Numer Algor
DOI 10.1007/s11075-010-9439-0

ORIGINAL PAPER

A projected–gradient interior–point algorithm for complementarity problems

**Roberto Andreani · Joaquim J. Júdice ·
José Mario Martínez · Joao Patrício**

Received: 25 July 2010 / Accepted: 15 November 2010

Abstract Interior–point algorithms are among the most efficient techniques for solving complementarity problems. In this paper, a procedure for globalizing interior–point algorithms by using the maximum stepsize is introduced. The algorithm combines exact or inexact interior–point and projected–gradient search techniques and employs a line–search procedure for the natural merit function associated with the complementarity problem. For linear problems, the maximum stepsize is shown to be acceptable if the Newton interior–point search direction is employed. Complementarity and optimization problems are discussed, for which the algorithm is able to process by either finding a solution or showing that no solution exists. A modification of the algorithm for dealing with infeasible linear complementarity problems is introduced

R. Andreani · J. M. Martínez
Instituto de Matemática, Estatística e Computação Científica,
Universidade Estadual de Campinas, Campinas, Brazil

R. Andreani
e-mail: andreani@ime.unicamp.br

J. M. Martínez
e-mail: martinez@ime.unicamp.br

J. J. Júdice
Departamento de Matemática, Universidade de Coimbra, Coimbra, Portugal
e-mail: Joaquim.Judice@co.it.pt

J. J. Júdice · J. Patrício
Instituto de Telecomunicações, Polo de Coimbra, Coimbra, Portugal

J. Patrício (✉)
Instituto Politécnico de Tomar, Tomar, Portugal
e-mail: Joao.Patricio@aim.estt.ipt.pt

Published online: 27 November 2010

which, in practice, employs only interior–point search directions. Computational experiments on the solution of complementarity problems and convex programming problems by the new algorithm are included.

Keywords Complementarity problems · Interior–point algorithms · Nonlinear programming

1 Introduction

A classical problem in Numerical Analysis consists of solving nonlinear systems of equations with unrestricted [9, 29, 30, 33] or nonnegative variables [4]. In this paper we address the case where complementarity conditions $x_i w_i = 0$ on nonnegative variables x_i and w_i are included in the definition of the system. Therefore the problem consists of finding $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $w \in \mathbb{R}^n$ such that

$$H(x, y, w) = 0, x_i w_i = 0, i = 1, \ldots, n, \tag{1}$$

and

$$x, w \geq 0, \tag{2}$$

where $H : \mathbb{R}^{n+m+n} \longrightarrow \mathbb{R}^{n+m}$ is continuously differentiable. Note that (1) is a system of $2n + m$ equations and unknowns.

Linear (LCP), Nonlinear (NCP) Complementarity Problems, Variational Inequalities and KKT conditions of constrained optimization problems may all be formulated in the form (1) and (2). For those particular cases many algorithms have been published [1, 2, 4, 6, 8, 10, 13, 15, 24, 27, 28, 34, 35, 37, 38]. Interior–point algorithms [10, 28, 35, 37, 38] exhibit very nice practical behavior for constrained optimization, complementarity and variational inequality problems. These algorithms compute, at each iteration, an exact or inexact Newton's direction for a system of nonlinear equations which contains a central parameter. Line–search or trust–region techniques are used to guarantee global convergence of the algorithm under suitable hypotheses. Interior–point algorithms that are quite efficient for solving monotone LCPs in practice, usually employ the maximum stepsize in each iteration [28, 37]. However, it is not possible to establish global convergence for such procedures [28, pp. 407–408].

In this paper we consider the natural (sum of squares) merit function associated with the nonlinear system (1) and (2), in order to design an algorithm that combines interior–point and projected–gradient techniques. A line search is included in the algorithm to guarantee global convergence to a stationary point of the natural merit function on the convex set defined by the linear constraints (2). For monotone CPs, such a stationary point is a solution of the CP. Hence the algorithm can process monotone variational inequalities and convex nonlinear programs when their feasible sets are nonempty and bounded. In case of a monotone linear complementarity problem, it is shown that the maximum stepsize can be used throughout the algorithm without destroying its global convergence property. It is also important to add that in this case the algorithm either finds a solution or shows that the LCP is infeasible. This has obvious consequences on the solution of the linear and convex quadratic program for which the algorithm either finds an optimal solution or shows that these problems are infeasible or unbounded.

The idea of the projected–gradient interior–point algorithm is to use the exact (or inexact) Newton's direction whenever it is possible and recommendable, and only move to the projected–gradient otherwise. In practice, for monotone LCPs, the algorithm never moves to the projected–gradient direction unless the problem is infeasible or there is no strictly feasible solution. If the LCP is infeasible or has no strictly feasible solution, the projected–gradient algorithm is activated. Since the convergence rate of the projected-gradient algorithm is much slower than the convergent rate of the interior–point algorithm, we propose a two-phase procedure that, in general, allows us to produce the correct diagnostic without using projected-gradient directions. Computational experience shows that the two-phase algorithm is able to find a solution of the LCP, or to show that the LCP is infeasible. Therefore, the algorithm can find an optimal solution for linear and convex programs or to show that they are primal or dual infeasible. Furthermore, no projected–gradient iterations are used in practice, provided that feasible LCPs have a strictly feasible solution (LP and QP have a strictly primal or dual feasible solution).

The structure of this paper is as follows. In Section 2 the projected–gradient interior–point algorithm is described and its convergence is analyzed in Section 3. Section 4 is devoted to the analysis of linear problems. Computational experience with the algorithm for solving LCPs, linear, quadratic and nonlinear programs are reported in Section 5. Conclusions about the efficiency of the proposed methodology are included in the last section.

Notation

The set of natural numbers is denoted by $\mathbb{N}$. Furthermore

$$\mathbb{R}^n_+ = \{x \in \mathbb{R}^n \ : \ x \geq 0\}.$$

For $v, w \in \mathbb{R}^n$, we define:

$$[v, w] = \left\{x \in \mathbb{R}^n : \ x = tv + (1-t)w \text{ for some } t \in [0, 1]\right\}.$$

The symbol $\|\cdot\|$ denotes the Euclidean norm, although many times it may be replaced by an arbitrary norm.

If $x, w \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, we denote $(x, y, w) = \begin{pmatrix} x \\ y \\ w \end{pmatrix}$, provided that this simplification does not lead to confusion.

2 Projected gradient inexact-Newton algorithm

Let

$$\Omega = \left\{ \begin{pmatrix} x \\ y \\ w \end{pmatrix} \in \mathbb{R}^{n+m+n} \, : \, x, w \geq 0 \right\}$$

and $z = \begin{pmatrix} x \\ y \\ w \end{pmatrix}$, $F(z) = \begin{pmatrix} H(x, y, w) \\ x_1 w_1 \\ \vdots \\ x_n w_n \end{pmatrix}$

$$f(z) = \frac{1}{2} \| F(z) \|_2^2 \tag{3}$$

For any $z \in \mathbb{R}^{2n+m}$ and $\eta > 0$ we define the scaled projected gradient as:

$$g(z, \eta) = P_\Omega(z - \eta \nabla f(z)) - z,$$

where P_Ω denotes the Euclidean projection on Ω. If $z^* \in \Omega$ and $g(z^*, 1) = 0$ then $g(z^*, \eta) = 0$ for all $\eta > 0$ and z^* is said to be stationary.

For each $z \in \Omega$, $\mu_i \geq 0$, $i = 1, \ldots, n$ and $\theta \in [0, 1[$, we define the inexact Newton direction $d = d(x, y, w) = (d_x, d_y, d_w)$ as a solution of

$$H'(x, y, w)d + H(x, y, w) = r \tag{4}$$

and

$$x_i w_i + x_i (d_w)_i + w_i (d_x)_i = \mu_i, \;\; i = 1, \ldots, n, \tag{5}$$

where $r \in \mathbb{R}^{n+m+n}$ is such that

$$\|r\| \leq \theta \| F(x, y, w) \|. \tag{6}$$

The following lemma shows that, if $\sum_{i=1}^n \mu_i^2$ is small enough, every solution d of (4), (5) and (6) is a descent direction for f.

Lemma 1 *Consider $z = (x, y, w) \in \Omega$ and $\mu \in \mathbb{R}^n_+$, such that*

$$\|\mu\|^2 < \left(1 - \theta^2\right) \sum_{i=1}^n (x_i w_i)^2 \, . \tag{7}$$

Assume that $d \in \mathbb{R}^n$ satisfies (4)–(6). Then d is a descent direction for f at z. Moreover d is always a descent direction if $\mu = 0$ and z is not a solution of (1) and (2).

