

# Infeasible Interior-Point Methods for Linear Optimization Based on Large Neighborhood

Alireza Asadi<sup>1</sup> · Cornelis Roos<sup>1</sup>

Published online: 29 October 2015

© The Author(s) 2015. This article is published with open access at Springerlink.com

**Abstract** In this paper, we design a class of infeasible interior-point methods for linear optimization based on large neighborhood. The algorithm is inspired by a full-Newton step infeasible algorithm with a linear convergence rate in problem dimension that was recently proposed by the second author. Unfortunately, despite its good numerical behavior, the theoretical convergence rate of our algorithm is worse up to square root of problem dimension.

**Keywords** Linear optimization · Primal-dual infeasible interior-point methods · Polynomial algorithms

**Mathematics Subject Classification** 90C05 · 90C51

## 1 Introduction

We consider the linear optimization (LO) problem in standard format, given by (P), and its dual problem given by (D) (see Sect. 3). We assume that the problems (P) and (D) are feasible. As is well known, this implies that both problems have optimal solutions and the same optimal value. In Sect. 8, we discuss how infeasibility and/or unboundedness of (P) and (D) can be detected.

---

✉ Alireza Asadi  
a.r.asadi@gmail.com  
Cornelis Roos  
C.Roos@tudelft.nl

<sup>1</sup> Faculty of Electrical Engineering, Computer Science and Mathematics, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands

Interior-point methods (IPMs) for LO are divided into two classes: feasible interior-point methods (FIPMs) and infeasible interior-point methods (IIPMs). FIPMs assume that a primal-dual strictly feasible point is available from which the algorithm can immediately start. In order to get such a solution several initialization methods have been presented, e.g., by Megiddo [1] and Anstreicher [2]. The so-called *big M method* of Megiddo has not been received well due to the numerical instabilities caused by the use of huge coefficients. A disadvantage of the so-called *combined phase-I and phase-II method* of Anstreicher is that the feasible region must be nonempty and a lower bound on the optimal objective value must be known.

IIPMs have the advantage that they can start with an arbitrary point and try to achieve feasibility and optimality, simultaneously. It is worth to mention that the best-known iteration bound for IIPMs is due to Mizuno [3]. We refer to, e.g., [4] for a survey of IIPMs.

Recently, the second author proposed an IIPM with full-Newton steps [4]. Each main iteration of the algorithm improves optimality and feasibility simultaneously, by a constant factor until a solution with a given accuracy is obtained. The amount of reduction is too small for practical purposes. The aim of this paper is to design a more aggressive variant of the algorithm which improves optimality and feasibility faster. We emphasize that this is our aim and also what happens in practice. As we will see, however, our algorithm suffers the same irony that occurs for FIPMs, namely that the theoretical convergence rate of large-update methods is worse than for full-Newton methods.

In the analysis of our algorithm, we use so-called *kernel function-based barrier function*. Any such barrier function is based on a univariate function, named its *kernel function*.<sup>1</sup> Such functions can be found in [5] and are closely related to the so-called *self-regular functions* introduced in [6]. In these references only FIPMs are considered, and it is shown that these functions are much more efficient for the process of re-centering, which is a crucial part in every FIPM, especially when an iterate is far from the central path. Not surprising, it turned out that these functions are also useful in our algorithm, where re-centering is also a crucial ingredient.

The paper is organized as follows. In Sect. 2, we describe the notations which we use throughout the paper. In Sect. 3, we explain the properties of a kernel function-based barrier function. In Sect. 4, as a preparation to our large neighborhood-based IIPM, we briefly recall the use of kernel-based barrier functions in large-update FIPMs, as presented in [5]. It will become clear in this section that the convergence rate highly depends on the underlying kernel function.

In Sect. 5, we describe our algorithm in detail. In our description we use a search direction based on a general kernel function. The algorithm uses two types of damped Newton steps: a so-called *feasibility* step and some *centering* steps. The feasibility step serves to reduce the infeasibility, whereas the centering steps keep the infeasibility fixed, but improve the optimality. This procedure is repeated until a solution with enough feasibility and optimality is obtained. Though many parts of our analysis are valid for general kernel function, at some places we restrict ourselves to a specific

---

<sup>1</sup> Throughout this paper, by a kernel function, we refer to the kernel functions introduced and discussed in [5].

kernel function. We show that the complexity of our algorithm based on this kernel function is slightly worse than the complexity of the algorithm introduced and discussed by Salahi et al. [7], who used another variant of this function. Note that the best-known iteration bound for the IIPMs that use a large neighborhood of the central path is due to Potra and Stoer [8] for a class of *superlinearly* convergent IIPMs for sufficient *linear complementarity problems* (LCP).

## 2 Notations

We use the following notational conventions. If  $x, s \in \mathbb{R}^n$ , then  $xs$  denotes the coordinatewise (or Hadamard) product of the vectors  $x$  and  $s$ . The nonnegative orthant and positive orthant are denoted as  $\mathbb{R}_+^n$  and  $\mathbb{R}_{++}^n$ , respectively. If  $z \in \mathbb{R}_+^n$  and  $f : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ , then  $f(z)$  denotes the vector in  $\mathbb{R}_+^n$  whose  $i$ -th component is  $f(z_i)$ , with  $1 \leq i \leq n$ . We write  $f(x) = O(g(x))$ , if for any  $x$  in the domain of  $f$ ,  $f(x) \leq cg(x)$  for some positive constant  $c$ . For  $z \in \mathbb{R}^n$ , we denote the  $l_1$ -norm by  $\|z\|_1$ , and the Euclidean norm by  $\|z\|$ .

## 3 Barrier Functions Based on Kernel Functions

Let the primal LO problem be defined as

$$\min \left\{ c^T x : Ax = b, \quad x \geq 0 \right\}, \tag{P}$$

and its dual problem as

$$\max \left\{ b^T y : A^T y + s = c, \quad s \geq 0 \right\}. \tag{D}$$

Here,  $A$  is a full row rank matrix in  $\mathbb{R}^{m \times n}$ ,  $b, y \in \mathbb{R}^m$ ,  $c, x, s \in \mathbb{R}^n$ ;  $x, y$  and  $s$  are vectors of variables.

It is well known that the triple  $(x, y, s)$  is optimal for (P) and (D) if and only if

$$\begin{aligned} Ax &= b, \quad x \geq 0, \\ A^T y + s &= c, \quad s \geq 0, \\ xs &= 0. \end{aligned} \tag{1}$$

The first two equations require primal and dual feasibility, respectively, and  $xs = 0$  is the so-called *complementarity condition*. Note that since  $x$  and  $s$  are nonnegative,  $xs = 0$  holds if and only if  $x^T s = 0$ . Therefore, since the feasibility conditions imply  $x^T s = c^T x - b^T y$ , the complementarity condition resembles the fact that at optimality the primal and dual objective values coincide.

Interior-point methods (IPMs) replace the complementarity condition by  $xs = \mu e$ , where  $\mu > 0$ . This gives rise to the following system:

$$\begin{aligned} Ax &= b, & x &\geq 0, \\ A^T y + s &= c, & s &\geq 0, \\ xs &= \mu e, \end{aligned} \tag{2}$$

where  $xs = \mu e$  is called the *centering condition*. It has been established (see, e.g., [9, Theorem II.4]) that the system (2) has a solution for some  $\mu > 0$  if and only if it has a solution for every  $\mu > 0$ . Moreover, this holds if and only if (P) and (D) satisfy the *interior-point condition* (IPC), i.e., there exist a feasible  $x > 0$  for (P) and a feasible  $(y, s)$  with  $s > 0$  for (D). If the matrix  $A$  has full row rank, then this solution is unique. It is denoted by  $(x(\mu), y(\mu), s(\mu))$  and called the  $\mu$ -center of (P) and (D). The  $\mu$ -centers form a virtual path inside the feasibility region leading to an optimal solution of (P) and (D). This path is called the *central path* of (P) and (D). When driving  $\mu$  to zero, the central path converges to an optimal triple  $(x, y, s)$  for (P) and (D).

We proceed by showing that the  $\mu$ -center can be characterized as the minimizer of a suitably chosen primal-dual barrier function. In fact we will define a wide class of such barrier functions, each of which is determined by a kernel function. A kernel function is just a univariate nonnegative function  $\psi(t)$ , where  $t > 0$ , which is strictly convex, minimal at  $t = 1$  and such that  $\psi(1) = 0$ , whereas  $\psi(t)$  goes to infinity both when  $t$  goes to zero and when  $t$  goes to infinity.

Now let  $(x, y, s)$  be a feasible triple with  $x > 0$  and  $s > 0$ . We call such a triple strictly feasible. We define the *variance vector* for this triple with respect to  $\mu$  as follows:

$$v := \sqrt{\frac{xs}{\mu}}.$$

Observe that  $v = e$  holds if and only if  $(x, y, s)$  is the  $\mu$ -center of (P) and (D). Given any kernel function  $\psi$  we extend its definition to  $\mathbb{R}_{++}^n$  according to

$$\Psi(v) := \sum_{i=1}^n \psi(v_i). \tag{3}$$

It is obvious that  $\Psi(v)$  is nonnegative everywhere, and  $\Psi(e) = 0$ . Yet we can define a barrier function  $\Phi(x, s, \mu)$  as follows:

$$\Phi(x, s, \mu) := \Psi(v), \tag{4}$$

where  $v$  is the variance vector of  $(x, y, s)$  with respect to  $\mu$ . It is now obvious that  $\Phi(x, s, \mu)$  is well-defined, nonnegative for every strictly feasible triple, and moreover,

$$\Phi(x, s, \mu) = 0 \Leftrightarrow \Psi(v) = 0 \Leftrightarrow v = e \Leftrightarrow (x, y, s) = (x(\mu), y(\mu), s(\mu)).$$

This implies that  $(x(\mu), y(\mu), s(\mu))$  is the (unique) minimizer of  $\Phi(x, s, \mu)$ .

As in [5], we call the kernel function  $\psi$  *eligible* iff it satisfies the following technical conditions.

$$\begin{aligned} t\psi''(t) + \psi'(t) &> 0, & t < 1, \\ t\psi''(t) - \psi'(t) &> 0, & t > 1, \\ \psi'''(t) &< 0, & t > 0, \\ 2\psi''(t)^2 - \psi'(t)\psi'''(t) &> 0, & t < 1. \end{aligned}$$

In the sequel it is always assumed that  $\psi$  is an eligible kernel function. Properties of eligible kernel functions will be recalled from [5] without repeating their proofs.

We would like to mention that a kernel function  $\psi$ , introduced and discussed in [5], has the following format

$$\psi(t) = \frac{t^2-1}{2} + \psi_b(t), \quad t > 0.$$

The term  $\frac{t^2-1}{2}$  is called the *growth term* which goes to infinity as  $t$  goes to infinity.  $\psi_b$  has the property that it goes to infinity as  $t$  tends to zero. This term is called the *barrier term* of  $\psi$ .

#### 4 A Class of Large-update FIPMs for LO

In this section we recall from [5] some results for a large-update FIPM for solving (P) and (D) using a kernel function-based barrier function.

