



# Proximal Stabilized Interior Point Methods and *Low-Frequency-Update* Preconditioning Techniques

Stefano Cipolla<sup>1</sup> · Jacek Gondzio<sup>1</sup>

Received: 3 May 2022 / Accepted: 1 March 2023 / Published online: 5 April 2023  
© The Author(s) 2023

## Abstract

In this work, in the context of Linear and convex Quadratic Programming, we consider Primal Dual Regularized Interior Point Methods (PDR-IPMs) in the framework of the Proximal Point Method. The resulting Proximal Stabilized IPM (PS-IPM) is strongly supported by theoretical results concerning convergence and the rate of convergence, and can handle degenerate problems. Moreover, in the second part of this work, we analyse the interactions between the regularization parameters and the computational footprint of the linear algebra routines used to solve the Newton linear systems. In particular, when these systems are solved using an iterative Krylov method, we are able to show—using a new rearrangement of the Schur complement which exploits regularization—that general purposes preconditioners remain attractive for a series of subsequent IPM iterations. Indeed, if on the one hand a series of theoretical results underpin the fact that the approach here presented allows a better re-use of such computed preconditioners, on the other, we show experimentally that such (re)computations are needed only in a fraction of the total IPM iterations. The resulting regularized second order methods, for which *low-frequency-update* of the preconditioners are allowed, pave the path for an alternative class of second order methods characterized by reduced computational effort.

**Keywords** Interior point methods · Proximal point methods · Regularized primal-dual methods · Convex quadratic programming

**Mathematics Subject Classification** 65K05 · 90C51 · 90C06

---

Communicated by Nobuo Yamashita.

---

✉ Stefano Cipolla  
scipolla@ed.ac.uk

✉ Jacek Gondzio  
j.gondzio@ed.ac.uk

<sup>1</sup> School of Mathematics, The University of Edinburgh, Edinburgh, UK

## 1 Introduction

In this work we consider the problem of solving the following primal-dual convex quadratic programs:

$$\begin{aligned}
 \min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x}) &:= \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{g}^T \mathbf{x} & \max_{\mathbf{x} \in \mathbb{R}^d, \mathbf{y} \in \mathbb{R}^m, \mathbf{s} \in \mathbb{R}^{|\mathcal{C}|}} & \mathbf{b}^T \mathbf{y} - \frac{1}{2} \mathbf{x}^T H \mathbf{x} \\
 \text{s.t. } A \mathbf{x} &= \mathbf{b} & & \\
 \mathbf{x} &= [\mathbf{x}_{\mathcal{F}}, \mathbf{x}_{\mathcal{C}}], \mathbf{x}_{\mathcal{F}} \in \mathbb{R}^{\bar{d}}, \text{ s.t. } H \mathbf{x} + \mathbf{g} - A^T \mathbf{y} - \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} = 0 & (1) & \\
 0 \leq \bar{d} &\leq d & & \mathbf{s} \geq 0, \\
 \mathbf{x}_{\mathcal{C}} \geq 0, & \mathbf{x}_{\mathcal{F}} \text{ free} & &
 \end{aligned}$$

where the dual problem has been obtained considering the Lagrangian function

$$\widehat{\mathcal{L}}(\mathbf{x}, \mathbf{y}, \mathbf{s}) = \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{g}^T \mathbf{x} - \mathbf{y}^T (A \mathbf{x} - \mathbf{b}) - \mathbf{s}^T \mathbf{x}_{\mathcal{C}}, \quad (2)$$

and where  $H \in \mathbb{R}^{d \times d}$  with  $H \succeq 0$ ,  $A \in \mathbb{R}^{m \times d}$  with  $m \leq d$  is not required to have full rank. We assume, moreover, for simplicity of exposition and w.l.g., that  $\mathcal{F} = \{1, \dots, \bar{d}\}$  and  $\mathcal{C} = \{\bar{d} + 1, \dots, d\}$  for some  $0 \leq \bar{d} \leq d$ , and define  $\mathcal{F} = \emptyset$  if  $\bar{d} = 0$  or  $\mathcal{C} = \emptyset$  if  $\bar{d} = d$ .

