A new stopping criterion for Krylov solvers applied in Interior Point Methods

Filippo Zanetti[*] Jacek Gondzio[†]

Abstract

A surprising result is presented in this paper with possible far reaching consequences for any optimization technique which relies on Krylov subspace methods employed to solve the underlying linear equation systems. In this paper the advantages of the new technique are illustrated in the context of Interior Point Methods (IPMs). When an iterative method is applied to solve the linear equation system in IPMs, the attention is usually placed on accelerating their convergence by designing appropriate preconditioners, but the linear solver is applied as a black box solver with a standard termination criterion which asks for a sufficient reduction of the residual in the linear system. Such an approach often leads to an unnecessary "oversolving" of linear equations. In this paper a new specialized termination criterion for Krylov methods used in IPMs is designed. It is derived from a deep understanding of IPM needs and is demonstrated to preserve the polynomial worst-case complexity of these methods. The new criterion has been adapted to the Conjugate Gradient (CG) and to the Minimum Residual method (MINRES) applied in the IPM context. The new criterion has been tested on a set of linear and quadratic optimization problems including compressed sensing, image processing and instances with partial differential equation constraints. Evidence gathered from these computational experiments shows that the new technique delivers significant improvements in terms of inner (linear) iterations and those translate into significant savings of the IPM solution time.

Keywords: Quadratic Programming, Interior Point Methods, Conjugate Gradient, MINRES, Stopping criterion.

1 Introduction

Interior Point Methods (IPMs) represent the state-of-the-art for the solution of convex optimization problems. Being second-order methods, they usually converge in merely a few iterations and if the cost of a single iteration is kept small they are able to outperform the first-order methods, especially when it comes to problems of very large dimensions. In these instances, the linear system that arises at each iteration is usually solved with an iterative Krylov subspace method, either Conjugate Gradient or MINRES, depending on the approach chosen. The ill-conditioning of the matrices involved has given rise to a wide collection of preconditioning strategies for various applications of IPMs (e.g. [7, 8, 9, 10, 18, 23, 32, 34, 39]). Recent developments have also been made regarding the impact of the numerical linear algebra inexactness (e.g. [13, 29, 30]) and in particular on the effect that an inexact linear solver has on the convergence properties of IPMs ([6, 11, 24, 28]). Many other improvements have been made regarding predictor-correctors

[*] School of Mathematics, University of Edinburgh, Edinburgh, UK. f.zanetti@sms.ed.ac.uk

[†] School of Mathematics, University of Edinburgh, Edinburgh, UK. j.gondzio@ed.ac.uk

strategies [12, 21], regularization strategies [3, 19, 36] or the use of quasi-Newton approaches [14, 15, 26].

When an iterative linear solver is used, the common approach is to employ a stopping criterion based on the reduction of the residual, i.e. the internal solver is stopped as soon as the initial residual is reduced by a certain predetermined factor. Different strategies have been developed in order to choose a stopping tolerance that allows the outer IPM iterations to converge, without requiring too many inner (linear solver) iterations. However, these techniques always rely on a tolerance imposed on the residual of the linear system. This approach does not necessarily represent the best choice, since the overall goal is not obtaining an accurate solution to the sequence of linear systems, but finding a suitable (though inexact) search direction for the optimization problem; in particular, it may be possible to obtain a Newton direction that would be considered too rough from a purely linear algebra perspective (i.e. its residual still would be too high and any standard stopping criterion would reject it) but that could be good enough to perform the next iteration of IPM successfully (i.e. the direction guarantees sufficient reductions of infeasibilities and the duality gap). If we are able to derive a stop criterion that accepts a direction not based on its *residual*, but based on a *potential improvement* it can bring to the outer IPM iterations, then we could reduce the number of inner iterations required at each outer step, with little or no disadvantage to the overall convergence properties of the IPM.

Early stopping strategies have been used in other fields: in [27, 31, 37] a CG stop criterion is applied to the Jacobi-Davidson eigensolver, when finding eigenvalues of large matrices; early stopping is also used in inverse problems and machine learning as a regularizer, to avoid the phenomenon known as semiconvergence (see e.g. [20]); in [4, 5] other stopping criteria are derived for various applications. However, to the best of our knowledge, an early stopping criterion, specifically designed to be applied in IPMs, has not yet been derived. This paper fills the gap.

In order to obtain such a criterion, we need to be able to estimate the convergence indicators of IPM (i.e. primal and dual infeasibility and complementarity) while performing the inner iterations with CG or MINRES. The main problem is that, to compute the full direction $(\Delta x, \Delta y, \Delta s)$ and to compute the infeasibilities, we would need to perform additional matrix-vector products at each inner iteration. Since in general we do only one matrix-vector product and one preconditioner application per iteration, adding extra matrix applications would slow down the linear solver immensely. Fortunately, with some clever implementation and exploiting the matrix operations that are already executed, we can estimate the IPM convergence indicators using only vector operations, resulting in a minimal increase in the cost of a single linear iteration.

We propose two stopping criteria for the solution of the linear system at each IPM iteration: a theoretical one, that we use to prove the polynomial complexity result and a practical one, that we use in the numerical tests. The latter, called *Interior Point Conjugate Gradient* (IPCG) or *Interior Point MINRES* (IPMINRES), depending on the approach chosen, is highly specialized for the specific task that it needs to solve and shows significant improvements with respect to the standard CG or MINRES on the problems that we tested, which include quadratic programs derived from image processing, compressed sensing and Partial Differential Equation (PDE) constrained optimization. In particular, we are able to avoid unnecessary inner iterations in the early stage of the IPM, while retaining the good behaviour of the method in its late iterations.

The rest of the paper is organized as follows: in Section 2 we present the Interior Point Method; in Section 3 we show the new IPCG and IPMINRES iterations, that allow to estimate the convergence of IPM; in Section 4 we introduce a theoretical stopping criterion, for which we prove polynomial complexity of IPM, and the criterion used in practice; in Section 5 we present the test problems and show the numerical results.

Notation In the following, e indicates the vector $(1, 1, \dots, 1)^T$ and I represents the identity matrix; their size will be clear from the context. Given a vector v, the diagonal matrix V is defined as $V = \operatorname{diag}(v)$ and we say that v_j^k represents the j-th component of vector v at the k-th iteration. The notation $v > 0$ indicates that each component v_j is strictly positive. $\|\cdot\|$ represents the Euclidean norm.

2 Interior Point Method

Let us consider a pair of primal-dual convex quadratic programming problems in standard form:

$$\min_x \; c^T x + \frac{1}{2} x^T Q x, \qquad \text{s.t. } Ax = b, \qquad x \geq 0, \tag{1}$$

$$\max_{y,\, s} \; b^T y - \frac{1}{2} x^T Q x, \qquad \text{s.t. } A^T y + s - Qx = c, \qquad s \geq 0, \tag{2}$$

where $x, s, c \in \mathbb{R}^n$, $y, b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$, $Q \in \mathbb{R}^{n \times n}$ positive semidefinite.

An Interior Point Method (IPM) looks for an approximation of the solution to (1)-(2) in the interior of the feasible region; the non-negativity constraint is enforced using a logarithmic barrier term, so that the Lagrangian takes the form

$$\mathcal{L}(x, y, \mu) = c^T x + \frac{1}{2} x^T Q x - y^T (Ax - b) - \mu \sum_{i=1}^{n} \log(x_j).$$

The optimality conditions for this perturbed problem are

$$\begin{cases} Ax = b \\ A^T y + s - Qx = c \\ XSe = \mu e \\ (x, s) \geq 0. \end{cases}$$

The Newton method applied to the previous mildly nonlinear system of equations produces the following linear system, to be solved at each IPM iteration

$$\begin{bmatrix} A & 0 & 0 \\ -Q & A^T & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} r_P \\ r_D \\ r_\mu \end{bmatrix} = \begin{bmatrix} b - Ax \\ c + Qx - A^T y - s \\ \sigma \mu e - XSe \end{bmatrix}, \tag{3}$$

where σ is the parameter responsible for the reduction in the complementarity measure μ.

System (3) is usually reduced to the augmented system

$$\begin{bmatrix} -Q - \Theta^{-1} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_D - X^{-1} r_\mu \\ r_P \end{bmatrix} \tag{4}$$

where $\Theta = XS^{-1}$, and solved using an indefinite factorization or an iterative method for symmetric indefinite systems (e.g. MINRES [33]), or it is further reduced to the normal equations

$$A(Q + \Theta^{-1})^{-1} A^T \Delta y = r_P + A(Q + \Theta^{-1})^{-1}(r_D - X^{-1} r_\mu) \tag{5}$$

and solved with a Cholesky factorization or using the Conjugate Gradient (CG) method. The direction is then used to compute the stepsize α and to find the next point $(x + \alpha\Delta x, y + \alpha\Delta y, s + \alpha\Delta s)$. The outer iterations are stopped as soon as the approximation satisfies the following IPM stopping criterion

$$\frac{\|b - Ax\|}{\|b\|} \leq \tau_P, \qquad \frac{\|c + Qx - A^T y - s\|}{\|c\|} \leq \tau_D, \qquad \mu \leq \tau_\mu, \tag{6}$$

where τ_P, τ_D and τ_μ are predetermined tolerances. In recent years a lot of effort has been put into designing efficient preconditioners for the augmented system and for the normal equations (e.g. [7, 8, 9, 10, 23, 32]). The major difficulty originates from the extreme ill conditioning of the matrices in (4) and (5) when μ gets close to zero; regularization is a common strategy employed to improve the conditioning of the problem (see [3, 19, 23, 36]).

The most successful implementations of the primal-dual Interior Point Methods are the path-following methods, where the approximations computed throughout the IPM iterations are forced to follow the central path and stay in the appropriately chosen neighbourhood of it. These algorithms show polynomial complexity, both in the feasible and infeasible case (see e.g. [22, 41]). A common neighbourhood of the central path used in the infeasible case is defined as follows: at iteration k, the point (x^k, y^k, s^k) is in the neighbourhood $\mathcal{N}_\infty(\gamma, \beta)$ if it satisfies

$$(x^k, s^k) > 0, \tag{7a}$$

$$\gamma\mu^k \leq x_j^k s_j^k \leq \mu^k/\gamma, \quad \forall j, \tag{7b}$$

$$\|r_P^k\| \leq \|r_P^0\|\beta\mu^k/\mu^0, \quad \|r_D^k\| \leq \|r_D^0\|\beta\mu^k/\mu^0, \tag{7c}$$

where $0 < \gamma < 1$ and $\beta \geq 1$ are two constants chosen at the beginning of the IPM algorithm. In this paper we focus both on the augmented system approach (4), when dealing with generic QPs, and on the normal equations approach (5), when dealing with LPs or special cases of QPs.

