High-Order Large-Step Interior Point Algorithms

For Linear Complementarity Problems

Gongyun Zhao

Department of Mathematics
National University of Singapore
10 Kent Ridge Crescent
Singapore 0511
Email: matzgy@leonis.nus.sg

October 1994

Abstract

In this paper we study a type of high-order large-step interior point algorithms for solving linear complementarity problems. These algorithms are implicitly associated with a large neighborhood whose size may depend on the dimension of the problems. The complexity of these algorithms depends on the size of the neighborhood. By using high-order power series (so the name high-order algorithm), the complexity can be reduced. We will show that the complexity upper bound for high-order large-step algorithms is asymptotically, as the order tends to infinity, equal to that for short-step algorithms.

Key Words: Interior point algorithm, High-order power series, Large-step, Complexity, Linear complementarity problem.

Abbreviated Title: High-order large-step interior point algorithms.

AMS(MOS) subject classifications: 90C05, 90C60, 49M35.

1. Introduction

Many works on large-step interior point methods have been appeared in the literature, cf. [2], [3], [4], [5], [7], [9], [10], [11], [12], [14], and [18]. Very recently, Zhao [20] presented and analysed a new type of large-step interior point algorithms (of first order). The present paper is a development of [20] in two aspects. First, the upper bound of the complexity for the first order large-step algorithm is improved. Secondly, results on high-order large-step algorithms are developed. It is well-known that the upper bound of the complexity for large-step algorithms is in general greater than that for short-step ones, (cf. [1], [13], [16], [17] and [21] for short-step algorithms). Using high-order approximation, the complexity upper bound for large-step algorithms can be reduced and usually it tends to the upper bound for short-step ones as the order tends to infinity. This idea was first presented by Monteiro et al. [11], and then also studied by Zhang and Zhang [18] and Hung and Ye [4]. In this paper we seek to improve the complexity upper bound by using high-order approximation in the large-step algorithms presented in [20].

Let $z = (x, s) \in R^{2n}$ be an interior point and $\mu > 0$ be a parameter. We denote $u = (xs/\mu)^{1/2}$ (the operation xs is defined in Section 2), and define the following functions:

$$\delta(z, \mu) = \|u - u^{-1}\|_2, \qquad \zeta(z, \mu) = \|u^{-1}\|_\infty.$$

The function $\delta(z, \mu)$ is called the *distance* from the point z to the point $z^c(\mu)$ on the central trajectory at μ. This definition of distance was first introduced by Jansen et al in [5], and also used in [6]. This distance is very suitable for large-step algorithms, since the distance $\delta(z, \mu) \to \infty$ if and only if z tends to the boundary of the feasible region. The function $\zeta(z, \mu)$ measures how far the components of u is bounded away from zero.

In what follows we describe a typical iteration of our first algorithm. Let $z^{[k]}$ and μ_k be the current feasible point and parameter respectively, with $\delta(z^{[k]}, \mu_k) = \beta n^\tau$. Here $\beta = O(1)$ and $\tau \geq 0$ are constants. We define a trajectory linking the current point $z^{[k]}$ and the target point $z^c(\mu_k)$. Along a high-order approximation (truncated power series) of the trajectory, we find the next point $z^{[k+1]}$ by minimizing the distance. So we have $\delta(z^{[k+1]}, \mu_k) < \delta(z^{[k]}, \mu_k)$. Then the parameter μ is reduced to μ_{k+1} such that $\delta(z^{[k+1]}, \mu_{k+1}) = \beta n^\tau$ again.

We study two algorithms which differ to each other in the way of finding $z^{[k+1]}$. The second algorithm (Algorithm 5.1) imposes a restriction on $z^{[k+1]}$ requiring $\zeta(z^{[k+1]}, \mu_k) \leq \zeta_*$ (some properly designated number), while the first algorithm (Algorithm 3.1) without this restriction. The first algorithm needs at most

$$O\Big(\min\{\zeta_u^{\frac{\nu+2}{\nu}} n^\omega,\, \zeta_u^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega}\} \ln(n\mu_0/\epsilon)\Big) \tag{1.1}$$

iterations starting from a parameter μ_0 to find a feasible point (x, s) with $\langle xs \rangle \leq \epsilon$. Here $\nu \geq 1$ is the order of the algorithm, $\omega = \max\{\tau, \frac{1}{2}\}$ and ζ_u is an upper bound of $\zeta(z^{[k]}, \mu_k)$ for $k = 0, 1, 2, \ldots$. The iteration complexity of the second algorithm is bounded by

$$O\Big(\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega} \ln(n\mu_0/\epsilon)\Big), \tag{1.2}$$

where $\zeta_* \geq 5$ can be designated to be any number which may depend on n.

The paper is organised as follows. In Section 2, we define notations which are used throughout this paper. In Section 3, we present the first high-order large-step algorithm while its complexity is estimated in Section 4. In Section 5, the second high-order large-step algorithm is presented and its complexity is analysed. In Section 6, we use another way to estimate the complexity of both algorithms. The final results are presented in Section 7.

2. Notations

Throughout this paper we use capital letters, English letters and Greek letters as matrices, column vectors and scalars, respectively. For any vector x, we denote by $[x]_i$ the i-th component of vector x. We denote $e = (1, \ldots, 1)^T$ regardless of dimension.

For any $x, s \in R^n$ and $\alpha \in R$ we define

$$xs = ([x]_1[s]_1, \ldots, [x]_n[s]_n)^T, \qquad x^\alpha = ([x]_1^\alpha, \ldots, [x]_n^\alpha)^T, \qquad |x| = (|[x]_1|, \ldots, |[x]_n|)^T.$$

$$\|x\|_2 = (\sum_{i=1}^n [x]_i^2)^{1/2}, \qquad \|x\|_\infty = \max_{1\leq i\leq n} |[x]_i|.$$

We extend the notation of inner product $\langle \cdot, \cdot \rangle$ to any number of vectors and omit comma. For example, for $a, b, c, d \in R^n$

$$\langle abcd \rangle = \sum_{i=1}^n [a]_i[b]_i[c]_i[d]_i.$$

Hence we have

$$\langle abcd \rangle = \langle a, bcd \rangle = \langle ab, cd \rangle = \langle e, abcd \rangle.$$

Especially, $\langle a \rangle = \langle e, a \rangle$.