For the past few decades, Interior Point Methods (IPMs) [13, 31] have gained wide appreciation due to their remarkable success in solving linear and convex quadratic programming problems (1). The computational cost of an IPM iteration is dominated by the solution of a Karush–Kuhn–Tucker (KKT) linear system of the form

$$\begin{bmatrix} H + \Theta^{-1} & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{y} \end{bmatrix} = \begin{bmatrix} f_{\mathbf{x}} \\ f_{\mathbf{y}} \end{bmatrix}, \quad (3)$$

where the diagonal matrix  $\Theta := \text{diag}(\mathbf{x})^{-1} \text{diag}(\mathbf{s})$  and the right hand side changes at every iteration. The diagonal correction  $\Theta$  represents, somehow, the core of the IPM methodology and acts, essentially, as a continuous approximation of the indicator function for labelling active and non-active variables based on the magnitude of its diagonal elements: in the limit, these elements approach 0 or  $+\infty$ .

A closer look at the KKT system in (3), reveals how the astonishing polynomial worst-case complexity of IPMs [13, 31] is counterbalanced by an intrinsic difficulty for the optimal tuning of the linear algebra solvers required for their implementation. We briefly summarize two such issues:

- (I1) near rank deficiency of  $A$ , or near singularity of  $H + \Theta^{-1}$ , can give rise to inefficient or unstable solutions of the KKT linear systems. This may occur regardless of whether a direct or iterative method is used for their solution. Moreover, it is important to mention at this point the related issue concerning the fact that, in case of a rank deficiency of  $A$ , the theory of convergence for IPM is not clear.

- (I2) for large scale problems, the unwelcome feature displayed by the diagonal elements of  $\Theta^{-1}$  represents a paradigmatic example of how Krylov methods may be easily made ineffective due to the fact that the conditioning of the involved linear systems deteriorates as the IPM iterations proceed. As a result, the robustness and efficiency of IPMs depend heavily on the use of preconditioners which should be recomputed/updated at every iteration due to the presence of the rapidly changing matrix  $\Theta^{-1}$ .

## 1.1 Motivations and Background

In the last 20 years there has been an intense research activity regarding items (I1) and (I2) mentioned in the previous section. In particular, in order to alleviate some of the numerical difficulties related to (I1), it has been proposed in [2] to systematically modify the linear system (3) using, in essence, a diagonal regularization. Despite the fact that this strategy has proven to be effective in practice, to the best of our knowledge, in literature few works are devoted to the complete theoretical understanding of these regularization techniques. We mention [11], where the global convergence of an *exact primal-dual regularized IPM* is shown under the somehow strong hypothesis that the computed Newton directions are uniformly bounded (see [11, Th. 5.4]) and [25], where regularization is interpreted in connection with the Augmented Lagrangian Method [17] and the Proximal Method of Multipliers [26] and where the convergence to the true solution is recovered when the regularization is driven to zero at a suitable speed.

Concerning (I2), the literature is quite extensive and it is not possible to give a short comprehensive outlook. We refer the interested reader to [13, Sec. 5] and [6] for a comprehensive survey. We prefer to stress, instead, the fact that the presence of the iteration dependent matrix  $\Theta$  and its diverging elements represents, somehow, the true challenge in the efficient implementation of IPMs for large scale problems. As a matter of fact, the computational costs related to the *necessary* re-computations of factorizations and/or preconditioners for the Newton linear systems represent the main bottleneck of the existing implementations.

It is important to note, moreover, that when factorizations are used as back-solvers for Newton systems, the issue of reducing systematically the number of such necessary factorizations has been successfully addressed using the *Multiple Centrality Corrections* (MCC) strategy, see [5, 15]. Indeed, in this context, the use of a given factorization is maximized by applying it to solve several linear systems and producing modified Newton directions in which larger steps can be made, ultimately reducing the number of IPM iterations. On the other hand, when a *very good preconditioner* is not available, the use of Krylov iterative methods [30] for the solution of the Newton systems makes the MCC strategy consistently less appealing. At large, the results presented in this work aim at achieving the same goals as the MCC strategy, i.e., reducing the number of factorization re-computations, but when Krylov methods are used as back-solvers for the Newton systems and when a given factorization is used as preconditioner.

As it will be clear from our theoretical developments, the main tool used to solve/alleviate simultaneously the issues outlined in items (I1) and (I2) is the regularization. Indeed, broadly speaking, this work can be viewed as a study of the

interactions between the regularization parameters used in the Primal Dual Regularized Interior Point Methods (PDR-IPMs) and the computational footprint of the linear algebra routines used to solve the related Newton linear systems.