Let us start with a definition. We call a triple  $(x, y, s)$ , with  $x > 0$ ,  $s > 0$ , an  $\varepsilon$ -solution of (P) and (D) iff the duality gap  $x^T s$  and the norms of the residual vectors  $r_b := b - Ax$ , and  $r_c := c - A^T y - s$ , do not exceed  $\varepsilon$ . In other words, defining  $\epsilon(x, y, s) := \max\{x^T s, \|r_b\|, \|r_c\|\}$ , we say that  $(x, y, s)$  is an  $\varepsilon$ -solution if  $\epsilon(x, y, s) \leq \varepsilon$ .

We assume, without any loss of generality, that the triple

$$(x^0, y^0, s^0) = (\mathbf{e}, 0, \mathbf{e}), \quad (5)$$

is a primal-dual feasible solution.<sup>2</sup> We then have  $x^0 s^0 = \mu^0 \mathbf{e}$  for  $\mu^0 = 1$ . This means that  $(x^0, y^0, s^0)$  is the 1-center, and hence  $\Phi(x^0, s^0, \mu^0) = 0$ . We use this triple to initialize our algorithm.

Each main (or outer) iteration of the algorithm starts with a strictly feasible triple  $(x, y, s)$  that satisfies  $\Phi(x, s, \mu) \leq \tau$  for some  $\mu \in ]0, 1]$ , where  $\tau$  is a fixed positive constant. It then constructs a new triple  $(x^+, y^+, s^+)$  such that  $\Phi(x^+, s^+, \mu^+) \leq \tau$  with  $\mu^+ < \mu$ . When taking  $\tau$  small enough, we obtain in this way a sequence of strictly feasible triples that belong to small neighborhoods of a sequence of  $\mu$ -centers,

<sup>2</sup> The problems (P) and (D) can be embedded into a self-dual problem for which the given triple is a feasible solution and that has an optimal solution that induces optimal solutions for (P) and (D).

for a decreasing sequence of  $\mu$ 's. As a consequence, the sequence of constructed triples  $(x, y, s)$  converges to an optimal solution of (P) and (D).

We will assume that  $\mu^+ = (1 - \theta)\mu$ , where  $\theta \in ]0, 1[$  is a fixed constant, e.g.,  $\theta = 0.5$  or  $\theta = 0.99$ . The larger the  $\theta$ , the more aggressive the algorithm is. In particular when  $\theta$  is large, each outer iteration will require several so-called inner iterations.

It is easy to compute the number the outer iterations. Using  $\mu^0 = 1$  one has the following result.

**Lemma 4.1** (cf. [9, Lemma II.17]) *The algorithm needs at most*

$$\left\lceil \frac{1}{\theta} \log \frac{n}{\varepsilon} \right\rceil,$$

*outer iterations to generate a strictly feasible  $\varepsilon$ -solution.*

The main task is therefore to get a sharp upper estimate for the number of inner iterations during an outer iteration. We now describe how such an estimate is obtained. We go into some detail, though without repeating proofs, because the results that we recall below are relevant for the IPM that we discuss in the next section.

As said before, at the start of each outer iteration we have a strictly feasible triple  $(x, y, s)$  and  $\mu > 0$  such that  $\Phi(x, s, \mu) \leq \tau$ . We first need to estimate the increase in  $\Phi$  when  $\mu$  is updated to  $\mu^+ = (1 - \theta)\mu$ . For this we need the following Lemma.

**Lemma 4.2** (cf. [5, Theorem 3.2]) *Let  $\rho : [0, \infty[ \rightarrow [1, \infty[$  be the inverse function of  $\psi(t)$  for  $t \geq 1$ . Then we have for any positive vector  $v$  and any  $\beta \geq 1$ :*

$$\Psi(\beta v) \leq n\psi\left(\beta\rho\left(\frac{\Psi(v)}{n}\right)\right).$$

Now let  $v$  be the variance vector of  $(x, y, s)$  with respect to  $\mu$ . Then one easily understands that the variance vector  $v^+$  of  $(x, y, s)$  with respect to  $\mu^+$  is given by  $v^+ := v/\sqrt{1 - \theta}$ . Hence, using Lemma 4.2 with  $\beta = 1/\sqrt{1 - \theta}$  we may write

$$\Phi(x, s, \mu^+) = \Psi(v^+) = \Psi\left(\frac{v}{\sqrt{1 - \theta}}\right) \leq n\psi\left(\frac{\rho\left(\frac{\Psi(v)}{n}\right)}{\sqrt{1 - \theta}}\right) \leq n\psi\left(\frac{\rho\left(\frac{\tau}{n}\right)}{\sqrt{1 - \theta}}\right),$$

where the last inequality holds because  $\rho$  is monotonically increasing and  $\Psi(v) := \Phi(x, s, \mu) \leq \tau$ . Hence the number  $\bar{\tau}$ , defined by

$$\bar{\tau} := n\psi\left(\frac{\rho\left(\frac{\tau}{n}\right)}{\sqrt{1 - \theta}}\right), \tag{6}$$

is an upper bound for the value of  $\Psi$  after a  $\mu$ -update. Note that this bound depends not on the triple  $(x, y, s)$ , but only on the kernel function  $\psi$  and the parameters  $n, \tau$  and  $\theta$ .

To simplify the notation we redefine  $\mu$  according to  $\mu := \mu^+$ . Thus we need to deal with the following question: given a triple  $(x, y, s)$  such that  $\Phi(x, s, \mu) \leq \bar{\tau}$ , how many inner iterations are needed to generate a triple  $(x, y, s)$  such that  $\Phi(x, s, \mu) \leq \tau$ . To answer this question we have to describe an inner iteration. It has been argued in Section 2.2. of [5] that it is natural to define search directions  $(\Delta x, \Delta y, \Delta s)$  by the system

$$\begin{aligned} A\Delta x &= 0, \\ A^T \Delta y + \Delta s &= 0, \\ s\Delta x + x\Delta s &= -\mu v \nabla \Psi(v). \end{aligned}$$

This system has a unique solution. It may be worth pointing out that if  $\psi(t) = \psi_1(t)$ , then  $-\mu v \nabla \Psi(v) = \mu \mathbf{e} - xs$ , and hence the resulting direction is the primal-dual Newton direction that is used in all primal-dual FIPMs. By doing a line search in this direction with respect to  $\Psi$  we get new iterates

$$(x, y, s) := (x, y, s) + \alpha(\Delta x, \Delta y, \Delta s),$$

where  $\alpha$  is the step size. According to [5, Lemma 4.4], we use below the following default step size:

$$\alpha = \frac{1}{\psi''(\rho(2\delta(v)))},$$

where  $\rho$  is the inverse function of  $-\frac{1}{2}\psi'(t)$ , and

$$\delta(v) := \frac{1}{2} \|\nabla \Psi(v)\|.$$

Fig. 1 shows a formal description of the algorithm.

The closeness of  $(x, y, s)$  to the  $\mu$ -center is measured by  $\Psi(v)$ , where  $v$  is the variance vector of  $(x, y, s)$  with respect to the current value of  $\mu$ . The initial triple  $(x, y, s)$  is as given by (5) and  $\mu = 1$ . So we then have  $\Psi(v) = 0 \leq \tau$ . After a

**Fig. 1** Large-update FIPM

```

Input:
a threshold parameter  $\tau > 0$ ;
an accuracy parameter  $\varepsilon > 0$ ;
a fixed barrier update parameter  $\theta, 0 < \theta < 1$ ;

begin
 $x := \mathbf{e}; y := 0; s := \mathbf{e}; \mu := 1$ ;
while  $\eta\mu \geq \varepsilon$ ,
   $\mu := (1 - \theta)\mu$ ;
  while  $\Psi(v) > \tau$ ,
     $(x, y, s) := (x, y, s) + \alpha(\Delta x, \Delta y, \Delta s)$ ;
     $v := \sqrt{\frac{xs}{\mu}}$ ;
  endwhile
endwhile
end
    
```

$\mu$ -update we have  $\Psi(v) \leq \bar{\tau}$ . Then a sequence of inner iterations is performed to restore the inequality  $\Psi(v) \leq \tau$ . Then  $\mu$  is updated again and so on. This process is repeated until  $n\mu$  falls below the accuracy parameter  $\varepsilon$  after which we have obtained an  $\varepsilon$ -solution.

To estimate the number of inner iterations we proceed as follows. Denoting the decrease in the value of  $\Psi$  as  $\Delta\Psi$ , it was shown in [5, Theorem 4.6] that

$$\Delta\Psi \geq \alpha\delta(v)^2 = \frac{\delta(v)^2}{\psi''(\rho(2\delta(v)))}. \tag{7}$$

Since the kernel function  $\psi$  is eligible, the last expression is increasing in  $\delta(v)$  [5, Lemma 4.7]. Besides, by [5, Theorem 4.9],  $\delta(v)$  is bounded from below as follows:

$$\delta(v) \geq \frac{1}{2}\psi'(\rho(\Psi(v))). \tag{8}$$

Combining (7) and (8), we arrive at

$$\Delta\Psi \geq \frac{(\psi'(\rho(\Psi(v))))^2}{4\psi''(\rho(\psi'(\rho(\Psi(v)))))}. \tag{9}$$

Following [5], let  $\gamma$  be the smallest number such that

$$\Psi(v)^{\gamma-1} \frac{(\psi'(\rho(\Psi(v))))^2}{4\psi''(\rho(\psi'(\rho(\Psi(v)))))} \geq \kappa \tag{10}$$

for some positive constant  $\kappa$ , whenever  $\Psi(v) \geq \tau$ . In [10], it is established that such constants  $\kappa$  and  $\gamma$  exist for each kernel functions. When denoting the value of the barrier function after the  $\mu$ -update as  $\Psi_0$  and the value after the  $k$ -th inner iteration as  $\Psi_k$ , it follows from (9) and (10) that

$$\Psi_0 \leq \bar{\tau}, \quad \Psi_k \leq \Psi_{k-1} - \kappa\Psi_{k-1}^{1-\gamma}, \quad k = 1, 2, \dots, \tag{11}$$

with  $\bar{\tau}$  as in (6). At this stage we may point out why the use of kernel functions other than the logarithmic kernel function may be advantageous. The reason is that if  $\psi$  is the logarithmic kernel function, then  $\gamma = 1$ , whence we obtain  $\Psi_k \leq \Psi_{k-1} - \kappa$  for each  $k \geq 1$ , provided that  $\Psi_{k-1} \geq \tau$ . This resembles the well-known fact that the best lower bound for the decrease in the logarithmic barrier function is a fixed constant, no matter what the value of  $\Psi(v)$  is. As we will see, smaller values of  $\gamma$  can be obtained for other kernel functions, which leads to larger reductions in the barrier function value, and hence lower iteration numbers.

