the expression of $\hat{b}_{i_k}$ (4.2). Therefore, $g(a_{i_k}) < \min_{x\geq 0} g(x)$, it is a contradiction. So that if $\hat{b}_{i_k} \neq 0$ then $[Y^j\hat{b}_{i_k}]_+ \neq 0$. The proof is completed. □

Proposition 4.1 tells that the invalid implementation of **HALS** happens if $\hat{b}_{i_k} = 0$ happens. To overcome this case, we propose an approach that

If $\hat{b}_{i_k} = 0$, skip updating of a_{i_k}.

Which is achieved by replacing line 8 and 9 in **HALS** with the following steps.

Adjusted Steps for HALS

```
1: if b_j ≠ 0
2:     a_j = [Y^j b_j]_+;
3:     a_j = a_j/||a_j||_2;
4: end if
```

We call **HALS** with this adjustment as **Adjusted HALS**. This adjustment will enrich the implementation and will not affect the reduction of objective value.

Lemma 4.1. *At the k-th iteration, a_{i_k} and b_{i_k} are updated to $\hat{a}_{i_k}$ and $\hat{b}_{i_k}$ with expressions* (4.2)*, then the following relations hold.*

$$a_{i_k}^T Y^{i_k} b_{i_k} \leq \hat{b}_{i_k}^T b_{i_k}, \quad a_{i_k}^T Y^{i_k} \hat{b}_{i_k} = \|\hat{b}_{i_k}\|^2 \leq \|[Y^{i_k}\hat{b}_{i_k}]_+\| a_{i_k}^T \hat{a}_{i_k}. \tag{4.3}$$

Theorem 4.1. *Assume that $\hat{b}_{i_k} \neq 0$ for every iteration, with a given start point (A_1, B_1), the sequence $\{f_k\}$ is decreasing to a limiting value f^*. And for every step*

$$f_k - f_{k+1} \geq \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\| \|a_{i_k} - \hat{a}_{i_k}^T\|^2 \geq 0.$$

Proof. The reduction of objective value at the k-th iteration is calculated as follows.

$$\begin{aligned}
& f_k - f_{k+1} \\
=& \|Y - A_k B_k^T\|^2 - \|Y - A_{k+1}B_{k+1}^T\|^2 \\
=& (\|Y - A_k B_k^T\|^2 - \|Y - A_k B_{k+1}^T\|^2) + (\|Y - A_k B_{k+1}^T\|^2 - \|Y - A_{k+1}B_{k+1}^T\|^2) \\
=& (b_{i_k}^T b_{i_k} - 2a_{i_k}^T Y^{i_k} b_{i_k}) - (\hat{b}_{i_k}^T \hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k}\hat{b}_{i_k}) + (2\hat{a}_{i_k}^T Y^{i_k}\hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k}\hat{b}_{i_k}) \\
\geq& (b_{i_k}^T b_{i_k} - 2\hat{b}_{i_k}^T b_{i_k}) - (\hat{b}_{i_k}^T \hat{b}_{i_k} - 2\hat{b}_{i_k}^T \hat{b}_{i_k}) + (2\hat{a}_{i_k}^T Y^{i_k}\hat{b}_{i_k} - 2\|[Y^{i_k}\hat{b}_{i_k}]_+\| a_{i_k}^T \hat{a}_{i_k}) \\
=& \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\| (2\hat{a}_{i_k}^T \frac{Y^{i_k}\hat{b}_{i_k}}{\|[Y^{i_k}\hat{b}_{i_k}]_+\|} - 2a_{i_k}^T \hat{a}_{i_k}) \\
=& \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\| (2\hat{a}_{i_k}^T \hat{a}_{i_k} - 2a_{i_k}^T \hat{a}_{i_k}) \\
=& \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\| \|a_{i_k} - \hat{a}_{i_k}^T\|^2 \\
\geq& 0.
\end{aligned}$$

The third equality is derived by substituting (4.1) with $\|a_{i_k}\| = \|\hat{a}_{i_k}\| = 1$. The first inequality is derived by (4.3). The last equality is due to $\|a_{i_k}\| = \|\hat{a}_{i_k}\| = 1$. Therefore, $\{f_k\}$ is converging to a limiting value f^*. □

Corollary 4.1. *Apply our* **Adjusted Steps for HALS***, with a given start point* (A_1, B_1)*, the sequence* $\{f_k\}$ *derived by the Adjusted HALS is decreasing to a limiting value* f^**, no need for the assumption that* $\hat{b}_{i_k} \neq 0$*. And for every step*

$$f_k - f_{k+1} \begin{cases} \geq \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\|\|a_{i_k} - \hat{a}_{i_k}^T\|^2 \geq 0, & \textit{if } \hat{b}_{i_k} \neq 0 \\ \geq \|b_{i_k}\|^2, & \textit{if } \hat{b}_{i_k} = 0. \end{cases}$$

Proof. The case when $\hat{b}_{i_k} \neq 0$ is analyzed in Theorem 4.1. If $\hat{b}_{i_k} = 0$, then a_{i_k} will remain unchanged at that iteration. Then,

$$\begin{aligned} & f_k - f_{k+1} \\ =& \|Y - A_k B_k^T\|^2 - \|Y - A_{k+1} B_{k+1}^T\|^2 \\ =& \|Y - A_k B_k^T\|^2 - \|Y - A_k B_{k+1}^T\|^2 \\ =& (b_{i_k}^T b_{i_k} - 2a_{i_k}^T Y^{i_k} b_{i_k}) - (\hat{b}_{i_k}^T \hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k} \hat{b}_{i_k}) \\ \geq& (b_{i_k}^T b_{i_k} - 2\hat{b}_{i_k}^T b_{i_k}) - (\hat{b}_{i_k}^T \hat{b}_{i_k} - 2\hat{b}_{i_k}^T \hat{b}_{i_k}) \\ =& \|b_{i_k}\|^2 \\ \geq& 0. \end{aligned}$$

This result will not affect the convergence of $\{f_k\}$. □

We should mention that $\hat{b}_{i_k} = 0$ happens if and only if a_{i_k} is in the null space of $(Y^{i_k})^T$. However, this case rarely happens since Y_{i_k} and a_{i_k} are changing for every iteration and there are many variables related to them. Theorem 4.1 and Corollary 4.1 describe the lower bound of the reduction and the convergence for the objective value. Next, we will analyze the properties of the accumulation points. To simplify the analysis, we will assume **HALS** can proceed smoothly in the followings.

Assumption 3. *Let (A^*, B^*) be any accumulation point for* **HALS***, the columns of B^* are non-zero vectors. Moreover, there exist $\epsilon > 0$, that $\|b_j^*\| > \epsilon, j = 1, \cdots, J$.*

The above assumption illustrates that the columns of stationary points are away from zero vector.

Theorem 4.2. *Under Assumption 3, the accumulation points of $\{(A_k, B_k)\}$ are absorption points that once it is achieved,* **HASL** *will stay at this point.*

Proof. Let (A^*, B^*) be an accumulation point that $f(A^*, B^*) = f^*$. Apply **HALS** to (A^*, B^*), we firstly see that b_1^* will be updated to $\hat{b}_1^*$ with

$$\hat{b}_1^* = \arg\min_{x \geq 0} \|Y - A^*(B^*)^T + a_1^*(b_1^*)^T - a_1^* x^T\|^2.$$

Since this problem has one unique solution, so that if $b_1^* \neq \hat{b}_1^*$, then

$$\|Y - A^*(B^*)^T + a_1^*(b_1^*)^T - a_1^*(\hat{b}_1^*)^T\|^2 < \|Y - A^*(B^*)^T\|^2 = f^*.$$

For the continuity, we can find a (A_k, B_k) close enough to (A^*, B^*) that

$$\|Y - A_k(B_k)^T + a_1(b_1)^T - a_1(\hat{b}_1^*)^T\|^2 < f^*.$$

Where a_1 and b_1 are the first column of A_k and B_k, respectively. Therefore

$$\min_{x \geq 0} \|Y - A_k(B_k)^T + a_1(b_1)^T - a_1 x^T\|^2 < f^*.$$

It is contradictory to $\{f_k\}$ will decreasingly converge to f^*, so that $b_1^* = \hat{b}_1^*$. Next, we analyze the update of a_1^*. Denote $Y_*^1 = Y - A^*(B^*)^T + a_1^*(b_1^*)^T$, the reduction after updating a_1^* is as follow

$$\begin{aligned}
&\|Y - A^*(B^*)^T\|^2 - \|Y - \sum_{i=2}^{n} a_i^*(b_i^*)^T - \hat{a}_1^*(b_1^*)^T\|^2 \\
=&\|Y_*^1 - a_1^*(b_1^*)^T\|^2 - \|Y_*^1 - \hat{a}_1^*(b_1^*)^T\|^2 \\
=&2(\hat{a}_1^*)^T Y_*^1 b_1^* - 2(a_1^*)^T Y_*^1 b_1^* \\
\geq&\|[Y_*^1 b_1^*]_+\| \|\hat{a}_1^* - a_1^*\|^2.
\end{aligned}$$

For the same reason, the reduction should be 0. So that $\hat{a}_1^* = a_1^*$. The same proof for $j = 2, \cdots, J$ can proceed on. So that (A^*, B^*) is an absorption point. □

Proposition 4.2 (Proposition 2.1.2 (a) in [3]). *If x^* is a local minimum of f over X, then*

$$\nabla f(x^*)^T(x - x^*) \geq 0, \forall x \in X. \tag{4.4}$$

A point x^* satisfying (4.4) is called a stationary point.