Proof Let

$$d = \begin{bmatrix} x' - x \\ y' - y \\ w' - w \end{bmatrix}.$$

By (4) and (5), we have

$$\left\| H(x,y,w) + H'(x,y,w) \begin{bmatrix} x'-x \\ y'-y \\ w'-w \end{bmatrix} \right\| \le \theta \sqrt{\|H(x,y,w)\|^2 + \sum_{i=1}^{n} (x_i w_i)^2} \tag{8}$$

and

$$x_i w_i + w_i(x_i' - x_i) + x_i(w_i' - w_i) = \mu_i, \; i = 1, \ldots, n. \tag{9}$$

Consider the convex quadratic function $q : \mathbb{R}^{n+m+n} \longrightarrow \mathbb{R}$ given by

$$q(\overline{x}, \overline{y}, \overline{w}) = \left\| H(x,y,w) + H'(x,y,w) \begin{bmatrix} \overline{x}-x \\ \overline{y}-y \\ \overline{w}-w \end{bmatrix} \right\|^2 + \sum_{i=1}^{n} \left[x_i w_i + w_i(\overline{x}_i - x_i) + x_i(\overline{w}_i - w_i) \right]^2 .$$

By (8) and (9),

$$q(x', y', w') \le \theta^2 \left[\|H(x,y,w)\|^2 + \sum_{i=1}^{n} (x_i w_i)^2 \right] + \sum_{i=1}^{n} \mu_i^2 .$$

On the other hand

$$q(x,y,w) = \|H(x,y,w)\|^2 + \sum_{i=1}^{n} (x_i w_i)^2 .$$

Let

$$\overline{\mu} = \sum_{i=1}^{n} \mu_i^2 .$$

Hence $q(x,y,w) > q(x',y',w')$ if $\overline{\mu} > 0$ satisfies (7) or if $\overline{\mu} = 0$ and (x,y,w) is not a solution of (1). Thus, by the convexity of q,

$$\nabla q(x,y,w)^\top d < 0.$$

Since $\nabla f(x,y,w) = \frac{1}{2} \nabla q(x,y,w)$, the lemma is proved. □

The following lemma provides a sufficient condition for generating a descent direction.

Lemma 2 *If* $(x,y,w) \in \Omega$ *is not a solution of* (1)*,* $\sigma_i \in]0, \sqrt{1-\theta^2}[$ *and*

$$\mu_i = \sigma_i \frac{x^\top w}{n} \tag{10}$$

for all $i = 1, \ldots, n$*, then any vector* d *satisfying* (4)*,* (5) *and* (6) *is a descent direction for* f*.*

Proof If $x^\top w > 0$ then μ_i defined by (10) satisfies (7). If $x^\top w = 0$ and (x, y, w) is not a solution of (1) the result follows by Lemma 1. □

The projected gradient Inexact Newton (PGIN) algorithm described below generates points lying in Ω and uses nonmonotone line–searches of Li–Fukushima type [25]. The line-search condition does not impose the objective function to decrease at every iteration.

Algorithm PGIN

Step 0: *Initial setup*: Consider $\gamma > 0$ and $\gamma_k > 0$ for all $k \in \mathbb{N}$ and such that

$$\sum_{k=0}^{\infty} \gamma_k = \gamma < \infty.$$

Let $\theta \in [0, 1[$, $\tau \in]0, 1]$, $\sigma \in]0, \sqrt{1-\theta^2}[$, $0 < \overline{\eta}_1 < \overline{\eta}_2$, $\rho > 0$, $\beta \in]0, \frac{1}{2}[$, $c_{\text{big}} > c_{\text{small}} > 0$, $c_{\text{small}} < 1$. Let $z^0 = (x^0, y^0, w^0) \in \Omega$. Assume that

$$z^k = \left(x^k, y^k, w^k\right) \in \Omega, \;\; \eta_k \in [\bar{\eta}_1, \bar{\eta}_2]$$

and $\theta_k \in [0, \theta[$. Then, the steps for obtaining

$$z^{k+1} = \left(x^{k+1}, y^{k+1}, w^{k+1}\right) \in \Omega$$

or declaring finite convergence are as follows:

Step 1: *Declare finite convergence if the scaled projected–gradient is zero*: If $\|g(z^k, \eta_k)\| = 0$, terminate the execution of the algorithm. Otherwise, compute $z^{k+1} = (x^{k+1}, y^{k+1}, w^{k+1})$ by the following steps 2–6:

Step 2: *Inexact Newton direction*: Compute

$$d^k = \left(d_x^k, d_y^k, d_w^k\right) \in \mathbb{R}^{n+m+n}$$

satisfying

$$\left\| H'\left(x^k, y^k, w^k\right) d^k + H\left(x^k, y^k, w^k\right) \right\| \le \theta_k \left\| F\left(x^k, y^k, w^k\right) \right\| \tag{11}$$

and

$$x_i^k w_i^k + x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i = \mu_{k,i} \tag{12}$$

where

$$\mu_{k,i} = \sigma_{k,i} \frac{\left(x^k\right)^\top w^k}{n}$$

and

$$\sigma_{k,i} \in [0, \sigma] \tag{13}$$

for $i = 1, \ldots, n$. If such a direction d^k does not exist or if $\|d^k\| > c_{\text{big}}$, choose $\tau_k \in]\tau, 1]$ and go to Step 4.

Step 3: *Compute the maximum steplength*: Choose $\tau_k \in [\tau, 1[$. Compute

$$\alpha_k^{\text{break}} = \max\left\{\alpha \ge 0 : \;\; z^k + \alpha d^k \in \Omega\right\} \tag{14}$$

and

$$\alpha_k^{\max} = \min\left\{1, \tau_k \alpha_k^{\text{break}}\right\}. \tag{15}$$

If $\alpha_k^{\max} \le c_{\text{small}} \min\left\{1, \|d^k\|\right\}$, go to Step 4. Otherwise, go to Step 5.

Step 4: *Projected gradient direction*: Compute (or re-define) $d^k = g(z^k, \eta_k)$ and set $\alpha_k^{\max} = \tau_k$.

Step 5: *Line–search*: Set $\alpha = \alpha_k^{\max}$.

Step 5.1: If

$$\left\|F\left(z^k + \alpha d^k\right)\right\| \le \left\|F\left(z^k\right)\right\| - \rho\left\|\alpha d^k\right\|^2 + \gamma_k \tag{16}$$

set $\alpha_k = \alpha$ and go to Step 6.

Step 5.2: Choose $\alpha_{\text{new}} \in [\beta\alpha, (1-\beta)\alpha]$, set $\alpha = \alpha_{\text{new}}$ and go to Step 5.1. (Note that the loop 5.1–5.2 necessarily finishes because $\gamma_k > 0$.)

Step 6: *Compute the new iterate*: choose $z^{k+1} \in \Omega$ such that

$$\left\|F\left(z^{k+1}\right)\right\| \le \left\|F\left(z^k + \alpha_k d^k\right)\right\|. \tag{17}$$

End.

Remarks

1. The choice of α_{new} in the interval $[\beta\alpha, (1-\beta)\alpha]$ allows us to use safeguarded parabolic interpolation for decreasing α, when the sufficient descent condition (16) does not hold. This is generally more efficient than decreasing α by a constant factor.
2. A sufficient (but not necessary) condition for the existence of d^k satisfying (11)–(13) is the nonsingularity of the matrix associated with the linear system (11) and (12). The boundedness of $\|d^k\|$ is guaranteed if the inverse of this matrix is bounded. However, the uniform boundedness of the matrix inverses is not a necessary condition for the boundedness of $\|d^k\|$. As an example, take the trivial monotone linear complementarity problem defined by:

$$x - w = 0, x \ge 0, w \ge 0, xw = 0.$$

Starting from $x^0 = w^0 = 1$, the algorithm converges with a linear rate to the degenerate solution $(0, 0)$, the matrix inverse norm tends to infinity but d^k remains bounded and the Newton direction is always accepted. The sequence of matrices tend to the null matrix but remain well-conditioned throughout the process.
3. Due to (15), if $\tau, \tau_k < 1$ the iterates z^{k+1} always remain in interior of Ω ($x^{k+1} > 0, w^{k+1} > 0$). Forcing the iterates to satisfy this property could be important because Newton directions do not change the zero value of a variable, that is, $x_i^{k+1} = 0$ ($w_i^{k+1} = 0$) provided $x_i^k = 0$ ($w_i^k = 0$). Since gradient projection iteration may be followed by Newton iterations, we

also maintain this requirement when gradient projections are used. It is important to stress that interiority will not play any role in the convergence theory. It is interesting to observe that, for monotone nonlinear complementarity problems (NCP) the Jacobian matrix is always nonsingular if the iterate is interior [10].

4. We require that the point $z^k + \alpha_k d^k$ must be obtained by the backtracking search (16). However, the next iterate z^{k+1} is not required to be $z^k + \alpha_k d^k$. Of course, if z^{k+1} is chosen as:

$$z^{k+1} = z^k + \alpha_k d^k$$

then, z^{k+1} satisfies (17) but, many times, better choices are possible. For example, if affordable, we can define:

$$z^{k+1} = z^k + \alpha_k^* d^k,$$

where α_k^* is a solution of the one–dimensional optimization problem

$$\begin{gathered} \min \left\| F\left(z^k + \alpha d^k\right) \right\| \\ \text{subject to } 0 \le \alpha \le \alpha_k^{\max}. \end{gathered}$$

5. Recent research in spectral projection algorithms for minimization on convex sets [4–6, 22] recommends the so-called spectral choice for η_k:

$$\begin{array}{lll} k = 0 & \Longrightarrow \eta_k = 1 & \\ \forall k > 0 : & \eta_k = \begin{cases} P_{[\eta_{\min}, \eta_{\max}]}\left(\frac{\xi_k}{\nu_k}\right) & \text{if } \nu_k > 0 \\ \eta_{\max} & \text{otherwise} \end{cases} \end{array} \tag{18}$$

where $P_{[l,u]}(\alpha)$ represents the projection of $\alpha \in \mathbb{R}$ on the interval $[l, u]$, $\xi_k = \left(z^k - z^{k-1}\right)^\top \left(z^k - z^{k-1}\right)$, $\nu_k = \left(z^k - z^{k-1}\right)^\top \left(\nabla f(z^k) - \nabla f(z^{k-1})\right)$ and $\eta_{\min}, \eta_{\max}$ are small and large positive numbers respectively. In practice, $\eta_{\min} = 10^{-2}$ is appropriate in general. Furthermore, $\eta_{\max} = 10^2$ usually works well but in many uses larger values for $\eta_{\max}$ provide better results [22].