By [5, Lemma 5.1], (11) implies that the number of inner iterations will not exceed

$$\frac{\bar{\tau}^\gamma}{\kappa\gamma} = \frac{1}{\kappa\gamma} \left( n\psi \left( \frac{\rho \left( \frac{\bar{\tau}}{n} \right)}{\sqrt{1-\theta}} \right) \right)^\gamma. \tag{12}$$



Multiplying this number by the number of outer iterations, as given by Lemma 4.1, we obtain the following upper bound for the total number of iterations:

$$\frac{1}{\theta\kappa\gamma} \left( n\psi \left( \frac{\rho \left( \frac{\tau}{n} \right)}{\sqrt{1-\theta}} \right) \right)^\gamma \log \frac{n}{\varepsilon}.$$

Given a kernel function  $\psi$ , it is now straightforward to compute the resulting iteration bound from this expression.

Let now  $\psi(t) = \psi_q(t)$ , where

$$\psi_q(t) := \frac{t^2 - 1}{2} + \frac{t^{1-q} - 1}{q - 1}, \quad \text{with } q > 1.$$

This function is introduced and discussed in [11] whereby it is established that  $\gamma = \frac{q+1}{2q}$  and  $\kappa = \frac{1}{124q}$ . As a result, the iteration number turns out to be

$$O \left( qn^{\frac{q+1}{2q}} \log \frac{n}{\varepsilon} \right).$$

The expression  $qn^{\frac{q+1}{2q}}$  is minimal at  $q = \frac{1}{2} \log n$  and then it is equal to  $\frac{1}{2}e \log n \sqrt{n}$ . Hence we obtain the iteration bound

$$O \left( \sqrt{n} \log n \log \frac{n}{\varepsilon} \right),$$

for the algorithm, which is the best-known iteration bound for large-update FIPMs based on kernel functions. It should be mentioned that the best-known iteration bound for the FIPMs that use a large neighborhood of the central path is due to Potra [12] who designed a *superlinearly* convergent predictor–corrector algorithm for linear complementarity problems that has an  $O(\sqrt{n}L)$  iteration bound, with  $L$  denoting the length of a binary string encoding the problem data.

In this paper, we consider a IIPM based on the use of a kernel function. Although many of the results below hold for any eligible kernel function, we will concentrate of the kernel  $\psi_q$ .

### 5 A Class of Large-Update IIPMs for LO

As usual in the theoretical analysis of IIPMs, we take the initial iterates as follows:

$$(x^0, y^0, s^0) = \zeta(\mathbf{e}, 0, \mathbf{e}), \tag{13}$$

where  $\mathbf{e}$  denotes the all-one vector in  $\mathbb{R}^n$ , and  $\zeta$  is a number such that

$$\|x^* + s^*\|_\infty \leq \zeta, \tag{14}$$

for some optimal solutions  $(x^*, y^*, s^*)$  of **(P)** and **(D)**. It is worth noting that if the data  $A, b, c$  are integral and  $L$  denotes the length of a binary string encoding the triple  $(A, b, c)$ , then it is well known that there exist optimal solutions  $x^*$  of **(P)** and  $(y^*, s^*)$  of **(D)** that satisfy (14) with  $\zeta = 2^L$  [4, Section 4.7].

Following [4], for any  $0 < \nu \leq 1$  we consider the primal-dual *perturbed pair*  $(P_\nu)$  and  $(D_\nu)$ , defined as follows:

$$\min \left\{ (c - \nu r_c^0)^T x : Ax = b - \nu r_b^0, x \geq 0 \right\}, \tag{P_\nu}$$

and

$$\max \left\{ (b - \nu r_b^0)^T y : A^T y + s = c - \nu r_c^0, s \geq 0 \right\}, \tag{D_\nu}$$

where  $r_b^0$  and  $r_c^0$  denote the initial primal and dual residual vectors, respectively. Note that, if  $\nu = 0$ , then the perturbed pair  $(P_\nu)$  and  $(D_\nu)$  coincides with the original pair **(P)** and **(D)**. Moreover,  $(x^0, y^0, s^0)$  is strictly feasible for the perturbed problems  $(P_\nu)$  and  $(D_\nu)$  if  $\nu = 1$ . In other words, if  $\nu = 1$ , then  $(P_\nu)$  and  $(D_\nu)$  satisfy the IPC and since  $x^0 s^0 = \mu^0 \mathbf{e}$ , with  $\mu^0 = \zeta^2$ , the  $\mu^0$ -center of  $(P_1)$  and  $(D_1)$  is given by  $(x^0, y^0, s^0)$ . The following Lemma is crucial.

**Lemma 5.1** (cf. [13, Theorem 5.13]) *The original problems **(P)** and **(D)** are feasible if and only if for each  $\nu$  satisfying  $0 < \nu \leq 1$  the perturbed problems  $(P_\nu)$  and  $(D_\nu)$  satisfy the IPC.*

Hence, for any  $\nu \in ]0, 1]$ , the problems  $(P_\nu)$  and  $(D_\nu)$  satisfy the IPC. This implies that the central path of  $(P_\nu)$  and  $(D_\nu)$  exists. The  $\mu$ -center of  $(P_\nu)$  and  $(D_\nu)$  is uniquely determined by the system

$$\begin{aligned} b - Ax &= \nu r_b^0, & x &\geq 0, \\ c - A^T y - s &= \nu r_c^0, & s &\geq 0, \\ xs &= \mu \mathbf{e}. \end{aligned}$$

In the sequel, the parameters  $\nu$  and  $\mu$  will always satisfy  $\mu = \nu \zeta^2$ . Therefore, we feel free to denote  $\mu$ -center of  $(P_\nu)$  and  $(D_\nu)$  as  $(x(\nu), y(\nu), s(\nu))$ . The set of these  $\mu$ -centers, as  $\nu$  runs through the interval  $]0, 1]$ , is called the *homotopy path* of **(P)** and **(D)** with respect to  $\zeta$ . By driving  $\nu$  to zero, the homotopy path converges to an optimal solution of **(P)** and **(D)** [14]. Our algorithm starts at the  $\mu$ -center for  $\nu = 1$  and then follows this homotopy path to obtain an  $\varepsilon$ -solution of **(P)** and **(D)**. Note that along the homotopy path the residual vectors are given by  $\nu r_b^0$  and  $\nu r_c^0$ , whereas the duality gap by  $\nu n \zeta^2$ . So these quantities converge to zero with the same speed. As a consequence we have

$$\varepsilon(x(\nu), y(\nu), s(\nu)) = \nu \varepsilon(x(1), y(1), s(1)) = \nu \varepsilon(\zeta \mathbf{e}, 0, \zeta \mathbf{e}), \quad \nu \in ]0, 1].$$

### 5.1 An Outer Iteration of the Algorithm

Just as in the case of large-update FIPMs we use a primal-dual barrier function to measure closeness of the iterates to the  $\mu$ -center of  $(P_v)$  and  $(D_v)$ . So, just as in Sect. 4,  $\Psi(v)$  will denote the barrier function based on the kernel function  $\psi(t)$ , as given in (3). Here,  $v$  denotes the variance vector of a triple  $(x, y, s)$  with respect to  $\mu > 0$ , and we define  $\Phi(x, s, \mu)$  as in (4). The algorithm is designed in such a way that at the start of each outer iteration we have  $\Psi(v) \leq \tau$  for some threshold value  $\tau = O(1)$ . As  $\Psi(v) = 0$  at the starting points (13), the condition  $\Psi(v) \leq \tau$  is certainly satisfied at the start of the first outer iteration.

Each outer iteration of the algorithm consists of a *feasibility step* and some *centering steps*. At the start of the outer iteration we have a triple  $(x, y, s)$  that is strictly feasible for  $(P_v)$  and  $(D_v)$ , for some  $v \in ]0, 1[$ , and that belongs to the  $\tau$ -neighborhood of the  $\mu$ -center of  $(P_v)$  and  $(D_v)$ , where  $\mu = v\zeta^2$ . We first perform a feasibility step during which we generate a triple  $(x^f, y^f, s^f)$  which is strictly feasible for the perturbed problems  $(P_{v^+})$  and  $(D_{v^+})$ , where  $v^+ = (1 - \theta)v$  with  $\theta \in ]0, 1[$ , and moreover, *close enough* to the  $\mu^+$ -center of  $(P_{v^+})$  and  $(D_{v^+})$ , with  $\mu^+ = v^+\zeta^2$ , i.e.,  $\Phi(x^f, s^f; \mu^+) \leq \tau^f$ , for some suitable value of  $\tau^f$ .

After the feasibility step, we perform some centering steps to get a strictly feasible triple  $(x^+, y^+, s^+)$  of  $(P_{v^+})$  and  $(D_{v^+})$  in the  $\tau$ -neighborhood of the  $\mu^+$ -center of  $(P_{v^+})$  and  $(D_{v^+})$ , where  $\mu^+ = (1 - \theta)\mu = v^+\zeta^2$ . During the centering steps the iterates stay feasible for  $(P_{v^+})$  and  $(D_{v^+})$ . Hence for the analysis of the centering steps we can use the analysis for FIPMs, presented in Sect. 4. From this analysis we derive that the number of centering steps will not exceed

$$\frac{(\Phi(x^f, s^f, \mu^+))^\gamma}{\kappa\gamma},$$

where the parameters  $\gamma$  and  $\kappa$  depend on the kernel function  $\psi$ . Hence we are left with the problem of defining a suitable search direction  $(\Delta^f x, \Delta^f y, \Delta^f s)$  for the feasibility step and to determine  $\theta$  such that after the feasibility step we have  $\Phi(x^f, s^f, \mu^+) \leq \tau^f$  for some suitable value of  $\tau^f$ . The number of outer iterations will be  $\frac{1}{\theta} \log\left(\frac{\epsilon(\zeta e, 0, \zeta e)}{\epsilon}\right)$ . Thus, the total number of iterations will not exceed

$$\frac{(\tau^f)^\gamma}{\theta\kappa\gamma} \log \frac{\epsilon(\zeta e, 0, \zeta e)}{\epsilon}. \tag{15}$$

### 5.2 Feasibility Step

For the search direction in the feasibility step, we use the triple  $(\Delta^f x, \Delta^f y, \Delta^f s)$  that is (uniquely) defined by the following system:

$$A\Delta^f x = \nu r_b^0, \tag{16a}$$

$$A^T \Delta^f y + \Delta^f s = \nu r_c^0, \tag{16b}$$

$$s\Delta^f x + x\Delta^f s = 0. \tag{16c}$$

**Fig. 2** Algorithm

```

Input:
accuracy parameter  $\varepsilon > 0$ ;
barrier update parameter  $\theta \in ]0, 1[$ ;
initialization parameter  $\zeta > 0$ ;
threshold parameter  $\tau = O(1)$ .

begin
 $x := \zeta e$ ;  $y := 0$ ;  $s := \zeta e$ ;  $v := 1$ ;  $\mu := \zeta^2$ ;
while  $\varepsilon(x, y, s) \geq \varepsilon$ 
  feasibility step:
   $(x, y, s) := (x, y, s) + \theta(\Delta^f x, \Delta^f y, \Delta^f s)$ ;
  update of  $\mu$  and  $v$ :
   $v := (1 - \theta) v$ ;
   $\mu := v \zeta^2$ ;
  centering steps:
  while  $\Phi(x, y, s; \mu) > \tau$ 
     $(x, y, s) := (x, y, s) + \alpha(\Delta x, \Delta y, \Delta s)$ ;
  endwhile
endwhile
end
    
```

Then, defining the new iterates according to

$$x^f = x + \theta \Delta^f x, \quad y^f = y + \theta \Delta^f y, \quad s^f = s + \theta \Delta^f s,$$

we have, due to (16a),

$$b - Ax^f = b - A(x + \theta \Delta^f x) = b - Ax - \theta v r_b^0 = v r_b^0 - \theta v r_b^0 = (1 - \theta) v r_b^0 = v^+ r_b^0.$$

In the same way one shows that  $c - A^T y^f - s^f = v^+ r_c^0$ . Hence it remains to find  $\theta$  such that  $x^f$  and  $s^f$  are positive and  $\Phi(x^f, s^f, \mu^+) \leq \tau^f$ . This is the hard part of the analysis of our algorithm, which we leave to the subsection below. We conclude this part with a formal description of the algorithm, in Fig. 2, and a graphical illustration in Fig. 3.