Theorem 4.3. *Under Assumption 3, the accumulation points of* **HALS** *are stationary points. Moreover, for any accumulation point (A^*, B^*), the following holds*

$$\|b_i^*\|^2 = \|[Y_*^i b_i^*]_+\|, i = 1, \cdots, J. \tag{4.5}$$

Where $Y_*^i = Y - A^*(B^*)^T + a_i^*(b_i^*)^T$.

Proof. Let (A^*, B^*) be a accumulation point, by Theorem 4.2, it satisfies

$$\begin{cases} b_i^* = [(Y_*^i)^T a_i^*]_+, \\ a_i^* = \frac{[Y_*^i b_i^*]_+}{\|[Y_*^i b_i^*]_+\|}. \end{cases}$$

Next, we prove the accumulation points are stationary points. Considering the following optimal solution

$$\widetilde{a} = \arg\min_{a \geq 0} \|Y_*^i - a(b_i^*)^T\|^2 = \frac{[Y_*^i b_i^*]_+}{\|b_i^*\|^2} = \frac{\|[Y_*^i b_i^*]_+\|}{\|b_i^*\|^2} a_i^*.$$

Then we know

$$\|Y_*^i - \widetilde{a}(b_i^*)^T\|^2 = \|Y_*^i - a_i^* \frac{\|[Y_*^i b_i^*]_+\|}{\|b_i^*\|^2}(b_i^*)^T\|^2 \leq \|Y_*^i - a_i^*(b_i^*)^T\|^2.$$

Since $b_i^* = \arg\min_{b \geq 0} \|Y_*^i - a_i^* b^T\|^2$, and the solution is unique, so that (4.5) holds. Then we derive

$$\begin{cases} b_i^* = [(Y^i)^T a_i^*]_+, \\ a_i^* = \frac{[Y^i b_i^*]_+}{\|b_i^*\|^2}. \end{cases}$$

Which means $a_i^* = \arg\min_{a \geq 0} \|Y_*^i - a(b_i^*)^T\|^2$ and $b_i^* = \arg\min_{b \geq 0} \|Y_*^i - a_i^* b^T\|^2$. Regarding $f(A, B) = \|Y - AB^T\|$ as the function of a_1 and fixing all the other variables, we know

$$f(a_1^*, a_2^*, \cdots, a_J^*, b_1^*, b_2^*, \cdots, b_J^*) \leq f(a_1, a_2^*, \cdots, a_J^*, b_1^*, b_2^*, \cdots, b_J^*), \forall a_1 \geq 0.$$

f is a convex function in respect to a_1, so that the above equation tells

$$\partial_{a_1} f(A^*, B^*)^T (a_1 - a_1^*) \geq 0, \forall a_1 \geq 0.$$

Where ∂_{a_1} is the gradient f in respect with a_1. In the same way, we derive

$$\partial_{a_j} f(A^*, B^*)^T (a_j - a_j^*) \geq 0, \forall a_j \geq 0,$$
$$\partial_{b_j} f(A^*, B^*)^T (b_j - b_j^*) \geq 0, \forall b_j \geq 0.$$

Therefore

$$\sum_{j=1}^{J} \partial_{a_j} f(A^*, B^*)^T (a_j - a_j^*) + \sum_{j=1}^{J} \partial_{b_j} f(A^*, B^*)^T (b_j - b_j^*) \geq 0, \forall A, B \geq 0.$$

Which satisfies (4.4). (A^*, B^*) is a stationary point. □

Next, we consider the upper bound of the reduction for every iteration.

Theorem 4.4. *The sequence* $\{B_k\}$ *generated by* **HALS** *is bounded and enjoys the following properties.*

(a) $\|B_k\| \leq f_1 + \|Y\|$,

(b) $\lim_{k \to +\infty} \|B_k - B_{k+1}\| = 0.$

Proof. First, by the non-negativity of A_k and B_k, and that every column of A_k is normalized, we know that

$$\|B_k\| = tr(B_k B_k^T)^{1/2} \leq tr(B_k A_k^T A_k B_k^T)^{1/2} = \|A_k B_k^T\|.$$

Therefore, we obtain

$$\|B_k\| - \|Y\| \leq \|A_k B_k^T\| - \|Y\| \leq \|Y - A_k B_k^T\| \leq f_1.$$

Which tells (a) holds. For (b), from Theorem 4.1, we know $\lim_{k \to +\infty} f_k = f^*$, then

$$\lim_{k \to +\infty} \|B_k - B_{k+1}\| = \lim_{k \to +\infty} \|b_{i_k} - \hat{b}_{i_{k+1}}\| = 0.$$

The proof is completed. □

Theorem 4.5. *The sequence* $\{A_k\}$ *generated by* **HALS**, *satisfies*

$$\lim_{k\to+\infty} \|A_k - A_{k+1}\| = 0.$$

Proof. Denote $z_k = (A_k, B_k)$ and the norm $\|z_k\| = (\|A_k\|^2 + \|B_k\|^2)^{\frac{1}{2}}$. Denote the algorithmic operator for every update of **HALS** as $\mathcal{H}$ that $z_{k+1} = \mathcal{H}(z_k)$, and we know that $\mathcal{H}$ is a continuous mapping.

Assume $\lim_{k\to+\infty} \|A_k - A_{k+1}\| = 0$ does not hold. Then there exist a $\delta > 0$, and infinity number of k in K that

$$K = \{k : \|A_k - A_{k+1}\| > \delta\}.$$

Let z^+ and z^* be a accumulation point of subsequence $\{z_k\}_{k\in I}$ and $\{z_{k+1}\}_{i\in I}$, respectively. Then $\|z^+ - z^*\| \geq \delta$, which means $z^+ \neq z^*$. However, $z^* = \mathcal{H}(z^+)$, it is contradictory to Theorem 4.2 that z^+ is an absorption point. The proof is completed. □

Lemma 4.2. *Let* $z_1, z_2 \in \mathbb{R}^n$ *satisfies*

$$\|z_1 - z_2\| \leq \varepsilon. \tag{4.6}$$

Then, we can derive that

$$[z_1]_-^T[z_2]_+ \leq n\varepsilon^2.$$

Where $[z_1]_- = \max\{0, -z_1\}$.

Proof. When the q-th elements of z_1 and z_2 are both non-negative or the q-th element of z_2 is non-positive, $[(z_1)_q]_-[(z_2)_q]_+ = 0$. Only for the case that $(z_2)_q$ is positive and $(z_1)_q$ is negative, $[(z_1)_q]_-[(z_2)_q]_+ > 0$. In this case, from (4.6), we know

$$0 < (z_2)_q \leq (z_1)_q + \varepsilon \Rightarrow (z_1)_q \geq -\varepsilon$$

$$(z_2)_q - \varepsilon \leq (z_1)_q < 0 \Rightarrow (z_2)_q \leq \varepsilon.$$

Therefore,

$$[z_1]_-^T[z_2]_+ = \sum_q [(z_1)_q]_-[(z_2)_q]_+ \leq n\varepsilon^2.$$

□

Theorem 4.6. *Under Assumption 3, with a given start point* (A_1, B_1)*, the sequence* $\{f_k\}$ *is decreasing to a limiting value* f^**. And with* k *goes to infinity, the following holds*

$$f_k - f_{k+1} = O(\|a_{i_k} - \hat{a}_{i_k}\|^2 + \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|A_{k-J} - A_k\|^2 + \|B_{k-J} - B_k\|^2).$$

Proof. With a similar proof of Theorem 4.1, we can derive

$$\begin{aligned} & f_k - f_{k+1} \\ =& (b_{i_k}^T b_{i_k} - 2a_{i_k}^T Y^{i_k} b_{i_k}) - (\hat{b}_{i_k}^T \hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k} \hat{b}_{i_k}) + (2\hat{a}_{i_k}^T Y^{i_k} \hat{b}_{i_k} - 2a_{i_k}^T Y^{i_k} \hat{b}_{i_k}) \\ =& \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|[Y^{i_k}\hat{b}_{i_k}]_+\| \|a_{i_k} - \hat{a}_{i_k}^T\|^2 + 2[a_{i_k}^T Y^{i_k}]_- b_{i_k} + 2a_{i_k}^T [Y^{i_k} \hat{b_{i_k}}]_-. \end{aligned} \tag{4.7}$$

Next, We analyze the term $2[a_{i_k}^T Y^{i_k}]_- b_{i_k}$ and $2a_{i_k}^T [Y^{i_k}\hat{b}_{i_k}]_-$. To calculate $2[a_{i_k}^T Y^{i_k}]_- b_{i_k}$, we make the denotation

$$y_{i_k} = (Y^{i_k})^T a_{i_k} = Y^T a_{i_k} - B_k A_k^T a_{i_k} + b_{i_k}.$$

Then, we know $[a_{i_k}^T Y^{i_k}]_- = [y_{i_k}^T]_-$ and $b_{i_k} = [y_{i_{k-J}}]_+$. What follows is to calculate $[y_{i_k}^T]_- [y_{i_{k-J}}]_+$. We derive the following results first.