6. The direction d^k may be computed using direct or iterative algorithms. Details on this computation will be given in Section 5.

3 Global convergence

Lemma 3 *For any sequence generated by Algorithm PGIN,*

$$\lim_{k \to \infty} \|\alpha_k d^k\| = 0.$$

Proof Since $\gamma_k \longrightarrow 0$, the negation of the thesis leads to $\|F(z^k)\| < 0$ for some k. □

Theorem 1 *Let $\{z^k\}$ be a sequence generated by Algorithm PGIN and let z^* be a cluster point such that*

$$\lim_{k\in K_1} z^k = z^*,$$

where $K_1 \subset \mathbb{N}$ is an infinite subsequence of indices. Then:

1. *z^* is a stationary point of*

$$\begin{array}{c}\min\ f(z)\\ \textit{subject to } z \in \Omega.\end{array}$$

2. *If K_1 contains infinite many indices k such that d^k is computed as an inexact Newton direction, then $F(z^*) = 0$.*

Proof We first prove 2. Let K_2 be an infinite subset of K_1 such that d^k is computed as an inexact Newton direction for every $k \in K_2$. We first consider the case in which

$$\lim_{k\in K_2} d^k = 0. \tag{19}$$

By (11) and (12) we have

$$\left\| H'\left(x^k, y^k, w^k\right)d^k + H\left(x^k, y^k, w^k\right)\right\| \le \theta_k \left\| F\left(x^k, y^k, w^k\right)\right\| \tag{20}$$

and

$$x_i^k w_i^k + x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i = \sigma_{k,i} \frac{\left(x^k\right)^\top w^k}{n}.$$

Therefore

$$\left(x^k\right)^\top w^k + \sum_{i=1}^n x_i^k \left(d_w^k\right)_i + \sum_{i=1}^n w_i^k \left(d_x^k\right)_i \le \sigma \left(x^k\right)^\top w^k. \tag{21}$$

Taking limits in (21) we obtain $\left(x^k\right)^\top w^k \longrightarrow 0$ and, by (20),

$$\left\| H\left(x^k, y^k, w^k\right)\right\| \longrightarrow 0.$$

Since $(x^*, y^*, w^*) \in \Omega$, it follows that z^* is a solution of (1).

Now, suppose that (19) does not hold. Then, there exists $\varepsilon > 0$ and K_3, an infinite subset of K_2, such that

$$\left\| d^k \right\| \ge \varepsilon, \ \text{ for all } k \in K_3. \tag{22}$$

By Lemma 3 we have $\alpha_k d^k \longrightarrow 0$. Therefore

$$\lim_{k\in K_3} \alpha_k = 0. \tag{23}$$

Since d^k is computed by the inexact Newton method, $\|d^k\| \le c_{\text{big}}$ for all $k \in K_2$. Therefore, by (22), we can extract an infinite subset K_4 of K_3 such that

$$\lim_{k\in K_4} d^k = \overline{d} \ne 0.$$

Since d^k is computed as an inexact Newton direction, then

$$\alpha_k^{\max} \geq c_{\text{small}} \min\left\{1, \|d^k\|\right\} \geq c_{\text{small}} \min\{1, \varepsilon\}.$$

Therefore $\alpha_k^{\max}$ is bounded away from zero for $k \in K_4$. On the other hand, by (23), the accepted stepsize α_k tends to zero. Hence, by Step 5 of the algorithm, for all $k \in K_4$ large enough there exists α'_k such that

$$\lim_{k \in K_4} \alpha'_k = 0$$

and

$$\left\| F\left(z^k + \alpha'_k d^k\right) \right\| - \left\| F\left(z^k\right) \right\| > -\rho \left\| \alpha'_k d^k \right\|^2. \tag{24}$$

Multiplying both sides of (24) by $\|F(z^k + \alpha'_k d^k)\| + \|F(z^k)\|$ we obtain

$$\left\| F\left(z^k + \alpha'_k d^k\right) \right\|^2 - \left\| F\left(z^k\right) \right\|^2 > -\rho \left\| \alpha'_k d^k \right\|^2 \left(\left\| F\left(z^k + \alpha'_k d^k\right) \right\| + \left\| F\left(z^k\right) \right\| \right). \tag{25}$$

Hence dividing by α'_k,

$$\frac{f\left(z^k + \alpha'_k d^k\right) - f\left(z^k\right)}{\alpha'_k} > -\rho \alpha'_k \left\| d^k \right\|^2 \left(\left\| F\left(z^k + \alpha'_k d^k\right) \right\| + \left\| F\left(z^k\right) \right\| \right) \tag{26}$$

Taking limits and using the Mean Value Theorem, we obtain:

$$\nabla f(x^*, y^*, w^*)^\top \overline{d} \geq 0, \tag{27}$$

so $\overline{d}$ is not a descent direction. But, by continuity,

$$\left\| H'(x^*, y^*, w^*)\overline{d} + H(x^*, y^*, w^*) \right\| \leq \theta \left\| F(x^*, y^*, w^*) \right\| \tag{28}$$

and there exist μ_i, $i = 1, \ldots, n$, such that

$$x_i^* w_i^* + x_i^* \left(\overline{d}_w\right)_i + w_i^* \left(\overline{d}_x\right)_i = \mu_i. \tag{29}$$

On the other hand, for $k = 0, 1, 2, \ldots$,

$$x_i^k w_i^k + x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i = \sigma_{k,i} \frac{\left(x^k\right)^\top w^k}{n}. \tag{30}$$

Therefore, taking limits appropriately,

$$\mu_i = \overline{\sigma}_i \frac{(x^*)^\top w^*}{n} \tag{31}$$

for some $\overline{\sigma}_i \in [0, \sigma]$, $i = 1, \ldots, n$. Therefore, $\overline{d}$ satisfies (28), (29), (31) and is not a descent direction. Then, by Lemma 2, (x^*, y^*, w^*) is a solution of (1).

We now prove 1. We only need to consider the case in which there exists k_0 such that d^k is the projected–gradient direction for all $k \in K_1$, $k \geq k_0$. We first suppose that, for some subsequence $K_5 \subset K_1$, $\lim_{k \in K_5} d^k = 0$. Considering a convenient subsequence of $\{\eta_k\}$ converging to $\eta \in [\overline{\eta}_1, \overline{\eta}_2]$, we obtain that

$$\left\| P(z^* - \eta \nabla f(z^*)) - z^* \right\| = 0,$$

so z^* is a stationary point of f over Ω.

Now suppose that $\|d^k\|$ is bounded away from 0 for $k \in K_1$. We take a convergent subsequence of $\{\eta_k\}$ so that d^k converges to $\bar{d} = P(z^* - \eta\nabla f(x^*)) - x^*$ along that subsequence.

By Lemma 3 we have that $\lim_{k\in K_1} \alpha_k = 0$. But $\alpha_k^{max} = \tau_k$ is bounded away from zero, therefore, by Steps 5.1 and 5.2, there exists α'_k such that

$$\lim_{k\in K_1} \alpha'_k = 0$$

and (24) holds for k large enough. Therefore, as in (24)–(27), we obtain:

$$\nabla f(x^*, y^*, w^*)^\top \bar{d} \geq 0.$$

Since $\bar{d} = g(z^*, \eta)$ this implies that z^* is stationary. □

Consider the Variational Inequality Problem over a convex set

$$\begin{array}{l}\text{Find } \overline{x} \in \mathcal{K} \text{ such that}\\ G(\overline{x})^\top (x - \overline{x}) \geq 0, \forall x \in \mathcal{K},\end{array} \tag{32}$$

where $G : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ is a continuously differentiable mapping, $g_i : \mathbb{R}^n \longrightarrow \mathbb{R}^1, i = 1, \ldots, l$ are convex twice smooth functions on $\mathbb{R}^n$ and

$$\mathcal{K} = \left\{x \in \mathbb{R}^n : \; Ax = b, x \geq 0, \; g_i(x) \leq 0, \; i = 1, \ldots, l\right\}.$$

If $\mathcal{K}$ satisfies a constraint qualification, this problem is equivalent to the following CP problem:

$$\begin{array}{l}G(x) = A^\top y - \nabla g(x)\mu + w\\ Ax = b\\ g(x) + \alpha = 0\\ x \geq 0, \;\; \mu \geq 0, \;\; w \geq 0, \;\; \alpha \geq 0\\ x^\top w = 0\\ \mu^\top \alpha = 0,\end{array}$$

where $g(x) = (g_1(x), \ldots, g_p(x))$, $\nabla g(x) = (\nabla g_1(x), \ldots, \nabla g_p(x)) \in \mathbb{R}^{n\times l}$, $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, $\mu \in \mathbb{R}^l$, $w \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}^l$ is a vector of slack variables for the constraints $g(x) \leq 0$. The natural merit function for this CP takes the form:

$$\Phi(x, w, y, \beta, \alpha) = \left\|G(x) + \nabla g(x)\mu - A^\top y - w\right\|^2 + \|Ax - b\|^2 + \|g(x) + \alpha\|^2$$
$$+ \sum_{i=1}^{n} (x_i w_i)^2 + \sum_{i=1}^{l} (\alpha_i \mu_i)^2.$$

Observe that, replacing x by (x, μ) and w by (w, α), the merit function Φ coincides with the merit function f defined in (3).