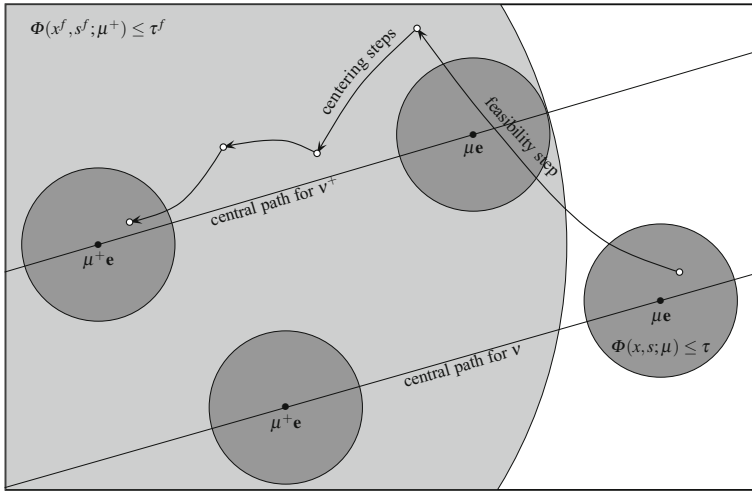
The straight lines in Fig. 3 depict the central paths of the pair  $(P_v)$  and  $(D_v)$  and the pair  $(P_{v^+})$  and  $(D_{v^+})$ . The  $\tau$ -neighborhoods of the  $\mu^-$  and  $\mu^+$ -centers are shown by the dark gray circles. The light gray region specifies the  $\tau^f$ -neighborhood of the  $\mu^+$ -center of  $(P_{v^+})$  and  $(D_{v^+})$ . The feasibility step is depicted by the first arrow at the right-hand side. The rest of the arrows stand for the centering steps. The iterates are shown by the circlets.

### 5.3 Analysis of the Feasibility Step

The feasibility step starts with some strictly feasible triple  $(x, y, s)$  for  $(P_v)$  and  $(D_v)$  and  $\mu = v \zeta^2$  such that

$$\Psi(v) \leq \tau \quad \text{with } v := \sqrt{\frac{xs}{\mu}}.$$

Our goal is to find  $\theta$  such that after the feasibility step, with step size  $\theta$ , the iterates  $(x^f, y^f, s^f)$  lie in the  $\tau^f$ -neighborhood of the  $\mu^+$ -center of the new perturbed pair  $(P_{v^+})$  and  $(D_{v^+})$ , with  $\tau^f = O(n)$ . This means that  $(x^f, y^f, s^f)$  are such that



**Fig. 3** An illustration of an iteration of the algorithm presented in Fig. 2

$$\Psi(v^f) \leq \tau^f \quad \text{where} \quad v^f := \sqrt{\frac{x^f s^f}{\mu^+}}, \quad \mu^+ = (1 - \theta)\mu.$$

To proceed, we define scaled feasibility directions  $d_x^f$  and  $d_s^f$  as follows:

$$d_x^f := \frac{v \Delta^f x}{x} \quad \text{and} \quad d_s^f := \frac{v \Delta^f s}{s}. \tag{17}$$

Using these notations, we may write

$$x^f = x + \theta \Delta^f x = x \left( \mathbf{e} + \theta \frac{\Delta^f x}{x} \right) = x \left( \mathbf{e} + \theta \frac{d_x^f}{v} \right) = \frac{x}{v} \left( v + \theta d_x^f \right), \tag{18}$$

$$s^f = s + \theta \Delta^f s = s \left( \mathbf{e} + \theta \frac{\Delta^f s}{s} \right) = s \left( \mathbf{e} + \theta \frac{d_s^f}{v} \right) = \frac{s}{v} \left( v + \theta d_s^f \right). \tag{19}$$

This shows that  $x^f$  and  $s^f$  are positive if and only if  $v + \theta d_x^f$  and  $v + \theta d_s^f$  are positive. On the other hand, using (17), we can reformulate (16c) as follows:

$$x \Delta^f s + s \Delta^f x = 0 \Leftrightarrow \frac{\Delta^f s}{s} + \frac{\Delta^f x}{x} = 0 \Leftrightarrow \frac{v \Delta^f s}{s} + \frac{v \Delta^f x}{x} = 0 \Leftrightarrow d_x^f + d_s^f = 0.$$

Therefore,  $d_s^f = -d_x^f$ . As a consequence,  $x^f$  and  $s^f$  are positive if and only if  $v \pm \theta d_x^f > 0$ . Since  $v > 0$ , this is equivalent to  $v^2 - \theta^2 (d_x^f)^2 > 0$ . We conclude that  $x^f$  and  $s^f$  are positive if and only if  $0 \leq \theta < \theta_{\max}$ , where

$$\theta_{\max} = \frac{1}{\left\| \frac{d_x^f}{v} \right\|_{\infty}}.$$

Yet we turn to the requirement that  $\Psi(v^f) \leq \tau^f$ . Using (18), (19) and  $xs = \mu v^2$ , we write

$$(v^f)^2 = \frac{x^f s^f}{\mu^+} = \frac{(v + \theta d_x^f)(v - \theta d_x^f)}{1 - \theta} = \frac{v^2 - \theta^2 (d_x^f)^2}{1 - \theta}.$$

Hence, if  $\theta < \theta_{\max}$  then we may write

$$v^f = \frac{\hat{v}}{\sqrt{1 - \theta}}, \quad \text{where } \hat{v} = \sqrt{v^2 - \theta^2 (d_x^f)^2}.$$

**Lemma 5.2** *Let  $\theta$  be such that  $1/\sqrt{1 - \theta} = O(1)$ . Then  $\Psi(\hat{v}) = O(n)$  implies  $\Psi(v^f) = O(n)$ .*

*Proof* By Lemma 4.2, we have

$$\Psi(v^f) \leq n\psi \left( \frac{\rho \left( \frac{\Psi(\hat{v})}{n} \right)}{\sqrt{1 - \theta}} \right).$$

Let  $\Psi(\hat{v}) = O(n)$ . Then  $\Psi(\hat{v}) \leq Cn$  for some positive constant  $C$ . Hence  $\Psi(\hat{v})/n \leq C$ . Recall that  $\rho(s) \geq 1$  for all  $s \geq 0$  and  $\rho(s)$  is monotonically increasing. Furthermore,  $\psi(t)$  is monotonically increasing for  $t \geq 1$ . Hence we obtain  $\Psi(v^f) \leq n\psi(\rho(C)/\sqrt{1 - \theta})$ . Since  $1/\sqrt{1 - \theta} = O(1)$ , the coefficient of  $n$  in the above upper bound for  $\Psi(v^f)$  does not depend on  $n$ . Hence the Lemma follows.  $\square$

Due to Lemma 5.2 it suffices for our goal to find  $\theta$  such that  $\Psi(\hat{v}) \leq \hat{\tau}$  where  $\hat{\tau} = O(n)$ . In the sequel, we consider  $\Psi(\hat{v})$  as a function of  $\theta$ , denoted as  $f_1(\theta)$ . So we have

$$f_1(\theta) := \Psi(\hat{v}) = \Psi \left( \sqrt{v^2 - \theta^2 (d_x^f)^2} \right).$$

We proceed by deriving a tight upper bound for  $f_1(\theta)$ , thereby using similar arguments as in [5]. Since the kernel function  $\psi(t)$  is eligible,  $\Psi(v)$  is e-convex (cf. [5, Lemma 2.1]), whence we have

$$f_1(\theta) \leq f(\theta) := \frac{1}{2} \left[ \Psi(v + \theta d_x^f) + \Psi(v - \theta d_x^f) \right].$$

The first and the second derivatives of  $f(\theta)$  are as follows:

$$f'(\theta) = \frac{1}{2} \sum_{i=1}^n \left[ \psi'(v_i + \theta d_{x_i}^f) - \psi'(v_i - \theta d_{x_i}^f) \right] d_{x_i}^f, \tag{20}$$

$$f''(\theta) = \frac{1}{2} \sum_{i=1}^n \left[ \psi''\left(v_i + \theta d_{x_i}^f\right) + \psi''\left(v_i - \theta d_{x_i}^f\right) \right] \left(d_{x_i}^f\right)^2. \tag{21}$$

Since  $\psi'''(t) < 0, \forall t > 0$ , it follows that  $\psi''(t)$  is monotonically decreasing. From this we deduce that

$$\psi''\left(v_i + \theta d_{x_i}^f\right) + \psi''\left(v_i - \theta d_{x_i}^f\right) \leq 2\psi''\left(v_i - \theta\|d_{x_i}^f\|\right) \leq 2\psi''\left(v_{\min} - \theta\|d_x^f\|\right),$$

where  $v_{\min} := \min(v)$  and  $\theta$  small enough, i.e., such that  $v_{\min} - \theta\|d_x^f\| > 0$ . Substitution into (21) gives

$$f''(\theta) \leq \|d_x^f\|^2 \psi''\left(v_{\min} - \theta\|d_x^f\|\right).$$

By integrating both sides of this inequality with respect to  $\theta$ , while using that  $f'(0) = 0$ , as follows from (20), we obtain

$$\begin{aligned} f'(\theta) &= f'(0) + \int_0^\theta f''(\xi) d\xi \leq \|d_x^f\|^2 \int_0^\theta \psi''\left(v_{\min} - \xi\|d_x^f\|\right) d\xi \\ &= -\|d_x^f\| \int_0^\theta \psi''\left(v_{\min} - \xi\|d_x^f\|\right) d\left(v_{\min} - \xi\|d_x^f\|\right) \\ &= \|d_x^f\| \left[ \psi'\left(v_{\min}\right) - \psi'\left(v_{\min} - \theta\|d_x^f\|\right) \right]. \end{aligned}$$

Integrating once more, we get

$$\begin{aligned} f(\theta) - f(0) &= \int_0^\theta f'(\xi) d\xi \leq \|d_x^f\| \int_0^\theta \left[ \psi'\left(v_{\min}\right) - \psi'\left(v_{\min} - \xi\|d_x^f\|\right) \right] d\xi \\ &= \psi'\left(v_{\min}\right)\theta\|d_x^f\| + \psi\left(v_{\min} - \theta\|d_x^f\|\right) - \psi\left(v_{\min}\right) \\ &\leq \left[ \psi'\left(v_{\min}\right) - \psi'\left(v_{\min} - \theta\|d_x^f\|\right) \right] \theta\|d_x^f\|, \end{aligned} \tag{22}$$

where the last inequality holds because  $\psi$  is convex.<sup>3</sup>

The first derivative with respect to  $v_{\min}$  of the right-hand side expression in this inequality is given by  $(\psi''(v_{\min}) - \psi''(v_{\min} - \theta\|d_x^f\|))\theta\|d_x^f\|$ . Since  $\psi''$  is (strictly) decreasing, this derivative is negative. Hence it follows that the expression is decreasing in  $v_{\min}$ . Therefore, when  $\theta$  and  $\|d_x^f\|$  are fixed, the less the  $v_{\min}$  is, the larger the expression will be. Below we establish how small  $v_{\min}$  can be when  $\delta(v)$  is given.