3 Estimating the convergence of the outer iterations

In this section, we want to understand how to estimate the IPM convergence indicators throughout the CG or MINRES iterations, so as to stop the inner solver as soon as the outer iteration can be performed successfully. The main indicators that are commonly used, as shown in (6), are the primal and dual infeasibilities and the complementarity gap. We develop algorithms and stopping criteria both for CG and MINRES, to be used for LPs and QPs respectively.

In the case of the normal equations for an LP, we apply the CG to system (5) with $Q = 0$; this means that at every inner iteration we update the approximation for Δy. In order to estimate the IPM indicators, we need also Δx and Δs, which are computed as follows

$$\Delta x = (S^{-1}r_\mu - \Theta r_D) + \Theta A^T \Delta y,$$
$$\Delta s = X^{-1}r_\mu - \Theta^{-1}\Delta x.$$

When using the augmented system instead, Δx and Δy are readily available and we just need to compute Δs.

We can see that these two formulas contain the first term, which is constant during the inner iterations, and the second term which varies as the Krylov method progresses. Once we know the full direction, the stepsizes can be computed as

$$\alpha_x = \min_{j\,:\,\Delta x_j<0} -\frac{x_j}{\Delta x_j}, \quad \alpha_s = \min_{j\,:\,\Delta s_j<0} -\frac{s_j}{\Delta s_j}. \tag{8}$$

3.1 IPCG for LP

Let us consider now the normal equations approach for an LP (i.e. $Q = 0$). If we suppose to stop the CG at a certain iteration for which we computed the full direction $(\Delta x, \Delta y, \Delta s)$ and also the stepsizes α_x and α_s, and if we indicate the new point that we obtain by $(\bar{x}, \bar{y}, \bar{s})$, then the infeasibilities could be written as

$$A\bar{x} - b = (Ax - b) + \alpha_x\Big((A\Theta A^T\Delta y) + (AS^{-1}r_\mu - A\Theta r_D)\Big),$$

$$A^T\bar{y} + \bar{s} - c = (A^Ty + s - c) + \alpha_s(A^T\Delta y + \Delta s).$$

The problematic terms in these formulas are given by $A^T\Delta y$ and $A\Theta A^T\Delta y$, and would require extra matrix operations to be computed. Let us define the vectors $v_1 = X^{-1}r_\mu$, $v_2 = S^{-1}r_\mu - \Theta r_D$, $v_3 = AS^{-1}r_\mu - A\Theta r_D$, $\xi_1 = A^T\Delta y$, $\xi_2 = A\Theta A^T\Delta y$; then, the previous expressions become

$$\Delta x = v_2 + \Theta\xi_1, \qquad \Delta s = v_1 - \Theta^{-1}\Delta x, \tag{9}$$

$$A\bar{x} - b = (Ax - b) + \alpha_x(\xi_2 + v_3), \tag{10}$$

$$A^T\bar{y} + \bar{s} - c = (A^Ty + s - c) + \alpha_s(\xi_1 + \Delta s). \tag{11}$$

Vectors v_1, v_2 and v_3 remain constant during the CG iterations and can be computed once at the beginning of the algorithm. Recall that, during the CG process, the approximation Δy is updated as

$$\Delta y = \Delta y + \alpha u$$

where α is the CG stepsize and u is the CG direction. Therefore, we can update also the quantities ξ_1 and ξ_2 in a similar way:

$$\xi_1 = \xi_1 + \alpha A^T u, \qquad \xi_2 = \xi_2 + \alpha A\Theta A^T u.$$

The quantity $A\Theta A^T u$ is already computed during the CG algorithm, because it is needed to find the stepsize α and to update the residual. While computing it, we can obtain as a byproduct also the quantity $A^T u$:

$$w_1 = A^T u, \qquad w_2 = A\Theta w_1.$$

In this way, we see that it is possible to update the quantities ξ_1 and ξ_2 at each inner iteration inexpensively, which in turn allows us to compute the IPM convergence indicators at each CG iteration using only vector operations. Notice that the products with matrix Θ needed to compute the directions in (9), in practice are performed as vector operations, since Θ is diagonal. Notice also that we need to compute w_1 and w_2 at the beginning of the CG process, to initialize the residual; thus, we do not add operations to initialize ξ_1 and ξ_2. However, we do add one single matrix-vector product with matrix A at the beginning of the algorithm, to compute the constant vector v_3. Algorithm IPCG summarizes the process just described: the main differences with the standard CG algorithm are in lines 2, 17, 18, 26, 27, 28, which are also highlighted in blue. We estimate the IPM convergence indicators only after a number `itstart` of iterations. The algorithm does not contain any stopping criterion for now, it simply computes the primal and dual infeasibilities and the duality gap at each CG iteration, if we were to stop the CG process at that iteration. The choice of the stopping criterion based on these indicators will be discussed in the following sections.

At each iteration, the standard CG algorithm performs one matrix-vector product, one preconditioner application, two scalar products and three `axpy` operations; what we propose to add in the IPCG algorithm requires, at each iteration, the equivalent of three scalar products (to compute μ, $\Theta\xi_1$, $\Theta^{-1}\Delta x$), approximately ten `axpy` operations and the computation of the stepsizes (which involves only vector operations). Therefore, we expect the computational cost of the IPCG iteration to be only slightly larger than that of the standard CG step, especially if the applications of the matrix or the preconditioner are particularly expensive.

Remark 1. *Notice that if we use a predictor-corrector strategy during the IPM, the algorithm just proposed works only when computing the predictor direction. For the corrector we need to modify equations* (10)-(11). *In particular, if we call* $(\Delta x^P, \Delta y^P, \Delta s^P)$ *the predictor computed previously, we need to add the term* $A\Delta x^P$ *to the expression for the primal residual* (10) *and the term* $A^T\Delta y^P + \Delta s^P$ *to the expression for the dual residual* (11)*; they can be computed at the beginning since they are constant, but they add matrix operations to be performed at every call of the algorithm. Alternatively, these can be avoided by saving the final values of the vectors* ξ_1 *and* ξ_2 *from the previous IPCG call that computed the predictor direction.*

3.2 IPMINRES for QP

Similarly, in the case of the augmented system for a QP, we see that the infeasibilities can be written as

$$A\bar{x} - b = (Ax - b) + \alpha_x(A\Delta x)$$
$$A^T\bar{y} + \bar{s} - Q\bar{x} - c = (A^Ty + s - Qx - c) + \alpha_s(A^T\Delta y + \Delta s) - \alpha_x(Q\Delta x).$$

Thus, at each inner iteration, we need to update the quantities $\xi_x = A\Delta x$, $\xi_y = A^T\Delta y$, $\xi_Q = Q\Delta x$; this can be done at little extra cost by exploiting the matrix-vector products already present in the MINRES algorithm, similarly to what we have earlier done for the CG. The implementation is slightly more complicated, since the MINRES updates the approximation using the two previous iterations; Algorithm IPMINRES shows the standard MINRES method, according to the implementation in [2], with the additional operations required: the main differences with the standard MINRES algorithm are in lines 3, 7, 17, 19, 21, 23, 24, 25, which are also highlighted in blue. The estimation of the residual is more complicated than in the CG case; we do not show it to avoid further overcomplicating of the displayed algorithm and because it is not affected by the new approach. As before, the additional cost is given only by vector operations (scalar products, `axpy` operations and stepsizes computation).

In the following, we will call IPM-I the IPM with inexact directions computed using either IPCG or IPMINRES.

Remark 2. *Notice that Algorithms IPCG and IPMINRES are just prototypical algorithms; in order to obtain the maximum efficiency, a specialized method can be derived for the specific problem that we have to solve. The structure of the method will resemble the one*

Algorithm IPCG Interior Point Conjugate Gradient method

Input: rhs f, tolerance τ_{CG}, max iterations `itmax`, matrices A, Θ, preconditioner P, initial approximation Δy, minimum iterations `itstart`

Input from IPM: current point (x, y, s), vectors r_P, r_D, r_μ

```
Initialize:
$v_1 = X^{-1} r_\mu$, $v_2 = \Theta(v_1 - r_D)$, $v_3 = A v_2$
$\xi_1 = A^T \Delta y$
$\xi_2 = A \Theta \xi_1$
$r_0 = f - \xi_2$
$r = r_0$
$z = P^{-1} r$
$u = z$
$\rho = r^T z$
$\texttt{iter} = 0$
while $\|r\| > \tau_{\mathrm{CG}} \|r_0\|$ and $\texttt{iter} < \texttt{itmax}$ do
    $\texttt{iter} = \texttt{iter} + 1$
    $w_1 = A^T u$
    $w_2 = A \Theta w_1$
    $\alpha = \rho / w_2^T u$
    $\Delta y = \Delta y + \alpha u$
    $\xi_1 = \xi_1 + \alpha w_1$
    $\xi_2 = \xi_2 + \alpha w_2$
    $r = r - \alpha w_2$
    $z = P^{-1} r$
    $\rho^N = r^T z$
    $\beta = \rho^N / \rho$
    $u = z + \beta u$
    $\rho = \rho^N$
    if $(\texttt{iter} \geq \texttt{itstart})$ then
        Compute Newton directions: $\Delta x = v_2 + \Theta \xi_1$,    $\Delta s = v_1 - \Theta^{-1} \Delta x$
        Compute stepsizes $\alpha_x$, $\alpha_s$ using $x$, $\Delta x$, $s$, $\Delta s$
        Compute convergence indicators:
            $$p_{\mathrm{inf}} = -r_P + \alpha_x(\xi_2 + v_3), \quad d_{\mathrm{inf}} = -r_D + \alpha_s(\xi_1 + \Delta s),$$
            $$\mu = (x + \alpha_x \Delta x)^T (s + \alpha_s \Delta s)$$
    end if
end while
```