Moreover, we may write:

$$\Omega = \left\{(x, y, w, \mu, \alpha) \in \mathbb{R}^{2n+m+2l} : \; x \geq 0, \; w \geq 0, \; \mu \geq 0, \; \alpha \geq 0\right\}.$$

The following result has been established in [3].

Theorem 2 *If G is monotone on the nullspace of A, $\mathcal{K} \neq \emptyset$ and $\mathcal{K}_1 = \{x \in \mathbb{R}^n : \ Ax = b\,, \ x \geq 0\}$ is bounded, then every stationary point of Φ over Ω is a solution of* (32).

Hence the algorithm is able to find a solution of a VI under the condition stated in the theorem. A KKT point of a Nonlinear Program (NLP) with a continuously differentiable function $h : \mathbb{R}^n \longrightarrow \mathbb{R}^1$ can be stated as a VI, where $G = \nabla h(x)$. Therefore the algorithm can also find KKT points of convex nonlinear programs.

4 The linear case

In this section we address the linear case of CP in which H is an affine mapping of the form

$$H(x, y, w) = A_1 x + A_2 y + A_3 w - q.$$

The first issue to investigate is whether the maximum stepsize $\alpha_{\max}$ can be accepted during the whole procedure without the need of the line–search in Step 5. In this case, Steps 6 and 7 could be replaced by:

$$z^{k+1} = z^k + \alpha^k_{\max} d^k.$$

To do this, it would be desirable that

$$\left\| F\left(z^k + \alpha^k_{\max} d^k\right) \right\| \leq \left\| F\left(z^k\right) \right\| \tag{33}$$

at each iteration of the algorithm. The following example shows that such a result does not hold in general.

Consider the strictly monotone linear complementarity problem defined by $n = 2, m = 0$ and

$$H(x, w) = \begin{bmatrix} 2x_1 - w_1 \\ \frac{4}{3} x_2 - w_2 \end{bmatrix}.$$

If $x^0 = (1, 3)$ and $w^0 = (2, 4)$, then $\left(x^0\right)^\top w^0 = 14$ and

$$\left(x^0_1 w_1\right)^2 + \left(x^0_2 w_2\right)^2 = 148.$$

Since $H(x^0, w^0) = 0$, then

$$\| F(x^0, w^0) \|^2 = 148.$$

We take $\theta = 0$ in (11) and we use the common choice for μ in interior–point algorithms

$$\mu_{0,i} = \mu = \sigma \frac{\left(x^0\right)^\top w^0}{2}$$

with $\sigma\in]0,1[$. Solving (11) and (12) we get

$$d_x^0=\begin{bmatrix}\frac{\mu+2}{4}-1\\ \frac{\mu+12}{8}-3\end{bmatrix}$$

$$d_w^0=\begin{bmatrix}\frac{\mu+2}{2}-2\\ \frac{\mu+12}{6}-4\end{bmatrix}$$

Therefore, $\alpha_k^{\max}=1$,

$$\|H(x^0+d_x^0,w^0+d_w^0)\|=0$$

and

$$\|F(x^0+d_x^0,w^0+d_w^0)\|^2=\frac{(\mu+2)^4}{64}+\frac{(\mu+12)^4}{2304}.$$

This implies that, for $\sigma\in]0.98,1[$,

$$\|F(x^0+d_x^0,w^0+d_w^0)\|^2\geq 151>148=\|F(x^0,w^0\|^2.$$

Hence (33) does not hold in this example.

The reason why the previous counterexample does not verify the descent property (33) is that μ is not small enough. In the following theorem we prove that, if H is an affine function and $\sigma_{k,i}$ is small enough so that

$$\mu_{k,i}<x_i^k w_i^k \tag{34}$$

for all $i=1,\dots,n$, then, not only (33) holds, but also the minimizer of $\|F(z^k+\alpha d^k)\|$ for $\alpha\in[0,\alpha_k^{\max}]$ occurs at $\alpha=\alpha_k^{\max}$.

It is interesting to observe that this property does not depend on the monotonicity of the problem. The theorem says that, if the direction d^k is defined satisfying the requirements of Step 2 of the algorithm with $\theta_k=0$ and (34), then the maximum allowed steplength minimizes the objective function and, so, the point defined by this steplength is admissible as z^{k+1}. The convergence theory requires $\|d^k\|\leq c_{big}$, but this assumption is not used in proof of the following theorem.

Theorem 3 *Let*

$$H(x,y,w)=A_1x+A_2y+A_3w-q \tag{35}$$

and suppose that, for $\theta=0$ and $\mu_{k,i}$ satisfying (34), the iteration (11)–(13) is well defined. If

$$\varphi(\alpha)=\|F(z^k+\alpha d^k)\|^2,$$

then $\varphi(\alpha)$ is strictly decreasing for $\alpha\in[0,\alpha_k^{\max}]$.

Proof We write the function $\varphi(\alpha)$ in the form

$$\varphi(\alpha)=\varphi_0(\alpha)^2+\sum_{i=1}^n\varphi_i(\alpha)^2,$$

where

$$\varphi_0(\alpha) = \left\| H\left(x^k + \alpha d_x^k, y^k + \alpha d_y^k, w^k + \alpha d_w^k\right) \right\|$$

and

$$\varphi_i(\alpha) = \left(x^k + \alpha d_x^k\right)_i \left(w^k + \alpha d_w^k\right)_i \tag{36}$$

for $i = 1, \ldots, n$. By (35) and (11) with $\theta = 0$, we have that $\varphi_0(1) = 0$. Since $\varphi_0(\alpha)$ is a nonnegative convex quadratic function in one variable, then it decreases monotonically between 0 and 1. Since $\alpha_k^{\max} \leq 1$, we have that $\varphi_0(\alpha)$ decreases monotonically between 0 and $\alpha_k^{\max}$.

Now we prove that each function $\varphi_i(\alpha)^2$, $i = 1, \ldots, n$, decreases monotonically between 0 and $\alpha_k^{\max}$. By (36),

$$\varphi_i(\alpha) = x_i^k w_i^k + \alpha \left[x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i\right] + \alpha^2 \left(d_w^k\right)_i \left(d_x^k\right)_i .$$

This function is quadratic in the variable α and its first and second derivatives are given by

$$\varphi_i'(\alpha) = \left[x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i\right] + 2\alpha \left(d_w^k\right)_i \left(d_x^k\right)_i \tag{37}$$

and

$$\varphi_i''(\alpha) = 2 \left(d_w^k\right)_i \left(d_x^k\right)_i \tag{38}$$

respectively. Furthermore, by (12) and (34),

$$\left[x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i\right] < 0. \tag{39}$$

Since $x_i^k > 0$ and $w_i^k > 0$, we have

$$\varphi_i'(0) < 0 \tag{40}$$

and

$$\left(d_w^k\right)_i < 0 \text{ or } \left(d_x^k\right)_i < 0. \tag{41}$$

If $\left(d_w^k\right)_i \left(d_x^k\right)_i = 0$, since $\varphi_i'(0) < 0$ and $\varphi_i(\alpha_k^{\max}) > 0$, then $\varphi_i(\alpha)^2$ decreases monotonically from 0 to $\alpha_k^{\max}$. On the other hand, if $(d_w^k)_i (d_x^k)_i \neq 0$, then the stationary point of $\varphi_i(\alpha)$ corresponding to a maximizer or minimizer is given by

$$\overline{\alpha} = -\frac{x_i^k \left(d_w^k\right)_i + w_i^k \left(d_x^k\right)_i}{2 \left(d_w^k\right)_i \left(d_x^k\right)_i} \tag{42}$$

and two cases may be considered.

Case 1: If $\left(d_w^k\right)_i \left(d_x^k\right)_i < 0$, then $\overline{\alpha}$ is a maximizer and $\overline{\alpha} < 0$. Since $\varphi_i(\alpha_k^{\max}) > 0$, then $\varphi_i(\alpha)^2$ decreases monotonically from 0 to $\alpha_k^{\max}$.