<sup>3</sup> We use that if  $f$  is convex and differentiable, then

$$(b - a)f'(a) \leq f(b) - f(a) \leq (b - a)f'(b).$$

For each coordinate  $v_i$  of  $v$  we have  $|\psi'(v_{\min})|/2 \leq \|\nabla\Psi(v)\|/2 = \delta(v)$ , which means that

$$-\delta(v) \leq -\frac{1}{2}\psi'(v_i) \leq \delta(v), \quad 1 \leq i \leq n.$$

Since the inverse function  $\rho$  of  $-\psi'/2$  is monotonically decreasing, this is equivalent to

$$\rho(\delta(v)) \leq v_i \leq \rho(-\delta(v)), \quad 1 \leq i \leq n. \tag{23}$$

Hence the smallest possible value of  $v_{\min}$  is  $\rho(\delta(v))$ , and this value is attained in the (exceptional) case where  $v_{\min}$  is the only coordinate of the vector  $v$  that differs from 1. So we may assume that  $v_{\min} = \rho(\delta(v))$ . This implies  $-\psi'(v_{\min})/2 = \delta(v)$  and hence  $\psi'(v_{\min}) \leq 0$ , whence  $v_{\min} \leq 1$ .

In the sequel, we denote  $\delta(v)$  simply as  $\delta$ . Substitution into (22) gives that

$$f(\theta) - f(0) \leq \left[ -2\delta - \psi'(\rho(\delta) - \theta\|d_x^f\|) \right] \theta\|d_x^f\|.$$

Hence we certainly have  $f(\theta) \leq \hat{\tau}$  if

$$f(0) + \left[ -2\delta - \psi'(\rho(\delta) - \theta\|d_x^f\|) \right] \theta\|d_x^f\| \leq \hat{\tau}.$$

Since  $f(0) = \Psi(v) \leq \tau$ , it can be verified that the last inequality holds if

$$-\frac{1}{2}\psi'(\rho(\delta) - \theta\|d_x^f\|) \leq \delta + \frac{\hat{\tau} - \tau}{2\theta\|d_x^f\|}.$$

Since  $\rho$  is decreasing, this is equivalent to

$$\rho(\delta) - \theta\|d_x^f\| \geq \rho\left(\delta + \frac{\hat{\tau} - \tau}{2\theta\|d_x^f\|}\right). \tag{24}$$

Note that if  $\theta$  approaches zero, then the left-hand side expression converges to  $\rho(\delta)$  and the right-hand side expression to zero. The left-hand side is decreasing in  $\theta$ , whereas the right-hand side is increasing. The largest possible  $\theta$  makes both sides equal. In order to get a tight approximation for this value, we first need to estimate  $\|d_x^f\|$ . This is done in the next section, where we follow a similar approach as in [15].

### 5.3.1 Upper Bound for $\|d_x^f\|$

One may easily check that the system (16), which defines the search directions  $\Delta^f x$ ,  $\Delta^f y$ , and  $\Delta^f s$ , can be expressed in terms of the scaled search directions  $d_x^f$  and  $d_s^f$  as follows:



$$\bar{A}d_x^f = \nu r_b^0, \tag{25}$$

$$\bar{A}^T \frac{\Delta^f y}{\mu} + d_s^f = \nu v s^{-1} r_c^0, \tag{26}$$

$$d_x^f + d_s^f = 0, \tag{27}$$

where

$$\bar{A} = AV^{-1}X, \quad V = \text{diag}(v), \quad X = \text{diag}(x).$$

At this stage we use that the initial iterates are given by (13) and (14), so we have

$$x^0 = s^0 = \zeta \mathbf{e}, \quad y^0 = 0, \quad \mu^0 = \zeta^2, \tag{28}$$

where  $\zeta > 0$  is such that

$$\|x^* + s^*\|_\infty \leq \zeta \tag{29}$$

for some optimal solutions  $x^*$  of (P) and  $(y^*, s^*)$  of (D).

Using a similar argument as given in [15, Section 4.3], it can be proven that

$$\|d_x^f\| \leq \frac{\mathbf{e}^T(x + s)}{\zeta v_{\min}}. \tag{30}$$

In [15, Lemma 4.3], it is shown that

$$\nu \zeta \mathbf{e}^T(x + s) \leq x^T s + \nu v \zeta^2. \tag{31}$$

Using  $xs = \mu v^2$  and (23) we may write

$$x^T s = \mu \mathbf{e}^T v^2 \leq \mu n \rho(-\delta)^2 = \nu v \zeta^2 \rho(-\delta)^2.$$

Substitution in (31) yields that

$$\zeta \mathbf{e}^T(x + s) \leq n \zeta^2 (1 + \rho(-\delta)^2).$$

Substitution in (30), also using that  $v_{\min} \geq \rho(\delta)$ , we obtain

$$\|d_x^f\| \leq \frac{n(1 + \rho(-\delta)^2)}{\rho(\delta)}. \tag{32}$$

### 5.3.2 Condition for $\theta$

Yet we return to the condition (24) on  $\theta$ :

$$\rho(\delta) \geq \theta \|d_x^f\| + \rho\left(\delta + \frac{\hat{\tau} - \tau}{2\theta \|d_x^f\|}\right).$$

The right-hand side is increasing in  $\|d_x^f\|$ . Therefore, due to (32), it suffices if

$$\rho(\delta) \geq \frac{\theta n(1 + \rho(-\delta)^2)}{\rho(\delta)} + \rho \left( \delta + \frac{\rho(\delta)(\hat{\tau} - \tau)}{2\theta n(1 + \rho(-\delta)^2)} \right). \tag{33}$$

Obviously, this implies that  $\theta n(1 + \rho(-\delta)^2) < \rho(\delta)^2$ . Therefore, there exists  $\alpha \in ]0, 1[$ , independent of  $n$ , such that

$$\theta = \frac{\alpha \rho(\delta)^2}{n(1 + \rho(-\delta)^2)}. \tag{34}$$

Clearly, using this  $\theta$ , the expression (33) can be restated as

$$\rho(\delta) \geq \alpha \rho(\delta) + \rho \left( \delta + \frac{\hat{\tau} - \tau}{2\alpha \rho(\delta)} \right). \tag{35}$$

Our objective is to find the largest possible  $\alpha$  satisfying this inequality. In order to proceed we need bounds for  $\delta = \delta(v)$  and  $\rho(\delta)$ . The next section deals with some technical lemmas which are useful in deriving these bounds.

### 6 Some Technical Lemmas

Recall that  $\rho$  is defined as the inverse function of  $-\frac{1}{2}\psi'(t)$  and  $\rho$  as the inverse function of  $\psi(t)$  for  $t \geq 1$ . We also need the inverse function of  $\psi(t)$  for  $t \in ]0, 1]$ , which we denote as  $\chi$ . To get tight estimates for these inverse functions, we need the inverse functions  $\bar{\chi}$  and  $\bar{\rho}$  of  $\psi_b(t)$  and  $-\psi'_b(t)$ , respectively. Recall that, as it was shown in [5], one has

$$\psi'_b(t) < 0, \quad \psi''_b(t) > 0, \quad \psi'''_b(t) < 0, \quad t > 0.$$

Hence  $\psi_b(t)$  is monotonically decreasing and  $\psi'_b(t)$  is monotonically increasing. This implies that  $\psi_b(t)$  and  $-\psi'_b(t)$  have inverse functions and these function are monotonically decreasing. We denote these inverse functions as  $\bar{\chi}$  and  $\bar{\rho}$ , respectively.

**Lemma 6.1** *With  $\bar{\chi}$  denoting the inverse function of  $\psi_b(t)$ , one has*

$$\bar{\chi} \left( s + \frac{1}{2} \right) \leq \chi(s) \leq \bar{\chi}(s), \quad s \geq 0.$$

*Proof* Let  $t \in ]0, 1]$ . Then one has

$$t = \chi(s) \iff \psi(t) = s \iff \psi_b(t) = s + \frac{1-t^2}{2} \iff \chi(s) = \bar{\chi} \left( s + \frac{1-\chi(s)^2}{2} \right).$$

Since  $\chi(s) \in ]0, 1]$ , this implies the inequalities in the lemma. □

**Lemma 6.2** With  $\bar{\rho}$  denoting the inverse function of  $-\psi'_b(t)$  for  $t > 0$ , one has

$$\bar{\rho}(1 + 2s) \leq \rho(s) \leq \bar{\rho}(2s), \quad s \geq 0.$$

*Proof* Since  $\psi'(t) = t + \psi'_b(t)$ , one has

$$\begin{aligned} t = \rho(s) &\Leftrightarrow -\frac{1}{2}\psi'(t) = s \\ &\Leftrightarrow -\psi'(t) = 2s \Leftrightarrow -\psi'_b(t) = 2s + t \Leftrightarrow \rho(s) = \bar{\rho}(2s + \rho(s)). \end{aligned}$$

If  $s \geq 0$ , then  $\rho(s) = t \in ]0, 1]$ , and hence  $\bar{\rho}(2s) \geq \rho(s) \geq \bar{\rho}(2s + 1)$ , proving the lemma.  $\square$

Recall that  $\rho$  is the inverse function of  $\psi(t)$  for  $t \geq 1$ . The following two results are less trivial than the preceding two lemmas.