$$\begin{aligned} & \|B_k A_k^T a_{i_k} - B_{k-J} A_{k-J}^T a_{i_{k-J}}\| \\ =& \|B_k A_k^T a_{i_k} - B_{k-J} A_k^T a_{i_k} + B_{k-J} A_k^T a_{i_k} - B_{k-J} A_{k-J}^T a_{i_{k-J}}\| \\ \leq& \|B_k A_k^T a_{i_k} - B_{k-J} A_k^T a_{i_k}\| + \|B_{k-J} A_k^T a_{i_k} - B_{k-J} A_{k-J}^T a_{i_{k-J}}\| \\ \leq& \|B_k - B_{k-J}\| \|A_k^T a_{i_k}\| + \|B_{k-J}\| \|A_k^T a_{i_k} - A_{k-J}^T a_{i_{k-J}}\| \\ \leq& \|B_k - B_{k-J}\| \|A_k^T\| \|a_{i_k}\| + \|B_{k-J}\| (\|(A_k^T - A_{k-J}^T) a_{i_k}\| + \|A_{k-J}^T (a_{i_k} - a_{i_{k-J}})\|) \\ \leq& \|B_k - B_{k-J}\| \sqrt{J} + \|B_{k-J}\| \|A_k^T - A_{k-J}^T\| + \|B_{k-J}\| \sqrt{J} \|A_k - A_{k-J}\| \\ =& \|B_k - B_{k-J}\| \sqrt{J} + \|B_{k-J}\| (\sqrt{J} + 1) \|A_k - A_{k-J}\| \end{aligned} \tag{4.8}$$

Then we can obtain

$$
\begin{aligned}
&\|y_{i_{k-J}} - y_{i_k}\| \\
=&\|Y^T a_{i_{k-J}} - B_{k-J} A_{k-J}^T a_{i_{k-J}} + b_{i_{k-J}} - (Y^T a_{i_k} - B_k A_k^T a_{i_k} + b_{i_k})\| \\
\leq&\|Y\| \|a_{i_{k-J}} - a_{i_k}\| + \|B_k A_k^T a_{i_k} - B_{k-J} A_{k-J}^T a_{i_{k-J}}\| + \|b_{i_{k-J}} - b_{i_k}\| \\
\leq&\|Y\| \|A_{k-J} - A_k\| + \|B_k A_k^T a_{i_k} - B_{k-J} A_{k-J}^T a_{i_{k-J}}\| + \|B_{k-J} - B_k\| \\
\leq&(\|Y\| + \|B_{k-J}\|(\sqrt{J}+1))\|A_k - A_{k-J}\| + (\sqrt{J}+1)\|B_k - B_{k-J}\|.
\end{aligned}
$$

From Theorem 4.4, we know $\|B_{k-J}\|$ is upper bounded. The last inequality is due to the result of (4.8). Following Lemma 4.2, we obtain

$$
2[a_{i_k}^T Y^{i_k}]_- b_{i_k} = 2[y_{i_k}]_-^T [y_{i_{k-J}}]_+ = O(\|A_{k-J} - A_k\|^2 + \|B_{k-J} - B_k\|^2). \tag{4.9}
$$

Then, we analyze the term $2a_{i_k}^T [Y^{i_k} \hat{b}_{i_k}]_-$. Denote

$$
z_{i_k} = Y^{i_k} \hat{b}_{i_k} = Y \hat{b}_{i_k} - A_k B_k^T \hat{b}_{i_k} + a_{i_k} b_{i_k}^T \hat{b}_{i_k}.
$$

Then we have

$$
2a_{i_k}^T [Y^{i_k} \hat{b}_{i_k}]_- = 2\frac{[Y^{i_{k-J}} \hat{b}_{i_{k-J}}]_+^T [Y^{i_k} \hat{b}_{i_k}]_-}{\|[Y^{i_{k-J}} \hat{b}_{i_{k-J}}]_+\|} = 2\frac{[z_{k-J}]_+^T [z_k]_-}{\|[Y^{i_{k-J}} \hat{b}_{i_{k-J}}]_+\|}. \tag{4.10}
$$

(A_k, B_k) will converge to the set of stationary points, and all the stationary points satisfy (4.5). Under the Assumption 3, there exists a large enough k_0 that for $k > k_0$ the following holds

$$
\|[Y^{i_k} \hat{b}_{i_{k-J}}]_+\| \geq \|\hat{b}_{i_{k-J}}\|^2 - \frac{\epsilon}{2} \geq \frac{\epsilon}{2} > 0. \tag{4.11}
$$

The above tells that $\|[Y^{i_{k-J}} \hat{b}_{i_{k-J}}]_+\|$ is lower bounded with k going to infinity. The following results will be used later.

$$\begin{aligned}
&\|A_{k-J}B_{k-J}^T\hat{b}_{i_{k-J}} - A_kB_k^T\hat{b}_{i_k}\| \\
=&\|A_{k-J}B_{k-J}^T\hat{b}_{i_{k-J}} - A_kB_{k-J}^T\hat{b}_{i_{k-J}} + A_kB_{k-J}^T\hat{b}_{i_{k-J}} - A_kB_k^T\hat{b}_{i_k}\| \\
\leq&\|A_{k-J} - A_k\|\|B_{k-J}^T\hat{b}_{i_{k-J}}\| + \|A_k\|\|B_{k-J}^T\hat{b}_{i_{k-J}} - B_k^T\hat{b}_{i_k}\| \\
\leq&\|A_{k-J} - A_k\|\|B_{k-J}^T\hat{b}_{i_{k-J}}\| + \|A_k\|(\|B_{k-J}^T\hat{b}_{i_{k-J}} - B_k^T\hat{b}_{i_{k-J}}\| + \|B_k^T\hat{b}_{i_{k-J}} - B_k^T\hat{b}_{i_k}\|) \\
\leq&\|A_{k-J} - A_k\|\|B_{k-J}^T\hat{b}_{i_{k-J}}\| + \|A_k\|(\|B_{k-J}^T - B_k^T\|\|\hat{b}_{i_{k-J}}\| + \|B_k^T\|\|\hat{b}_{i_{k-J}} - \hat{b}_{i_k}\|) \\
=&\|B_{k-J}^T\hat{b}_{i_{k-J}}\|\|A_{k-J} - A_k\| + \sqrt{J}\|\hat{b}_{i_{k-J}}\|\|B_{k-J}^T - B_k^T\| + \sqrt{J}\|B_k\|\|b_{i_k} - \hat{b}_{i_k}\|. \quad (4.12)
\end{aligned}$$

Where $\|B_{k-J}^T\hat{b}_{i_{k-J}}\|, \|B_{k-J}^T\|$ and $\|\hat{b}_{i_{k-J}}\|$ are bounded by Theorem 4.4. Also, we have

$$\begin{aligned}
&\|a_{i_k}b_{i_k}^T\hat{b}_{i_k} - a_{i_{k-J}}b_{i_{k-J}}^T\hat{b}_{i_{k-J}}\| \\
\leq&\|a_{i_k} - a_{i_{k-J}}\|b_{i_k}^T\hat{b}_{i_k} + \|a_{i_{k-J}}\||b_{i_k}^T\hat{b}_{i_k} - b_{i_{k-J}}^T\hat{b}_{i_{k-J}}| \\
\leq&\|a_{i_k} - a_{i_{k-J}}\|b_{i_k}^T\hat{b}_{i_k} + |b_{i_k}^T\hat{b}_{i_k} - b_{i_{k-J}}^T\hat{b}_{i_{k-J}}| \\
\leq&\|a_{i_k} - a_{i_{k-J}}\|b_{i_k}^T\hat{b}_{i_k} + \|b_{i_k}^T - b_{i_{k-J}}^T\|\|\hat{b}_{i_k}\| + \|b_{i_{k-J}}^T\|\|\hat{b}_{i_k} - \hat{b}_{i_{k-J}}\| \\
\leq&\|a_{i_k} - a_{i_{k-J}}\|b_{i_k}^T\hat{b}_{i_k} + \|b_{i_k} - b_{i_{k-J}}\|\|\hat{b}_{i_k}\| + \|b_{i_{k-J}}\|\|\hat{b}_{i_k} - b_{i_k}\| \\
\leq&b_{i_k}^T\hat{b}_{i_k}\|A_k - A_{k-J}\| + \|\hat{b}_{i_k}\|\|B_k - B_{k-J}\| + \|b_{i_{k-J}}\|\|\hat{b}_{i_k} - b_{i_k}\|. \quad (4.13)
\end{aligned}$$

Where $b_{i_k}^T\hat{b}_{i_k}$, $\|\hat{b}_{i_k}\|$ and $\|b_{i_{k-J}}\|$ are upper bounded by Theorem 4.4. Next, we make the calculation.