We specify the following symbols:

$e = (1, \ldots, 1)^T$;

x, s: variables in Problems (LCP), or in System $(CT)_\mu$;

$\mu > 0$: the parameter in System $(CT)_\mu$;

$\nu \geq 1$: order of algorithm;

$z = (x, s)$;

$z^c(\mu) = (x^c(\mu), s^c(\mu))$: the central trajectory defined by $(CT)_\mu$;

$z(\alpha) = (x(\alpha), s(\alpha))$: the trajectory linking a point (x, s) to the point $(x^c(\mu), s^c(\mu))$;

$z_i = (x_i, s_i)$: derivatives of $z(\alpha)$ defined by (3.2) and (3.3);

$h_k = \mu^{-1} \sum_{i=k-\nu}^{\nu} x_i s_{k-i}$;

$u = (xs/\mu)^{1/2}$;

$u(\alpha) = \left(\frac{(\sum_{i=0}^{\nu} x_i \alpha^i)(\sum_{i=0}^{\nu} s_i \alpha^i)}{\mu} \right)^{1/2}$;

$w(\alpha), r(\alpha)$: defined by (4.4);
$\delta(z, \mu) = \|u - u^{-1}\|_2$: distance from z to $z(\mu)$, defined by (3.4);
$\delta = \delta(z, \mu)$;
$\delta(\alpha) = \|u(\alpha) - u(\alpha)^{-1}\|_2$;
$\zeta(z, \mu) = \|u^{-1}\|_\infty$;
$\zeta = \zeta(z, \mu)$;
$\zeta(\alpha) = \|u(\alpha)^{-1}\|_\infty$;
ζ_u: an upper bound of $\zeta(z^{[k]}, \mu_k)$ for $k = 0, 1, 2, \ldots$;
ζ_*: a designated number used in subproblem (5.2);
$\tilde{\alpha}$: a seleted step length defined by (4.8) or (6.1);
$\beta, \tau \geq 0$: $\delta(z, \mu) \leq \beta n^\tau$;
$\omega = \max\{1/2, \tau\}$.

3. The first algorithm

The purpose of this paper is to study a type of high-order large-step interior point algorithms for solving the following linear complementarity problem:

find $(x, s) \in R^{2n}$ satisfying

$$\begin{aligned} &s = Mx + b \\ &\langle xs \rangle = 0 \qquad\qquad (LCP) \\ &x, s \geq 0, \end{aligned}$$

where $M \in R^{n \times n}$ and $b \in R^n$ are given matrix and vector. For a survey of interior point methods for (LCP), the reader is referred to the excellent paper [8] by Kojima, Megiddo, Noma and Yoshise.

Throughout the paper, we make the following assumptions:

A 1: Matrix M is positive semi-definite, i.e., matrix M satisfies $\langle x(Mx) \rangle \geq 0$ for every $x \in R^n$. Note that skew-symmetric matrix, i.e., matrix M satisfying $\langle x(Mx) \rangle = 0$ for every $x \in R^n$, is included as a special case.

A 2: The feasible region of (LCP) has a nonempty interior, i.e., there exists $(x_0, s_0) > 0$ such that $s_0 = Mx_0 + b$.

We write $z = (x, s)$. A point z is said to be *strictly feasible* if it is feasible to (LCP) and $z > 0$.

By relaxing the complementary slackness $xs = 0$, we obtain the following system of equations

$$\begin{aligned} &xs = \mu e \\ &s = Mx + b \qquad\qquad (CT)_\mu \\ &x, s \geq 0, \end{aligned}$$

where $\mu > 0$ is a parameter. The solutions to $(CT)_\mu$ define a curve $z^c(\mu) = (x^c(\mu), s^c(\mu))$ with parameter μ, called *central trajectory*. For more details about the central trajectory, the reader may refer to [15] and [21].

For any $\mu > 0$ and any point (x_0, s_0), we will construct a trajectory which links the points (x_0, s_0) and $(x^c(\mu), s^c(\mu))$. Following this trajectory we can approximate the point $(x^c(\mu), s^c(\mu))$ from (x_0, s_0). For this purpose, we observe the following equation which defines the Newton direction (d_x, d_s) at point (x, s) aiming at the point $(x^c(\mu), s^c(\mu))$:

$$sd_x + xd_s = \mu e - xs.$$

A trajectory $(x(\bar{\alpha}), s(\bar{\alpha}))$ whose tangent constantly coincides with the Newton direction is then define by the following differential equation:

$$s\frac{dx}{d\bar{\alpha}} + x\frac{ds}{d\bar{\alpha}} = \mu e - xs.$$

The above equation can be rewritten as

$$\frac{d(xs)}{d\bar{\alpha}} = \mu e - xs.$$

The solution of this equation with the initial value $(x(0), s(0)) = (x_0, s_0)$ is given by

$$x(\bar{\alpha})s(\bar{\alpha}) = \mu e - (\mu e - x_0 s_0)\exp(-\bar{\alpha}).$$

Since $\lim_{\bar{\alpha}\to\infty} x(\bar{\alpha})s(\bar{\alpha}) = \mu e$, the trajectory defined by the above equation converges to the point $(x^c(\mu), s^c(\mu))$. Using the parameter transformation $\alpha = 1 - \exp(-\bar{\alpha})$, we get

$$x(\alpha)s(\alpha) = x_0 s_0 + (\mu e - x_0 s_0)\alpha. \tag{3.1}$$

Now we have $(x(0), s(0)) = (x_0, s_0)$ and $(x(1), s(1)) = (x^c(\mu), s^c(\mu))$. To determine the derivatives of the trajectory, we write $(x(\alpha), s(\alpha))$ as power series:

$$x(\alpha) = \sum_{i=0}^{\infty} x_i \alpha^i, \qquad s(\alpha) = \sum_{i=0}^{\infty} s_i \alpha^i.$$

Substituting the above expansions into (3.1), and comparing coefficients of each power of α, we obtain recursive formulas as follows:

$$\begin{aligned} s_0 x_1 + x_0 s_1 &= \mu e - x_0 s_0 \\ s_0 x_k + x_0 s_k &= -\sum_{i=1}^{k-1} x_i s_{k-i} \qquad \text{for } k = 2, 3, \ldots. \end{aligned} \tag{3.2}$$

For the feasibility, the following equations should also be satisfied.

$$s_i = Mx_i \qquad \text{for } i = 1, 2, \ldots. \tag{3.3}$$

The algorithms presented in this paper use the derivatives up to order ν determined by (3.2) and (3.3) to construct a truncated power series to approximate the point $(x^c(\mu), s^c(\mu))$.

For any strictly feasible point (x,s) and parameter $\mu > 0$, we denote vector

$$u = \left(\frac{xs}{\mu}\right)^{1/2}.$$

In order to measure the distance of a point $z=(x,s)$ from the point $z^c(\mu) = (x^c(\mu), s^c(\mu))$ on the central trajectory, we define a *distance* as follows:

$$\delta(z,\mu) = \|u - u^{-1}\|_2. \tag{3.4}$$

This distance was first introduced to interior point methods by Jansen et al. [5]. A significant feature of this distance is that it tends to infinity as some component $[x]_i[s]_i/\mu \to 0$ or $\to \infty$, that is, the distance tends to infinity if and only if the point (x,s) tends to the boundary of the feasible region.

Now we present the first algorithm in this paper.

Algorithm 3.1: Given $\epsilon > 0$, $\beta > 0$ and an integer $\nu \geq 1$. Let $z^{[0]} = (x^{[0]}, s^{[0]})$ be a strictly feasible solution of (LCP), and $\mu_0 \in (0, \langle x^{[0]}s^{[0]}\rangle/n)$ (e.g., $\mu_0 = 0.1\langle x^{[0]}s^{[0]}\rangle/n$) be a parameter which satisfy

$$\delta(z^{[0]}, \mu_0) \leq \beta\sqrt{n}.$$

Let $k=0$. Implement the following steps.