Algorithm IPMINRES Interior Point Minimum Residual method

Input: rhs f, tolerance τ_{minres}, max iterations `itmax`, matrices A, Θ,Q, preconditioner P, minimum iterations `itstart`

Input from IPM: current point (x, y, s), vectors r_P, r_D, r_μ

```
Initialize:
$\psi = P^{-1}f,\ r_1 = f,\ r_2 = r_1,\ \beta = \sqrt{f^T\psi},\ w = 0,\ w_2 = 0,\ c_s = -1,\ s_n = 0,\ \bar\varphi = \beta,\ \epsilon = 0,$
$\Delta = 0,\ \texttt{iter} = 0$
$w^v = 0,\ w_2^v = 0,\ \xi = 0,\ \zeta = X^{-1}r_\mu$
while residual $> \tau_{\text{minres}}$ and iter $<$ itmax do
    $\texttt{iter} = \texttt{iter} + 1$
    $v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \frac{1}{\beta}\psi$
    $\psi = \begin{bmatrix} -Q - \Theta^{-1} & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$, with byproduct $z^v = \begin{bmatrix} Qv_1 \\ Av_1 \\ A^T v_2 \end{bmatrix}$
    if iter $\geq 2$ then $\psi = \psi - (\beta/\beta_0)r_1$ end if
    $\alpha = v^T\psi$
    $\psi = \psi - (\alpha/\beta)r_2$
    $r_1 = r_2,\ r_2 = \psi$
    $\psi = P^{-1}r_2$
    $\beta_0 = \beta,\ \beta = \sqrt{r_2^T\psi}$
    $\epsilon_0 = \epsilon,\ \delta = c_s\bar\delta + s_n\alpha,\ \bar g = s_n\bar\delta - c_s\alpha,\ \epsilon = s_n\beta,\ \bar\delta = -c_s\beta,\ r = \sqrt{\bar g^2 + \bar\delta^2}$
    $\gamma = \max(\sqrt{\bar g^2 + \beta^2}, \epsilon)$
    $c_s = \bar g/\gamma,\ s_n = \beta/\gamma,\ \varphi = c_s\bar\varphi,\ \bar\varphi = s_n\bar\varphi$
    $w_1 = w_2,\ w_2 = w,\ w_1^v = w_2^v,\ w_2^v = w^v$
    $w = (v - \epsilon_0 w_1 - \delta w_2)/\gamma$
    $w^v = (z^v - \epsilon_0 w_1^v - \delta w_2^v)/\gamma$
    $\Delta = \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \Delta + \varphi w$
    $\xi = \begin{bmatrix} \xi_Q \\ \xi_x \\ \xi_y \end{bmatrix} = \xi + \varphi w^v$
    if (iter $\geq$ itstart) then
        Compute Newton direction: $\Delta s = \zeta - \Theta^{-1}\Delta x$
        Compute stepsizes $\alpha_x,\ \alpha_s$ using $x,\ \Delta x,\ s,\ \Delta s$
        Compute convergence indicators:
            $p_{\text{inf}} = -r_P + \alpha_x\xi_x, \quad d_{\text{inf}} = -r_D + \alpha_s(\xi_y + \Delta s) - \alpha_x\xi_Q,$
            $\mu = (x + \alpha_x\Delta x)^T(s + \alpha_s\Delta s)$
    end if
end while
```

that we propose, but there may be differences. Notice also that IPCG can be applied to QPs for which it is possible to form the normal equations, e.g. because there are no linear constraints or because matrix Q is diagonal.

4 Stopping criterion

We now address the problem of how to choose a stopping criterion based on the convergence indicators that are estimated by the IPCG or IPMINRES algorithms. In this section, we propose a theoretical criterion, that allows us to prove a polynomial complexity result for the IPM-I. Then, we propose a practical criterion, based on the experimental evidence.

Remark 3. *In this section we use α to indicate the IPM stepsize, while in the previous section we used it for the CG stepsize. Since they are both standard notation widely used in these fields, we did not change them.*

4.1 Polynomial complexity

In this section, we will follow [41, chapter 6], with the difference that here we are dealing with a quadratic program. We make some standard assumptions: the neighbourhood is defined by (7); we choose $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$, $\sigma_{\max} \leq 0.5$; we consider a single stepsize α_k instead of two different ones for the primal and dual direction; we choose the stepsize such that the next point is inside the neighbourhood and it satisfies the Armijo condition

$$\mu_{k+1} \leq (1 - 0.01\alpha_k)\mu_k. \tag{12}$$

We already know that, when dealing with an LP and using an exact method to find the direction, there is a minimum stepsize that can be taken, $\alpha_k \geq \bar{\alpha}$, and both the primal and dual infeasibilities are reduced by a factor $(1 - \alpha_k)$. Moreover, from the third equation in (3), we can obtain that

$$\frac{\Delta x_j^k}{x_j^k} + \frac{\Delta s_j^k}{s_j^k} = -1 + \frac{\sigma_k \mu_k}{x_j^k s_j^k}, \quad \forall j. \tag{13}$$

Notice that the right hand side in the last equation is $\mathcal{O}(1)$, due to condition (7b). Therefore, given these facts, we choose our stopping criterion as follows: we accept the direction produced by the inner solver as soon as

$$\max_j \left| \frac{\Delta x_j^k}{x_j^k} \right| \leq M, \qquad \max_j \left| \frac{\Delta s_j^k}{s_j^k} \right| \leq M \tag{14}$$

for some fixed constant M. Moreover, we also require that

$$\|r_P^{k+1}\| \leq \eta_k \|r_P^k\|, \quad \|r_D^{k+1}\| \leq \eta_k \|r_D^k\|, \tag{15}$$

where $\eta_k \geq 1 - \alpha_k$, since we cannot reasonably assume that an inexact direction would do as well as the exact one. Thus, we suppose that $\eta_k = 1 - \omega_k \alpha_k$, for some $\omega_k \leq 1$; the choice of ω_k will be clarified in the next Lemma. We also suppose that the equation

$$S^k \Delta x^k + X^k \Delta s^k = \sigma_k \mu_k e - X^k S^k e \tag{16}$$

continues to hold even if the direction is inexact; this is the case if we build Δs from Δx, as it usually happens when employing the normal equations or the augmented system.

Algorithm IPM-I Interior Point Method with early stopping of the linear solver

```
Input: $\gamma \in [0,1]$, $\beta \geq 1$, $0 < \sigma_{\min} < \sigma_{\max} \leq 0.5$; $M > 0$, $0 < \delta < \sigma_{\min}$
Choose $(x^0, y^0, s^0)$ with $(x^0, s^0) > 0$
while (6) is not satisfied do
    Choose $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$
    Choose $\omega_k \in [1 - \sigma_k + \delta, 1]$
    Apply IPCG or IPMINRES to find a direction $(\Delta x^k, \Delta y^k, \Delta s^k)$ such that
        $\max_j \left| \frac{\Delta x_j^k}{x_j^k} \right| \leq M, \qquad \max_j \left| \frac{\Delta s_j^k}{s_j^k} \right| \leq M,$
        $\|r_P^{k+1}\| \leq (1 - \omega_k \alpha_k) \|r_P^k\|, \quad \|r_D^{k+1}\| \leq (1 - \omega_k \alpha_k) \|r_D^k\|,$
        $S^k \Delta x^k + X^k \Delta s^k = \sigma_k \mu_k e - X^k S^k e.$
    Choose $\alpha_k$ as the largest $\alpha \in [0,1]$ such that
        $(x^k + \alpha \Delta x^k, y^k + \alpha \Delta y^k, s^k + \alpha \Delta s^k) \in \mathcal{N}_\infty(\gamma, \beta),$
        $(x^k + \alpha \Delta x^k)^T (s^k + \alpha \Delta s^k) \leq (1 - 0.01\alpha)(x^k)^T (s^k).$
    Set
        $(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k + \alpha_k \Delta x^k, y^k + \alpha_k \Delta y^k, s^k + \alpha_k \Delta s^k)$
end while
```

Algorithm IPM-I summarizes the choices made here and shows also some other features that will be clear during the proof of Lemma 1.

We will now prove that, if we choose the direction using the stopping criterion defined by (14)-(15), then there still exists a minimum stepsize $\tilde{\alpha}$ that we can take at each iteration. In the following, we will omit the iteration index k, for sake of clarity.

Lemma 1. *There exists a value $\tilde{\alpha} \in (0,1)$ such that the following conditions are satisfied for all $\alpha \in [0, \tilde{\alpha}]$ at each IPM iteration and for all components j:*

$$(x_j + \alpha \Delta x_j)(s_j + \alpha \Delta s_j) \geq \gamma (x + \alpha \Delta x)^T (s + \alpha \Delta s)/n, \tag{17a}$$

$$(x_j + \alpha \Delta x_j)(s_j + \alpha \Delta s_j) \leq (1/\gamma)(x + \alpha \Delta x)^T (s + \alpha \Delta s)/n, \tag{17b}$$

$$(x + \alpha \Delta x)^T (s + \alpha \Delta s)/n \leq (1 - 0.01\alpha)\mu, \tag{17c}$$

$$(x + \alpha \Delta x)^T (s + \alpha \Delta s) \geq \eta x^T s. \tag{17d}$$

Proof. Let us start by noticing that (14) implies these two facts

1. the positivity constraints $x + \alpha \Delta x > 0$ and $s + \alpha \Delta s > 0$ are automatically satisfied for any $\alpha \in [0, \frac{1}{M}]$;

2. the following bounds hold

$$|\Delta x_j \Delta s_j| \leq \frac{M^2}{\gamma} \mu, \qquad |\Delta x^T \Delta s| \leq M^2 n \mu. \tag{18}$$