$$\begin{aligned}
&\|z_{i_{k-J}} - z_{i_k}\| \\
\leq&\|Y\|\|\hat{b}_{i_{k-J}} - \hat{b}_{i_k}\| + \|A_{k-J}B_{k-J}^T\hat{b}_{i_{k-J}} - A_kB_k^T\hat{b}_{i_k}\| + \|a_{i_k}b_{i_k}^T\hat{b}_{i_k} - a_{i_{k-J}}b_{i_{k-J}}^T\hat{b}_{i_{k-J}}\| \\
\leq&(\|Y\| + \sqrt{J}\|B_k\| + \|b_{i_{k-J}}\|)\|b_{i_k} - \hat{b}_{i_k}\| + (\|B_{k-J}^T\hat{b}_{i_{k-J}}\| + b_{i_k}^T\hat{b}_{i_k})\|A_{k-J} - A_k\| \\
&+ (\sqrt{J}\|\hat{b}_{i_{k-J}}\| + \|\hat{b}_{i_k}\|)\|B_{k-J} - B_k\|
\end{aligned} \quad (4.14)$$

The second inequality combines the results of (4.12) and (4.13). Considering (4.10), (4.11) and (4.14), following Lemma 4.2, we obtain

$$2a_{i_k}^T[Y^{i_k}\hat{b_{i_k}}]_- = 2\frac{[z_{k-J}]_+^T[z_k]_-}{\|[Y^{i_{k-J}}\hat{b}_{i_{k-J}}]_+\|} = O(\|b_{i_k} - \hat{b}_{i_k}\|^2 + \|A_{k-J} - A_k\|^2 + \|B_{k-J} - B_k\|^2). \quad (4.15)$$

Finally, combining (4.7), (4.9), (4.11), and (4.15), we can derive

$$f_k - f_{k+1} = O(\|a_{i_k} - \hat{a}_{i_k}\|^2 + \|b_{i_k} - \hat{b}_{i_k}\|^2 + \|A_{k-J} - A_k\|^2 + \|B_{k-J} - B_k\|^2).$$

□

4.3 Convergence for Fast Hierarchical Alternating Least Squares Algorithm

The k-th iteration of **Fast HALS** updating process can be viewed as

$$(A_k, B_k) \to (A_k, B_{k+1}) \to (A_{k+1}, B_{k+1}).$$

Every column of B_k is first updated, and then all the columns of A_k are updated. Denote a_i^k the i-th column of A_k, b_i^k the i-th column of B_k. After the first i columns are updated, A_k is changed to A_k^i. The same with B_k^i. Therefore $A_k^J = A_{k+1}, A_k^0 = A_k$, $B_k^J = B_{k+1}$, and $B_k^0 = B_k$. Denote

$$Y_k^i = Y - A_k(B_k^{i-1})^T + a_i^k(b_i^k)^T = Y - A_k(B_k^i)^T + a_i^k(b_i^{k+1})^T. \tag{4.16}$$

The second equality holds because B_k^{i-1} and B_k^i are only different in the i-th column. The update for B_k to B_{k+1} can be expressed as

$$b_i^k \to b_i^{k+1} = [(Y_k^i)^T a_i^k]_+, \ i \text{ is from 1 to } J. \tag{4.17}$$

Denote

$$\overline{Y}_k^i = Y - A_k^{i-1}B_{k+1}^T + a_i^k(b_i^{k+1})^T = Y - A_k^i B_{k+1}^T + a_i^{k+1}(b_i^{k+1})^T. \tag{4.18}$$

The second equality holds because A_k^{i-1} and A_k^i are only different in the i-th column. The update process for A_k to A_{k+1} is

$$a_i^k \to a_i^{k+1} = \frac{[\overline{Y}_k^i b_i^{k+1}]_+}{\|[\overline{Y}_k^i b_i^{k+1}]_+\|}, \ i \text{ is from 1 to } J. \tag{4.19}$$

In the execution of **Fast HALS**, if $[\overline{Y}_k^i b_i^{k+1}]_+ = 0$, the normalization process for a_j will be a problem. To avoid this problem, we usually replace the operation $[x]_+ = max\{0, x\}$ with $[x]_+ = max\{\varepsilon, x\}$, where ε is a small positive number, such as 10^{-14}. Here, we propose another approach to get rid of this problem. Replace the steps of **Fast HALS** in lines 12 and 13 with the following steps.

Ajusted Steps for FAST HALS

$\hat{a}_j = [a_j q_{jj} + p_j - A q_j]_+$;

if $\hat{a}_j = 0$

    $b_j = 0$

else

    $a_j = \hat{a}_j / \|\hat{a}_j\|_2$;

end if;

The above steps describe the adjustments that if $[\overline{Y}_k^i b_i^{k+1}]_+ = 0$ occurs, b_j^k is set to be 0, and a_j^k remains unchanged. This approach will validate the implementation and does not influence the convergence. To simplify the case, we will assume **Fast HALS** can proceed smoothly in the following analysis.

Theorem 4.7. *With a given start point (A_1, B_1), the sequence $\{f_k\}$ generated by* **Fast HALS** *is decreasing to a limiting value f^*. And for every step*

$$f(A_k, B_k) - f(A_{k+1}, B_{k+1}) \geq \|B_k - B_{k+1}\|_F^2 + \|D_k(A_k - A_{k+1})\|_F^2,$$

where $D_k = \mathrm{diag}(\|[\overline{Y}_k^i b_i^{k+1}]_+\|^{\frac{1}{2}})$ and $\overline{Y}_k^i = Y - A_k^{i-1} B_{k+1} + a_i^k (b_i^{k+1})^T$.

Proof. From the proof of Theorem 4.1 we know

$$\begin{aligned}
&f(A_k, B_k^{i-1}) - f(A_k, B_k^i) \\
=&\|Y - A_k(B_k^{i-1})^T\|^2 - \|Y - A_k(B_k^i)^T\|^2 \\
=&\|Y_k^i - a_i^k(b_i^k)^T\|^2 - \|Y_k^i - a_i^k(b_i^{k+1})^T\|^2 \\
=&(\|b_i^k\|^2 - 2(a_i^k)^T Y_k^i b_i^k) - (\|b_i^{k+1}\|^2 - 2(a_i^k)^T Y_k^i b_i^{k+1}) \\
\geq&(\|b_i^k\|^2 - 2[(a_i^k)^T Y_k^i]_+ b_i^k) - (\|b_i^{k+1}\|^2 - 2(a_i^k)^T Y_k^i b_i^{k+1}) \\
=&\|b_i^k\|^2 - 2(b_i^{k+1})^T b_i^k + \|b_i^{k+1}\|^2 \\
=&\|b_i^k - b_i^{k+1}\|^2.
\end{aligned}$$

The second equality is due to (4.16). The third equality comes from (4.1) and $\|a_i^k\| = 1$. The first inequality is due to the non-negativity of b_i^k. The fourth equality holds because (4.17) and $(a_i^k)^T Y_k^i b_i^{k+1} = \|b_i^{k+1}\|^2$. Therefore

$$\begin{aligned}
f(A_k, B_k) - f(A_k, B_{k+1}) &= \sum_{i=1}^{J} f(A_k, B_k^{i-1}) - f(A_k, B_k^i) \\
&\geq \sum_{i=1}^{J} \|b_i^k - b_i^{k+1}\|^2 \\
&= \|B_k - B_{k+1}\|^2.
\end{aligned}$$

From the proof of Theorem 4.1 we know

$$\begin{aligned}
&f(A_k^{i-1}, B_{k+1}) - f(A_k^i, B_{k+1}) \\
=&\|Y - A_k^{i-1} B_{k+1}^T\|^2 - \|Y - A_k^i B_{k+1}^T\|^2 \\
=&\|\overline{Y}_k^i - a_i^k(b_i^{k+1})^T\|^2 - \|\overline{Y}_k^i - a_i^{k+1}(b_i^{k+1})^T\|^2 \\
=&2(a_i^{k+1})^T \overline{Y}_k^i b_i^{k+1} - 2(a_i^k)^T \overline{Y}_k^i b_i^{k+1} \\
\geq&2(a_i^{k+1})^T \overline{Y}_k^i b_i^{k+1} - 2(a_i^k)^T [\overline{Y}_k^i b_i^{k+1}]_+ \\
=&\|[\overline{Y}_k^i b_i^{k+1}]_+\|(2\|a_i^{k+1}\|^2 - 2(a_i^k)^T a_i^{k+1}) \\
=&\|[\overline{Y}_k^i b_i^{k+1}]_+\|\|a_i^k - a_i^{k+1}\|^2.
\end{aligned} \tag{4.20}$$

The second equality is due to (4.18). The third equality is the simplification after applying (4.1) and $\|a_i^k\| = \|a_i^{k+1}\| = 1$. The first inequality is due to the non-negativity of a_i^k. The fourth equality is due to (4.19). Therefore

$$\begin{aligned} f(A_k, B_{k+1}) - f(A_{k+1}, B_{k+1}) &= \sum_{i=1}^{J} f(A_k^{i-1}, B_{k+1}) - f(A_k^i, B_{k+1}) \\ &\geq \sum_{i=1}^{J} \|[\overline{Y}_k^i b_i^{k+1}]_+\| \|a_i^k - \hat{a}_i^k\|^2 \\ &= \|D_k(A_k - A_{k+1})\|_F^2 \end{aligned}$$