Step 1: Calculate derivatives z_i, $i=1,2,\ldots,\nu$ at $z^{[k]}$ and μ_k by (3.2) and (3.3). Set $z^{[k+1]} = z^{[k]} + \sum_{i=1}^{\nu} z_i \alpha_k^i$, where α_k minimizes $\delta(z^{[k]} + \sum_{i=1}^{\nu} z_i\alpha^i, \mu_k)$ for $\alpha \geq 0$.

Step 2: Determine μ_{k+1} as the smallest $\mu \in (0, \mu_k)$ which satisfies

$$\delta(z^{[k+1]}, \mu) \leq \beta\sqrt{n}. \tag{3.5}$$

If $\langle x^{[k+1]}s^{[k+1]}\rangle \leq \epsilon$, stop; $(x^{[k+1]}, s^{[k+1]})$ is an approximate solution. Otherwise, set $k = k+1$, and go to Step 1. #

Remark: As a point tends to the boundary of the feasible region, the distance tends to infinity. Hence the minimum α_k in Step 1 exists and point $z^{[k+1]}$ is strictly feasible. #

If all components of u and u^{-1} are of size $O(1)$, then $\delta(z,\mu) \leq \beta\sqrt{n}$ for some β. But $\delta(z,\mu) \leq \beta\sqrt{n}$ also allows some components of u and u^{-1} to be as large as $O(\sqrt{n})$. So the neighborhood which is implicitly imposed in Algorithm 3.1 may be larger than the generally used neighborhood

$$\mathcal{N}_\infty^-(\beta) = \{(x,s) \mid xs \geq \beta^{-1}n^{-1}\langle xs\rangle e\},$$

(cf. [4], [7] and [10]).

4. Complexity of the first algorithm

The estimation of the reduction of $\delta(z^{[k]}, \mu_k) - \delta(z^{[k+1]}, \mu_k)$ is the main part in the process of the complexity analysis of the algorithm. For the purpose of presenting a polynomial upper bound on the complexity, it is not necessary to find the optimal step length α_k minimizing $\delta(z^{[k]} + \sum_{i=1}^{\nu} z_i\alpha^i, \mu_k)$, because α_k depends on the data of the problem and its analytic representation may be very complicated. What we should do is to select a suitable step length to establish a lower bound on the reduction $\delta(z^{[k]}, \mu_k) - \delta(z^{[k+1]}, \mu_k)$.

In the complexity analysis, we only consider a typical iteration. That is, the index k for the iteration is fixed and the variation of every quantity only depends on the step length α. Hence in what follows, we omit the subscript k and write every quantity as a function of α. For example:

$$u(\alpha) = \Big(\frac{(\sum_{i=0}^{\nu} x_i\alpha^i)(\sum_{i=0}^{\nu} s_i\alpha^i)}{\mu}\Big)^{1/2}, \tag{4.1}$$

where (x_i, s_i), $i = 1, 2, \ldots, \nu$, are the derivatives at $(x_0, s_0) = (x, s)$ defined by (3.2) and (3.3),

$$\delta(\alpha) = \|u(\alpha) - u(\alpha)^{-1}\|_2, \qquad \text{and} \qquad \zeta(\alpha) = \|u(\alpha)^{-1}\|_\infty. \tag{4.2}$$

With these notations, we have

$$\delta(0) = \delta(z^{[k]}, \mu), \qquad \delta(\alpha_k) = \delta(z^{[k+1]}, \mu),$$

and for any $\alpha \geq 0$,

$$\delta(\alpha_k) \leq \delta(\alpha).$$

In order to estimate how small $\delta(\alpha_k)$ is, we will selete a suitable $\tilde{\alpha}$ and find an upper bound of $\delta(\tilde{\alpha})$ in the form $\delta(0) - \sigma$.

We denote

$$h_k = \mu^{-1} \sum_{i=k-\nu}^{\nu} x_i s_{k-i}.$$

Using (3.2) we can write

$$u(\alpha)^2 = w(\alpha)^2 + r(\alpha), \tag{4.3}$$

where

$$w(\alpha)^2 = u^2 - (u^2 - e)\alpha, \quad \text{and} \quad r(\alpha) = \sum_{k=\nu+1}^{2\nu} h_k\alpha^k. \tag{4.4}$$

In what follows, $|x|$ denotes the vector whose components are the absolute value of components of a vector x.

Lemma 4.1: *For any $\alpha \in [0,1]$ such that $|w(\alpha)^{-2}r(\alpha)| < e$, the following inequality holds true.*

$$|u(\alpha) - u(\alpha)^{-1}| \leq |w(\alpha) - w(\alpha)^{-1}| + \Big(w(\alpha)^{-1} + w(\alpha)^{-3}\Big)|r(\alpha)|.$$

Proof: For any $\lambda \in R$ with $|\lambda| < 1$, one can verify that

$$|(1+\lambda)^{1/2} - 1| \le |\lambda|, \qquad \text{and} \qquad |(1+\lambda)^{-1/2} - 1| \le |\lambda|.$$

So we have

$$\begin{aligned} |u(\alpha) - u(\alpha)^{-1}| &\le |w(\alpha) - w(\alpha)^{-1}| + w(\alpha)|(e + w(\alpha)^{-2} r(\alpha))^{1/2} - e| \\ &\quad + w(\alpha)^{-1}|(e + w(\alpha)^{-2} r(\alpha))^{-1/2} - e| \\ &\le |w(\alpha) - w(\alpha)^{-1}| + (w(\alpha)^{-1} + w(\alpha)^{-3})|r(\alpha)|. \qquad \# \end{aligned}$$

Let $y \in R^n$ be any vector and $N \subset \{1, 2, \ldots, n\}$ be any index set, we denote by y_N the subvector consisting of those components of y whose indices belong to N.

Lemma 4.2: *For any index set $N \subset \{1, 2, \ldots, n\}$ and any $\alpha \in [0, 1]$, the following inequality holds true:*

$$\|(w(\alpha) - w(\alpha)^{-1})_N\|_2 \le \left(1 - \frac{\alpha}{2}\right) \|(u - u^{-1})_N\|_2.$$

Proof: One can verify that the function

$$\xi(\alpha) = \|(w(\alpha) - w(\alpha)^{-1})_N\|_2^2$$

is convex, and

$$\xi(0) = \|(u - u^{-1})_N\|_2^2, \qquad \xi(1) = 0.$$

This implies that for any $\alpha \in [0, 1]$,

$$\xi(\alpha) \le \xi(0)(1 - \alpha) + \xi(1)\alpha = \|(u - u^{-1})_N\|_2^2 (1 - \alpha).$$

So we have

$$\|(w(\alpha) - w(\alpha)^{-1})_N\|_2 \le (1 - \alpha)^{1/2} \|(u - u^{-1})_N\|_2 \le \left(1 - \frac{\alpha}{2}\right) \|(u - u^{-1})_N\|_2. \qquad \#$$

To estimate $r(\alpha)$, we should first estimate the derivatives (x_i, s_i) of the trajectory from a point (x, s) to the point $(x^c(\mu), s^c(\mu))$. Notice that now $(x_0, s_0) = (x, s)$. We denote

$$D = \text{ diag } (x^{-1/2} s^{1/2}), \qquad u = \left(\frac{xs}{\mu}\right)^{1/2}.$$

Then (3.2) can be rewritten as

$$\begin{aligned} Dx_1 + D^{-1} s_1 &= -\mu^{1/2}(u - u^{-1}) \\ Dx_k + D^{-1} s_k &= -\mu^{-1/2} u^{-1} \sum_{i=1}^{k-1} x_i s_{k-i} \qquad \text{for } k = 2, 3, \ldots. \end{aligned} \tag{4.5}$$

Denote

$$\eta_k = \|Dx_k + D^{-1}s_k\|_2, \qquad \zeta = \|u^{-1}\|_\infty, \qquad \delta = \|u - u^{-1}\|_2.$$

Throughout this paper we assume $\zeta \geq 1$. Otherwise, ζ is assumed to be replaced by $\max\{\zeta, 1\}$. This replacement does not affect the complexity upper bounds.