Case 2: If

$$\left(d_w^k\right)_i < 0 \text{ and } \left(d_x^k\right)_i < 0, \tag{43}$$

then, by the definition of $\alpha_k^{\max}$, we have

$$\alpha_k^{\max} < \min\left\{-\frac{x_i^k}{\left(d_x^k\right)_i}, -\frac{w_i^k}{\left(d_w^k\right)_i}\right\}. \tag{44}$$

As $\overline{\alpha} > 0$ is a stationary point, then

$$\overline{\alpha} = \frac{1}{2}\left[-\frac{x_i^k}{\left(d_x^k\right)_i} - \frac{w_i^k}{\left(d_w^k\right)_i}\right] > \alpha_k^{\max}.$$

Since $\varphi_i(\alpha_k^{\max}) > 0$ we can conclude that $\varphi_i(\alpha)^2$ decreases monotonically from 0 to $\alpha_k^{\max}$.

Then, the theorem is proved. □

Consider again the set

$$\mathcal{K} = \left\{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\right\} \tag{45}$$

with $A \in \mathbb{R}^{m\times n}$ full rank and $m < n$. Let the columns of $Z \in \mathbb{R}^{n\times(n-m)}$ be a basis of the nullspace of A. Let us consider the Affine Variational Inequality Problem

$$\begin{array}{l}\text{Compute } \overline{x} \in \mathcal{K} \\ \text{such that } (M\overline{x} + q)^\top (x - \overline{x}) \geq 0, \ \forall x \in \mathcal{K}\end{array} \tag{46}$$

where $M \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$. As before, $\overline{x}$ is a solution of (46) if and only if $(\overline{x}, \overline{y}, \overline{w})$ is a solution of the problem:

$$\begin{aligned} w &= q + Mx - A^\top y \\ 0 &= Ax - b \\ x^\top w &= 0 \\ x, w &\geq 0. \end{aligned} \tag{47}$$

The following result has been established in [3].

Theorem 4 *Let $(\overline{x}, \overline{y}, \overline{w})$ be a stationary point over Ω of the merit function*

$$f(x, y, w) = \|q + Mx - A^\top y - w\|_2^2 + \|Ax - b\|_2^2 + \sum_{i=1}^{n}(x_i w_i)^2 \tag{48}$$

If the columns of Z form a basis of the nullspace of A and $Z^\top MZ$ is a positive semi–definite matrix, then

(i) *If $f(\bar{x}, \bar{y}, \bar{w}) = 0$, then $(\bar{x}, \bar{y}, \bar{w})$ is a solution of (47).*
(ii) *If $f(\bar{x}, \bar{y}, \bar{w}) > 0$, then the problem (47) is infeasible.*

The same result holds if M is a P–matrix [8], that is, if all the principal minors of M are positive. Therefore this theorem confirms a well known result [8] that the LCP (47) with a P or PSD matrix has at least one solution provided is feasible. In particular an Affine Variational Inequality with a P–matrix has at least a solution provided it is feasible. Furthermore, this solution is unique [8].

Let us now consider a Quadratic Program (QP):

$$\begin{aligned} &\text{Minimize } q^\top x + \tfrac{1}{2} x^\top M x = f(x) \\ &\text{subject to } Ax = b \\ &\qquad\qquad x \geq 0. \end{aligned} \tag{49}$$

The KKT conditions for this Program consist of a LCP of the form of (47). By Theorem 4, if f is convex over the nullspace of A, then a stationary point $(\overline{x}, \overline{y}, \overline{w})$ of the merit function (48) over Ω solves the convex QP, in the sense that:

1. If $f(\overline{x}, \overline{y}, \overline{w}) = 0$, then $\overline{x}$ is a global minimum of the quadratic program;
2. If $f(\overline{x}, \overline{y}, \overline{w}) > 0$, then the quadratic program is primal or dual infeasible.

The same conclusion applies for a Linear Program (LP)

$$\begin{aligned} &\text{Minimize } c^\top x \\ &\text{subject to } Ax = b \\ &\qquad\qquad x \geq 0. \end{aligned} \tag{50}$$

Computational experience to be discussed in the next section shows that for feasible monotone LCPs, convex quadratic programs and linear programs with at least a strictly feasible solution, the algorithm never uses projected–gradient directions during the whole iterations. However, the projected–gradient iterations must be required for LCPs that are infeasible or feasible without a strictly feasible solution. In particular, infeasible LPs and convex quadratic programs also require the projected–gradient iterations. In this latter case, the algorithm should terminate with a stationary point with a positive value for the natural merit function.

Since the projected–gradient algorithm is in general quite slow, we recommend the following two–phase procedure for dealing with monotone LCPs and convex quadratic and linear programs:

Phase 1: Apply the algorithm PGIP. If a solution is found by solely using interior–point directions, terminate.

Phase 2: If an interior–point search direction cannot be computed, switch to the following feasibility problem

$$\begin{aligned} \min\ &\|H(x, y, w)\|_2^2 = f(x, y, w) \\ \text{subject to } &x \geq 0,\ w \geq 0. \end{aligned}$$

Solve this Quadratic Program by the PGIP algorithm with the current point as initial one to find a stationary point (global minimum over Ω) $(\overline{x}, \overline{y}, \overline{w})$. Two terminations are then possible:

Termination 1: $f(\overline{x}, \overline{y}, \overline{w}) > 0$ so the original LCP is infeasible (LP or convex QP are either primal or dual infeasible).

Termination 2: $f(\overline{x}, \overline{y}, \overline{w}) = 0$, so a feasible solution of the LCP (a primal and dual feasible solution of the LP or the QP) has been found. If this solution is complementary, then a solution of the LCP (optimal solution of the LP or convex QP) is at hand. Otherwise, projected–gradient iterations of the PGIP algorithm have to be applied until the end.

Computational experience has shown that Phase 2 when applied to infeasible linear and convex quadratic programs generally terminates in Termination 1, showing primal or dual infeasibility of the convex QP or LP. Furthermore the process never uses projected–gradient iterations. Other efficient and popular implementations of interior–point algorithms are not conclusive with respect to infeasibility or unboundedness of convex QPs and LPs.

5 Computational experiments

In this section we report some computational experiments that have been performed to highlight the strengths and weaknesses of PGIP algorithm. For that purpose, we coded the algorithm in Fortran 77, using Intel Fortran Compiler version 7.0 [21], with options `-O3 -tpp7 -xM -ip`, and ran all the experiments on a Linux 2.4.25 system featuring an AMD Athlon processor running at 1.6 Ghz with 256 M of RAM. For CPU time measuring the function `etime()` was used. The stopping criterion is $\|g(z^k, 1)\| < 10^{-6}$. We consider $c_{\text{big}} = 10^4$, $c_{\text{small}} = 10^{-4}$, $\gamma_k = \frac{1}{k^2}$, $\theta = 0.5$, $\sigma_{ki} = \sigma = \frac{1}{\sqrt{n}}$, $\rho = 10^{-1}$, $\beta = \frac{1}{4}$ and $\tau_k = \tau = 0.9995$. The initial iteration z^0 as a vector of ones of appropriate dimension, unless otherwise stated. All the implementations were done in double precision.

We have organized our experience in four sections, namely:

Experience 1: Solution of feasible monotone LCPs, and convex QPs and LPs with an optimal solution.
Experience 2: Solution of Nonmonotone LCPs.
Experience 3: Inexact versus Exact Newton's iteration for a Structured Convex Quadratic Optimization Problem.
Experience 4: Infeasible Linear Programs.

5.1 Experience 1—Solution of feasible monotone LCPs, convex QPs and LPs with an optimal solution

We started by testing the performance of the PGIP algorithm on the solution of the famous Murty's LCP [27]:

$$\begin{aligned} w &= q + Mx \\ x,\ w &\geq 0 \\ x^{\top} w &= 0 \end{aligned}$$

where

$$M = \begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ \vdots & \vdots & \ddots & \\ 2 & 2 & \dots & 1 \end{bmatrix}$$

and

$$q_i = \begin{cases} -1 & \text{if } i \in J \\ 0 & \text{otherwise} \end{cases}, \quad i = 1, \dots, n,$$

with $J \subset \{1, \dots, n\}$. So $A_1 = -M$, $A_2 = 0$ and $A_3 = I_n$ in the definition (35) of $H(x, y, w)$, where I_n is the identity matrix of order n. Due to the specific structure of the matrix M, it was possible to create an implementation of the algorithm that takes this into account and is able to solve quite large LCPs in a small amount of time. Furthermore, since M is a lower triangular matrix, an exact Newton direction ($\theta = 0$) can be computed in a very efficient manner.

Tables 1 and 2 display the results with problems of orders from 2,500 to 15,000 and degenerate components of the solution ranging from 0 to 75% of the variables. In this table, IT stands for the number of PGIP iterations performed, and CPU corresponds to the CPU time required for the execution. The PGIP algorithm showed a quite robust performance. In each iteration k, the exact Newton direction d^k has always been used, that is, no projected–gradient direction was required. Furthermore, $\alpha_k = \alpha_k^{\max}$ was always chosen, as the problem is linear. Finally the values FMERIT of the merit function at the solutions of the LCP computed by the algorithm are always quite small, which indicates that these solutions are accurate in terms of feasibility and complementarity gap.