Using(18), (16) and (7b), it is easy to show that the following inequalities hold

$$(x_j + \alpha\Delta x_j)(s_j + \alpha\Delta s_j) \geq (1-\alpha)\gamma\mu + \alpha\sigma\mu - \alpha^2 M^2 \frac{\mu}{\gamma}, \tag{19a}$$

$$(x_j + \alpha\Delta x_j)(s_j + \alpha\Delta s_j) \leq (1-\alpha)\frac{\mu}{\gamma} + \alpha\sigma\mu + \alpha^2 M^2 \frac{\mu}{\gamma}, \tag{19b}$$

$$(x + \alpha\Delta x)^T (s + \alpha\Delta s)/n \geq (1-\alpha)\mu + \alpha\sigma\mu - \alpha^2 M^2 \mu, \tag{19c}$$

$$(x + \alpha\Delta x)^T (s + \alpha\Delta s)/n \leq (1-\alpha)\mu + \alpha\sigma\mu + \alpha^2 M^2 \mu. \tag{19d}$$

Using (19a) and (19d), we obtain

$$\begin{aligned}
&(x_j + \alpha\Delta x_j)(s_j + \alpha\Delta s_j) - \gamma(x + \alpha\Delta x)^T (s + \alpha\Delta s)/n \geq \\
&\geq (1-\alpha)\gamma\mu + \alpha\sigma\mu - \alpha^2 M^2 \frac{\mu}{\gamma} - \gamma((1-\alpha)\mu + \alpha\sigma\mu + \alpha^2 M^2 \mu) \geq \\
&\geq \alpha\sigma_{\min}\mu(1-\gamma) - \alpha^2 M^2 \mu(\gamma + 1/\gamma)
\end{aligned}$$

and thus (17a) is satisfied if the final expression is non-negative, i.e.

$$\alpha \leq \frac{\sigma_{\min}\gamma(1-\gamma)}{M^2(1+\gamma^2)}.$$

Using (19b) and (19c), we obtain

$$\begin{aligned}
&(1/\gamma)(x + \alpha\Delta x)^T (s + \alpha\Delta s)/n - (x_j + \alpha\Delta x_j)(s_j + \alpha\Delta s_j) \geq \\
&\geq (1/\gamma)((1-\alpha)\mu + \alpha\sigma\mu - \alpha^2 M^2 \mu) - (1-\alpha)\frac{\mu}{\gamma} - \alpha\sigma\mu - \alpha^2 M^2 \frac{\mu}{\gamma} \geq \\
&\geq \alpha\sigma_{\min}\mu(1/\gamma - 1) - 2\alpha^2 M^2 \mu/\gamma
\end{aligned}$$

and thus (17b) is satisfied if the final expression is non-negative, i.e.

$$\alpha \leq \frac{\sigma_{\min}(1-\gamma)}{2M^2}.$$

Using (19d), we obtain

$$\begin{aligned}
&(1 - 0.01\alpha)\mu - (x + \alpha\Delta x)^T (s + \alpha\Delta s)/n \geq \\
&\geq (1 - 0.01\alpha)\mu - (1-\alpha)\mu - \alpha\sigma\mu - \alpha^2 M^2 \mu \geq \\
&\geq 0.99\alpha\mu - \alpha\sigma_{\max}\mu - \alpha^2 M^2 \mu
\end{aligned}$$

and thus (17c) is satisfied if the final expression is non-negative, i.e.

$$\alpha \leq \frac{0.99 - \sigma_{\max}}{M^2}.$$

Using (19c) and setting $\eta = 1 - \omega\alpha$, we obtain

$$\begin{aligned}
&(x + \alpha\Delta x)^T (s + \alpha\Delta s) - (1 - \omega\alpha)x^T s \geq \\
&\geq (1-\alpha)x^T s + \alpha\sigma x^T s - \alpha^2 M^2 x^T s - (1 - \omega\alpha)x^T s \geq \\
&\geq \alpha(\sigma + \omega - 1)x^T s - \alpha^2 M^2 x^T s
\end{aligned}$$

and thus (17d) is satisfied if the final expression is non-negative, i.e.

$$\alpha \leq \frac{\sigma + \omega - 1}{M^2}.$$

This condition makes sense only if $\omega > 1 - \sigma$; therefore, at each IPM iteration, after choosing σ, we should choose $\omega \in]1 - \sigma, 1]$. Notice what this means: in the early IPM iterations, σ is closer to 1 and thus ω can be closer to 0, which makes the stop criterion easier to satisfy. In the later iterations, σ might get closer to 0 and thus ω is closer to 1, which makes the stop criterion harder to satisfy. If we choose ω such that $\omega \geq 1 - \sigma + \delta$, with δ a fixed constant, $\delta < \sigma_{\min}$, then $\omega + \sigma - 1 \geq \delta$ and we obtain

$$\alpha \leq \frac{\delta}{M^2} \quad \Rightarrow \quad \alpha \leq \frac{\sigma + \omega - 1}{M^2}.$$

This explains the choice of ω made in Algorithm IPM-I.

Therefore, the minimum stepsize that we can take at each IPM iteration is given by

$$\tilde{\alpha} = \min\Big(\frac{\sigma_{\min}\gamma(1-\gamma)}{M^2(1+\gamma^2)}, \frac{\sigma_{\min}(1-\gamma)}{2M^2}, \frac{0.99 - \sigma_{\max}}{M^2}, \frac{\delta}{M^2}, 1\Big). \tag{20}$$

□

We highlight that inequalities (17a)-(17b) imply that the next IPM iteration satisfies condition (7b); the inequality (17c) represents the Armijo condition, while inequality (17d) implies that

$$\frac{\|r_P^k\|}{\mu^k} \leq \frac{\eta_k \|r_P^{k-1}\|}{\mu^k} \leq \frac{\|r_P^{k-1}\|}{\mu^{k-1}} \leq \frac{\beta\|r_P^0\|}{\mu^0}$$

and similarly for the dual residual, which is equivalent to condition (7c). If α is larger than $1/M$, then also condition (7a) is satisfied. Therefore, we can say that we can always take a minimum stepsize equal to $\min(\tilde{\alpha}, 1/M)$ and be sure to end up still inside the neighbourhood $\mathcal{N}_\infty(\gamma, \beta)$.

To obtain a polynomial complexity result, we need to specify the value of M. The next lemma helps us in this sense.

Lemma 2. *There exists a starting point* (x^0, y^0, s^0) *for which, when solving an LP or a QP, the exact direction satisfies* (14) *with* $M = \mathcal{O}(n^2)$.

Proof. We know that for a specific choice of the starting point, the minimum stepsize when using the exact direction satisfies $\bar{\alpha} \geq C_1 n^{-2}$, for some $C_1 > 0$. This is a standard result that can be found in [41, Lemma 6.7] for the LP case or in [40, Theorem 3.5] for the QP case.

Using (8), this implies that

$$\frac{-x_j}{\Delta x_j} \geq \frac{C_1}{n^2} \quad \forall j \text{ s.t. } \Delta x_j < 0, \qquad \frac{-s_j}{\Delta s_j} \geq \frac{C_1}{n^2} \quad \forall j \text{ s.t. } \Delta s_j < 0.$$

Therefore

$$\frac{-\Delta x_j}{x_j} \leq \frac{n^2}{C_1} \quad \forall j \text{ s.t. } \Delta x_j < 0, \qquad \frac{-\Delta s_j}{s_j} \leq \frac{n^2}{C_1} \quad \forall j \text{ s.t. } \Delta s_j < 0.$$

There are now two possibilities for each component j:

- $\Delta x_j > 0$ and $\Delta s_j > 0$ (or $\Delta x_j < 0$ and $\Delta s_j < 0$); then to satisfy (13) it must be

$$\frac{\Delta x_j}{x_j} = \mathcal{O}(1), \qquad \frac{\Delta s_j}{s_j} = \mathcal{O}(1).$$

- $\Delta x_j < 0$ and $\Delta s_j > 0$ (or vice versa); then it follows from (13) that

$$\frac{\Delta s_j}{s_j} = \mathcal{O}(1) - \frac{\Delta x_j}{x_j} \leq \mathcal{O}(1) + \frac{n^2}{C_1} = \mathcal{O}(n^2).$$

Thus, we see that

$$\Big|\frac{\Delta x_j}{x_j}\Big| \leq \mathcal{O}(n^2), \qquad \Big|\frac{\Delta s_j}{s_j}\Big| \leq \mathcal{O}(n^2), \qquad \forall j.$$

□

If the exact direction satisfies (14) with $M = \mathcal{O}(n^2)$, then it is reasonable to assume that in our stop criterion we can choose $M = \mathcal{O}(n^q),\ q \geq 2$, since we cannot do better than the exact direction. Moreover, since we know that the linear solver will find the exact solution, which satisfies (14) with $q = 2$, at some point (in exact arithmetic), we can expect that before this final iteration it is possible to stop choosing $q \geq 2$.

We can thus characterize better the minimum stepsize in (20): choosing $M = \mathcal{O}(n^q)$, it follows that $\tilde{\alpha} \geq \mathcal{O}(n^{-2q})$. We can now prove the polynomial complexity of IPM-I.

Theorem 1. *Suppose that the starting point satisfies $\mu_0 \leq 1/\nu^\kappa$ for some positive constant κ. Suppose to use a constant $M = \mathcal{O}(n^q),\ q \geq 2$. Then there exists an index K with*

$$K = \mathcal{O}(n^{2q}|\log \nu|)$$

such that $\mu_k \leq \nu$ for all $k \geq K$.

Proof. We notice that the Armijo condition (12), together with the previous characterization of $\tilde{\alpha}$, implies that

$$\mu_{k+1} \leq \Big(1 - \frac{C_2}{n^{2q}}\Big)\mu_k.$$

for some $C_2 > 0$. From here, we apply [41, Theorem 3.2] and conclude. □

We have thus shown that the proposed inexact IPM retains polynomial convergence. In the best case where $M = \mathcal{O}(n^2)$, the result suggests that the number of iterations is proportional to n^4, instead of n^2 when using the exact IPM. We underline that our approach is different from the ones in e.g. [6, 24], because we do not assume to solve the linear system (3) obtaining a small residual, but we merely require that the direction satisfies (14)-(15). This means that we have moved the attention from the residual of the linear system (that has little to do with the convergence of the outer IPM iterations), to other quantities that are directly related to the convergence of IPM. In this way, we hope to obtain a stopping criterion for the linear solver that avoids unnecessary inner iterations.