Because

$$\langle Dx_k D^{-1}s_k\rangle = \langle x_k s_k\rangle = \langle x_k(Mx_k)\rangle \geq 0, \tag{4.6}$$

we have

$$\max\{\|Dx_k\|_2, \|D^{-1}s_k\|_2\} \leq \eta_k.$$

The lemma below gives an upper bound on η_k, (also see [4] and [19]).

Lemma 4.3:

$$\max\{\|Dx_k\|_2, \|D^{-1}s_k\|_2\} \leq \eta_k \leq \theta_k \mu^{1/2}\zeta^{k-1}\delta^k.$$

where

$$\theta_k = \frac{1}{k}\binom{2k-2}{k-1} \leq \frac{1}{k}4^k$$

is the well-known solution to the following equations:

$$\theta_1 = 1, \qquad\qquad \theta_k = \sum_{i=1}^{k-1} \theta_i\theta_{k-i}, \qquad k = 2, 3, \ldots.$$

Proof: We use the induction to prove the inequality $\eta_k \leq \theta_k\mu^{1/2}\zeta^{k-1}\delta^k$. Obviously, it holds true for $k = 1$. Now for $k \geq 2$ suppose it holds for all $1 \leq i < k$, then from (4.5) we have

$$\begin{aligned}
\eta_k &\leq \mu^{-1/2}\zeta\sum_{i=1}^{k-1}\eta_i\eta_{k-i} \\
&\leq \mu^{-1/2}\zeta\sum_{i=1}^{k-1}\theta_i\theta_{k-i}\mu\zeta^{k-2}\delta^k \\
&= \theta_k\mu^{1/2}\zeta^{k-1}\delta^k.
\end{aligned}$$

This shows that the inequality also holds true for k. The proof is completed. #

Using Lemma 4.3 we can estimate h_k for $k = \nu+1, \nu+2, \ldots, 2\nu$

$$\begin{aligned}
\|h_k\|_2 &\leq \mu^{-1}\sum_{i=k-\nu}^{\nu}\eta_i\eta_{k-i} \\
&\leq \zeta^{k-2}\delta^k\sum_{i=k-\nu}^{\nu}\theta_i\theta_{k-i} \\
&\leq \theta_k\zeta^{k-2}\delta^k \\
&\leq \frac{(4\zeta\delta)^k}{k\zeta^2}.
\end{aligned} \tag{4.7}$$

For the complexity analysis (but not for implementation of the algorithm) we make the following choice of step length $\tilde{\alpha}$.

$$\tilde{\alpha} = \frac{1}{4^{\frac{2\nu+1}{\nu}}\zeta^{\frac{\nu+2}{\nu}}\delta}, \qquad \text{for } \nu \geq 1. \tag{4.8}$$

Now we estimate some quantities related to this step length.

Lemma 4.4: *For the vector $r(\alpha)$ defined by (4.4) and the step length $\tilde{\alpha}$ defined by (4.8), we have*

$$\|r(\tilde{\alpha})\|_2 \leq \frac{\delta\tilde{\alpha}}{8\zeta^3}.$$

Proof: Using (4.7) we have

$$\|r(\tilde{\alpha})\|_2 \leq \sum_{k=\nu+1}^{2\nu} \|h_k\|_2 \tilde{\alpha}^k \leq \frac{1}{(\nu+1)\zeta^2} \sum_{k=\nu+1}^{2\nu} (4\zeta\delta\tilde{\alpha})^k.$$

For $\nu = 1$, we have by (4.8)

$$\|r(\tilde{\alpha})\|_2 \leq \frac{1}{2\zeta^2}(4\zeta\delta\tilde{\alpha})^2 = \frac{\delta\tilde{\alpha}}{8\zeta^3}.$$

For $\nu \geq 2$,

$$\|r(\tilde{\alpha})\|_2 \leq \frac{(4\zeta\delta\tilde{\alpha})^{\nu+1}}{(\nu+1)\zeta^2(1-4\zeta\delta\tilde{\alpha})} = \frac{(\frac{1}{4})^\nu \delta\tilde{\alpha}}{(\nu+1)\zeta^3(1-4\zeta\delta\tilde{\alpha})} \leq \frac{(\frac{1}{4})^2\delta\tilde{\alpha}}{3\zeta^3(1-\frac{1}{4})} \leq \frac{\delta\tilde{\alpha}}{8\zeta^3}. \qquad \#$$

We are ready to estimate an upper bound on $\delta(\tilde{\alpha})$.

Lemma 4.5: *Let step length $\tilde{\alpha}$ be defined by (4.8). Then the following inequality holds:*

$$\delta(\tilde{\alpha}) \leq \delta - \frac{\tilde{\alpha}}{4}\delta.$$

Proof: By Lemma 4.1 we have

$$\delta(\tilde{\alpha}) \leq \|w(\tilde{\alpha}) - w(\tilde{\alpha})^{-1}\|_2 + \|w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3}\|_\infty \|r(\tilde{\alpha})\|_2.$$

By the definition of $w(\tilde{\alpha})$ in (4.4) and because function $\xi(\lambda) = \lambda(1-\alpha+\lambda^2\alpha)^{-1/2}$ is monotonely increasing for any fixed $\alpha \in [0,1]$, we have

$$\begin{aligned} \|w(\alpha)^{-1}\|_\infty &= \|u^{-1}(e - e\alpha + u^{-2}\alpha)^{-1/2}\|_\infty \\ &= \zeta(1-\alpha+\zeta^2\alpha)^{-1/2} \\ &\leq \zeta. \end{aligned} \tag{4.9}$$

It follows

$$\|w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3}\|_\infty \le 2\zeta^3.$$

Then by Lemmas 4.2 and 4.4, we obtain

$$\delta(\tilde{\alpha}) \le \Big(1 - \frac{\tilde{\alpha}}{2}\Big)\delta + \frac{1}{4}\delta\tilde{\alpha} = \delta - \frac{\tilde{\alpha}}{4}\delta. \qquad \#$$

By the above lemma and using Lemma 3.3 in [20], we can show that at each iteration in Step 2 of Algorithm 3.1 the parameter μ can be reduced at least as much as

$$\mu_{k+1} \le (1-\rho)\mu_k, \qquad \text{where } \rho = \frac{\delta\tilde{\alpha}}{4(\beta n^\tau + \sqrt{n})}. \tag{4.10}$$

Also in [20], a relation between $\langle xs\rangle$ and μn was shown as follows:

$$\langle xs\rangle \le (\beta^2 n^{2\tau} + \beta n^{\tau+1/2} + n)\mu, \tag{4.11}$$

if $\delta = \delta(x, s, \mu) \le \beta n^\tau$. Using (4.10) and (4.11), we obtain a complexity estimate as follows. (The proof is referred to Theorem 3.4 in [20].)