Table 1 Performance of algorithm PGIP solving Murty LCPs

n		PGIP			
		0%	25%	50%	75 %
2,500	IT	20	24	27	25
	CPU	0.65	0.78	0.86	0.79
	FMERIT	2×10^{-7}	3×10^{-7}	3×10^{-7}	2×10^{-7}
5,000	IT	20	30	31	28
	CPU	2.36	3.32	3.56	3.23
	FMERIT	4×10^{-7}	2×10^{-7}	3×10^{-7}	3×10^{-7}
7,500	IT	20	31	31	26
	CPU	5.04	7.98	7.86	6.61
	FMERIT	5×10^{-7}	5×10^{-7}	4×10^{-7}	6×10^{-7}
10,000	IT	20	26	31	32
	CPU	8.14	10.62	12.71	13.01
	FMERIT	6×10^{-7}	4×10^{-7}	6×10^{-7}	5×10^{-7}
12,500	IT	20	22	32	32
	CPU	19.03	29.05	30.47	30.66
	FMERIT	8×10^{-7}	7×10^{-7}	7×10^{-7}	8×10^{-7}

Table 2 Performance of algorithms PATHLCP1 (PATHLCP with presolve) and PATHLCP2 (PATHLCP without presolve) solving Murty LCPs

n		PATHLCP 1				PATHCP 2			
		0%	25%	50%	75 %	0%	25%	50%	75 %
2,500	IT	–	–	–	–	26	24	30	30
	CPU	1.24	1.24	1.24	1.24	11.93	11.54	12.01	12.01
	FMERIT	4×10^{-7}	6×10^{-7}	5×10^{-7}	4×10^{-7}	4×10^{-7}	6×10^{-7}	5×10^{-7}	4×10^{-7}
5,000	IT	–	–	–	–	26	26	30	30
	CPU	6.81	6.81	6.81	6.81	83.29	83.29	99.12	99.12
	FMERIT	6×10^{-7}	8×10^{-7}	7×10^{-7}	6×10^{-7}	6×10^{-7}	8×10^{-7}	7×10^{-7}	6×10^{-7}
7,500	IT	–	–	–	–	*	*	*	*
	CPU	20.00	20.00	20.00	20.00	*	*	*	*
	FMERIT	8×10^{-7}	7×10^{-7}	6×10^{-7}	7×10^{-7}	*	*	*	*

In order to gain a better idea of the performance of the PGIP algorithm we have also tried to solve all the Murty's problems by the code PATHLCP [14]. Two versions of this code have been tested, which differ on the use of a Presolve procedure (PATHLCP1) or not (PATHLCP2). The numerical performance of these codes is displayed in Table 2. Both versions have been unable to solve the LCPs of orders 10,000 and 12,500 due to lack of memory of our hardware resources, and the same happened with PATHLCP2 for $n = 7{,}500$ (this is marked by "*" in Table 2). For the remaining problems PATHLCP2 seems to be competitive with PGIP in terms of the number of iterations but not in CPU time. The presolve technique of PATHLCP has been able to solve the LCPs in all those instances but (this is denoted by "–" in Table 2) with a higher execution time than PGIP.

Since PATHLCP solved these LCPs in the presolve phase, we decided to test the LCPs mentioned in [27, p. 380], where

$$M = \begin{bmatrix} 6 & -4 & 1 & & & & \\ -4 & 6 & -4 & 1 & & & \\ 1 & -4 & 6 & -4 & 1 & & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots & \ddots & 1 \\ & & & 1 & -4 & 6 & -4 \\ & & & & 1 & -4 & 6 \end{bmatrix}$$

and q is given by $q_i = a_{i+1} - a_i$, $i = 1, \ldots, n$, with $a \in \mathbb{R}^{n+1}$ a vector whose components are random numbers in the interval [0, 30]. We also developed an implementation of PGIP that took advantage of the structure of this problem for the computation of the Newton direction. Table 3 displays the performance of PGIP and PATHLCP (with presolve) algorithms for solving these LCPs for different dimensions n. As before, PATHLCP was not capable of solving the largest instances (this is marked by $*$ in Table 3). Furthermore, PGIP was much more efficient than PATHLCP for all the problems.

We now discuss the performance of PGIP on linear programs and convex quadratic programs with an optimal solution taken from the well–known `NETLIB` collection [17]. In Table 4, we show the performance of the PGIP algorithm for eight of these test problems. Again the algorithm performed

Table 3 Performance of PATHLCP and PGIP for pentadiagonal LCPs

n	PATHLCP			PGIP		
	IT	CPU	FMERIT	IT	CPU	FMERIT
500	324	0.05	6×10^{-7}	32	0.01	2×10^{-7}
1,000	474	0.16	1×10^{-7}	41	0.01	3×10^{-7}
2,000	5,660	4.17	2×10^{-7}	53	0.02	1×10^{-7}
3,000	7,020	7.93	1×10^{-7}	61	0.04	5×10^{-7}
4,000	$*$	$*$	$*$	67	0.05	7×10^{-7}
5,000	$*$	$*$	$*$	72	0.07	7×10^{-7}

Table 4 Performance of algorithm PGIP on some linear programs from the NETLIB collection

Name	n	m	nza	IT	CPU	FMERIT
AFIRO	51	27	102	13	0.05	2.0×10^{-8}
BLEND	114	74	522	21	0.26	3.1×10^{-8}
DIET	9	3	21	22	0.02	2.6×10^{-8}
SHARE2B	162	96	777	24	0.44	1.6×10^{-8}
PILOT	6,103	2,684	46,861	62	105.35	2.0×10^{-7}
PILOT4	1,400	687	5,818	41	14.22	3.5×10^{-6}
PILOTNOV	2,990	1,519	14,419	76	89.12	4.7×10^{-6}
PILOT87	8,478	3,828	78,545	94	210.14	2.8×10^{-6}

quite well and has always used the Newton iteration. Furthermore $\alpha^k_{\max}$ has always been used, as the corresponding complementarity problem is linear.

The same performance has been shown on the solution of four convex quadratic programs (CQP) with linear constraints taken from the CUTEr collection [19]. As before the CP represents the KKT conditions associated to these CQPs.

For all the tests the value FMERIT of the merit function at the solution is quite small, which indicates that the algorithm has been able to find accurate solutions in terms of primal and dual infeasibility and complementarity gap (Table 5).

A third set of experiments has been done on the solution of a Convex Quadratic Program with linear constraints and two convex quadratic constraints of the form

$$\begin{aligned} \min\ & c^\top x + \tfrac{1}{2} x^\top M x \\ \text{subject to}\ & Ax = b \\ & h^\top x + \tfrac{1}{2} x^\top H x \le h_0 \\ & g^\top x + \tfrac{1}{2} x^\top G x \le g_0 \\ & 0 \le x \le u \end{aligned}$$

where H, G and M are PSD matrices of order n, $c, h, g \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m\times n}$ and $h_0, g_0 \in \mathbb{R}^1$. As discussed in [31], this problem has been taken from a a thermal dispatch model with environmental and take–or–pay constraints. The KKT conditions lead into a nonlinear CP. The results of the experience of processing this algorithm for a smaller (PSMALL) and a larger (PLARGE) instance of the model are included in Table 6, where algorithm PGIP is compared with MINOS (an active-set solver) [26], that was run within GAMS [7]

Table 5 Performance of algorithm PGIP on some convex quadratic problems from the CUTEr collection

Problem	m	n	IT	ITPRJ	NFKS	CPU	FMERIT
QAFIRO	27	51	27	0	0	0.02	2×10^{-9}
QPCBLEND	74	114	32	0	0	0.08	3×10^{-14}
QRECIPE	91	204	67	0	0	0.12	9×10^{-15}
QSHARE1B	117	253	24	0	0	0.18	6×10^{-7}

Table 6 Comparative performance of PGIP, MINOS and LOQO on problems PSMALL and PLARGE

	PSMALL $n = 600, m = 60$		PLARGE $n = 18{,}720, m = 1{,}872$	
	IT	CPU	IT	CPU
PGIP	36	0.08	94	148.34
MINOS	304	6.22	NS	NS
LOQO	29	0.10	69	284.75

and LOQO [35, 36], a predictor–corrector interior–point solver coded in ANSI C. The starting point for both IP and LOQO was set as $x_i = \frac{u_i}{2}, i = 1, \ldots, n$.

The results presented in Table 6 seem to indicate that LOQO performs less iterations to attain the solution, although PGIP is less time–consuming. These two algorithms are much more efficient than MINOS, that was not able to solve Problem PLARGE within our hardware resources. This is marked by NS in Table 6. As before, PGIP algorithm has always used the Newton search direction in each iteration, that is, no projected–gradient iteration was required. Since the CP is nonlinear, the line–search procedure had to be used in each iteration. However, the results indicate that, as in the linear case, $\alpha^k_{\max}$ has always been chosen as stepsize in each iteration of the procedure.

To terminate the first set of experiments, we have processed some linear and convex quadratic programs that are feasible but have no strictly feasible solutions. For this type of problems the algorithm required projected–gradient iterations to terminate in a stationary point of the natural merit function over Ω that gives an optimal solution for the linear or convex quadratic program. However, preprocessing usually avoids this type of occurrence by eliminating some constraints or variables of the program. So, as final conclusion of this set of experiments, we claim that in practice the projected–gradient interior–point algorithm always finds the optimal solution of linear and convex quadratic programs and solutions of LCPs when they exist, by simply using the interior–point direction and the maximum stepsize in each iteration.