## 1.2 Contribution and Organization

The investigation which aims at addressing both issues (I1) and (I2) is naturally organized into two main threads. Indeed, in the first part of this work we aim at clarifying how alleviating the (near) rank deficiency of matrix  $A$  using regularization affects the convergence of the underlying IPM scheme. To this end, we build a bridge between Primal Dual Regularized IPMs (PDR-IPMs) and the Proximal Point Method (PPM) [28] giving a precise description of the synergies occurring between them. In particular, our analysis contributes to the understanding of the hidden machinery which controls the convergence of the PDR-IPM and clarifies, finally, the influence of regularization for PDR-IPMs: our proposal, the Proximal Stabilized IPM (PS-IPM), is strongly supported by theoretical results concerning convergence/rate of convergence and can handle degenerate problems as those described in (I1).

In the second part of this work, building the momentum from the developed convergence theory, we address (I2) using a PS-IPM perspective. Here we prove that regularization can be used, in fact, as a tool to pursue the challenging aspiration of reducing systematically the number of necessary preconditioner re-computations needed for the iterative solution of IPM Newton linear systems. Indeed, using an equivalent formulation of the LP/QP problem and a new rearrangement of the Schur complements for the related Newton systems, when such systems are solved using an iterative Krylov method, we are able to prove a series of theoretical results showing that the computed preconditioners are *re-usable* during the IPM iterations. The presented numerical results further underpin this claim showing that the number of sufficient preconditioner re-computations equals just a fraction of the total IPM iterations when general purpose preconditioners are used. As a straightforward consequence of the above findings, we are able to devise a class of IPM-type methods characterized by the fact that the re-computation of any given preconditioner can be performed in *low-frequency* regime, hence the linear algebra footprint of the method is significantly reduced.

The precise outline of the contribution and the organization of the work can be summarized as follows.

In Sects. 2 and 3, using the PPM [28] in its inexact variant [20], we prove the convergence of the PDR-IPM-type schemes for the solution of problem (1) without any further assumptions on the uniform boundedness of the Newton directions or assuming that the regularization is driven to zero. Indeed, if on one hand our PS-IPM sheds further light on the experimental evidence that regularization is of extreme importance for robust and efficient IPMs implementations, on the other, it is supported by a precise result, see Theorem 2.1, tying the magnitude of the regularization parameters to its rate of convergence. The experimental evidence of the goodness of the proposed framework and of the resulting implementation is presented in Sect. 5.2 where we show that, when direct methods are used for the solution of the Newton system, fixing the regularization

parameters to small values is preferable to a strategy in which a decreasing sequence of regularization parameters is employed.

In Sect. 4, we are able to depict a precise quantitative picture on the relations intervening between the regularization parameters and the necessity of recomputing any given preconditioner. Indeed, heavily relying on the form of the primal-dual regularized Newton systems and using a novel rearranging of their Schur complement which is based on a *separation of variables trick*, we propose and analyse a new preconditioning technique for which the *frequency of re-computation* depends inversely on the magnitude of the regularization parameters. As a result, in the proposed PS-IPM scheme, the overall computational footprint of the linear algebra solvers can be decreased at the cost of slowing down its rate of convergence.

Finally, in Sect. 5, we carry out an experimental analysis of PS-IPMs when the corresponding Newton systems are solved using a Krylov iterative method preconditioned as proposed in Sect. 4. The presented results show that the proposed algorithm can be tuned to obtain a number of preconditioner updates roughly equal to one third of the total IPM iterations while maintaining an IPM-type rate of convergence, leading, hence, to improvements of the computational performance in large scale settings.

### 1.3 Notation

In the following  $\mathbf{x}, \mathbf{y}, \mathbf{z}, \dots$  are vectors and, given  $Q \in \mathbb{R}^{u \times u}$ ,  $\|Q\|_2$  is the spectral norm and  $\lambda_i(Q)$  is its  $i$ -th eigenvalue. Given a set  $C \subset \mathbb{R}^u$ ,  $I_C(\cdot)$  denotes the characteristic function of the set, whereas  $\mathcal{B}(\cdot)$  and  $Cl(\cdot)$  denote, respectively, the boundary operator and the closure operator. When  $C \subset \mathbb{R}^u$  is convex,  $N_C(\mathbf{u})$  denotes the normal cone to  $C$  in  $\mathbf{u} \in \mathbb{R}^u$  (see [9, Sec. 2.1] or [4, Sect. 4]). The operators  $\partial_{\mathbf{x}} f(\mathbf{x}, \mathbf{y})$  and  $\partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$  denote the partial sub-differential operators of a given function  $f(\mathbf{x}, \mathbf{y})$ , see [27, Sec. 2] or [4, Sect. 4]. Moreover, given  $\mathbf{u} \in \mathbb{R}^u$  and a closed set  $C \subset \mathbb{R}^u$ , we define  $dist(\mathbf{u}, C) := \min\{\|\mathbf{z} - \mathbf{c}\| \text{ for } \mathbf{c} \in C\}$  and  $\Pi_C(\cdot)$  the associated projection operator. Finally, given  $\mathbf{u} \in \mathbb{R}^u$ , we define  $B(\mathbf{u}, \delta) := \{\mathbf{z} \in \mathbb{R}^u \text{ s.t. } \|\mathbf{z} - \mathbf{u}\| < \delta\}$  and we use the capital letter  $U$  to denote the matrix  $U := \text{diag}(\mathbf{u}) \in \mathbb{R}^{u \times u}$ .