By the above result, we derive

$$\begin{aligned} & f(A_k, B_k) - f(A_{k+1}, B_{k+1}) \\ =& f(A_k, B_k) - f(A_k, B_{k+1}) + f(A_k, B_{k+1}) - f(A_{k+1}, B_{k+1}) \\ \geq& \|B_k - B_{k+1}\|_F^2 + \|D_k(A_k - A_{k+1})\|_F^2. \end{aligned}$$

Since f_k is decreasing with lower bound, so that $\lim_{k\to\infty} f_k = f^*$. □

If $[\overline{Y}_k^i b_i^{k+1}] = 0$ happens and our adjustment steps are applied, then we know a_j^k will remain unchanged while b_j^k is changed to be 0. In this case, we denote $\hat{B}_{k+1}$ for B_{k+1} after the j-th column is forced to be 0. Then, the reduction, as mentioned in (4.20), is adjusted as follows

$$\begin{aligned} & f(A_k^{i-1}, B_{k+1}) - f(A_k^i, \hat{B}_{k+1}) \\ =& \|Y - A_k^{i-1} B_{k+1}^T\|^2 - \|Y - A_k^{i-1} \hat{B}_{k+1}^T\|^2 \\ =& \|\overline{Y}_k^i - a_i^k (b_i^{k+1})^T\|^2 - \|\overline{Y}_k^i\|^2 \\ =& \|b_i^{k+1}\|^2 - 2(a_i^k)^T \overline{Y}_k^i b_i^{k+1} \\ \geq& \|b_i^{k+1}\|^2 \\ \geq& 0. \end{aligned}$$

The above result will not affect the convergence of $\{f_k\}$.

Assumption 4. *Let (A^*, B^*) be any accumulation point for* **HALS**, *the columns of A^* and B^* are non-zero vectors. Moreover, there exist $\epsilon > 0$, that*

$$\|a_j^*\| \geq \epsilon, \|b_j^*\| > \epsilon, j = 1, \cdots, J.$$

Theorem 4.8. *Under Assumption 4, the accumulation points of* $\{(A_k, B_k)\}$ *are absorption points that once it is achieved,* **Fast HASL** *will stay at this point.*

Proof. The proof is similar to Theorem 4.2 with small adjustments. Let (A^*, B^*) be an accumulation point that $f(A^*, B^*) = f^*$. Applying **Fast HALS** to (A^*, B^*), $\hat{b}_1^* = b_1^*$ is proved in Theorem 4.2. For the same reason, $\hat{b}_j^* = b_j^*, j = 2, \cdots, J$ will be derived. Then, the proof in Theorem 4.2 tells $\hat{a}_1^* = a_1^*$. With the same analysis, we can see $\hat{a}_j^* = a_j^*, j = 2, \cdots, J$. So (A^*, B^*) is an absorption point. □

Theorem 4.9. *Under Assumption 4, the accumulation points of* **Fast HALS** *are stationary points. Moreover, for any accumulation point* (A^*, B^*)*, the following holds*

$$\|b_i^*\|^2 = \|[Y_*^i b_i^*]_+\|, i = 1, \cdots, J. \tag{4.21}$$

Where $Y_*^i = Y - A^*(B^*)^T + a_i^*(b_i^*)^T$.

Proof. The proof is the same with Theorem 4.3. □

Theorem 4.4 and Theorem 4.5 for **FALS** still hold for **Fast HALS** with the same proof. These theorems tell that

(a) $\|B_k\|$ is bounded;

(b) $\lim_{k\to\infty} \|A_k - A_{k+1}\| = 0$ and $\lim_{k\to\infty} \|B_k - B_{k+1}\| = 0$.

Theorem 4.10. *Under the Assumption 4, with a given start point* (A_1, B_1)*, the sequence* $\{f_k\}$ *is decreasing to a limiting value* f^**. And for every step*

$$f_k - f_{k+1} = O(\|A_k - A_{k-1}\|^2 + \|A_{k+1} - A_k\|^2 + \|B_k - B_{k+1}\|^2 + \|B_{k+1} - B_k\|^2).$$

Proof. First, the update of b_i^k makes the following reduction.

$$\begin{aligned} & f(A_k, B_k^{i-1}) - f(A_k, B_k^i) \\ =& \|Y - A_k(B_k^{i-1})^T\|^2 - \|Y - A_k(B_k^i)^T\|^2 \\ =& \|Y_k^i - a_i^k(b_i^k)^T\|^2 - \|Y_k^i - a_i^k(b_i^{k+1})^T\|^2 \\ =& (\|b_i^k\|^2 - 2(a_i^k)^T Y_k^i b_i^k) - (\|b_i^{k+1}\|^2 - 2(a_i^k)^T Y_k^i b_i^{k+1}) \\ =& (\|b_i^k\|^2 - 2[(a_i^k)^T Y_k^i]_+ b_i^k) - (\|b_i^{k+1}\|^2 - 2(a_i^k)^T Y_k^i b_i^{k+1}) + 2[(a_i^k)^T Y_k^i]_- b_i^k \\ =& \|b_i^k - b_i^{k+1}\|^2 + 2[(a_i^k)^T Y_k^i]_- b_i^k. \end{aligned} \tag{4.22}$$

Then, we analyze the term $2[(a_i^k)^T Y_k^i]_- b_i^k$. As we know

$$b_i^k = [(Y_{k-1}^i)^T a_i^{k-1}]_+,$$

what we need to calculate is $[(a_i^k)^T Y_k^i]_- [(Y_{k-1}^i)^T a_i^{k-1}]_+$. The following results will be used later.

$$\begin{aligned} & \|B_k^{i-1} A_k^T a_i^k - B_{k-1}^{i-1} A_{k-1}^T a_i^{k-1}\| \\ \leq & \|B_k^{i-1} - B_{k-1}^{i-1}\| \|A_k^T a_i^k\| + \|B_{k-1}^{i-1}\| \|A_k^T a_i^k - A_{k-1}^T a_i^{k-1}\| \\ \leq & \|B_k^{i-1} - B_{k-1}^{i-1}\| \|A_k^T a_i^k\| + \|B_{k-1}^{i-1}\| \|A_k^T - A_{k-1}^T\| \|a_i^k\| + \|B_{k-1}^{i-1}\| \|A_{k-1}^T\| \|a_i^k - a_i^{k-1}\| \\ \leq & \sqrt{J} \|B_k^{i-1} - B_{k-1}^{i-1}\| + \|B_{k-1}^{i-1}\| \|A_k^T - A_{k-1}^T\| + \sqrt{J} \|B_{k-1}^{i-1}\| \|a_i^k - a_i^{k-1}\|. \end{aligned} \tag{4.23}$$

Then we calculate the difference between $(Y_{k-1}^i)^T a_i^{k-1}$ and $(Y_k^i)^T a_i^k$.

$$\begin{aligned} & \|(Y_{k-1}^i)^T a_i^{k-1} - (Y_k^i)^T a_i^k\| \\ =& \|Y^T a_i^{k-1} - B_{k-1}^{i-1} A_{k-1}^T a_i^{k-1} + b_i^{k-1} - (Y^T a_i^k - B_k^{i-1} A_k^T a_i^k + b_i^k)\| \\ \leq & \|Y^T\| \|a_i^{k-1} - a_i^k\| + \|B_k^{i-1} A_k^T a_i^k - B_{k-1}^{i-1} A_{k-1}^T a_i^{k-1}\| + \|b_i^{k-1} - b_i^k\| \\ \leq & (\|Y^T\| + \sqrt{J}\|B_{k-1}^{i-1}\|) \|a_i^{k-1} - a_i^k\| + \sqrt{J}\|B_k^{i-1} - B_{k-1}^{i-1}\| + \|B_{k-1}^{i-1}\| \|A_k^T - A_{k-1}^T\| + \|b_i^{k-1} - b_i^k\| \\ =& O(\|B_k^{i-1} - B_{k-1}^{i-1}\| + \|A_k - A_{k-1}\|). \end{aligned}$$

The second inequality is due to (4.23). Then, following Lemma 4.2, we can derive

$$2[(a_i^k)^T Y_k^i]_- b_i^k = O(\|B_k^{i-1} - B_{k-1}^{i-1}\|^2 + \|A_k - A_{k-1}\|^2).$$

Combining (4.22), we obtain

$$f(A_k, B_k^{i-1}) - f(A_k, B_k^i) = O(\|b_i^k - b_i^{k+1}\|^2 + \|B_k^{i-1} - B_{k-1}^{i-1}\|^2 + \|A_k - A_{k-1}\|^2). \quad (4.24)$$

Next, the update of a_i^k makes the following reduction.