5.2 Experience 2—solution of nonmonotone LCPs

As is discussed in [11, 24], the Knapsack Problem consists of finding a vector $z \in \mathbb{R}^n$ such that

$$a^\top z = b\,,\ z_i \in \{0, 1\},\ i = 1, \ldots, n,$$

where $a \in \mathbb{R}^n$ and $b > 0$ are given. This problem is equivalent to a LCP [24]

$$\begin{gathered} w = q + Mx \\ x \geq 0,\ w \geq 0 \\ x^\top w = 0 \end{gathered}$$

where

$$q = \begin{bmatrix} p \\ p \\ \vdots \\ p \\ -b \\ b \end{bmatrix}, \quad M = \begin{bmatrix} B & 0 & \dots & 0 & 0 & 0 \\ 0 & B & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & B & 0 & 0 \\ & & \bar{a}^T & & & \\ & & -\bar{a}^T & & & \end{bmatrix}$$

and

$$p = \begin{bmatrix} 0 \\ 0 \\ -1 \\ 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{a} = (\bar{a}_1, \dots, \bar{a}_{4n+2})^\top \in \mathbb{R}^{4n+2}$$

with

$$\bar{a}_i = \begin{cases} a_i \text{ if } i = 4j-3, \ \ j = 1, \dots, n \\ 0 \text{ otherwise} \end{cases}$$

It is possible to prove that M is P_0 matrix, that is, all principal minors of M are nonnegative. However, this matrix is not PSD and the LCP is nonmonotone. We solved these LCPs with orders of $n = 2$ (KNAP2) and $n = 10$ (KNAP10). The computational experience indicates that the algorithm converged to a stationary point of the merit function over Ω that is not a complementarity solution. Furthermore projected–gradient iterations were used from the beginning until the end.

A second experience with a nonmonotone LCP has been done with a nonconvex quadratic program (NCQP) [16] of the form (49), where

$$M = \begin{bmatrix} 0 & -1 & -1 & 0 & -1 & 0 & 0 & 0 & -1 & -1 \\ -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 \\ -1 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & -1 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$b = 1, \quad q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^\top .$$

It is easy to see that $Z^\top M Z$ is not a PSD matrix for any basis Z of the nullspace of A. For this problem, the projected–gradient direction was used in all 52 iterations of the PGIP algorithm, leading to a stationary point of the merit function with a value of 2.5×10^{-1}. This means that the algorithm was not able to find a solution of the LCP associated with the Karush–Kuhn–Tucker conditions of the quadratic program. Furthermore, $\eta_{\max} = 10^8$ was the value that produced the best performance. These two examples clearly demonstrate that the PGIP algorithm should not be recommended to process nonmonotone complementarity problems and nonconvex programs. On the other hand, PATHLCP has been able to solve all these LCPs in a small number of iterations and execution time (see Table 7).

5.3 Experience 3—exact versus Inexact Newton Iteration

Next we show the results of the application of PGIP on the solution of a large–scale Structured Convex Quadratic Programming (CQP) that arises on the so–called Linear Plate Obstacle Problem [20, 23], describing the equilibrium of a thin elastic clamped plate, that may come into contact with a rigid obstacle, by the action of a vertical force. This problem can be formulated [12] as a strictly convex Quadratic Program of the form

$$\begin{array}{l} \text{Minimize } \frac{1}{2} x_J^\top C_{JJ} z_J - x_J^\top F_J \\ \text{subject to } x_{J_1} \geq \Psi \end{array}$$

with $J, L \subseteq \{1, \ldots, 4n\}$ such that $\{1, \ldots, 4n\} = J \cup L$ and $J \cap L = \emptyset$, and $J_1 \subset J$. Furthermore $C \in \mathbb{R}^{4n \times 4n}$ and $F \in \mathbb{R}^{4n}$ are obtained assembling a given finite element matrix $C^e \in \mathbb{R}^{16 \times 16}$ and a given element force vector $F^e \in \mathbb{R}^{16}$ [12]. As before, the PGIP algorithm solves a CP given by the KKT conditions associated with this CQP.

Since C_{JJ} is a symmetric positive definite matrix, an inexact Newton direction for the PGIP algorithm may be computed by using the Preconditioned Conjugate Gradient algorithm [18]. As discussed in [32], the stopping criterion for the Preconditioned Conjugate Gradient (PCG) algorithm plays an important role on the overall performance of the Interior Point algorithm. In our experiments the PCG is stopped whenever the residual $\|r_p\|$ satisfies the following inequality

$$\|r_p\| \leq \beta \big(\|C_{JJ} x_J - w_J - F_J\|_\infty + \max_{i \in J_1} \{(x_i - \Psi_i) w_i\} \big)$$

with $\beta = 10^{-2}$.

The implementation of the linear system solver takes advantage of the structure of this problem. For a given iterate $x = (x_{J_1}, x_{J_f})$ and $w = (w_{J_1}, w_{J_f})$ let $x \circ w = (x_{J_1} \circ w_{J_1}, x_{J_f} \circ w_{J_f}) = (x_i w_i)_{i \in J_1 \cup J_f}$. Then it is possible to prove

Table 7 Performance of PATHLCP (with presolve) on nonmonotone LCPs

	Test problems		
	KNAP2	KNAP10	NCQP
IT	18	32	20
CPU	0.35	0.81	0.40

that the search direction in this iteration can be found by solving a linear system of the form

$$\begin{bmatrix} X_{J_1}C_{J_1J_1}X_{J_1} + V_{J_1} & X_{J_1}C_{J_1J_f}X_{J_f} \\ X_{J_f}C_{J_fJ_1}X_{J_1} & X_{J_f}C_{J_fJ_f}X_{J_f} + V_{J_f} \end{bmatrix} \begin{bmatrix} \Delta x_{J_1} \\ \Delta x_{J_f} \end{bmatrix} = \begin{bmatrix} p_{J_1} \\ p_{J_f} \end{bmatrix}$$

where X_{J_1}, X_{J_f}, V_{J_1} and V_{J_f} are diagonal matrices whose elements are the components of the vectors x_{J_1}, x_{J_f}, $x_{J_1} \circ w_{J_1}$ and $x_{J_f} \circ w_{J_f}$ respectively, $C_{J_1J_1}$ and $C_{J_fJ_f}$ are band matrices and p is an appropriate vector. We consider two different preconditioning strategies for this linear system:

DIAG: Diagonal preconditioner;
BAND: The preconditioner is the band matrix

$$\begin{bmatrix} X_{J_1}C_{J_1J_1}X_{J_1} + V_{J_1} & 0 \\ 0 & X_{J_f}C_{J_fJ_f}X_{J_f} + V_{J_f} \end{bmatrix}$$

We now discuss some issues related to the implementation of the Interior Point algorithm for this specific problem. The central path parameter μ_{kl} at the k-th iteration of the Interior Point algorithm is given by

$$\mu_{kl} = \sigma_{kl} \frac{\sum_{i \in J_1} \left(x_i^{(k)} - \Psi_i^{(k)}\right) w_i^{(k)}}{|J_1|}$$

with $0 < \sigma_{kl} < 1$ given, and $l \in \{1, 2, \ldots, 4n\}$.

The stopping criterion that is implemented in the Interior Point algorithm is

$$\mu_{kl}\left((n_x - 2) \times \left(n_y - 2\right)\right) \leq \varepsilon_1 \text{ and } \|C_{JJ}x_J - w_J - F_J\|_\infty \leq \varepsilon_2,$$

where $n_x \times n_y$ is the number of nodes of the grid associated with the finite element discretization of the problem. In the forthcoming experiences we

Table 8 Performance of the interior point algorithm on the solution of the linear plate obstacle problem

	Grid	N	IT	CPU	ITAVG
DIAG	25×25	2,404	13	0.71	1.35×10^{-2}
	50×50	9,804	13	11.71	1.127×10^{-2}
	75×75	22,204	7	23.11	7.141×10^{-3}
	100×100	39,604	6	46.98	5.209×10^{-3}
	125×125	62,004	6	98.94	4.155×10^{-3}
	150×150	89,404	6	193.26	2.27×10^{-3}
	175×175	121,804	5	288.93	3.011×10^{-3}
BAND	25×25	2,404	12	0.88	1.454×10^{-2}
	50×50	9,804	15	9.99	4.391×10^{-3}
	75×75	22,204	11	47.76	5.953×10^{-3}
	100×100	39,604	8	65.98	5.49×10^{-3}
	125×125	62,004	6	125.87	2.909×10^{-3}
	150×150	89,404	6	162.07	2.27×10^{-3}
	175×175	121,804	5	234.70	2.148×10^{-3}

Table 9 Performance of algorithm PGIP on primal or dual infeasible linear problems

Name	n	m	IT	CPU	FMERIT
ITEST2	13	9	59	0.03	202.98
ITEST6	17	11	11	0.14	1.90
BGPRTR	40	20	401	1.42	36,246
KLEIN1	108	54	325	2.61	406.21
VOL1	459	177	371	9.12	8,125.5

consider $\varepsilon_1 = \varepsilon_2 = 10^{-6}$. These experiences are reported in Table 8, for which ITAVG is given by

$$\text{ITAVG} = \frac{\text{ITPCGA}}{|J| \times \text{IT}}$$

and ITPCGA corresponds to the total number of iterations of the Preconditioned Conjugate algorithm performed during the application of the Interior Point algorithm. This table seems to indicate that the Interior Point algorithm is quite adequate for this type of problems, as this number ITAVG is always quite small. Furthermore the band preconditioner seems to be superior to the diagonal one. Finally, this inexact version is much more efficient than the exact one used in [12]. So this computational study shows that for quite large–scale and structured linear complementarity problems, linear and convex quadratic programs that appear quite often in applications, the use of inexact Newton direction may be quite worthwhile.