## 2 Convex Formulation and the Proximal Point Algorithm

For problem (1) we consider the following Lagrangian function:

$$\mathcal{L}(\mathbf{x}, \mathbf{y}) = \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{g}^T \mathbf{x} - \mathbf{y}^T (A \mathbf{x} - \mathbf{b}) + I_D(\mathbf{x}, \mathbf{y}), \tag{4}$$

where  $D$  is the convex closed set

$$D := \mathbb{R}^{\bar{d}} \times \mathbb{R}_{\geq 0}^{d-\bar{d}} \times \mathbb{R}^m.$$

We notice that the particular Lagrangian considered in (4) is, somehow, non-standard in the IPM literature due to the presence of the term  $I_D(\mathbf{x}, \mathbf{y})$ . This choice allows us to consider variational formulation of problem (1) [see (7)] suitable for the application

of the PPM [28]. In the following Lemma 2.1, we briefly summarize the properties of the function  $I_D(\mathbf{x}, \mathbf{y})$  needed to obtain such a variational formulation.

**Lemma 2.1** *The following properties are true:*

1.

$$[\mathbf{v}, \mathbf{w}] \in N_D(\mathbf{x}, \mathbf{y}) \Leftrightarrow \begin{cases} \mathbf{v}_i = 0 & \text{for } i = 1, \dots, \bar{d} \\ \mathbf{v}_i = 0 & \text{if } \mathbf{x}_i > 0 \text{ and } i = \bar{d} + 1, \dots, d \\ \mathbf{v}_i \leq 0 & \text{if } \mathbf{x}_i = 0 \text{ and } i = \bar{d} + 1, \dots, d \\ \mathbf{w}_i = 0 & \text{for } i = 1, \dots, m \end{cases}, \quad (5)$$

*i.e.*,

$$N_D(\mathbf{x}, \mathbf{y}) = N_{\mathbb{R}^{\bar{d}} \times \mathbb{R}_{\geq 0}^{d-\bar{d}}}(\mathbf{x}) \times \{0\}.$$

2.

$$\partial_{\mathbf{x}} I_D(\mathbf{x}, \mathbf{y}) = N_{\mathbb{R}^{\bar{d}} \times \mathbb{R}_{\geq 0}^{d-\bar{d}}}(\mathbf{x}) \text{ and } \partial_{\mathbf{y}} I_D(\mathbf{x}, \mathbf{y}) = \{0\}.$$

**Proof** *Part 1* of the thesis follows using the definition of normal cone and observing that, since

$$\begin{aligned} \mathcal{B}(A \times B \times C) &= (\mathcal{B}(A) \times \text{Cl}(B) \times \text{Cl}(C)) \cup (\text{Cl}(A) \\ &\quad \times \mathcal{B}(B) \times \text{Cl}(C)) \cup (\text{Cl}(A) \times \text{Cl}(B) \times \mathcal{B}(C)) \end{aligned}$$

and  $\mathcal{B}(\mathbb{R}^{\bar{d}}) = \mathcal{B}(\mathbb{R}^m) = \emptyset$ , we have  $\mathcal{B}(D) = \mathbb{R}^{\bar{d}} \times \mathcal{B}(\mathbb{R}_{\geq 0}^{d-\bar{d}}) \times \mathbb{R}^m$ .