Remark 4. *In this section, we have used the results from [41, Chapter 6]; it is worth pointing out that the results presented there are obtained using a neighbourhood without the upper bound in* (7b)*. However, with some simple calculations, it is possible to see that the final results do not change adding the upper bound. A similar conclusion was obtained in [12], where the upper bound was added in the case of a feasible algorithm.*

4.2 Practical stopping criterion

In this section, we discuss the behaviour of the IPM convergence indicators for one of the test problems and we derive the stopping criterion used in practice. The problem used here, which will be formally introduced in the next section, is a QP without linear constraints that arises from tomographic imaging; thus, we can use the normal equations approach, but only consider the dual infeasibility.

We start by displaying in Figure 1 the behaviour of the dual infeasibility, complementarity, primal and dual stepsizes at the tenth and twentieth IPM iteration; they are computed at every inner CG iteration, using Algorithm IPCG.

We can see that, for all these indicators, there is a stagnation during the CG iterations. This means that after approximately half of the number of CG iterations required, the approximate direction is already able to produce a new IPM point with values of the convergence indicators very close to the ones that we would obtain using the standard residual test. It is therefore reasonable to think of stopping the CG iterations as soon as the stagnation occurs. However, we expect to need a very accurate direction in the late phase of IPM, which means that stagnation may fail to appear in the last iterations. Thus, we propose to use a stopping criterion that keeps the standard residual reduction test, but that is able to stop in advance if a stagnation in the IPM convergence indicators is detected.

In order to detect stagnation, we need to compute a measure of the deviation of the most recent values of the convergence indicators from their mean. We used the following heuristic: at CG iteration j, we compute the new convergence indicators p^j_{inf}, d^j_{inf}, μ^j, and then we evaluate the relative variations with respect to the values of the previous iteration

$$\texttt{var}^P_j = \left|\frac{\|p^j_{\text{inf}}\| - \|p^{j-1}_{\text{inf}}\|}{\|p^{j-1}_{\text{inf}}\|}\right|, \quad \texttt{var}^D_j = \left|\frac{\|d^j_{\text{inf}}\| - \|d^{j-1}_{\text{inf}}\|}{\|d^{j-1}_{\text{inf}}\|}\right|, \quad \texttt{var}^\mu_j = \left|\frac{\mu^j - \mu^{j-1}}{\mu^{j-1}}\right|.$$

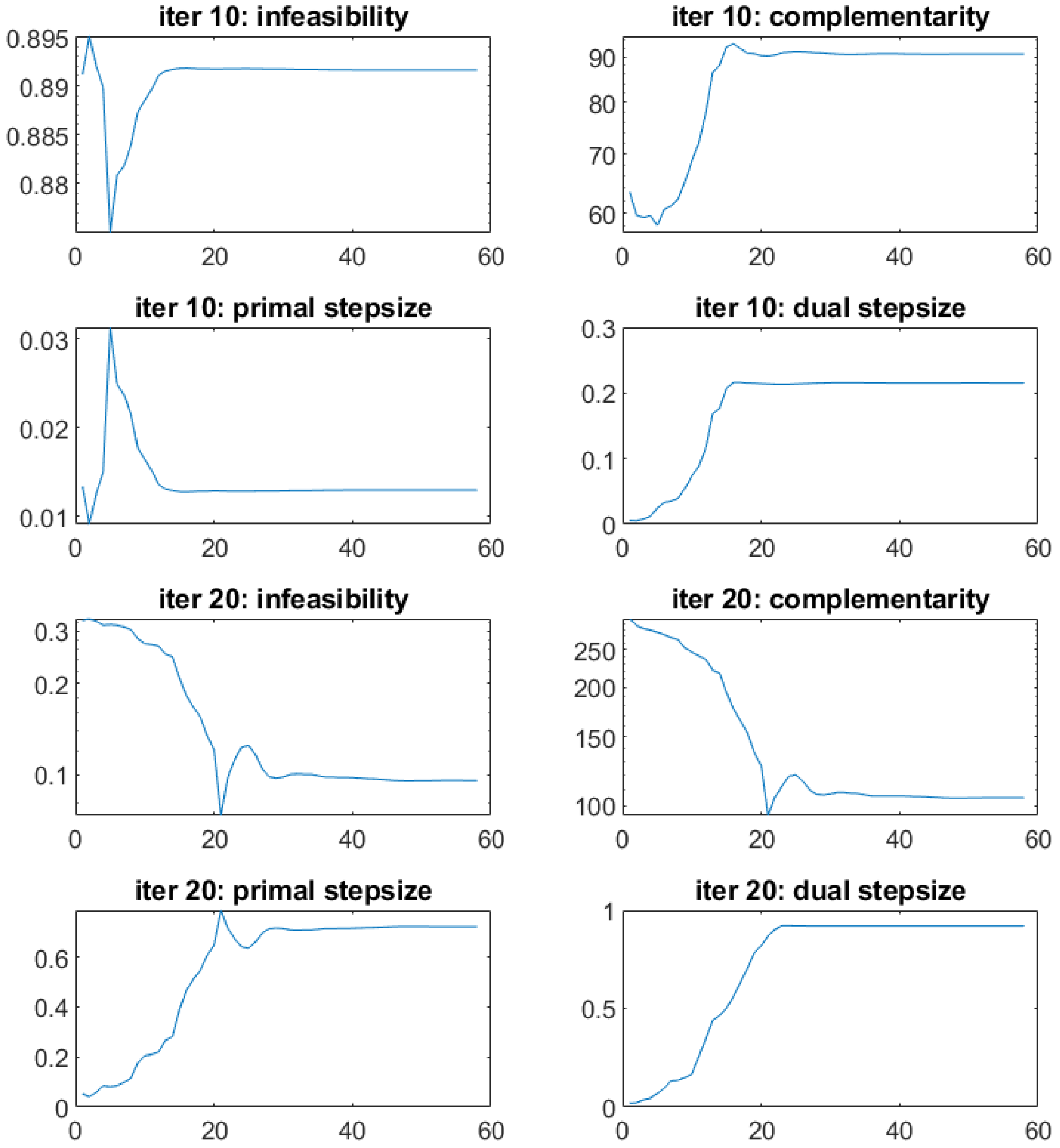
In order to decide if the stagnation occurred, we compute the averages of the last five variations; as soon as those fall below a prescribed tolerance ε we activate the stopping criterion

$$\frac{1}{5}\sum_{i=0}^{4}\texttt{var}^P_{j-i} < \varepsilon \quad \wedge \quad \frac{1}{5}\sum_{i=0}^{4}\texttt{var}^D_{j-i} < \varepsilon \quad \wedge \quad \frac{1}{5}\sum_{i=0}^{4}\texttt{var}^\mu_{j-i} < \varepsilon.$$

Typical values of ε that we used are 10^{-2}, 10^{-3} or 10^{-4}. We could equivalently use a test on the variance of the last five values of the indicators, but we preferred to use this criterion, since it has an immediate interpretation which makes it easier to tune the tolerance for the various problems. Indeed, a value $\varepsilon = 0.01$ means that we stop the CG as soon as the values of the convergence indicators of IPM have changed on average less that 1% in the last five iterations.

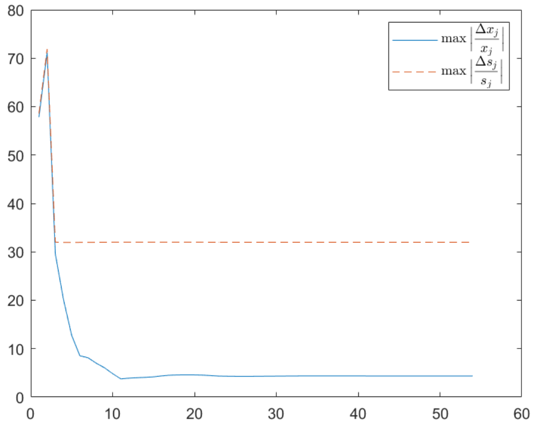
The practical criterion presented above is clearly different from the theoretical one shown in Algorithm IPM-I. However, the condition $S\Delta x + X\Delta s = \sigma\mu e - XSe$ is automatically satisfied if we reconstruct Δx from Δs; moreover, due to the stagnation when we decide to stop the CG, the values of the stepsizes and of the infeasibilities are close to the ones that we would obtain computing the exact direction; thus, it is reasonable to assume that the theoretical criterion on the residual (15) is also satisfied, with ω close to the value 1. Concerning the remaining theoretical condition (14), we notice that there is also a stagnation of the quantities $\max|\Delta x_j/x_j|$ and $\max|\Delta s_j/s_j|$, as shown in Figure 2. Thus, the practical stopping criterion, despite being different from the one that we use to prove polynomial complexity, allows to obtain a direction that is likely to satisfy also the theoretical stopping criterion. Therefore, we expect the new IPM-I, that exploits the novel

Figure 1: Infeasibility, complementarity and stepsizes computed at every CG iteration, for the 10th and 20th IPM iteration.



stopping criterion, to converge in a number of iterations similar or slightly larger than the original IPM, provided that the tolerance ε is chosen appropriately.

Figure 2: Values of $\max|\Delta x_j/x_j|$ and $\max|\Delta s_j/s_j|$ throughout the CG iterations at the 20th IPM iteration.



5 Numerical results

In this section we introduce test problems and show the results that we obtained with the standard CG or MINRES and with the novel IPCG or IPMINRES. We show overall results in terms of IPM iterations, inner iterations and computational time, and then provide also an insight into the individual IPM iterations to demonstrate where the gains resulting from the new method are the most significant.

All the numerical experiments were performed using MATLAB R2019a, on a computer with a quad-core 1.7GHz processor and 16GB of RAM.

5.1 Tomographic reconstruction

Our first test problem involves the reconstruction of an image obtained with a dual-energy x-ray tomography [25]. This is a classical inverse problem in many practical fields, from medicine to industrial applications. The noise in the measurements and the requirement of using as few angles of measurement as possible (e.g. to minimize the radiation dose to a patient), make this kind of problem challenging. The goal is to understand a spatial distribution of two different materials, for example the bone and soft tissue; to do so, we discretize the domain of interest and introduce two vectors $x_1, x_2 \in \mathbb{R}^n$, which contain information about the concentration of the two materials in the points of the discretization. In [25], the authors propose a new regularization technique which replaces the standard Joint Total Variation approach and exploits the inner product $x_1^T x_2$ to enforce the separation of the two materials.