Theorem 4.6: *Let $\beta = O(1)$, and let $\zeta_u \ge 1$ be a constant such that*

$$\zeta_k := \|(u^{[k]})^{-1}\|_\infty \le \zeta_u \qquad \textit{for all } k = 0, 1, 2, \ldots,$$

where $u^{[k]} = (x^{[k]}s^{[k]}/\mu_k)^{1/2}$. Then Algorithm 3.1 needs at most $O(\zeta_u^{\frac{\nu+2}{\nu}} n^\omega \ln(n\mu_0/\epsilon))$ iterations to find a feasible point (x, s) with $\langle xs\rangle \le \epsilon$, where $\omega = \max\{1/2, \tau\}$. #

Remarks: The following are more details about Theorem 4.6.

i) First let us observe the power of n in the complexity upper bound (omitting the others) which is $O(n^\tau)$ for $\tau \ge 1/2$. This shows that the complexity is a polynomial of n with its degree linearly depending on the radius of the neighborhood, βn^τ. As $\tau \to \infty$, the complexity tends to an exponential one. So we can imagine that a large-step algorithm as $\tau \to \infty$ should behave as the simplex method whose worst-case complexity is known to be exponential. The observation on numerical and theoretical results seems to support this conjecture.

ii) In what follows, we consider the case $\tau = 1/2$. The upper bound of the iteration complexity (omitting logarithmic term) is then

$$O(\zeta_u^{\frac{\nu+2}{\nu}} n^{1/2}). \tag{4.12}$$

As the order of the algorithm, ν, grows, this upper bound asymptotically equals to

$$O(\zeta_u n^{1/2}).$$

It is worth noting that as long as $\nu \in [1, n]$, the arithmatic operations needed at each iteration will not exceed $O(n^3)$, since we need only one matrix factorization (costing $O(n^3)$)

at each iteration and one more order derivative consequently adds only $O(n^2)$ arithmatic operations.

iii) The number ζ_u is either $O(1)$ or $O(n^\gamma)$ with some $\gamma > 0$. In the former case, we achieve an upper bound $O(n^{1/2})$ by (4.12). On the other hand, the latter case implies that some points (x_k, s_k) generated by Algorithm 3.1 should go beyond the neighborhood $\mathcal{N}_\infty^-(\beta)$, which has been used for large-step algorithms in, e.g., [4], [7] and [10].

iv) In general, ζ_u may lie between $O(1)$ and $O(n^\tau)$, since $\|u_k^{-1}\|_\infty \le \delta(z_k, r_k) + 1 \le \beta n^\tau + 1$. Hence the complexity is of magnitude between

$$O(n^\omega \ln(n\mu_0/\epsilon)) \qquad \text{and} \qquad O\big(n^{\frac{(\nu+2)\tau}{\nu}+\omega} \ln(n\mu_0/\epsilon)\big).$$

But, how large ζ_u is exactly is still an open problem.

5. The second algorithm and its complexity

For Algorithm 3.1, Theorem 4.6 provides an upper bound $O(\zeta_u^{\frac{\nu+2}{\nu}} n^\omega \ln(n\mu_0/\epsilon))$ on the iteration complexity. This upper bound depends on ζ_u which is still unknown for us. In what follows we will present another algorithm which differs slightly from Algorithm 3.1 by adding a safeguard in (5.2) on the determination of step length. We will show that all major large-step features of the first algorithm are retained. If we designate $\zeta_* = O(n^{\tau/3})$ in the following algorithm, its iteration complexity can be shown to be bounded by $O(n^{\frac{(\nu+2)\tau}{3\nu}+\omega} \ln(n\mu_0/\epsilon))$. We denote

$$\zeta(z,\mu) = \|u^{-1}\|_\infty = \left\| \left(\frac{xs}{\mu}\right)^{-1/2} \right\|_\infty . \tag{5.1}$$

Algorithm 5.1: Given $\epsilon > 0$, $\beta > 0$ and $\zeta_* > 0$. Given a strictly feasible solution $z^{[0]}$ and a parameter $\mu_0 \in (0, \langle x^{[0]} s^{[0]}\rangle/n)$ (e.g., $\mu_0 = 0.1\langle x^{[0]} s^{[0]}\rangle/n$) satisfying $\delta(z^{[0]}, \mu_0) \le \beta n^\tau$ and $\zeta(z^{[0]}, \mu_0) \le \zeta^*$.

Let $k = 0$. Implement the following steps.

Step 1: Calculate z_i, $i = 1, 2, \ldots, \nu$ at $z^{[k]}$ and μ_k by (3.2) and (3.3). Set $z^{[k+1]} = z^{[k]} + \sum_{i=1}^{\nu} z_i \alpha_k^i$, where α_k is a solution to the following subproblem:

$$\begin{aligned} &\min\ \delta(z^{[k]} + \sum_{i=1}^{\nu} z_i \alpha^i, \mu_k) \\ &\text{s. t. } \zeta(z^{[k]} + \sum_{i=1}^{\nu} z_i \alpha^i, \mu_k) \le \zeta_* \\ &\qquad \alpha \ge 0. \end{aligned} \tag{5.2}$$

Step 2: Determine μ_{k+1} as the smallest $\mu \in (0, \mu_k)$ which satisfies

$$\delta(z_{k+1}, \mu) \le \beta n^\tau. \tag{5.3}$$

If $\langle x_{k+1} s_{k+1} \rangle \leq \epsilon$, stop; z_{k+1} is an approximate solution. Otherwise, set $k = k+1$, and go to Step 1. #

Remark: By (5.1) we see that for any $0 < \mu' \leq \mu$, $\zeta(z, \mu') \leq \zeta(z, \mu)$. Since $\zeta(z_{k+1}, \mu_k) \leq \zeta_*$ holds as required by the constraint of (5.2) and $\mu_{k+1} < \mu_k$, we have $\zeta(z_{k+1}, \mu_{k+1}) \leq \zeta_*$ for all $k = 0, 1, 2, \ldots$. #

To analyse the complexity of Algorithm 5.1, we first show that the step length chosen as (4.8) satisfies the constraints in subproblem (5.2).