*Part 2* follows by the definition of normal cones and sub-differentials, see [4, Lemma 4.8].  $\square$

In our particular case, using Lemma 2.1, the saddle sub-differential operator related to (4) can be written as

$$T_{\mathcal{L}}(\mathbf{x}, \mathbf{y}) := \begin{bmatrix} \partial_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{y}) \\ \partial_{\mathbf{y}}(-\mathcal{L}(\mathbf{x}, \mathbf{y})) \end{bmatrix} = \begin{bmatrix} H\mathbf{x} + \mathbf{g} - A^T \mathbf{y} + \partial_{\mathbf{x}} I_D(\mathbf{x}, \mathbf{y}) \\ A\mathbf{x} - \mathbf{b} \end{bmatrix}. \quad (6)$$

The proper saddle function  $\mathcal{L}(\mathbf{x}, \mathbf{y})$  satisfies the hypothesis of [27, Cor. 2, p. 249] and hence the associated saddle operator, namely  $T_{\mathcal{L}}$ , is maximal monotone. In particular, the solutions  $(\mathbf{x}^*, \mathbf{y}^*) \in \mathbb{R}^{d+m}$  of the problem  $0 \in T_{\mathcal{L}}(\mathbf{x}, \mathbf{y})$  are just the saddle points of  $\mathcal{L}$ , if any. It is important to note, at this stage, that since  $T_{\mathcal{L}}$  is maximal monotone,  $T_{\mathcal{L}}^{-1}(0)$  is closed and convex.

Moreover, the problem of finding  $(\mathbf{x}^*, \mathbf{y}^*)$  s.t.  $0 \in T_{\mathcal{L}}(\mathbf{x}^*, \mathbf{y}^*)$  can be alternatively written as the one of finding a solution for the problem

$$-\begin{bmatrix} H & -A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \begin{bmatrix} -\mathbf{g} \\ \mathbf{b} \end{bmatrix} \in N_D(\mathbf{x}, \mathbf{y}), \quad (7)$$

which represents the variational inequality formulation of problem (6) (see [9, Sec. 2.1]).

Given  $(\mathbf{x}^*, \mathbf{y}^*)$  a solution of (7), we can recover a solution  $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$  of (1) defining  $\mathbf{s}^* := -\mathbf{v}^*(\bar{d} + 1 : d)$  where  $(\mathbf{v}^*, \mathbf{w}^*) \in N_D(\mathbf{x}^*, \mathbf{y}^*)$ . Indeed, it is easy to see using equations (2) and (5), that the vector  $\mathbf{s}^*$  corresponds to the standard Lagrange multiplier for the inequality constraints.

Before concluding this section, let us stress the fact that in the remainder of this work we make the following

**Assumption 1** The (convex) set of saddle points of  $\mathcal{L}$  is non-empty, i.e.,  $T_{\mathcal{L}}^{-1}(0) \neq \emptyset$ , and hence there exists at least one solution  $(\mathbf{x}^*, \mathbf{y}^*, \mathbf{s}^*)$  of the problem (1).

### 2.1 Proximal Point Method

In this section we follow essentially the developments from [7, 19] specializing our discussion for the operator  $T_{\mathcal{L}}$ . The Proximal Point Method (PPM) [28] finds zeros of maximal monotone operators by recursively applying their proximal operator. In particular, starting from an initial guess  $[\mathbf{x}_0, \mathbf{y}_0]$ , a sequence  $[\mathbf{x}_k, \mathbf{y}_k]$  of primal-dual pairs is generated as follows:

$$(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) = \mathcal{P}(\mathbf{x}_k, \mathbf{y}_k), \text{ where } \mathcal{P} = (I + \Sigma^{-1}T_{\mathcal{L}})^{-1} \tag{8}$$

$$\text{and } \Sigma := \text{blockdiag}(\rho I_d, \delta I_m), \text{ with } \rho > 0, \delta > 0.$$

Since  $\Sigma^{-1}T_{\mathcal{L}}$  is also a maximal monotone operator,  $\mathcal{P}$  is single valued, non expansive and the generated sequence converges to a solution  $[\mathbf{x}^*, \mathbf{y}^*] \in T_{\mathcal{L}}^{-1}(0)$  [28].