If we stack together the vectors x_1 and x_2 into a single vector $x \in \mathbb{R}^{2n}$, the optimization problem that arises takes the following form

$$\min_{x\geq 0} \;\|m - \mathcal{G}x\|^2 + \rho\|x\|^2 + 2\eta\, x_1^T x_2,$$

where m is the measurement vector and $\mathcal{G}$ is an operator that incorporates information about the geometry of the problem and the materials used; we have used two coefficients,

ρ for the Tikhonov regularization and η for the novel regularizer. Written as a standard QP, the problem reads

$$\min_{x\geq 0}\ \frac{1}{2}x^T\mathcal{Q}x - m^T\mathcal{G}x,$$

where

$$\mathcal{Q} = \begin{bmatrix} c_{11}^2 + c_{21}^2 & c_{11}c_{12} + c_{21}c_{22} \\ c_{11}c_{12} + c_{21}c_{22} & c_{12}^2 + c_{22}^2 \end{bmatrix} \otimes R^TR + \begin{bmatrix} \rho & \eta \\ \eta & \rho \end{bmatrix} \otimes I.$$

Here c_{11}, c_{12}, c_{21} and c_{22} describe the attenuation constants of the two materials for the two x-ray energies used, while R contains information about the geometry of the measurements and can only be accessed via matrix-vector products performed using the Radon transform; $\otimes$ denotes the Kronecker product.

This optimization problem does not have linear equality constraints; if we apply an interior point method and formulate the normal equations, we obtain a linear system with matrix $\mathcal{Q} + X^{-1}S$. The structure of matrix R^TR allows the use of a block-diagonal preconditioner

$$P = \begin{bmatrix} (c_{11}^2 + c_{21}^2)\nu I + \rho I & (c_{11}c_{12} + c_{21}c_{22})\nu I + \eta I \\ (c_{11}c_{12} + c_{21}c_{22})\nu I + \eta I & (c_{12}^2 + c_{22}^2)\nu I + \rho I \end{bmatrix} + X^{-1}S,$$

where ν approximates the main diagonal of the blocks in R^TR. Therefore, it is possible to apply the CG with this positive definite preconditioner to find the IPM direction. The application of the matrix of the system is particularly expensive, since it involves the call of the Radon and inverse Radon transforms, to apply R and R^T respectively; thus, a single CG iteration is particularly expensive and we expect that the IPCG can bring a substantial benefit.

We show in Tables 1 and 2 the results of the application of IPM to this problem: we used 3 centrality correctors and an IPM tolerance of 10^{-8}; the CG tolerance was set to 10^{-6}, the IPCG tolerance ε was 0.01 or 0.001 and we chose a value `itstart` $= 5$, which means that we perform 5 standard CG iterations before starting to use the new stop criterion. The parameter q indicates how fine the discretization is; we show also the size of the matrix, equal to $2q^2$. We report the number of IPM iterations, CG iterations and the computational time for both methods; in the last two columns, we show the reduction of the number of CG iterations and time when using IPCG instead of the standard CG. (Both methods CG and IPCG use the same preconditioner.)

Table 1: Comparison of the results with the standard CG and the IPCG with $\varepsilon = 1\%$.

		Standard CG			IPCG, $\varepsilon = 0.01$			Inner It	Time
q	Size	IPM	Inner It	Time	IPM	Inner It	Time	red %	red %
32	2,048	18	3,810	7.46	19	586	1.44	84.6	80.7
64	8,192	20	6,301	35.04	24	1,149	6.29	81.8	82.0
128	32,768	23	9,249	140.91	26	1,366	23.02	85.2	83.7
256	131,072	26	15,115	817.45	32	1,763	106.36	88.3	87.0
512	524,288	29	25,112	5,174.26	49	2,639	639.92	89.5	87.6

It is worth observing that when using IPM with the new stopping criterion, the number of outer (IPM) iterations is very close to the one obtained with the original IPM; this confirms that the inexact direction that we compute is sufficiently precise so as not to destroy the convergence properties of IPM. In particular, we see that using a lower tolerance ε guarantees an IPM iteration count closer to the original one, as we could expect.

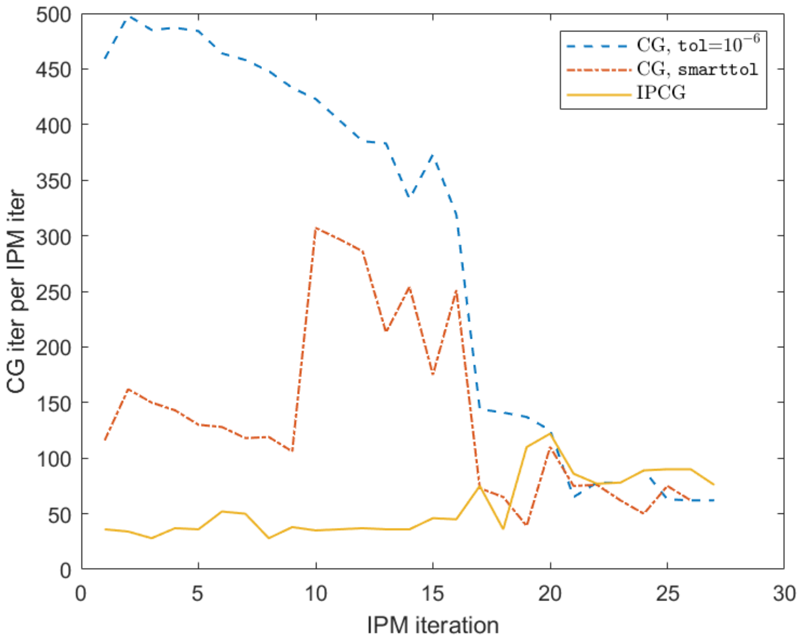
Table 2: Comparison of the results with the standard CG and the IPCG with $\varepsilon = 0.1\%$.

		Standard CG			IPCG, $\varepsilon = 0.001$			Inner It	Time
q	Size	IPM	Inner It	Time	IPM	Inner It	Time	red %	red %
32	2,048	18	3,810	7.46	19	1,038	2.25	72.8	69.8
64	8,192	20	6,301	35.04	24	1,484	7.90	76.4	77.5
128	32,768	23	9,249	140.91	25	1,986	32.69	78.5	76.8
256	131,072	26	15,115	817.45	28	2,678	157.79	82.3	80.7
512	524,288	29	25,112	5,174.26	34	3,772	881.90	85.0	83.0

Next, we notice that the overall CG iterations are reduced by a factor going from 72 to 90%; this suggests that most of the inner iterations usually performed inside the IPM are not necessary at all, since an accurate direction can be found much more easily. The lower iteration count produces a strong reduction of the computational time, from 70 to 88%: this reflects the high cost of the matrix-vector product inside the CG, which implies a very low cost of the operations added in Algorithm IPCG. If we compare the results in the last row of Table 1, we can see that the time per CG iteration goes from 206ms in the case of standard CG to 243ms for the IPCG, an increase of only 18% against a 90% reduction of the number of iterations.

We can see that using $\varepsilon = 0.001$ produces a smaller reduction of CG iterations, but a more stable IPM iterations count; we also highlight the fact that the problem is initialized randomly and with the more inaccurate version ($\varepsilon = 0.01$) it might happen in some runs that the IPM does not converge. We thus think that it is better to use a smaller value of ε especially when the dimension of the problem becomes very large.

Figure 3: Comparison of the number of CG iterations needed at each IPM iteration, for the case $q = 128$ and $\varepsilon = 0.001$.



Next, we want to understand how the gain of IPCG is distributed during the IPM iterations. To do this, we recorded the number of CG iterations at each IPM iteration (summing together the inner iterations for predictors and correctors) in three different situations: when using CG with tolerance 10^{-6}; when using IPCG with $\varepsilon = 10^{-3}$; when using CG with a variable tolerance, so that for a third of the outer iterations we set the tolerance at 10^{-2}, for another third at 10^{-4} and for the last third at 10^{-6}. We highlight that this last approach, that we call `smarttol`, is not viable in practice, since we would need to know a priori an estimate of the number of IPM iterations; however, it is interesting to compare this approach with the IPCG. Figure 3 shows the comparison of the iterations for the problem with $q = 128$.

We can see that, when using standard CG with fixed tolerance, the number of iterations decreases at the end, since less correctors are computed; when using IPCG, we do not observe this decrease, since the smaller number of correctors is balanced by the increased accuracy needed. Indeed, in the late IPM phase, the new stop criterion is not triggered and IPCG stops with the standard reduction test; the reader may observe that the two graphs overlap in the last iterations. However, in the initial phase, we observe a significant advantage of IPCG over the standard CG. Even if a variable tolerance is used, the standard (varying) stopping criterion is still unable to match the performance of the method proposed in this paper. In particular, the IPCG delivers a significant gain in the initial phase of IPM.

We also analyzed the CG final relative residual at each CG call (for predictors and correctors), in the case of standard CG with tolerance 10^{-6} and IPCG, which is shown in Figure 4; since we compute on average 3 or 4 directions at each IPM iteration, the number of CG calls is approximately three times larger than the number of outer iterations. We can see that the final residual of IPCG displays noticeable irregularity: in particular, it seems that it is possible to stop the computation of the corrector directions much earlier than that of the predictors. Given this surprising behaviour of the final residual, it is impossible to match the performance of the new stopping criterion using a standard residual test.

The graphs displayed in these two figures undeniably confirm that a high accuracy in the first IPM iterations is not needed at all, and that the best method to decide when a direction is sufficiently precise to perform the next IPM iteration successfully should be based on the IPM indicators and not on the residual of the linear system.