**Lemma 6.3** (cf. [5, Lemma 6.2]) For  $s \geq 0$ , one has

$$\sqrt{1 + 2s} \leq \rho(s) \leq 1 + \sqrt{2s}.$$

**Lemma 6.4** One has, for any  $v \in \mathbb{R}^n_{++}$ ,

$$\frac{1}{2}\psi'(\rho(\Psi(v))) \leq \delta(v) \leq -\frac{1}{2}\psi'(\chi(\Psi(v))).$$

*Proof* The left-hand side inequality in the lemma is due to [5, Theorem 4.9]. The proof of the right-hand side inequality can be obtained by slightly changing the proof of [5, Theorem 4.9] and is therefore omitted.  $\square$

### 7 Iteration Bound of the IIPM Based on $\psi_q$

Since the kernel function  $\psi_q$  led to the best-known iteration bound for large-update FIPMs based on kernel functions, from now on we restrict ourselves to the case where  $\psi = \psi_q$ . Thus, we have

$$\psi_b(t) = \frac{t^{1-q} - 1}{q - 1}, \quad \psi'_b(t) = -t^{-q}, \quad \psi''_b(t) = qt^{-q-1}, \quad \psi'''_b(t) = -q(q + 1)t^{-q-2}.$$

In the current case the inverse functions  $\bar{\chi}$  and  $\bar{\rho}$  are defined as follows:

$$\bar{\chi}(s) = \frac{1}{[1 + (q - 1)s]^{\frac{1}{q-1}}}, \quad s > \frac{-1}{q - 1}, \tag{36}$$

$$\bar{\rho}(s) = \frac{1}{s^{\frac{1}{q}}}, \quad s > 0. \tag{37}$$

Since  $\chi$  and  $-\frac{1}{2}\psi'$  are decreasing,  $-\frac{1}{2}\psi'\chi$  is increasing. Hence, letting  $\Psi(v) \leq \tau$ , Lemma 6.4 implies that

$$\delta = \delta(v) \leq -\frac{1}{2}\psi'(\chi(\tau)). \tag{38}$$

By Lemma 6.1 we have  $\chi(\tau) \geq \bar{\chi}(\tau + \frac{1}{2})$ . Using once more that  $-\frac{1}{2}\psi'$  is decreasing we obtain

$$2\delta \leq -\psi'(\bar{\chi}(\tau + \frac{1}{2})).$$

Since  $-\psi'(t) = t^{-q} - t \leq t^{-q}$ , and due to (36), it follows that

$$\begin{aligned} 2\delta &\leq \bar{\chi}(\tau + \frac{1}{2})^{-q} \\ &= [1 + (q - 1)(\tau + \frac{1}{2})]^{q-1} \\ &= [1 + (q - 1)(\tau + \frac{1}{2})][1 + (q - 1)(\tau + \frac{1}{2})]^{q-2} \\ &\leq [1 + (q - 1)(\tau + \frac{1}{2})]e^{\tau + \frac{1}{2}}, \end{aligned}$$

where the last inequality is due to  $(1 + ax)^{1/x} \leq e^a$  for  $x > 0$  and  $1 + ax > 0$ . Hence, when taking  $\tau \leq 1/2$ , we have

$$\delta \leq \frac{1}{2}qe. \tag{39}$$

Since  $\rho$  is decreasing, by applying  $\rho$  to both sides of (38), and using Lemma 6.1 and (36) we obtain

$$\rho(\delta) \geq \chi(\tau) \geq \bar{\chi}(\tau + \frac{1}{2}) = \frac{1}{[1 + (q - 1)(\tau + \frac{1}{2})]^{q-1}}.$$

If  $\tau \leq 1/2$  this implies

$$\rho(\delta) \geq \frac{1}{q^{\frac{1}{q-1}}} = \frac{1}{e^{\frac{\log q}{q-1}}} \geq \frac{1}{e}. \tag{40}$$

Using that  $\rho$  is decreasing and also Lemma 6.2 and (37) we have

$$\rho\left(\delta + \frac{\hat{\tau} - \tau}{2\alpha\rho(\delta)}\right) \leq \rho\left(\frac{\hat{\tau} - \tau}{2\alpha\rho(\delta)}\right) \leq \bar{\rho}\left(\frac{\hat{\tau} - \tau}{\rho(\delta)}\right) = \left(\frac{\rho(\delta)}{\hat{\tau} - \tau}\right)^{\frac{1}{q}} \leq \frac{1}{(\hat{\tau} - \tau)^{\frac{1}{q}}}.$$

Furthermore, using  $\rho \geq 1/e$  we conclude that (35) certainly will hold if

$$\frac{1 - \alpha}{e} \geq \frac{1}{(\hat{\tau} - \tau)^{\frac{1}{q}}}.$$

Now taking  $\alpha = 1/2$  it follows that (35) will be satisfied if

$$q \leq \frac{\log(\hat{\tau} - \tau)}{\log(2e)}. \tag{41}$$

Substitution of  $\alpha = 1/2$  into (34) yields

$$\theta = \frac{\rho(\delta)^2}{2n(1 + \rho(-\delta)^2)}.$$

Due to (40) we have  $\rho(\delta) \geq 1/e$ . In order to find an upper bound for  $\rho(\delta)$  we do as follows. For  $s \geq 0$ ,

$$t = \rho(-s) \Leftrightarrow -\frac{1}{2}\psi'(t) = -s \Leftrightarrow 2s = t - t^{-q} \Leftrightarrow t = 2s + t^{-q}, \quad t \geq 1.$$

Since  $t \geq 1$  we have  $t^{-q} \in ]0, 1]$ . Hence  $t = \rho(-s)$  implies  $2s \leq \rho(-s) \leq 2s + 1$ . Thus,  $\rho(-\delta) \leq 2\delta + 1$ . By (39), one has  $\rho(-\delta) \leq 1 + qe$ . We therefore may conclude that (33) certainly holds if we take

$$\theta = \frac{1}{2e^2n(1 + (1 + qe)^2)}. \tag{42}$$

This is the value that will be used in the sequel.

### 7.1 Complexity Analysis

As we established in (15), the total number of iterations is at most

$$\frac{(\tau^f)^\gamma}{\theta\kappa\gamma} \log \frac{n\xi^2}{\varepsilon}.$$

We assume that  $\hat{\tau} = O(n)$ . Due to Lemma 5.2 we then also have  $\tau^f = O(n)$ , provided that  $1/\sqrt{1 - \theta} = O(1)$ . Due to (42) the latter condition is satisfied. To simplify the presentation we use  $\tau^f = n$  in the analysis below, but our results can easily be adapted to the case where  $\hat{\tau} = O(n)$ . When choosing  $\theta$  maximal such that (42) holds, we have

$$\theta = \frac{1}{2e^2n(1 + (1 + qe)^2)}.$$

Hence, substitution of  $\gamma = \frac{q+1}{2q}$  and  $\kappa = \frac{1}{124q}$  yields that the total number of iterations does not exceed

$$248e^2qn(1 + (1 + qe)^2)n^{\frac{q+1}{2q}} \log \frac{n\xi^2}{\varepsilon} = O\left(q^3n\sqrt{n}n^{\frac{1}{2q}} \log \frac{n\xi^2}{\varepsilon}\right). \tag{43}$$

The expression  $q^3n^{\frac{1}{2q}}$  is minimal if  $q = \log n/6$ , and then it is equal to  $e^4(\log n)^3/512$ . This value of  $q$  satisfies (41), since  $\log(2e) \leq 6$ . Hence we obtain the following iteration bound:

$$O\left(n\sqrt{n}(\log n)^3 \log \frac{n\xi^2}{\varepsilon}\right). \tag{44}$$

It is worth mentioning that if  $\psi_1$  were used, then the iteration bound of our algorithm would be  $O\left(n^2 \log \frac{n\xi^2}{\varepsilon}\right)$ , which is a factor  $\sqrt{n}/(\log n)^3$  worse than (44).

### 8 Detecting Infeasibility or Unboundedness

As indicated in the introduction, the algorithm will detect infeasibility or/and unboundedness of (P) and (D) if no optimal solutions exist. In that case Lemma 5.1 implies the existence of  $\bar{\nu} > 0$  such that the perturbed pair  $(P_\nu)$  and  $(D_\nu)$  satisfy the IPC if and only if  $\nu \in ]\bar{\nu}, 1]$ . As long as  $\nu^+ = (1 - \theta)\nu > \bar{\nu}$  the algorithm will run as it should, with  $\theta$  given by (42). However, if  $\varepsilon$  is small enough, at some stage it will happen that  $\nu > \bar{\nu} \geq \nu^+$ . At this stage the new perturbed pair does not satisfy the IPC. This will reveal itself since at that time we necessarily have  $\theta_{\max} < \tilde{\theta}$ . If this happens we may conclude that there is no optimal pair  $(x^*, s^*)$  satisfying  $\|x^* + s^*\|_\infty \leq \zeta$ . We refer to [4] for a more detailed discussion.

### 9 Computational Results

In this section, we present the numerical results for a practical variant of the algorithm described in Sect. 5. Theoretically, the barrier parameter  $\mu$  is updated by a factor  $(1 - \theta)$  with  $\theta$  given by (42), and the iterates are kept very close to the  $\mu$ -centers, namely the  $\tau$ -neighborhood of the  $\mu$ -centers, with  $\tau = \frac{1}{8}$ . In practice, it is not efficient to do in this way and not even necessary either. We present a variant of the algorithm which uses a predictor–corrector step in the feasibility step. Moreover, for the parameter  $\tau$ , defined in Sect. 5.1, we allow some larger value than  $\frac{1}{8}$ , e.g.,  $\tau = O(n)$ . We set  $\tau = \hat{\tau} = O(n)$  with  $\hat{\tau}$  defined as in Sect. 5.1. As a consequence, the algorithm does not need centering steps. We choose  $\hat{\tau}$  according to the following heuristics: if  $n \leq 500$ , then  $\hat{\tau} = 100n$ , for  $500 \leq n \leq 5000$ , we choose  $\hat{\tau} = 10n$  and for  $n \geq 5000$ , we set  $\hat{\tau} = 3n$ . We compare the performance of the algorithm with the well-known LIPSOL package [16].

#### 9.1 Starting Point and Stopping Criterion

A critical issue when implementing a primal-dual method is to find a suitable starting point. It seems sensible to obtain a starting point which is well centered and as close to a feasible primal-dual point as possible. The one suggested by theory, i.e., given by (13), being nicely centered, it may be quite far from the feasibility region. Moreover, to find a suitable  $\zeta$  is another issue.

In our implementation, we use a starting point  $(x^0, y^0, s^0)$  which is proposed by Lustig et al. [17] and inspired by the starting point used by Mehrtora [18]. Since we are interested in a point which is in the  $\tau$ -neighborhood of the  $\mu^0$ -center. If  $\Phi(x^0; s^0; \mu_0) > \tau$ , we increase  $\mu_0$  until  $\Phi(x^0; s^0; \mu_0) \leq \tau$ .

As in LIPSOL, our algorithm terminates if, for  $\varepsilon = 10^{-6}$ , either

$$E(x, y, s) := \frac{\|r_b\|}{\max(1, \|b\|)} + \frac{\|r_c\|}{\max(1, \|c\|)} + \frac{|c^T x - b^T y|}{\max(1, |c^T x|, |b^T y|)} \leq \varepsilon, \tag{45}$$

or  $|x^T s - (x^+)^T s^+| < \varepsilon$  occurs. The condition (45) measures the total relative errors in the optimality conditions (1), while the later criterion terminates the program if only a tiny improvement is obtained on the optimality. In fact, it prevents the program from stalling. We include this criterion following Lustig [19].