$$\begin{aligned}
&f(A_k^{i-1}, B_{k+1}) - f(A_k^i, B_{k+1})\\
=&\|Y - A_k^{i-1}B_{k+1}^T\|^2 - \|Y - A_k^i B_{k+1}^T\|^2\\
=&\|\overline{Y}_k^i - a_i^k(b_i^{k+1})^T\|^2 - \|\overline{Y}_k^i - a_i^{k+1}(b_i^{k+1})^T\|^2\\
=&2(a_i^{k+1})^T\overline{Y}_k^i b_i^{k+1} - 2(a_i^k)^T\overline{Y}_k^i b_i^{k+1}\\
=&2(a_i^{k+1})^T\overline{Y}_k^i b_i^{k+1} - 2(a_i^k)^T[\overline{Y}_k^i b_i^{k+1}]_+ + 2(a_i^k)^T[\overline{Y}_k^i b_i^{k+1}]_-\\
=&\|[\overline{Y}_k^i b_i^{k+1}]_+\|\|a_i^k - a_i^{k+1}\|^2 + 2(a_i^k)^T[\overline{Y}_k^i b_i^{k+1}]_-.
\end{aligned} \quad (4.25)$$

Then, we analyze the term $2(a_i^k)^T[\overline{Y}_k^i b_i^{k+1}]_-$. We know that

$$a_i^k = \frac{[\overline{Y}_{k-1}^i b_i^k]_+}{\|[\overline{Y}_{k-1}^i b_i^k]_+\|}.$$

We calculate the term $[\overline{Y}_{k-1}^i b_i^k]_+^T[\overline{Y}_k^i b_i^{k+1}]_-$ first and then calculate $2(a_i^k)^T[\overline{Y}_k^i b_i^{k+1}]_-$. The following results will be used later.

$$\begin{aligned}
&\|A_{k-1}^{i-1}B_k^T b_i^k - A_k^{i-1}B_{k+1}^T b_i^{k+1}\|\\
\leq&\|A_{k-1}^{i-1} - A_k^{i-1}\|\|B_k^T b_i^k\| + \|A_k^{i-1}\|\|B_k^T b_i^k - B_{k+1}^T b_i^{k+1}\|\\
\leq&\|A_{k-1}^{i-1} - A_k^{i-1}\|\|B_k^T b_i^k\| + \|A_k^{i-1}\|\|B_k^T - B_{k+1}^T\|\|b_i^k\| + \|A_k^{i-1}\|\|B_{k+1}^T\|\|b_i^{k+1} - b_i^k\|\\
=&\|B_k^T b_i^k\|\|A_{k-1}^{i-1} - A_k^{i-1}\| + \sqrt{J}\|b_i^k\|\|B_k^T - B_{k+1}^T\| + \sqrt{J}\|B_{k+1}^T\|\|b_i^{k+1} - b_i^k\|.
\end{aligned} \quad (4.26)$$

Also, the following result is needed.

$$\begin{aligned}
&\|a_i^k(b_i^{k+1})^T b_i^{k+1} - a_i^{k-1}(b_i^k)^T b_i^k\|\\
\leq&\|a_i^k - a_i^{k-1}\|\|b_i^{k+1}\|^2 + \|a_i^{k-1}\|\|(b_i^{k+1})^T b_i^{k+1} - (b_i^k)^T b_i^k\|\\
\leq&\|a_i^k - a_i^{k-1}\|\|b_i^{k+1}\|^2 + \|b_i^k - b_i^{k+1}\|\|b_i^k + b_i^{k+1}\|.
\end{aligned} \quad (4.27)$$

We calculate the difference between $\overline{Y}_k^i b_i^{k+1}$ and $\overline{Y}_{k-1}^i b_i^k$.

$$
\begin{aligned}
&\|\overline{Y}_k^i b_i^{k+1} - \overline{Y}_{k-1}^i b_i^k\| \\
\leq &\|Y\|\|b_i^{k+1} - b_i^k\| + \|A_{k-1}^{i-1} B_k^T b_i^k - A_k^{i-1} B_{k+1}^T b_i^{k+1}\| + \|a_i^k (b_i^{k+1})^T b_i^{k+1} - a_i^{k-1} (b_i^k)^T b_i^k\| \\
\leq &(\|Y\| + \sqrt{J}\|B_{k+1}\| + 1 + \|b_i^{k+1} + b_i^k\|)\|b_i^{k+1} - b_i^k\| + \|b_i^{k+1}\|^2 \|a_i^k - a_i^{k-1}\| \\
&+ \|B_k^T b_i^k\|\|A_{k-1}^{i-1} - A_k^{i-1}\| + \sqrt{J}\|b_i^k\|\|B_k - B_{k+1}\| \\
= &O(\|A_{k-1}^{i-1} - A_k^{i-1}\| + \|B_k - B_{k+1}\|)
\end{aligned}
$$

The second inequality comes from the results of (4.26) and (4.27). Here $\|B_{k+1}\|$, $\|b_i^{k+1}\|$, $\|b_i^k\|$, and $\|B_k^T b_i^k\|$ are bounded. Following Lemma 4.2, we obtain

$$
[\overline{Y}_{k-1}^i b_i^k]_+^T [\overline{Y}_k^i b_i^{k+1}]_- = O(\|A_{k-1}^{i-1} - A_k^{i-1}\|^2 + \|B_k - B_{k+1}\|^2). \tag{4.28}
$$

(A_k, B_k) will converge to the set of stationary points, and all the stationary points satisfy (4.21). Under Assumption 4, there exists a large enough k_0 such that for $k > k_0$ the following holds

$$
\|[\overline{Y}_{k-1}^i b_i^k]_+\| \geq \|b_i^k\|^2 - \frac{\epsilon}{2} \geq \frac{\epsilon}{2} > 0.
$$

The above tells that $\|[\overline{Y}_{k-1}^i b_i^k]_+\|$ is lower bounded with k going to infinity. Therefore, in view of (4.28), we derive

$$
2(a_i^k)^T [\overline{Y}_k^i b_i^{k+1}]_- = \frac{[\overline{Y}_{k-1}^i b_i^k]_+^T [\overline{Y}_k^i b_i^{k+1}]_-}{\|[\overline{Y}_{k-1}^i b_i^k]_+\|} = O(\|A_{k-1}^{i-1} - A_k^{i-1}\|^2 + \|B_k - B_{k+1}\|^2).
$$

Combining (4.25), we derive

$$
f(A_k^{i-1}, B_{k+1}) - f(A_k^i, B_{k+1}) = O(\|a_i^k - a_i^{k+1}\|^2 + \|A_{k-1}^{i-1} - A_k^{i-1}\|^2 + \|B_k - B_{k+1}\|^2). \tag{4.29}
$$

Then we derive

$$
\begin{aligned}
& f(A_k, B_k) - f(A_{k+1}, B_{k+1}) \\
=& f(A_k, B_k) - f(A_k, B_{k+1}) + f(A_k, B_{k+1}) - f(A_{k+1}, B_{k+1}) \\
=& \sum_{i=1}^{J} f(A_k, B_k^{i-1}) - f(A_k, B_k^{i}) + \sum_{i=1}^{J} f(A_k^{i-1}, B_{k+1}) - f(A_k^{i}, B_{k+1}) \\
=& \sum_{i=1}^{J} O(\|b_i^k - b_i^{k+1}\|^2 + \|B_k^{i-1} - B_{k-1}^{i-1}\|^2 + \|A_k - A_{k-1}\|^2) \\
& + \sum_{i=1}^{J} O(\|a_i^k - a_i^{k+1}\|^2 + \|A_{k-1}^{i-1} - A_k^{i-1}\|^2 + \|B_k - B_{k+1}\|^2) \\
=& O(\|A_k - A_{k-1}\|^2 + \|A_{k+1} - A_k\|^2 + \|B_k - B_{k+1}\|^2 + \|B_{k+1} - B_k\|^2).
\end{aligned}
$$

The third equality is due to (4.24) and (4.29). The last equality holds because

$$
\begin{aligned}
\|B_k^{i-1} - B_{k-1}^{i-1}\|^2 \leq \|B_k - B_{k+1}\|^2 + \|B_{k+1} - B_k\|^2, \\
\|A_{k-1}^{i-1} - A_k^{i-1}\|^2 \leq \|A_k - A_{k-1}\|^2 + \|A_{k+1} - A_k\|^2.
\end{aligned}
$$

The proof is completed. □

From the above analysis, we know that b_i^k is updated with an exact solution of the subproblem $\min_{x\geq 0} \|Y - A_k(B_k) + a_i^k(b_i^k)^T - a_i^k x^T\|^2$. However, since the normalization step, the updated a_i^{k+1} is not the exact solution to the problem $\min_{x\geq 0} \|Y - A_k B_{k+1}^T + a_i^k(b_i^{k+1})^T - x(b_i^{k+1})\|^2$. Therefore, we propose a scaling step, making a_i^{i+1} be the exact optimal solution of a subproblem.

The scaled algorithm is as follows.