Lemma 5.1: *Let $\zeta_* \geq \max\{\delta^{1/3}, 10\}$ be any number. If $\zeta = \zeta(z, \mu) \leq \zeta_*$, then for $\tilde{\alpha}$ defined as (4.8), we have*

$$\zeta(\tilde{\alpha}) = \zeta(z + \sum_{i=1}^{\nu} z_i \tilde{\alpha}^i, \mu) \leq \zeta_*.$$

Proof: Similar to Lemma 4.1, we have

$$u(\tilde{\alpha})^{-1} \leq w(\tilde{\alpha})^{-1} + w(\tilde{\alpha})^{-3} |r(\tilde{\alpha})|. \tag{5.4}$$

Using (4.9) and Lemma 4.4, we obtain

$$\zeta(\tilde{\alpha}) \leq \zeta(1 - \tilde{\alpha} + \zeta^2 \tilde{\alpha})^{-1/2} + \frac{\delta \tilde{\alpha}}{8}. \tag{5.5}$$

If $\delta \leq 8$, then $\zeta \leq 9$ and

$$\zeta(\tilde{\alpha}) \leq \zeta + \frac{\delta \tilde{\alpha}}{8} \leq 10 \leq \zeta_*.$$

Otherwise, $\delta \geq 8$, then one can verify that $(\zeta^2 - 1)\tilde{\alpha}$ is small enough such that

$$\Big(1 + (\zeta^2 - 1)\tilde{\alpha}\Big)^{-1/2} \leq 1 - \frac{2}{5}(\zeta^2 - 1)\tilde{\alpha}. \tag{5.6}$$

Hence,

$$\zeta(\tilde{\alpha}) \leq \zeta - \frac{2}{5}\zeta(\zeta^2 - 1)\tilde{\alpha} + \frac{\delta \tilde{\alpha}}{8}. \tag{5.7}$$

If $\zeta + \delta\tilde{\alpha}/8 \leq \delta^{1/3}$, it is obvious that $\zeta(\tilde{\alpha}) \leq \zeta_*$. If $\zeta + \delta\tilde{\alpha}/8 > \delta^{1/3}$, then

$$\zeta > \delta^{1/3} - \frac{\delta\tilde{\alpha}}{8} \geq \delta^{1/3} - \frac{1}{128} \geq \frac{255}{256}\delta^{1/3} \geq \frac{255}{128},$$

since $\delta \geq 8$. Now it is easy to verify that

$$1 - \zeta^{-2} - \frac{5\delta}{16\zeta^3} \geq 0.$$

Therefore, by (5.7) it follows $\zeta(\tilde{\alpha}) \leq \zeta \leq \zeta_*$. #

Similar to Theorem 5.2 in [20], we have the following result.

Theorem 5.2: *Let $\beta = O(1)$, and $\zeta_* = \max\{(\beta n^\tau)^{1/3}, 10\}$. Then Algorithm 5.1 needs at most $O(n^{\frac{(\nu+2)\tau}{3\nu}+\omega}\ln(n\mu_0/\epsilon))$ iterations to find a feasible solution (x,s) with $\langle xs\rangle \le \epsilon$.*

Remark: For $\tau \ge 1/2$, the upper bound of the iteration complexity (omitting $\ln(n\mu_0/\epsilon)$) is $O(n^{\frac{(4\nu+2)\tau}{3\nu}})$. If $\tau = 1/2$ and $\nu \to \infty$, then it asymptotically equals to $O(n^{2/3})$. This upper bound is greater than that obtained by Hung and Ye [4], because here we (implicitly) use a larger neighborhood $\mathcal{N}_\infty^-(\zeta_*)$.

6. Another upper bound of the complexity

In this section we choose another step length $\tilde{\alpha}$ which leads to another upper bound of the complexity. For the first order algorithm, i.e. $\nu = 1$, we will show that the iteration complexity of Algorithm 3.1 (without restriction on $\zeta(z(\alpha),\mu)$) is bounded by $O(n^{\tau+\omega}\ln(n\mu_0/\epsilon))$. For high-order ones, i.e. $\nu > 1$, the complexity upper bound depends on ζ. We will give complexity bounds for both Algorithms 3.1 and 5.1 in Theorems 6.3 and 6.5 respectively.

We choose a step length $\tilde{\alpha}$ as follows.

$$\tilde{\alpha} = \frac{1}{3\cdot 4^{\frac{\nu+1}{\nu}}\zeta^{\frac{\nu-1}{\nu}}\delta^{\frac{\nu+1}{\nu}}}, \qquad \text{for } \nu \ge 1. \tag{6.1}$$

For this step length we have a similar result as Lemma 4.4.

Lemma 6.1: *Suppose that $\delta \ge 1$. Then for the step length $\tilde{\alpha}$ defined by (6.1), we have*

$$\|r(\tilde{\alpha})\|_2 \le \frac{\tilde{\alpha}}{6}.$$

Proof: The proof is an analogue of the proof for Lemma 4.4. Here we only need to notice that $\zeta \le \delta + 1$, hence for $\nu \ge 2$,

$$4\zeta\delta\tilde{\alpha} \le \frac{1}{3}\left(\frac{\zeta}{\delta}\right)^{1/\nu} \le \frac{\sqrt{2}}{3}. \qquad \#$$

In what follows, we present a similar result as Lemma 4.5. However, the proof is totally different.

Lemma 6.2: *For the step length $\tilde{\alpha}$ defined by (6.1), we have the following inequality.*

$$\delta(\tilde{\alpha}) \le \delta - \frac{\tilde{\alpha}}{4}\delta + \frac{5\tilde{\alpha}}{3}.$$

Proof: We define two index sets for the given u as follows:

$$L = \{i \mid 0 < [u]_i < 1/2\} \qquad \text{and} \qquad S = \{i \mid [u]_i \ge 1/2\}.$$

Let p be any vector, then p_L and p_S denote the subvectors consisting of those components of p whose indices belong to L and S, respectively.

Using the notation $w(\tilde{\alpha}) = u(e+(u^{-2}-e)\tilde{\alpha})^{1/2}$, and noticing that $w(\tilde{\alpha})_L^{-1} \le u_L^{-1}$, we have

$$\begin{aligned}
\left(u^{-1}-w(\tilde{\alpha})^{-1}\right)_L &= w(\tilde{\alpha})_L^{-1}\left((e+(u^{-2}-e)\tilde{\alpha})^{1/2}-e\right)_L \\
&= w(\tilde{\alpha})_L^{-1}\left((e+(u^{-2}-e)\tilde{\alpha})^{1/2}+e\right)_L^{-1}(u^{-2}-e)_L\tilde{\alpha} \\
&\ge \frac{1}{2}w(\tilde{\alpha})_L^{-1}\left(e+(u^{-2}-e)\tilde{\alpha}\right)_L^{-1/2}(u^{-2}-e)_L\tilde{\alpha} \\
&= \frac{1}{2}w(\tilde{\alpha})_L^{-2}(u^{-1}-u)_L\tilde{\alpha} \\
&\ge \frac{1}{2}w(\tilde{\alpha})_L^{-3}(e-u^2)_L\tilde{\alpha}.
\end{aligned}$$

Similarly, we have

$$(w(\tilde{\alpha})-u)_L \ge \frac{1}{2}w(\tilde{\alpha})_L^{-1}(e-u^2)_L\tilde{\alpha}.$$

It follows from Lemma 6.1 that

$$|r(\tilde{\alpha})_L| \le \|r(\tilde{\alpha})\|_2 e \le \frac{\tilde{\alpha}}{6}e \le \frac{\tilde{\alpha}}{4}(e-u^2)_L.$$