### 9.2 Feasibility Step Size

As in other efficient numerical experiments, e.g., [16, 17], regardless of the theoretical result, we apply different step sizes along the primal step  $\Delta x$  and the dual step  $(\Delta y, \Delta s)$ . This implies that the feasibility improves much faster than when identical step sizes are used. Letting  $(x, y, s)$  be the current iterates and  $(\Delta x, \Delta y, \Delta s)$  the Newton step, we obtain the maximum step sizes  $\theta_{\max}^p$  and  $\theta_{\max}^d$  in, respectively, the primal and the dual spaces as follows:

$$\theta_{\max}^p = \min_{\Delta x_i < 0} \left\{ 1, -\frac{x_i}{\Delta x_i} \right\}, \quad \theta_{\max}^d = \min_{\Delta s_i < 0} \left\{ 1, -\frac{s_i}{\Delta s_i} \right\}.$$

The goal is to keep the iterates close to the  $\mu$ -center, i.e., in its  $\hat{\tau}$ -neighborhood where  $\hat{\tau}$  is defined in Sect. 5.3. Thus, letting  $\bar{\theta}$  be such that

$$\Phi(x + \bar{\theta}\theta_{\max}^p \Delta x, s + \bar{\theta}\theta_{\max}^d \Delta s, \mu) \leq \hat{\tau},$$

the primal and the dual step sizes  $\theta_p$  and  $\theta_d$  are defined as follows:

$$\theta_p = \bar{\theta}\theta_{\max}^p \quad \text{and} \quad \theta_d = \bar{\theta}\theta_{\max}^d.$$

### 9.3 Solving the Linear System

We apply the backslash command of MATLAB (`'\'`) to solve the normal equations

$$AD^2 A^T \Delta y = \hat{b}, \tag{46}$$

where  $\hat{b}$  is some right-hand side and  $D$ , in LO case, has the form

$$D := \text{diag} \left( \sqrt{\frac{x}{s}} \right).$$

Denoting  $M := AD^2A^T$  in (46), whenever the matrix  $M$  is ill-conditioned, we could obtain some more accurate solution by perturbing  $M$  as

$$M = M + 10^{-9}I,$$

where  $I$  is the identity matrix with size of  $M$ .

### 9.4 A Rule for Updating $\mu$

Motivated by the numerical results, and considering the fact that the Mehrotra’s PC method has become the most efficient in practice and used in most IPM-based software packages, e.g., [16,20–22], we present the numerical results of the variant of our algorithm which uses Mehrotra’s PC direction at the feasibility step.

At the feasibility step, we apply the system

$$\begin{aligned} A\Delta_x^a &= r_b, \\ A^T\Delta_y^a + \Delta_s^a &= r_c, \\ s\Delta_x^a + x\Delta_s^a &= -xs, \end{aligned}$$

to obtain the affine-scaling directions  $(\Delta_x^a, \Delta_y^a, \Delta_s^a)$ . Then, the maximum step sizes  $\theta_{\max}^p$  and  $\theta_{\max}^d$  in, respectively, primal and dual spaces are calculated as described in Sect. 9.2. Then, defining

$$\mu^a := \frac{(x + \theta_{\max}^p\Delta_x^a)^T(s + \theta_{\max}^d\Delta_s^a)}{n},$$

we let

$$\sigma = \left(\bar{\sigma} \frac{\mu^a}{\mu}\right)^3, \quad \bar{\sigma} \in ]0, 1[.$$

We use  $\bar{\sigma} = 0.3$ , as the default value of  $\bar{\sigma}$ . If  $\sigma < 1$ , we calculate the new barrier update parameter  $\mu$  as follows:

$$\mu_{\text{new}} = \sigma\mu^a.$$

Then, if necessary, by increasing  $\mu_{\text{new}}$  by a constant factor, say 1.1, we derive some  $\mu_{\text{new}}$  for which

$$\Phi(x, s, \mu_{\text{new}}) \leq \hat{\tau}.$$

Note that  $\mu_{\text{new}}$  is acceptable only if  $\mu_{\text{new}} < \mu$ .

In the case where  $\mu_{\text{new}} \geq \mu$  or  $\sigma \geq 1$ , we do not perform any  $\mu$ -update and proceed with the following corrector step which attempts to center toward the current  $\mu$ -center:



$$\begin{aligned}
 A\Delta_s^c &= 0, \\
 A^T \Delta_y^c + \Delta_s^c &= 0, \\
 s\Delta_s^c + x\Delta_s^c &= \mu\mathbf{e} - \Delta_x^a \Delta_s^a.
 \end{aligned}
 \tag{47}$$

Recall that if  $\sigma \geq 1$ , then we have an increase in the duality gap and hence the use of  $\mu_{\text{new}}$  is no longer sensible. If  $\sigma < 1$  and  $\mu_{\text{new}} < \mu$ , we use the system (47) with  $\mu = 0$  as a corrector step.

The feasibility step  $(\Delta^f x, \Delta^f y, \Delta^f s)$  is obtained as follows:

$$\Delta^f x = \Delta_x^a + \Delta_s^c, \quad \Delta^f y = \Delta_y^a + \Delta_y^c, \quad \Delta^f s = \Delta_s^a + \Delta_s^c.$$

Next, we calculate the maximum step sizes  $\theta_{\max}^p$  and  $\theta_{\max}^d$  in, respectively, the primal and the dual spaces along the step  $(\Delta^f x, \Delta^f y, \Delta^f s)$ , as described in Sect. 9.2. Assuming that  $\mu$  is the outcome of the predictor step, we obtain a  $\theta$  such that

$$\Phi(x + \theta\theta_{\max}^p \Delta^f x, s + \theta\theta_{\max}^d \Delta^f s, \mu) \leq \hat{\tau},$$

with  $\hat{\tau}$  chosen as described in Sect. 9.4. By setting  $\theta_p = \theta\theta_{\max}^d$  and  $\theta_d = \theta\theta_{\max}^p$ , we calculate the new iterates  $(x^f, y^f, s^f)$  as follows:

$$x^f = x + \theta_p \Delta^f x, \quad y^f = y + \theta_d \Delta^f y \quad \text{and} \quad s^f = s + \theta_d \Delta^f s.$$

### 9.5 Results

In this section, we present our numerical results. Motivated by the theoretical result which says that the kernel function  $\psi_q$  gives rise to the best-known theoretical iteration bound for large-update IIPMs based on kernel functions, we compare the performance of the algorithm described in the previous subsection based on both the logarithmic barrier function and the  $\psi_q$ -based barrier function. As the theory suggests, we use  $q = \frac{\log n}{6}$  in  $\psi_q$ .

Our test was done on a standard PC with Intel® Core™ 2 Duo CPU and 3.25 GB of RAM. The code was implemented by version 7.11.0 (R2010b) of MATLAB® on Windows XP Professional operating system. The problems chosen for our test are from the NETLIB set. To simplify the study, we choose the problems which are in (P) format, i.e., there is no nonzero lower bound or finite upper bound on the decision variables.

Before solving a problem using the algorithm, we first solve it for singleton variables. This helps to reduce the size of the problem.

Numerical results are tabulated in Table 1. In second and the fourth columns, we listed the total number of iterations of the algorithm based on, respectively,  $\psi_1$ , the kernel function of the logarithmic barrier function, and  $\psi_q$ . The third and fifth columns stand for the quantity  $E(x, y, s)$ . The iteration numbers of the LIPSOL package are given in the sixth column of these tables. Finally, in the seventh column, we listed the

**Table 1** Numerical results ( $q = \frac{\log n}{6} \ln \psi_q$ )

Problem			$\psi_1$		$\psi_q$		Lipsol	
Name	Rows	Cols	It.	$E(x, y, s)$	It.	$E(x, y, s)$	It.	$E(x, y, s)$
25fv47	822	1571	<b>26</b>	1.8e−007	27	6.1e−007	25	2.8e−007
Adlitttle	57	97	12	6.8e−008	12	8.9e−008	11	2.4e−011
Afiro	28	32	8	1.0e−007	8	1.6e−007	7	3.7e−009
Agg	489	163	17	8.8e−007	18	7.2e−008	18	1.1e−008
Agg2	517	302	17	9.5e−007	17	4.9e−007	18	2.6e−010
Agg3	517	302	18	3.0e−007	18	5.7e−007	16	6.2e−008
Bandm	306	472	20	2.6e−007	<b>19</b>	3.5e−007	16	3.6e−007
Beaconfd	174	262	11	1.1e−007	11	3.0e−007	11	1.2e−010
Blend	75	83	13	6.2e−007	13	2.7e−007	12	5.7e−011
Bnl1	644	1175	32	5.0e−007	32	1.2e−007	25	5.3e−008
Bnl2	2325	3489	33	4.1e−007	31	9.3e−007	31	1.3e−007
Brandy	221	249	19	2.5e−007	19	8.9e−008	18	2.0e−008
D2q06c	2172	5167	28	5.6e−001	31	7.0e−007	28	4.8e−007
Degen2	445	534	25	1.3e−004	<b>18</b>	1.2e−004	13	4.2e−007
Degen3	1504	1818	<b>23</b>	1.4e−004	24	3.0e−005	19	1.4e−007
e226	224	282	<b>22</b>	7.4e−007	23	2.7e−007	20	8.9e−007
ffff800	525	854	26	1.0e−006	27	5.7e−007	26	3.0e−007
Israel	175	142	22	2.0e−007	<b>21</b>	7.1e−007	20	2.2e−007
Lotfi	154	308	16	3.7e−007	15	5.4e−007	15	4.6e−008
Maros-r7	3137	9408	19	8.0e−007	19	7.6e−008	14	1.0e−009
Sc105	106	103	10	1.7e−008	10	2.3e−008	9	4.2e−008
Sc205	206	203	11	2.5e−007	11	2.5e−007	11	6.5e−009
Sc50a	51	48	9	1.4e−007	9	1.2e−007	9	2.8e−009
Sc50b	51	48	7	5.4e−007	7	4.9e−007	7	1.6e−007
Scagr7	130	140	13	3.1e−007	13	3.4e−008	11	3.5e−007
Scfxm1	331	457	<b>18</b>	4.2e−007	19	2.0e−007	16	3.7e−007
scfxm2	661	914	<b>21</b>	1.4e−006	22	5.9e−008	19	1.6e−008
Scfxm3	991	1371	23	1.3e−007	<b>22</b>	8.5e−007	20	3.0e−010
Scsd1	78	760	<b>13</b>	3.9e−007	14	1.2e−007	10	3.3e−011
Scsd6	148	1350	15	3.8e−007	15	4.2e−007	11	7.8e−008
Scsd8	398	2750	13	6.4e−008	<b>12</b>	3.3e−007	11	4.0e−011
Sctap1	301	480	18	5.5e−007	18	6.9e−007	16	1.2e−008
Sctap2	1091	1880	19	1.3e−007	18	3.5e−007	18	3.5e−009
Sctap3	1481	2480	<b>19</b>	6.4e−007	20	9.6e−008	17	2.4e−008
Share1b	118	225	23	6.0e−007	23	1.5e−006	21	1.9e−010
Share2b	97	79	12	1.3e−008	12	6.8e−008	11	1.7e−007
Ship04l	403	2118	15	7.8e−007	15	8.0e−007	12	5.6e−011
Ship04s	403	1458	15	3.2e−007	15	4.3e−007	12	3.6e−007
Ship12l	1152	5427	19	9.1e−007	19	3.7e−007	15	7.7e−009