5.2 Compressed sensing

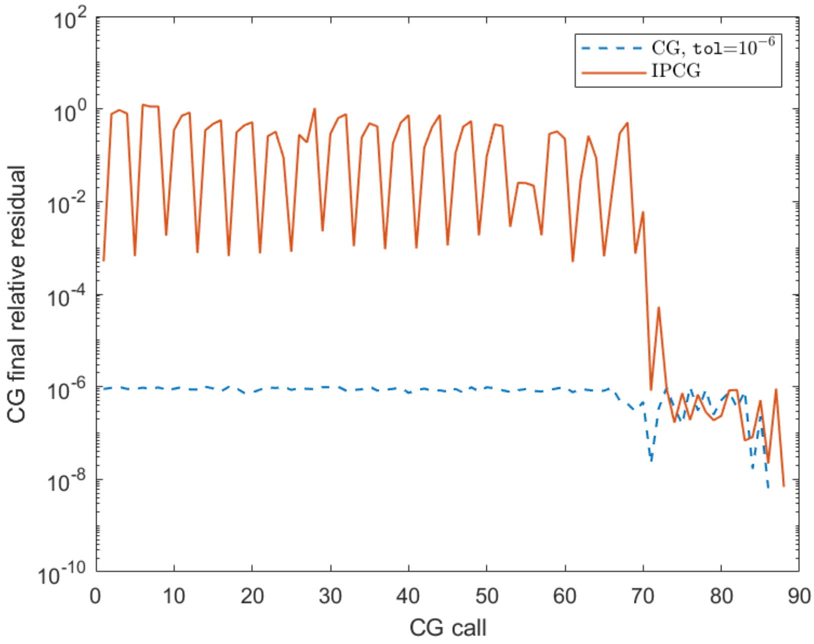
The second test problem arises from compressed sensing [18]: we look for a sparse solution to an undetermined linear system $Ax = b$, where sparsity is enforced by means of a 1-norm regularization. If we linearize the 1-norm by adding extra variables, the optimization problem that arises is the following

$$\min_{z \geq 0} \tau e^T z + \frac{1}{2} \|F^T z - b\|^2,$$

where $\tau > 0$, $z = \begin{bmatrix} u \ ; & v \end{bmatrix}$, u and v being the positive and negative parts of vector x, and $F^T = \begin{bmatrix} A & -A \end{bmatrix}$. Rewriting it as a standard quadratic program and formulating the IPM normal equations, we see that the matrix of the linear system to be solved is

$$H = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \otimes A^T A + \Theta^{-1}.$$

Figure 4: Comparison of the final relative residual for CG and IPCG.



Due to the structure of matrix A, which satisfies the restricted isometry property (see [18] for all the details), matrix H can be efficiently preconditioned by the block diagonal matrix

$$P = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \otimes \eta I + \Theta^{-1}$$

for an appropriate constant η. The difference with respect to the first test problem is that now the IPM direction is computed using a very low accuracy for the CG: the residual tolerance is just 10^{-2} throughout all the IPM iterations.

The test problems are taken from the Sparco collection [38]; of the 18 problems considered in [18], 5 did not show any improvement when using IPCG instead of CG (in part because they were easy enough and the CG was already performing a low number of iterations). We show the results for the remaining 13 that did show improvement in Table 3. The IPM tolerance varies between 10^{-6} and 10^{-10} according to the problem being solved and no corrector direction is used. The default values for IPCG are $\varepsilon = 0.01$ and $\texttt{itstart} = 5$, but some problems required different parameters, which are indicated in the Table. The dual residual did not show consistent stagnation during the CG iterations, thus we applied IPCG using only the stagnation of complementarity as the stop criterion.

We can see that all these problems display an impressive reduction in the number of CG iterations and CPU time, even if the standard CG tolerance is very high. The added cost of IPCG varies throughout the problems, but on average is roughly $35 - 40\%$ of the original iteration cost. Sometimes we see also a reduction in IPM iterations; this may be because our inexact method is finding by chance a direction that is better than the exact one.

5.3 PDE constrained optimization

As a last test problem, we considered PDE constrained optimization problems (see e.g. [34]) and we used the augmented system approach, in order to test Algorithm IPMINRES.

Table 3: Results for Compressed sensing IPM; stop criterion based only on complementarity.

		Standard CG			IPCG			Inner It	Time
ID	Size	IPM	Inner It	Time	IPM	Inner It	Time	red %	red %
6	4,096	22	2,128	41.53	19	180	4.38	91.5	89.5
9	256	11	382	0.13	11	165	0.08	56.8	37.6
10	2,048	12	2,210	0.64	12	768	0.28	65.2	56.3
11 †*	2,048	19	663	1.55	21	536	1.22	19.2	21.3
401	114,688	14	160	15.20	12	55	6.31	65.6	58.5
402 †	172,032	14	238	29.70	12	59	9.61	75.2	67.6
403	393,216	19	2,282	203.55	25	280	34.56	87.7	83.0
601	8,192	20	2,146	105.90	17	462	19.33	78.5	81.7
602	8,192	22	2,280	118.38	20	453	18.83	80.1	84.1
603	8,192	16	1,085	17.53	12	72	2.19	93.4	87.5
701 †	131,072	12	1,028	39.25	12	216	12.39	79.0	68.4
702	32,768	8	926	14.75	8	128	3.48	86.2	76.4
903 †	2,048	13	1,794	2.18	17	979	1.27	45.4	41.7

†: $\varepsilon = 0.1\%$ instead of 1%, *: `itstart` = 20 instead of 5

In this section, we will indicate with $\hat{v}$ and v respectively the continuous and discretized version of a variable v. The kind of problems we used involve PDE as constraints and they take the standard form

$$\begin{aligned} \min_{y,u} \quad & \frac{1}{2}\|\hat{y} - \hat{y}_0\|_{L^2}^2 + \frac{\beta}{2}\|\hat{u}\|_{L^2}^2 \\ \text{s.t.} \quad & -\nabla^2 \hat{y} = \hat{u} + \hat{f}, \quad \hat{y} \in \Omega \\ & \hat{y} = \hat{g}, \quad \hat{y} \in \partial\Omega \\ & \hat{u}_a \leq \hat{u} \leq \hat{u}_b \end{aligned}$$

where Ω is the domain of evolution of the problem, $\hat{y}, \hat{u}$ are the state and control variables, $\hat{y}_0$ is the desired state function, $\hat{f}, \hat{g}, \hat{u}_a, \hat{u}_b$ are given functions and $\beta > 0$ is the regularization parameter. The objective of this formulation is to keep the state $\hat{y}$ close to the fixed desired state $\hat{y}_0$ and minimize the control $\hat{u}$, while satisfying the PDE and bound constraints.

We apply a standard IPM to this problem, using the discretize-than-optimize approach, as described in [34], and obtain the discretized quantities y, u, y_0, u_a, u_b; we introduce the variables z_a and z_b defined as $(z_a)_j = \mu/(u - u_a)_j$ and $(z_b)_j = \mu/(u_b - u)_j$. We use a standard Q1 finite elements discretization, formulate the augmented system, and obtain

$$\begin{bmatrix} M & 0 & K \\ 0 & \beta M + \Theta & -J \\ K & -J & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta u \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} r_y \\ r_u \\ r_\lambda \end{bmatrix},$$

where $M \in \mathbb{R}^{n \times n}$ is the finite elements mass matrix, $J \in \mathbb{R}^{n \times n}$ is the same matrix but with boundary conditions applied, $K \in \mathbb{R}^{n \times n}$ is the stiffness matrix, $\Theta = Z_a(U - U_b)^{-1} + Z_b(U_a - U)^{-1}$, $\lambda \in \mathbb{R}^n$ is the vector of Lagrange multipliers. The dimension of the matrices n is determined by the discretization parameter n_c, as $n = (2^{n_c} + 1)^2$; the whole augmented system has size $3n$.

We can solve this linear system using MINRES, provided that we use a positive definite preconditioner; exploiting the ideas in [34] and [35], we use the following preconditioner

$$P = \begin{bmatrix} \tilde{M} & 0 & 0 \\ 0 & \beta\tilde{M}+\Theta & 0 \\ 0 & 0 & \tilde{S} \end{bmatrix},$$

where $\tilde{M}$ contains only the diagonal of M and $\tilde{S}$ is an approximation of the Schur complement

$$\tilde{S} = \Big(K + \frac{1}{\sqrt{\beta}}J\Big)M^{-1}\Big(K + \frac{1}{\sqrt{\beta}}J\Big).$$

The Schur complement preconditioner is constant throughout the IPM iterations and to apply it we compute the Cholesky factorization of $(K + J/\sqrt{\beta})$ once at the beginning of the algorithm.

We computed the finite element matrices using the IFISS package [1, 16, 17]. We used an IPM tolerance of 10^{-8} and a MINRES standard tolerance of 10^{-8}, with 400 maximum iterations; we did not apply any predictor-corrector strategy. Applying the IPMINRES, we noticed that only the complementarity μ was able to reach stagnation, like in the case of compressed sensing problems; hence, we used a stopping criterion that checks only the complementarity μ. In Table 4, we show the parameters of the problems that we considered, in terms of values of n_c and β, size of the linear system, IPMINRES parameters ε and `itstart` that produced the best results. In Table 5 instead we show the comparison of the results obtained using MINRES and IPMINRES, in terms of IPM iterations, MINRES iterations and computational time.

Table 4: Size of the augmented system, IPMINRES tolerance ε and parameter `itstart` for all the combinations of β and n_c used.

		$\beta = 10^{-4}$		$\beta = 10^{-5}$		$\beta = 10^{-6}$	
n_c	Size	ε	`itstart`	ε	`itstart`	ε	`itstart`
5	3,267	10^{-3}	15	10^{-3}	15	10^{-4}	20
6	12,675	10^{-3}	15	10^{-3}	15	10^{-4}	15
7	49,932	10^{-3}	15	10^{-3}	15	10^{-4}	15
8	198,147	10^{-3}	15	10^{-3}	15	10^{-4}	15

We can see that using the IPMINRES produces a significant reduction of inner iterations and computational time, like in the previous test problems; the IPM iterations grow slightly as a result of the less exact directions used. We highlight also that, when β becomes smaller, making the linear system harder to solve, the gain of IPMINRES grows; moreover, the gain in computational time is larger for larger problems. This is encouraging, because it means that the method proposed in this paper is not adversely affected when the linear systems become more ill conditioned.

This last test problem shows that the IPMINRES approach applied to the augmented system is viable and can produce significant improvements with respect to the standard techniques; it also shows that our stopping criterion can be applied to a wide range of different problems, delivering consistent improvement across the board.