Hence

$$\begin{aligned}
\left(w(\tilde{\alpha})^{-1}+w(\tilde{\alpha})^{-3}\right)_L|r(\tilde{\alpha})_L| &\le \frac{1}{4}\left(w(\tilde{\alpha})^{-1}+w(\tilde{\alpha})^{-3}\right)_L(e-u^2)_L\tilde{\alpha} \\
&\le \frac{1}{2}\left((u^{-1}-w(\tilde{\alpha})^{-1})+(w(\tilde{\alpha})-u)\right)_L.
\end{aligned}$$

It follws from Lemma 4.1 and the above inequality that

$$\begin{aligned}
|\left(u(\tilde{\alpha})^{-1}-u(\tilde{\alpha})\right)_L| &\le (w(\tilde{\alpha})^{-1}-w(\tilde{\alpha}))_L + (w(\tilde{\alpha})^{-1}+w(\tilde{\alpha})^{-3})_L|r(\tilde{\alpha})_L| \\
&\le \frac{1}{2}(u^{-1}-u)_L + \frac{1}{2}(w(\tilde{\alpha})^{-1}-w(\tilde{\alpha}))_L.
\end{aligned}$$

By Lemma 4.2 and denoting $q = u - u^{-1}$ and $q(\alpha) = u(\alpha) - u(\alpha)^{-1}$, we have

$$\|q(\tilde{\alpha})_L\|_2 \le \frac{1}{2}\|q_L\|_2 + \frac{1}{2}\left(1-\frac{\tilde{\alpha}}{2}\right)\|q_L\|_2 = \left(1-\frac{\tilde{\alpha}}{4}\right)\|q_L\|_2.$$

For the index set S we have the following estimation by Lemma 4.1.

$$\|q(\tilde{\alpha})_S\|_2 \le \|(w(\tilde{\alpha})-w(\tilde{\alpha})^{-1})_S\|_2 + \|(w(\tilde{\alpha})^{-1}+w(\tilde{\alpha})^{-3})_S\|_\infty\|r(\tilde{\alpha})\|_2.$$

Similar to (4.9) we have $\|w(\tilde{\alpha})_S^{-1}\|_\infty \le 2$ since $\|u_S^{-1}\|_\infty \le 2$. Then by Lemmas 4.2 and 6.1, we have

$$\begin{aligned}
\|q(\tilde{\alpha})_S\|_2 &\le \|(w(\tilde{\alpha})-w(\tilde{\alpha})^{-1})_S\|_2 + 10\|r(\tilde{\alpha})\|_2 \\
&\le \left(1-\frac{\tilde{\alpha}}{2}\right)\|q_S\|_2 + \frac{5\tilde{\alpha}}{3}.
\end{aligned}$$

Now we sum up the inequalities for L and S below

$$\begin{aligned}\|q(\tilde\alpha)\|_2 &= \sqrt{\|q(\tilde\alpha)_L\|_2^2 + \|q(\tilde\alpha)_S\|_2^2}\\ &\le \sqrt{\Big(1-\frac{\tilde\alpha}{4}\Big)^2\|q_L\|_2^2 + \Big((1-\frac{\tilde\alpha}{4})\|q_S\|_2 + \frac{5\tilde\alpha}{3}\Big)^2}\\ &\le \sqrt{\Big(1-\frac{\tilde\alpha}{4}\Big)^2\|q\|_2^2 + \frac{10\tilde\alpha}{3}(1-\frac{\tilde\alpha}{4})\|q_S\|_2 + \Big(\frac{5\tilde\alpha}{3}\Big)^2}\\ &\le \Big(1-\frac{\tilde\alpha}{4}\Big)\|q\|_2 + \frac{5\tilde\alpha}{3}. \qquad \#\end{aligned}$$

Similar to Theorem 4.6, we have the following result.

Theorem 6.3: *Let $\beta = O(1)$ and suppose that $\beta n^\tau > 12/5$. Let $\zeta_u \ge 1$ be a constant such that*

$$\zeta_k := \|(u^{[k]})^{-1}\|_\infty \le \zeta_u \qquad \text{for all } k = 0, 1, 2, \dots,$$

where $u^{[k]} = (x^{[k]}s^{[k]}/\mu_k)^{1/2}$. Then Algorithm 3.1 needs at most $O(\zeta_u^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega}\ln(n\mu_0/\epsilon))$ iterations to find a feasible point (x, s) with $\langle xs\rangle \le \epsilon$, where $\omega = \max\{1/2, \tau\}$. #

We notice that for the first order algorithm, i.e. $\nu = 1$, the upper bound of the iteration complexity is

$$O(n^{\tau+\omega}\ln(n\mu_0/\epsilon)),$$

which does not depend on ζ_u. Hence wherever the iterates move to does not affect the complexity. It is interesting to compare this result with the complexity bound presented in Theorem 4.6 and the remarks after that.

The complexity bound for higher-order is dependent of ζ_u. So we should consider Algorithm 5.1, for which we need a lemma similar to Lemma 5.1.

Lemma 6.4: *Let $\zeta_* \ge 5$ be any number and suppose that $\delta \ge 1$. If $\zeta = \zeta(z,\mu) \le \zeta_*$, then for $\tilde\alpha$ defined as (6.1), we have*

$$\zeta(\tilde\alpha) = \zeta(z + \sum_{i=1}^{\nu} z_i\tilde\alpha^i, \mu) \le \zeta_*.$$

Proof: Same as the proof of Lemma 5.1 we use (5.4), (4.9) and Lemma 6.1 to derive

$$\zeta(\tilde\alpha) \le \zeta(1-\tilde\alpha+\zeta^2\tilde\alpha)^{-1/2} + \frac{\zeta^3\tilde\alpha}{6}.$$

If $\zeta \le 4$, then

$$\zeta(\tilde\alpha) \le \zeta + \frac{\zeta^3\tilde\alpha}{6} \le 5 \le \zeta_*,$$

since $\tilde\alpha \le 1/12$. If $\zeta \ge 4$, then using $\zeta - 1 \le \delta$ one can verify that $(\zeta^2-1)\tilde\alpha$ is small enough such that

$$\Big(1 + (\zeta^2-1)\tilde\alpha\Big)^{-1/2} \le 1 - \frac{2}{5}(\zeta^2-1)\tilde\alpha.$$

So we have

$$\zeta(\tilde{\alpha}) \leq \zeta - \frac{7}{30}\zeta^3\tilde{\alpha} + \frac{2}{5}\zeta\tilde{\alpha} \leq \zeta \leq \zeta_*. \qquad \#$$

As Theorem 5.2, we have the following result.

Theorem 6.5: *Let $\beta = O(1)$ with $\beta n^\tau > 12/5$, and let $\zeta_* \geq 5$ be any number. Then Algorithm 5.1 needs at most $O(\zeta_*^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega} \ln(n\mu_0/\epsilon))$ iterations to find a feasible solution (x, s) with $\langle xs\rangle \leq \epsilon$.* #

Remark: Although the complexity upper bounds in Theorems 6.3 and 6.5 have the same form, they present different bounds. In Theorem 6.3, ζ_u represents an upper bound on ζ_k and it is unknown. In Theorem 6.5, however, ζ_* can be designated to any number (≥ 5).