Scaled Fast HALS

Initialize nonnegative matrix A and B using ALS;

Normalize the vectors a_j to unit ℓ_2-norm length;

repeat

  $W = Y^T A$;

  $V = A^T A$;

  for $j = 1$ to J **do**

    $b_j = [b_j + w_j - Bv_j]_+$;

    $\overline{b}_j = b_j / \|b_j\|$;

    $a_j = a_j \|b_j\|$;

  end for

  $B = \overline{B}$;

  $P = YB$;

  $Q = B^T B$;

  for $j = 1$ to J **do**

    $a_j = [a_j q_{jj} + p_j - Aq_j]_+$;

    $b_j = b_j \|a_j\|$;

    $\overline{a}_j = a_j / \|b_j\|$;

  end for

  $A = \overline{A}$;

until convergence criterion is reached

Here $\overline{A} = [\overline{a}_1, \cdots, \overline{a}_J]$ and $\overline{B} = [\bar{b}_1, \cdots, \bar{b}_J]$. We perform a numerical test in the following. Let $I = K = 600, J = 60$, Y is randomly generated that

$$Y_1 = 100 * randn(I, J), Y_2 = 100 * randn(K, J), Y = Y_1 Y_2^T + randn(I, K).$$

The convergence performance of **Scaled Fast HALS** and **Fast HALS** for 100 iterations is presented. The numerical result is an average of 50 tests.

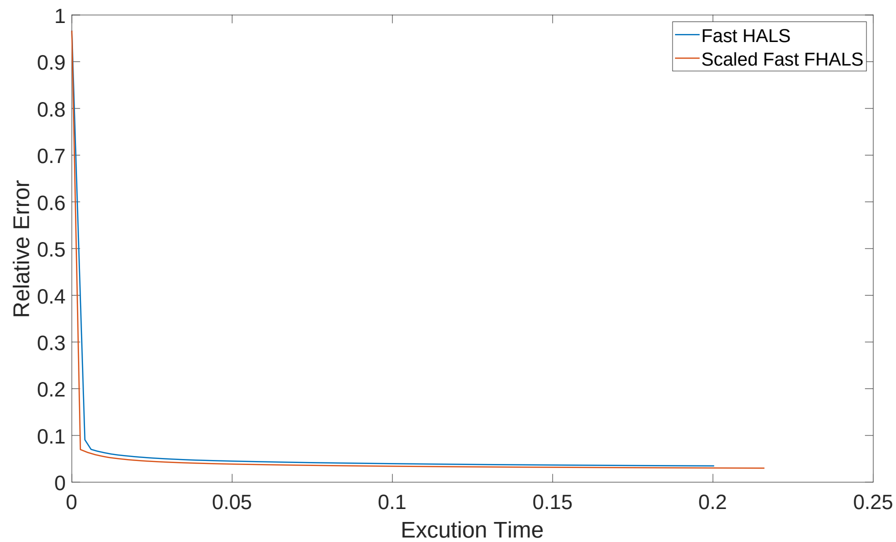

Figure 4.1: Comparison for **Fast HALS** and **Scaled Fast HALS**

In the figure, the curve of **Scaled Fast HALS** is longer than **Fast HALS** means the **Scaled Fast HALS** cost more time for 100 iterations. And, the curve of **Scaled Fast HALS** is under the curve of **HALS** tells **Scaled Fast HALS** performs better if given the same execution time.

We can conclude that: (1) A single iteration of **Scaled Fast HALS** costs more time than **Fast HALS**; (2) For the same execution time, **Scaled Fast HALS** performs slightly better than **Fast HALS**.

Bibliography

[1] N. Andrei, *Predictor-corrector interior-point methods for linear constrained optimization*, Stud. Inform. Control., 7 (1998), pp. 155–177.

[2] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, *Algorithms and applications for approximate nonnegative matrix factorization*, Comput. Statist. Data Anal., 52 (2007), pp. 155–173.

[3] D. P. Bertsekas, W. Hager, and O. Mangasarian, *Nonlinear programming*, Athena Scientific Belmont, MA, 1998.

[4] E. H. Bowman, *Production scheduling by the transportation method of linear programming*, Oper. Res., 4 (1956), pp. 100–103.

[5] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, *Metagenes and molecular pattern discovery using matrix factorization*, Proc. Nat. Acad. sciences, 101 (2004), pp. 4164–4169.

[6] J. V. Burke and S. Xu, *The global linear convergence of a noninterior path-following algorithm for linear complementarity problems*, Math. Oper. Res., 23 (1998), pp. 719–734.

[7] ——, *A non-interior predictor-corrector path following algorithm for the monotone linear complementarity problem*, Math. Program., 87 (2000), pp. 113–130.

[8] ——, *Complexity of a noninterior path-following method for the linear complementarity problem*, J. Optim. Theory Appl., 112 (2002), pp. 53–76.

[9] B. Chen and X. Chen, *A global linear and local quadratic continuation smoothing method for variational inequalities with box constraints*, Comput. Optim. Appl., 17 (2000), pp. 131–158.

[10] B. Chen and P. T. Harker, *A non-interior-point continuation method for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1168–1190.

[11] B. Chen and N. Xiu, *A global linear and local quadratic noninterior continuation method for nonlinear complementarity problems based on Chen-Mangasarian smoothing functions*, SIAM J. Optim., 9 (1999), pp. 605–623.

[12] X. Chen and P. Tseng, *Non-interior continuation methods for solving semidefinite complementarity problems*, Math. Program., 95 (2003), pp. 431–474.

[13] M. Chu, F. Diele, R. Plemmons, and S. Ragni, *Optimality, computation, and interpretation of nonnegative matrix factorizations*, in SIAM J. Matrix Anal., Citeseer, 2004.

[14] A. Cichocki and A.-H. Phan, *Fast local algorithms for large scale nonnegative matrix and tensor factorizations*, IEICE Trans. Electron., 92 (2009), pp. 708–721.

[15] A. Cichocki, R. Zdunek, and S.-i. Amari, *Hierarchical ALS algorithms for nonnegative matrix and 3d tensor factorization*, in International Conference on Independent Component Analysis and Signal Separation, Springer, 2007, pp. 169–176.

[16] G. B. Dantzig, *Linear programming and extensions*, Princeton Landmarks in Mathematics, Princeton University Press, Princeton, NJ, corrected ed., 1998.

[17] D. den Hertog, *Interior point approach to linear, quadratic and convex programming*, vol. 277 of Mathematics and its Applications, Kluwer Academic Publishers Group, Dordrecht, 1994. Algorithms and complexity.

[18] D. den Hertog, C. Roos, and T. Terlaky, *On the classical logarithmic barrier function method for a class of smooth convex programming problems*, J. Optim. Theory Appl., 73 (1992), pp. 1–25.

[19] I. I. Dikin, *Iterative solution of problems of linear and quadratic programming*, Dokl. Akad. Nauk SSSR, 174 (1967), pp. 747–748.

[20] R. Färe, F. R. Førsund, S. Grosskopf, K. Hayes, and A. Heshmati, *A note on decomposing the Malmquist productivity index by means of subvector homotheticity*, Econom. Theory, 17 (2001), pp. 239–245.

[21] P. E. Gill, W. Murray, M. A. Saunders, J. A. Tomlin, and M. H. Wright, *On projected Newton barrier methods for linear programming and an equivalence to Karmarkar's projective method*, Math. Program., 36 (1986), pp. 183–209.

[22] N. Gillis and F. Glineur, *Nonnegative factorization and the maximum edge biclique problem*, arXiv preprint arXiv:0810.4225, (2008).

[23] J. Gondzio, *Interior point methods 25 years later*, European J. Oper. Res., 218 (2012), pp. 587–601.

[24] E. F. Gonzalez and Y. Zhang, *Accelerating the Lee-Seung algorithm for nonnegative matrix factorization*, Tech. Rep., (2005). Houston, TX: Department of Computational and Applied Mathematics, Rice University.

[25] J. Gorski, F. Pfeuffer, and K. Klamroth, *Biconvex sets and optimization with biconvex functions: a survey and extensions*, Math. Methods Oper. Res., 66 (2007), pp. 373–407.

[26] C. Grossmann, *Asymptotic analysis of a path-following barrier method for linearly constrained convex problems*, Optimization, 45 (1999), pp. 69–87.

[27] C. Grossmann, *Penalty/Barrier path-following in linearly constrained optimization*, Discuss. Math. Dier. Incl. Control Optim., 20 (2000), pp. 7–26.

[28] F. Jarre, *On the method of analytic centers for solving smooth convex programs*, in Optimization (Varetz, 1988), vol. 1405 of Lecture Notes in Math., Springer, Berlin, 1989, pp. 69–85.

[29] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 851–868.

[30] N. Karmarkar, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.

[31] L. G. Khachiyan, *A polynomial algorithm in linear programming*, Yingyong Shuxue yu Jisuan Shuxue, (1980), pp. 1–3.

[32] H. Kim and H. Park, *Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 713–730.

[33] J. Kim, Y. He, and H. Park, *Algorithms for nonnegative matrix and tensor factorizations: a unified view based on block coordinate descent framework*, J. Global Optim., 58 (2014), pp. 285–319.

[34] V. Klee and G. J. Minty, *How good is the simplex algorithm?*, in Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif., 1969; dedicated to the memory of Theodore S. Motzkin), 1972, pp. 159–175.

[35] M. Kojima, N. Megiddo, and S. Mizuno, *A primal-dual infeasible-interior-point algorithm for linear programming*, Math. Program., 61 (1993), pp. 263–280.

[36] ——, *A primal-dual infeasible-interior-point algorithm for linear programming*, Math. Program., 61 (1993), pp. 263–280.