6 Conclusion

We have shown that it is possible to stop the inner Krylov iterations during an interior point method much earlier than it was previously thought, provided that the stopping

Table 5: Results with standard MINRES and the new IPMINRES for the PDE constrained optimization problem.

		MINRES			IPMINRES			Inner It	Time
β	n_c	IPM	Inner It	Time	IPM	Inner It	Time	red %	red %
10^{-4}	5	24	958	0.55	29	504	0.34	47.4	38.2
	6	25	1,003	5.08	29	494	2.83	50.7	44.3
	7	26	1,041	45.12	31	523	24.51	49.8	45.7
	8	30	1,159	450.91	34	592	237.58	48.9	47.3
10^{-5}	5	23	1,631	0.87	32	745	0.47	54.3	46.0
	6	27	1,944	9.64	34	751	4.40	61.4	54.4
	7	29	2,125	90.69	37	835	38.08	60.7	58.0
	8	31	2,262	878.72	40	959	378.55	57.6	56.9
10^{-6}	5	26	3,702	1.75	37	1,645	0.98	55.6	44.0
	6	30	4,502	27.80	39	1,626	9.01	63.9	67.6
	7	31	4,950	210.03	41	1,869	82.13	62.2	60.9
	8	35	5,640	2,121.84	41	1,817	714.84	67.8	66.3

criterion used is based on the IPM convergence indicators and not only on the reduction of the residual of the linear system. We have given a polynomial complexity proof in the case of a quadratic program with an ideal stopping criterion and we have proposed two practical algorithms for the normal equations and augmented system approaches. They exploit the stagnation of some of the IPM indicators, depending on the problem, and are only marginally more computationally expensive then the original algorithms.

We have provided computational evidence for a wide range of problems, from image processing, compressed sensing and PDE-constrained applications; they all display a significant reduction in the number of inner Krylov iterations and computational time. In particular, the largest gain appears in the early IPM phase, where it is already known that a lower accuracy of Newton directions is sufficient; however, we have shown that it is extremely difficult to mimic the behaviour of our stopping criterion using only a residual test, since the residual of the optimal stopping point may vary drastically during the IPM iterations. This fact strongly supports our claim that a good stopping criterion for CG or MINRES should be based on the IPM convergence indicators. For some test problems, the numerical results seem to suggest that the relative performance of the new stopping criterion improves when the size of the problem grows and/or its ill-conditioning increases.

We strongly believe that many other practical optimization algorithms in which a Krylov subspace method is used to solve the linear equation systems are likely to benefit from a specialized stopping criterion developed with an understanding of the specific needs of the method.

Acknowledgements

We thank John W. Pearson for providing codes and fruitful discussions about PDE problems.

References

[1] *IFISS*. https://personalpages.manchester.ac.uk/staff/david.silvester/ifiss/.

[2] *MINRES*. https://web.stanford.edu/group/SOL/software/minres/.

[3] A. Altman and J. Gondzio, *Regularized symmetric indefinite systems in interior point methods for linear and quadratic optimization*, Optimization Methods and Software, 11-12 (1999), pp. 275–302.

[4] M. Arioli, *A stopping criterion for the conjugate gradient algorithm in a finite element method framework*, Numerische Mathematik, 97 (2004), pp. 1–24.

[5] O. Axelsson and I. Kaporin, *Error norm estimation and stopping criteria in preconditioned conjugate gradient iterations*, Numerical Linear Algebra with Applications, 8 (2001), pp. 265–286.

[6] S. Bellavia, *Inexact Interior-Point Method*, Journal of Optimization Theory and Applications, 96 (1998), pp. 109–121.

[7] L. Bergamaschi, J. Gondzio, A. Martínez, J. W. Pearson, and S. Pougkakiotis, *A new preconditioning approach for an interior point-proximal method of multipliers for linear and convex quadratic programming*, Numerical Linear Algebra with Applications.

[8] L. Bergamaschi, J. Gondzio, M. Venturin, and G. Zilli, *Inexact constraint preconditioners for linear systems arising in interior point methods*, Computational Optimization and Applications, 36 (2007), pp. 137–147.

[9] L. Bergamaschi, J. Gondzio, and G. Zilli, *Preconditioning Indefinite Systems in Interior Point Methods for Optimization*, Computational Optimization and Applications, 28 (2004), pp. 149–171.

[10] S. Bocanegra, F. Campos, and A. Oliveira, *Using a Hybrid Preconditioner for Solving Large-Scale Linear Systems arising from Interior Point Methods*, Computational Optimization and Applications, 36 (2007), pp. 149–164.

[11] S. Cafieri, M. D'Apuzzo, V. De Simone, and D. di Serafino, *Stopping criteria for inner iterations in inexact potential reduction methods: a computational study*, Computational Optimization and Applications, 36 (2007), pp. 165–193.

[12] M. Colombo and J. Gondzio, *Further Development of Multiple Centrality Correctors for Interior Point Methods*, Computational Optimization and Applications, 41 (2008), pp. 277–305.

[13] M. D'Apuzzo, V. De Simone, and D. di Serafino, *On mutual impact of numerical linear algebra and large-scale optimization with focus on interior point methods*, Computational Optimization and Applications, 45 (2010), pp. 283–310.

[14] D. Ek and A. Forsgren, *Approximate solution of system of equations arising in interior-point methods for bound-constrained optimization*, Computational Optimization and Applications, 79 (2021), pp. 155–191.

[15] ——, *A structured modified Newton approach for solving systems of nonlinear equations arising in interior-point methods for quadratic programming*, 2021. arXiv:2009.07913[math.OC].

[16] H. C. Elman, A. Ramage, and D. J. Silvester, *IFISS: a Matlab toolbox for modelling incompressible flow*, ACM Transactions on Mathematical Software, 33 (2007).

[17] ——, *IFISS: A computational laboratory for investigating incompressible flow problems*, SIAM Review, 56 (2014), pp. 261–273.

[18] K. Fountoulakis, J. Gondzio, and P. Zhlobich, *Matrix-free interior point method for compressed sensing problems*, Mathematical Programming Computation, 6 (2014), pp. 1–31.

[19] M. P. Friedlander and D. Orban, *A primal-dual regularized interior-point method for convex quadratic programs*, Mathematical Programming Computation, 4 (2012), pp. 71–107.

[20] S. Gazzola and M. Sabaté Landman, *Krylov methods for inverse problems: Surveying classical, and introducing new, algorithmic approaches*, GAMM-Mitteilungen, 43.

[21] J. Gondzio, *Multiple Centrality Corrections in a Primal-Dual Method for Linear Programming*, Computational Optimization and Applications, 6 (1996), pp. 137–156.

[22] ——, *Interior Point Methods 25 Years Later*, European Journal of Operational Research, 218 (2012), pp. 587–601.

[23] ——, *Matrix-Free Interior Point Method*, Computational Optimization and Applications, 51 (2012), pp. 457–480.

[24] ——, *Convergence Analysis of an Inexact Feasile Interior Point Method for Convex Quadratic Programming*, SIAM Journal on Optimization, 23 (2013), pp. 1510–1527.

[25] J. Gondzio, L. Matti, S.-M. Latva-Aijo, S. Siltanen, and F. Zanetti, *Material-separating regularizer for multi-energy X-ray tomography*, 2021. In preparation.

[26] J. Gondzio and F. N. C. Sobral, *Quasi-Newton approaches to interior point methods for quadratic problems*, Computational Optimization and Applications, 74 (2019), pp. 93–120.

[27] M. E. Hochstanbach and Y. Notay, *Controlling inner iterations in the Jacobi-Davidson method*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 460–477.

[28] B. Morini and V. Simoncini, *Stability and Accuracy of Inexact Interior Point Methods for Convex Quadratic Programming*, Journal of Optimization Theory and Applications, 175 (2017), pp. 450–477.

[29] B. Morini, V. Simoncini, and M. Tani, *Spectral estimates for unreduced symmetric KKT systems arising from interior point methods*, Numerical Linear Algebra with Applications, 23 (2016), pp. 776–800.

[30] ——, *A comparison of reduced and unreduced KKT systems arising from interior point method*, Computational Optimization and Applications, 68 (2017), pp. 1–27.

[31] Y. Notay, *Combination of Jacobi–Davidson and conjugate gradients for the partial symmetric eigenproblem*, Numerical Linear Algebra with Applications, 9 (2002), pp. 21–44.

[32] A. R. L. Oliveira and D. C. Sorensen, *A New Class of Preconditioners for Large-Scale Linear Systems from Interior Point Methods for Linear Programming*, Linear Algebra and its Applications, 394, pp. 1–24.

[33] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM Journal of Numerical Analysis, 12 (1975), pp. 617–629.

[34] J. W. Pearson and J. Gondzio, *Fast interior point solution of quadratic programming problems arising from PDE-constrained optimization*, Numerische Mathematik, 137 (2017), pp. 959–999.

[35] J. W. Pearson and A. J. Wathen, *A new approximation of the Schur complement in preconditioners for PDE-constrained optimization*, Numerical Linear Algebra with Applications, 19 (2011), pp. 816–829.

[36] S. Pougkakiotis and J. Gondzio, *Dynamic Non-diagonal Regularization in Interior Point Methods for Linear and Convex Quadratic Programming*, Journal of Optimization Theory and Applications, 181 (2019), pp. 905–945.

[37] A. Stathopoulos, *Neraly optimal preconditioned methods for Hermitian eigenproblems under limited memory. Part i: seeking one eigenvalue*, SIAM J. Sci. Comput., 29 (2007), pp. 481–514.

[38] E. Van Den Berg, M. P. Friedlander, G. Hennenfent, F. J. Herrman, R. Saab, and O. Yilmaz, *Sparco: a testing framework for sparse reconstruction*, ACM Trans. Math. Softw., 35 (2009), pp. 1–16.

[39] M. I. Velazco, A. R. L. Oliveira, and F. F. Campos, *A note on hybrid preconditioners for large-scale normal equations arising from interior-point methods*, Optimization Methods and Software, 25 (2010), pp. 321–332.

[40] S. J. Wright, *A path-following interior-point algorithm for linear and quadratic problems*, Annals of Operations Research, 62 (1996), pp. 103–130.

[41] ——, *Primal-Dual Interior-Point Methods*, SIAM, 1997.