**Table 1** continued

Problem			$\psi_1$		$\psi_q$		Lipsol	
Name	Rows	Cols	it.	$E(x, y, s)$	it.	$E(x, y, s)$	it.	$E(x, y, s)$
Ship12s	1152	2763	<b>17</b>	1.3e–007	18	6.0e–008	15	3.6e–007
Stocfor1	118	111	<i>14</i>	4.9e–007	16	2.5e–007	16	1.1e–007
Stocfor2	2158	2031	25	1.8e–007	25	2.1e–007	21	2.3e–008
Truss	1001	8806	18	3.9e–007	18	1.7e–007	17	8.4e–007
Wood1p	245	2594	17	8.3e–007	17	1.2e–005	14	7.0e–010
Woodw	1099	8405	25	3.1e–007	<b>24</b>	1.0e–006	23	5.1e–010
Total			816		815		725	

quantity  $E(x, y, s)$  of the LIPSOL package. In each row, the Italic number denotes the smallest of the iteration numbers of the three algorithms, namely our algorithm based on  $\psi_1$  and  $\psi_q$ , and LIPSOL, and the bold number denotes the smallest of the iteration numbers of  $\psi_1$ -based and  $\psi_q$ -based algorithms. As it can be noticed from the last row of the table, the overall performance of the algorithm based on  $\psi_1$  is much better than that the variant based on  $\psi_q$ . However, in some of the problems, the  $\psi_q$ -based algorithm outperforms the  $\psi_1$ -based algorithm. This amounts to the problems AGG, BANDM, DEGEN2, DEGEN3, SCSD1, SCSD6, SCSD8 and SHARE2B. The columns related to LIPSOL prove that it is still the best; however, our  $\psi_1$ -based algorithm saved one iteration compared with LIPSOL, to solve AGG2 and AGG3, and two iterations when solving STOCFOR1.

## 10 Conclusions

In this paper, we analyze infeasible interior-point methods (IIPMs) for LO based on a large neighborhood. Our work is motivated by [4] in which Roos presents a full-Newton IIPM for LO. Since the analysis of our algorithms requires properties of barrier functions based on kernel functions that are used in large-update feasible interior-point methods (FIPMs), we present primal-dual large-update FIPMs for LO based on kernel functions, as well.

In Roos' algorithm, the iterates move within small neighborhood of the  $\mu$ -centers of the perturbed problem pairs. In many IIPMs, the algorithm reduces the infeasibility and the duality gap at the same rate. His algorithm has the advantages that it uses full-Newton steps and hence no calculation of step size is needed, and moreover its theoretical iteration bound is  $O(n \log(\epsilon(\zeta \mathbf{e}, 0, \zeta \mathbf{e})/\epsilon))$  which coincides with the best-known iteration bound for IIPMs. Nevertheless, it has the deficiency that it is too slow in practice as it reduces the quantity  $\epsilon(x, y, s)$  by a factor  $(1 - \theta)$  with  $\theta = O(\frac{1}{n})$  at an iteration.

We attempt to design a large-update version of Roos' algorithm which allows some larger amount of reduction of  $\epsilon(x, y, s)$  at an iteration than  $1 - \theta$  with  $\theta = O(1/n)$ . This requires that the parameter  $\theta$  is larger than  $O(1/n)$ , even  $\theta = O(1)$ .

Unfortunately, the result of Sect. 5 implies that  $\theta$  is  $O(1/(n(\log n)^2))$  which yields  $O(n\sqrt{n}(\log n)^3 \log(\epsilon(\zeta \mathbf{e}, 0, \zeta \mathbf{e})/\epsilon))$  iteration bound for a variant. Since the theoretical result of the algorithm is disappointing, we rely on the numerical results to establish that our algorithm is a large-update method. A practically efficient version of the algorithm is presented and its numerical result is compared with the well-known LIPSOL package. Fortunately, the numerical results seem promising as our algorithm has iteration numbers close to those of LIPSOL and, in a few cases, outperforms it. This means that IIPMs suffer from the same irony as FIPMs, i.e., regardless of their nice practical performance, the theoretical complexity of large-update methods is worse. Recall that the best-known iteration bound for large-update IIPMs is  $O(n\sqrt{n} \log n \log(\epsilon(\zeta \mathbf{e}, 0, \zeta \mathbf{e})/\epsilon))$  which is due to Salahi et al. [7].

As in LIPSOL [16], different step sizes in the primal and the dual spaces are used in our implementation. This gives rise to a faster achievement in feasibility than when identical step sizes are used. Moreover, inspired by the LIPSOL package, we use a predictor–corrector step in the feasibility step of the algorithm.

Future research may focus on the following areas:

- As mentioned before, our algorithm has a factor  $(\log n)^2$  worse iteration bound than the best-known iteration bound for large-update IIPMs. One may consider how to modify the analysis such that the iteration bound of our algorithm is improved by a factor  $(\log n)^2$ .
- As mentioned in Sect. 10, according to the analysis of our algorithm presented in Sect. 5, the barrier-updating parameter  $\theta$  is  $O(1/(n(\log n)^2))$ . This yields the loose iteration bound given by (44). This slender value of  $\theta$  is obtained because of some difficulties in the analysis of the algorithm which uses the largest value of  $\theta$ , satisfying (24), to assure that  $\Psi(\hat{v}) = O(n)$ . This value of  $\theta$  is much smaller than the best value we may choose. Assuming  $n = 60$ , the largest value of  $\theta$  satisfying  $\Psi(\hat{v}) = n$  is 0.849114 while the value of  $\theta$  suggested by theory is 0.141766. A future research may focus on some new analysis of the algorithm which yields some larger value of  $\theta$ .
- Roos' full-Newton step IIPM was extended to semidefinite optimization (SDO) by Mansouri and Roos [23], to symmetric optimization (SO) by Gu et al. [24] and to LCP by Mansouri et al. [25]. An extension of large-update FIPMs based on kernel functions to SDO was presented by El Ghami [10]. One may consider how our algorithm behaves in theory and practice when it is extended to case of SDO, SO and LCP.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

1. Megiddo, N. (ed.): Pathways to the optimal set in linear programming. In: Progress in Mathematical Programming (Pacific Grove, CA, 1987), pp. 131–158. Springer, New York (1989)

2. Anstreicher, K.M.: A combined phase I-phase II projective algorithm for linear programming. *Math. Progr.* **43**(2, (Ser. A)), 209–223 (1989)
3. Mizuno, S.: Polynomiality of infeasible-interior-point algorithms for linear programming. *Math. Progr.* **67**(1, Ser. A), 109–119 (1994)
4. Roos, C.: A full-Newton step  $O(n)$  infeasible interior-point algorithm for linear optimization. *SIAM J. Optim.* **16**(4), 1110–1136 (2006)
5. Bai, Y., El Ghami, M., Roos, C.: A comparative study of Kernel functions for primal-dual interior-point algorithms in linear optimization. *SIAM J. Optim.* **15**(1), 101–128 (2004)
6. Peng, J., Roos, C., Terlaky, T.: *Self-Regularity: A New Paradigm for Primal-Dual Interior-Point Algorithms*. Princeton Series in Applied Mathematics. Princeton University Press, Princeton (2002)
7. Salahi, M., Peyghami, M.R., Terlaky, T.: New complexity analysis of IIPMs for linear optimization based on a specific self-regular function. *Eur. J. Oper. Res.* **186**(2), 466–485 (2008)
8. Potra, F.A., Stoer, J.: On a class of superlinearly convergent polynomial-time interior-point methods for sufficient LCP. *SIAM J. Optim.* **20**(3), 1333–1363 (2009)
9. Roos, C., Terlaky, T., Vial, J.-Ph.: *Interior-Point Methods for Linear Optimization*. Springer, New York (2006). In: Second edition of Theory and Algorithms for Linear Optimization [Wiley, Chichester, 1997]
10. El Ghami, M.: *New Primal-Dual Interior-Point Methods Based on Kernel Functions*. Ph.D. thesis, Delft University of Technology, The Netherlands (2005)
11. Peng, J., Roos, C., Terlaky, T.: A new and efficient large-update interior-point method for linear optimization. *Vychisl. Tekhnol.* **6**(4), 61–80 (2001)
12. Potra, F.A.: A superlinearly convergent predictor-corrector method for degenerate LCP in a wide neighborhood of the central path with  $O(\sqrt{n}L)$ -iteration complexity. *Math. Progr.* **100**(2, Ser. A), 317–337 (2004)
13. Ye, Y.: *Interior Point Algorithms*. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley, New York (1997)
14. Asadi, A., Gu, G., Roos, C.: Convergence of the homotopy path for a full-Newton step infeasible interior-point method. *Oper. Res. Lett.* **38**(2), 147–151 (2010)
15. Gu, G., Mansouri, H., Zangiabadi, M., Bai, Y., Roos, C.: Improved full-Newton step  $O(n)$  infeasible interior-point method for linear optimization. *J. Optim. Theory Appl.* **145**(2), 271–288 (2010)
16. Zhang, Y.: Solving large-scale linear programs by interior-point methods under the MATLAB environment. *Optim. Methods Softw.* **10**(1), 1–31 (1998)
17. Lustig, I., Marsten, R.E., Shanno, D.F.: On implementing Mehrotra’s predictor-corrector interior-point method for linear programming. *SIAM J. Optim.* **2**(3), 435–449 (1992)
18. Mehrotra, S.: On the implementation of a primal-dual interior-point method. *SIAM J. Optim.* **2**(4), 575–601 (1992)
19. Lustig, I.: Feasibility issues in a primal-dual interior-point method for linear programming. *Math. Progr.* **49**(2, (Ser. A)):145–162 (1990/91)
20. Vanderbei, R.J.: LOQO: an interior-point code for quadratic programming. *Optim. Methods Softw.* **11/12**(1–4), 451–484 (1999)
21. Andersen, E.D., Andersen, K. D.: The Mosek interior-point optimizer for linear programming: an implementation of the homogeneous algorithm. In: Frenk, H., Roos, K., Terlaky, T., Zhang, S. (eds.) *High Performance Optimization*, vol. 33 of *Appl. Optim.* pp. 197–232. Kluwer Academy Publication, Dordrecht (2000)
22. Czyzyk, J., Mehrotra, S., Wagner, M., Wright, S.: PCx: an interior-point code for linear programming. *Optim. Methods Softw.* **11/12**(1–4), 397–430 (1999)
23. Mansouri, H., Roos, C.: A new full-Newton step  $O(n)$  infeasible interior-point algorithm for semidefinite optimization. *Numer. Algorithms* **52**(2), 225–255 (2009)
24. Gu, G., Zangiabadi, M., Roos, C.: Full Nesterov–Todd step infeasible interior-point method for symmetric optimization. *Eur. J. Oper. Res.* **3**, 473–484 (2011)
25. Mansouri, H., Zangiabadi, M., Pirhaji, M.: A full-Newton step  $O(n)$  infeasible-interior-point algorithm for linear complementarity problems. *Nonlinear Anal. Real World Appl.* **12**(1), 545–561 (2011)