[37] M. Kojima, N. Megiddo, and T. Noma, *Homotopy continuation methods for nonlinear complementarity problems*, Math. Oper. Res., 16 (1991), pp. 754–774.

[38] M. Kojima, S. Mizuno, and A. Yoshise, *A polynomial-time algorithm for a class of linear complementarity problems*, Math. Program., 44 (1989), pp. 1–26.

[39] ——, *An $O(\sqrt{n}L)$ iteration potential reduction algorithm for linear complementarity problems*, Math. Program., 50 (1991), pp. 331–342.

[40] K. O. Kortanek, F. Potra, and Y. Ye, *On some efficient interior point methods for nonlinear convex programming*, Linear Algebra Appl., 152 (1991), pp. 169–189.

[41] K. O. Kortanek and J. Zhu, *A polynomial barrier algorithm for linearly constrained convex programming problems*, Math. Oper. Res., 18 (1993), pp. 116–127.

[42] ——, *On controlling the parameter in the logarithmic barrier term for convex programming problems*, J. Optim. Theory Appl., 84 (1995), pp. 117–143.

[43] D. D. Lee and H. S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, 401 (1999), pp. 788–791.

[44] ——, *Algorithms for non-negative matrix factorization*, in Advances in neural information processing systems, 2001, pp. 556–562.

[45] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng, *Learning spatially localized, parts-based representation*, in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, vol. 1, IEEE, 2001, pp. 207–212.

[46] C.-J. Lin, *On the convergence of multiplicative update algorithms for nonnegative matrix factorization*, IEEE Trans. Neural Netw., 18 (2007), pp. 1589–1596.

[47] ——, *Projected gradient methods for nonnegative matrix factorization*, Neural Comput., 19 (2007), pp. 2756–2779.

[48] J. López and H. Ramírez C, *On the central paths and cauchy trajectories in semidefinite programming*, Kybernetika, 46 (2010), pp. 524–535.

[49] C. K. Lovell, S. Grosskopf, E. Ley, J. T. Pastor, D. Prior, and P. V. Eeckaut, *Linear programming approaches to the measurement and analysis of productive efficiency*, Top, 2 (1994), pp. 175–248.

[50] I. J. Lustig, R. E. Marsten, and D. F. Shanno, *On implementing Mehrotra's predictor-corrector interior-point method for linear programming*, SIAM J. Optim., 2 (1992), pp. 435–449.

[51] ——, *Interior point methods for linear programming: computational state of the art*, ORSA J. Comput., 6 (1994), pp. 1–14.

[52] N. Megiddo, *Pathways to the optimal set in linear programming*, in Progress in mathematical programming (Pacific Grove, CA, 1987), Springer, New York, 1989, pp. 131–158.

[53] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

[54] S. Mehrotra and J. Sun, *An interior point algorithm for solving smooth convex programs based on Newton's method*, in Mathematical developments arising from linear programming (Brunswick, ME, 1988), vol. 114 of Contemp. Math., Amer. Math. Soc., Providence, RI, 1990, pp. 265–284.

[55] S. Mizuno, M. J. Todd, and Y. Ye, *On adaptive-step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.

[56] R. D. C. Monteiro, *On the continuous trajectories for a potential reduction algorithm for linear programming*, Math. Oper. Res., 17 (1992), pp. 225–253.

[57] ——, *A globally convergent primal-dual interior point algorithm for convex programming*, Math. Program., 64 (1994), pp. 123–147.

[58] R. D. C. Monteiro and I. Adler, *Interior path following primal-dual algorithms. Part I: Linear programming*, Math. Program., 44 (1989), pp. 27–41.

[59] ——, *Interior path following primal-dual algorithms. Part II: Convex quadratic programming*, Math. Program., 44 (1989), pp. 43–66.

[60] ——, *An extension of Karmarkar type algorithm to a class of convex separable programming problems with global linear rate of convergence*, Math. Oper. Res., 15 (1990), pp. 408–422.

[61] R. D. C. Monteiro, T. Tsuchiya, and Y. Wang, *A simplified global convergence proof of the affine scaling algorithm*, Ann. Oper. Res., 46/47 (1993), pp. 443–482.

[62] I. Necoara and J. A. K. Suykens, *Interior-point Lagrangian decomposition method for separable convex optimization*, J. Optim. Theory Appl., 143 (2009), pp. 567–588.

[63] Y. Nesterov and A. Nemirovskii, *Interior-point polynomial algorithms in convex programming*, vol. 13 of SIAM Studies in Applied Mathematics, SIAM, Philadelphia, PA, 1994.

[64] W. Ogryczak, *Multiple criteria linear programming model for portfolio selection*, Ann. Oper. Res., 97 (2000), pp. 143–162.

[65] P. Paatero, *The multilinear engine—a table-driven, least squares program for solving multilinear problems, including the n-way parallel factor analysis model*, J. Comput. Graph. Statist., 8 (1999), pp. 854–888.

[66] P. Paatero and U. Tapper, *Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics, 5 (1994), pp. 111–126.

[67] V. P. Pauca, F. Shahnaz, M. W. Berry, and R. J. Plemmons, *Text mining using non-negative matrix factorizations*, in Proceedings of the Fourth SIAM International Conference on Data Mining, SIAM, Philadelphia, PA, 2004, pp. 452–456.

[68] F. Potra and Y. Ye, *A quadratically convergent polynomial algorithm for solving entropy optimization problems*, SIAM J. Optim., 3 (1993), pp. 843–860.

[69] L. Qi and D. Sun, *Improving the convergence of non-interior point algorithms for nonlinear complementarity problems*, Math. Comp., 69 (2000), pp. 283–304.

[70] X. Qian, L.-Z. Liao, and J. Sun, *Analysis of some interior point continuous trajectories for convex programming*, Optimization, 66 (2017), pp. 589–608.

[71] R. L. Sheu, *A generalized interior-point barrier function approach for smooth convex programming with linear constraints*, J. Inform. Optim. Sci., 20 (1999), pp. 187–202.

[72] R. L. Sheu and S. C. Fang, *On the generalized path-following methods for linear programming*, Optimization, 30 (1994), pp. 235–249.

[73] Y. Shi, *A combination of potential reduction steps and steepest descent steps for solving convex programming problems*, Numer. Linear Algebra Appl., 9 (2002), pp. 195–203.

[74] ——, *A projected-steepest-descent potential-reduction algorithm for convex programming problems*, Numer. Linear Algebra Appl., 11 (2004), pp. 883–893.

[75] S. Sra and I. S. Dhillon, *Generalized nonnegative matrix approximations with bregman divergences*, in Advances in neural information processing systems, 2006, pp. 283–290.

[76] J. Sun, *A convergence analysis for a convex version of Dikin's algorithm*, Ann. Oper. Res., 62 (1996), pp. 357–374.

[77] J. Sun, J. Zhu, and G. Zhao, *A predictor-corrector algorithm for a class of nonlinear saddle point problems*, SIAM J. Control Optim., 35 (1997), pp. 532–551.

[78] M. J. Todd and Y. Ye, *A centered projective algorithm for linear programming*, Math. Oper. Res., 15 (1990), pp. 508–529.

[79] P. Tseng and Z.-Q. Luo, *On the convergence of the affine-scaling algorithm*, Math. Program., 56 (1992), pp. 301–319.

[80] S. A. Vavasis, *On the complexity of nonnegative matrix factorization*, SIAM J. Optim., 20 (2009), pp. 1364–1377.

[81] R. Villasana, L. Garver, and S. Salon, *Transmission network planning using linear programming*, IEEE Trans. Power Appar. Syst., (1985), pp. 349–356.

[82] P. Wolfe, *A duality theorem for non-linear programming*, Quart. Appl. Math., 19 (1961), pp. 239–244.

[83] S. Xu and J. V. Burke, *A polynomial time interior-point path-following algorithm for LCP based on Chen-Harker-Kanzow smoothing techniques*, Math. Program., 86 (1999), pp. 91–103.

[84] W. Xu, X. Liu, and Y. Gong, *Document clustering based on non-negative matrix factorization*, in Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, 2003, pp. 267–273.

[85] Y. Ye, *An $O(n^3L)$ potential reduction algorithm for linear programming*, Math. Program., 50 (1991), pp. 239–258.

[86] ——, *Interior point algorithms: theory and analysis*, vol. 44, John Wiley & Sons, New York, 2011.

[87] Y. Ye and K. Anstreicher, *On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP*, Math. Program., 62 (1993), pp. 537–551.

[88] Y. Ye, O. Güler, R. A. Tapia, and Y. Zhang, *A quadratically convergent $O(\sqrt{n}L)$-iteration algorithm for linear programming*, Math. Program., 59 (1993), pp. 151–162.

[89] M. R. Young, *A minimax portfolio selection rule with linear programming solution*, Manag. Sci., 44 (1998), pp. 673–683.

[90] J. Zhu, *A path following algorithm for a class of convex programming problems*, Z. Oper. Res., 36 (1992), pp. 359–377.

CURRICULUM VITAE

Academic qualifications of the thesis author, Mr. HOU Liangshao:

- Received the degree of Bachelor of Science from Zhejiang University of Technology, June 2014.
- Received the degree of Master of Science from Shanghai University of Finance and Economics, January 2017.

July 2020