If we impose the restriction $\zeta(\alpha) \leq \zeta_* = O(1)$ in Algorithm 5.1, the iteration complexity is $O(n^{\frac{\tau}{\nu}+\omega} \ln(n\mu_0/\epsilon))$, which is the same as what was obtained in [4] for the case $\tau = 1/2$. As $\nu \to \infty$, this complexity bound asymptotically equals to $O(n^\omega \ln(n\mu_0/\epsilon))$. For $\tau = 1/2$, it is $O(\sqrt{n}\ln(n\mu_0/\epsilon))$, the well-known best upper bound of the iteration complexity for short-step algorithms. The restriction $\zeta(\alpha) \leq \zeta_*$ essentially represent the neighborhood $\mathcal{N}_\infty^-(\zeta_*)$, so Algorithm 5.1 generates a sequence in $\mathcal{N}_\infty^-(\zeta_*)$ using line search. Theorem 6.5 tells us how the complexity depends on the radius ζ_* of the neighborhood $\mathcal{N}_\infty^-(\zeta_*)$.

7. Conclusion

In Sections 4, 5 and 6, we have used two different step lengths to analyse the complexity and achieved two distinct complexity bounds. Summing them up, we present the following results for Algorithm 3.1 and Algorithm 5.1 respectively.

Theorem 7.1: *Let $\beta = O(1)$ with $\beta n^\tau > 12/5$. Let $\zeta_u \geq 1$ be an upper bound on ζ_k for all iteration k. Then for every $\nu \geq 1$, the iteration complexity of Algorithm 3.1 is bounded from above by*

$$O(\min\{\zeta_u^{\frac{\nu+2}{\nu}} n^\omega, \zeta_u^{\frac{\nu-1}{\nu}} n^{\frac{\tau}{\nu}+\omega}\} \ln(n\mu_0/\epsilon)). \qquad \#$$

Remark: Let $\tau = 1/2$ and $\nu = 1$. If we can prove that all iterates are confined in $\mathcal{N}_\infty^-(\beta)$, then $\zeta_u = \beta = O(1)$, and the complexity is bounded by $O(\sqrt{n}\ln(n\mu_0/\epsilon))$ as the best result up to date for short-step algorithms. Otherwise, wherever the iterates move to, as long as they satisfy $\delta(z^{[k]}, \mu_k) \leq \beta\sqrt{n}$, the complxity bound is $O(n\ln(n\mu_0/\epsilon))$ as the best result up to date for large-step ones.

For Algorithm 5.1, we notice that if ζ_* is designated to be $O(n^{\tau/3})$, then the complexity bound in Theorem 6.5 is the same as that in Theorem 5.2, that is, Theorem 5.2 is actually a special case of Theorem 6.5. So our final result on complexity upper bound of Algorithm 5.1 is given by Theorem 6.5, (see also the remarks to Theorem 6.5).

References

[1] R.M. Freund, "Polynomial-time algorithms for linear programming based only on primal scaling and projected gradients of a potential function," *Mathematical Programming* 51 (1991) 203–222.

[2] C. Gonzaga, "Large-step path-following methods for linear programming, part I: Barrier function methods," *SIAM J. Optim.*, 1 (1991) 268–279.

[3] C. Gonzaga, "Large-step path-following methods for linear programming, part II: Potential reduction method," *SIAM J. Optim.*, 1 (1991) 280–292.

[4] P. Hung and Y. Ye, "An asymptotical $O(\sqrt{n}L)$-iteration path-following linear programming algorithm that uses long steps," Technical Report No. 53, on Computational Mathematics, Dept. of Mathematics, University of Iowa, (1994).

[5] B. Jansen, C. Roos, T. Terlaky and J.-Ph. Vial, "Primal-dual algorithms for linear programmming based on the logarithmic barrier method," Technical report No. 92–104, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands (1992).

[6] B. Jansen, C. Roos, T. Terlaky and J.-Ph. Vial, "Primal-dual target-following algorithms for linear programmming," Technical report No. 93–107, Faculty of Technical Mathematics and Informatics, Delft University of Technology, The Netherlands (1993).

[7] M. Kojima, N. Megiddo and S. Mizuno, "Theoretical convergence of large-step primal-dual interior point algorithms for linear programming," *Mathematical Programming* 59 (1993) 1–21.

[8] M. Kojima, N. Megiddo, T. Noma and A. Yoshise, "A unified approach to interior point algorithms for linear complementarity problems," volume 538 of *Lecture Notes in Computer Science.* Springer Verlag, Berlin, Germany, 1991.

[9] M. Kojima, S. Mizuno and A. Yoshise, "A primal-dual interior point algorithm for linear programming," in: N. Megiddo, ed., *Progress in Mathematical Programming: Interior-Point and Related Methods* (Springer, New York, 1989) pp. 29–47.

[10] S. Mizuno, M.J. Todd and Y. Ye, "On adaptive-step primal-dual interior-point algorithms for linear programming," *Mathematics of Operations Research* 18 (1993) 964–981.

[11] R.C. Monteiro, I. Adler and M.G.C. Resende, "A polynomial-time primal-dual affine scaling algorithm for linear and convex quadratic programming and its power series extension," *Mathematics of Operations Research* 15 (1990) 191–214.

[12] F.A. Potra, "On a predictor-corrector method for solving linear programs from infeasible starting points," Technical Report 34, Department of Mathematics, The University of Iowa (Iowa City, IA 52242, USA, 1992).

[13] J. Renegar, "A polynomial-time algorithm based on Newton's method for linear programming," *Mathematical Programming* 40 (1988) 59–93.

[14] C. Roos and J.-Ph. Vial, "Long steps with the logarithmic penalty barrier function in linear programming," in *Economic Decision-Making: Games, Economics and Optimization*, (J. Gabszeywicz, J.F. Richard, and L. Wolsey, eds.), Elsevier Science Publisher B. V., Amsterdam, 1989, 433–441.

[15] G. Sonnevend, "An analytic center for polyhedrons and new classes for linear (smooth-convex) programming," *Lect. Notes in Control and Inf. Sci.*, Vol. 84 (Springer, 1985) 866–876.

[16] M. Todd, "Recent developments and new directions in linear programming," *Mathematical Programming – Recent Developments and Applications* (M. Iri and K. Tanabe, eds.), Klwer Academic Publishers, 1989, 109–157.

[17] Y. Ye, "An $O(n^3L)$ potential reduction algorithm for linear programming," *Mathematical Programming* 50 (1991) 239–258.

[18] L. Zhang and Y. Zhang, "On polynomiality of the Mehrotra-type predictor-corrector interior-point algorithms," Technical Report 93–12, Department of Mathematics and Statistics, University of Maryland at Baltimore County (Baltimore, MD, 1993).

[19] G. Zhao, "On the relationship between the curvature integral and the complexity of path-following methods in linear programming," Research Report No. 566, Department of Mathematics, National University of Singapore, (Singapore, 1993), to appear in *SIAM Journal on Optimization.*

[20] G. Zhao, "Large-step path-following primal-dual algorithms for linear programming," Research Report No. 613, Department of Mathematics, National University of Singapore, (Singapore, 1994).

[21] G. Zhao and J. Stoer, "Estimating the complexity of a class of path-following methods for solving linear programming by curvature integrals," *Appl. Math. and Optimization*, 27 (1993) 85–103.