5.4 Experience 4—infeasible linear programs

We now report a set of experiments with a set of infeasible and unbounded linear programs taken from the NETLIB collection, which are presented in Table 9.

For these problems, the projected–gradient direction was always used by the PGIP algorithm. As before, $\eta_{\max} = 10^8$ was the value that produced the best performance. The algorithm was able to converge slowly to a stationary point of the merit function over Ω, with an optimal positive value, given in the column FMERIT. We also solved these problems by the Two Phase procedure described in the previous section. The results of the performance of this last technique are displayed in Table 10, where ITPG and ITQP represent the number of exact Newton iterations used by the PGIP algorithm in each of

Table 10 Performance of the merit function check on primal or dual infeasible linear programs

Problem	ITPG	ITQP	CPU	FQP
ITEST2	0	7	0.05	202.98
ITEST6	0	6	0.02	1.90
BGPRTR	0	4	0.27	36,246
KLEIN1	0	8	0.32	406.21
VOL1	0	13	3.98	8,125.5

the two phases respectively. We notice that the algorithm was able to establish infeasibility in a small number of interior-point iterations. We also add as the last column FQP the positive optimal value of the Phase 2 convex quadratic program. As expected, the values of columns FMERIT of Table 9 and FQP of Table 10 are similar.

6 Conclusions

In this paper we considered the Complementarity Problem (CP) in the form (1) and (2), as a general square nonlinear system that includes complementarity constraints. This form is more general than the ones that represent optimality conditions and variational inequality problems. We used the natural squared norm as merit function, to introduce a projected gradient interior point (PGIP) algorithm for solving CP, and proved that cluster points of PGIP are stationary and are solutions provided that for infinitely many indices the inexact Newton direction is used. Under additional assumptions we proved convergence of the whole sequence. When H is an affine function, we showed that the Newton direction has the property that the merit function is monotonically decreasing along the steplength. Therefore, the maximum steplength is admissible. Under a weak monotonicity assumption, if the merit function is positive at a stationary point of the associated optimization problem, then the problem is infeasible. We suggested an alternative two-phase algorithm for dealing with the "possibly infeasible" linear case.

Numerical results on different instances in this paper indicate that if a monotone LCP has a strictly feasible solution or is infeasible, then no projected-gradient iteration is required during the whole procedure. The same conclusion applies for linear and convex quadratic programs. It will be interesting to study whether this type of behavior is supported by theoretical arguments. The results also show that the algorithm is in general unable to process nonmonotone complementarity problems. The solution of these complementarity problems and general linear complementarity problems [11] by this type of approach should also be interesting topics for future research.

References

1. Andreani, R., Birgin, E.G., Martínez, J.M., Schuverdt, M.L.: Augmented lagrangian methods under the constant positive linear dependence constraint qualification. Math. Program. **111**, 5–32 (2008)
2. Andreani, R., Friedlander, A., Santos, S.A.: On the resolution of the generalized nonlinear complementarity problem. SIAM J. Optim. **12**, 303–321 (2001)
3. Andreani, R., Júdice, J., Martinez, J.M., Patrício, J.: On the natural merit function for solving complementarity problems. Math. Program. (to appear)
4. Birgin, E.G., Martinez, J.M., Raydan, M.: Nonmonotone spectral projected gradient methods on convex sets. SIAM J. Optim. **10**, 1196–1211 (2000)
5. Birgin, E.G., Martinez, J.M., Raydan, M.: Algorithm 813 SPG: software for convex–constrained optimization. ACM Trans. Math. Softw. **27**, 340–349 (2001)

6. Birgin, E.G., Martinez, J.M., Raydan, M.: Inexact spectral projected gradient methods on convex sets. IMA J. Numer. Anal. **23**, 539–559 (2003)
7. Brooke, A., Kendrick, D., Meeraus, A., Raman, R.: GAMS—A User's Guide. GAMS Development Corporation (1998)
8. Cottle, R., Pang, J., Stone, R.: The Linear Complementarity Problem. Academic, New York (1992)
9. Dennis, J.E., Schnabel, R.B.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Prentice Hall Series in Computational Mathematics, Englewood Cliffs, New Jersey (1983)
10. Facchinei, F., Pang, J.S.: Finite-Dimensional Inequalities and Complementarity Problems. Springer, New York (2003)
11. Fernandes, L., Friedlander, A., Guedes, M.C., Júdice, J.: Solution of a general linear complementarity problem using smooth optimization and its applications to bilinear programming and LCP. Appl. Math. Optim. **43**, 1–19 (2001)
12. Fernandes, L., Júdice, J., Figueiredo, I.: On the solution of a finite element approximation of a linear obstacle plate problem. Int. J. Appl. Math. Comput. Sci. **12**, 27–40 (2002)
13. Ferris, M.C., Kanzow, C., Munson, T.S.: Feasible descent algorithms for mixed complementarity problems. Math. Program. **86**, 475–497 (1999)
14. Ferris, M.C., Munson, T.S.: Interfaces to PATH 3.0: design, implementation and usage. Comput. Optim. Appl. **12**, 207–227 (1999)
15. Fischer, A.: New constrained optimization reformulation of complementarity problems. J. Optim. Theory Appl. **12**, 303–321 (2001)
16. Floudas, C.A., Pardalos, P.M.: A Collection of Test Problems for Constrained Global Optimization Algorithms, vol. 455. Springer, New York (1990)
17. Gay, D.M.: Electronic Mail Distribution of Linear Programming Test Problems. Technical report, Mathematical Programming Society COAL Newsletter (1985)
18. Golub, G., Van Loan, C.: Matrix Computations. Johns Hopkins University Press, Baltimore (1983)
19. Gould, N.I., Orban, D., Toint, P.: General CUTEr Documentation. Technical Report TR/PA/02/13, CERFACS (2003)
20. Haslinger, J., Hlavacek, I., Necas, J.: Numerical methods for unilateral problems in solid mechanics. In: Ciarlet, P.G., Lions, J.L. (eds.) Handbook of Numerical Analysis, vol. IV. North–Holland, Amsterdam (1996)
21. Intel Corporation: Intel Fortran Compiler User's Guide (2002)
22. Júdice, J., Raydan, M., Rosa, S., Santos, S.: On the solution of the symmetric eigenvalue complementarity problem by the spectral projected gradient algorithm. Numer. Algorithms **45**, 391–407 (2008)
23. Kikuchi, N., Oden, J.T.: Contact Problems in Elasticity: A Study of Variational Inequalities and Finite Elements. SIAM, Philadelphia (1988)
24. Kojima, M., Megiddo, N., Noma, T., Yoshise, A.: A unified approach to interior-point algorithms for linear complementarity problems. IN: Lecture Notes in Computer Science, vol. 538. Springer, Berlin (1991)
25. Li, D.-H., Fukushima, M.: Derivative–free line search and global convergence of Broyden–like method for nonlinear equations. Optim. Methods Softw. **13**, 181–201 (2000)
26. Murtagh, B.A., Saunders, M.A.: MINOS 5.1 User Guide. Technical Report, Department of Operations Research, Stanford University (1987)
27. Murty, K.: Linear Complementarity, Linear and Nonlinear Programming. Heldermann, Berlin (1988)
28. Nocedal, J., Wright, S.: Numerical Optimization. Springer, New York (2000)
29. Ortega, J.M., Rheinboldt, W.C.: Iterative Solution of Nonlinear Equations in Several Variables. SIAM, Philadelphia (2000)
30. Ostrowski, A.M.: Solution of Equations and Systems of Equations. Academic, New York (1967)
31. Patrício, J.: Algoritmos de Pontos Interiores para Problemas Complementares Monótonos e Suas Aplicações. PhD thesis, Universidade de Coimbra, Portugal (2007). In Portuguese
32. Portugal, L.F., Resende, M.G.C., Veiga, G., Júdice, J.J.: A truncated primal-infeasible dual-feasible network interior point method. Networks **35**, 91–108 (2000)

33. Schwetlick, H.: Numerische Lösung Nichtlinearer Gleichungen. Deutscher Verlag der Wissenschaften, Berlin (1979)
34. Solodov, M.V.: Stationary points of bound constrained minimization reformulations. J. Optim. Theory Appl. **94**, 449–467 (1997)
35. Vanderbei, R.J.: LOQO: an interior point code for quadratic programming. Optim. Methods Softw. **12**, 451–484 (1999)
36. Vanderbei, R.J.: LOQO's User Manual, Version 4.05. School of Engineering and Applied Science, Princeton University (2000)
37. Wright, S.: Primal–Dual Interior–Point Methods. SIAM, Philadelphia (1997)
38. Wright, S.J., Ralph, D.: A superlinear infeasible interior-point algorithm for monotone complementarity problems. Math. Oper. Res. **21**, 815–838 (1996)