Evaluating the proximal operator  $\mathcal{P}$  is equivalent to finding a solution  $(\mathbf{x}, \mathbf{y})$  to the problem

$$0 \in T_{\mathcal{L}}(\mathbf{x}, \mathbf{y}) + \Sigma((\mathbf{x}, \mathbf{y}) - (\mathbf{x}_k, \mathbf{y}_k)),$$

which is guaranteed to have a unique solution. In particular, evaluating the proximal operator is equivalent to finding a solution of

$$0 \in \begin{bmatrix} H\mathbf{x} + \mathbf{g} - A^T\mathbf{y} + \partial_x I_D(\mathbf{x}, \mathbf{y}) + \rho(\mathbf{x} - \mathbf{x}_k) \\ A\mathbf{x} - \mathbf{b} + \delta(\mathbf{y} - \mathbf{y}_k) \end{bmatrix}$$

which, in turn, corresponds to solving the primal dual regularized problem in (RP):

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^d, \mathbf{y} \in \mathbb{R}^m} & \frac{1}{2}\mathbf{x}^T H\mathbf{x} + \mathbf{g}^T\mathbf{x} + \frac{\rho}{2}\|\mathbf{x} - \mathbf{x}_k\|^2 + \frac{\delta}{2}\|\mathbf{y}\|^2 \\ \text{s.t.} & A\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) = \mathbf{b} \\ & \mathbf{x}_{\mathcal{C}} \geq 0, \mathbf{x}_{\mathcal{F}} \text{ free} . \end{aligned} \tag{RP}$$

Moreover, the problem at hand (RP), can be written in the following variational form:

$$-\begin{bmatrix} H + \rho I & -A^T \\ A & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \begin{bmatrix} -\mathbf{g} + \rho \mathbf{x}_k \\ \mathbf{b} + \delta \mathbf{y}_k \end{bmatrix} \in N_D(\mathbf{x}, \mathbf{y}). \quad (9)$$

## 2.2 Inexact PPM

The inexact PPM has been originally analysed in [28] but we follow here the developments of [20]. We consider an approximate version of the PPM scheme in (8) where  $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$  satisfies the criterion  $(A_1)$  in [20], i.e.,

$$\begin{aligned} \|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - \mathcal{P}(\mathbf{x}_k, \mathbf{y}_k)\| &\leq \varepsilon_k \min\{1, \|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - (\mathbf{x}_k, \mathbf{y}_k)\|\}, \\ \text{where } \sum_{k=0}^{+\infty} \varepsilon_k &< \infty. \end{aligned} \quad (10)$$

Theorem 2.1 summarizes the results we are going to use in this work (the statements are specialized for our case):

**Theorem 2.1** 1. *The sequence  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k \in \mathbb{N}}$  generated by the recursion in (8) and using as inexactness criterion (a relaxed version of (10))*

$$\|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - \mathcal{P}(\mathbf{x}_k, \mathbf{y}_k)\| \leq \varepsilon_k, \text{ where } \sum_{k=0}^{+\infty} \varepsilon_k < \infty,$$

*is bounded if and only if Assumption 1 holds. Moreover it converges to a point  $(\mathbf{x}^*, \mathbf{y}^*) \in T_{\mathcal{L}}^{-1}(0)$  and*

$$0 = \lim_{k \rightarrow \infty} \|(I - \mathcal{P})(\mathbf{x}_k, \mathbf{y}_k)\| = \lim_{k \rightarrow \infty} \|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - (\mathbf{x}_k, \mathbf{y}_k)\|,$$

*see [28, Th. 1].*

2. *Suppose that*

$$\begin{aligned} \exists a > 0, \exists \delta > 0 : \forall \mathbf{w} \in B(0, \delta), \forall \mathbf{z} \in T_{\mathcal{L}}^{-1}(\mathbf{w}) \\ \text{we have } \text{dist}(\mathbf{z}, T_{\mathcal{L}}^{-1}(0)) &\leq a \|\mathbf{w}\|. \end{aligned} \quad (11)$$

*Then, the sequence  $\{(\mathbf{x}_k, \mathbf{y}_k)\}_{k \in \mathbb{N}}$  generated by the recursion in (8) using as inexactness criterion (10), is such that  $\text{dist}((\mathbf{x}_k, \mathbf{y}_k), T_{\mathcal{L}}^{-1}(0)) \rightarrow 0$  linearly. Moreover, the rate of convergence is bounded by  $a/(a^2 + (1/\max\{\rho, \delta\})^2)^{1/2}$ , i.e.,*

$$\limsup_{k \rightarrow \infty} \frac{\text{dist}((\mathbf{x}_{k+1}, \mathbf{y}_{k+1}), T_{\mathcal{L}}^{-1}(0))}{\text{dist}((\mathbf{x}_k, \mathbf{y}_k), T_{\mathcal{L}}^{-1}(0))} \leq \frac{a}{(a^2 + (1/\max\{\rho, \delta\})^2)^{1/2}} < 1, \quad (12)$$

*see [20, Th. 2.1].*

**Remark 2.1** Since the operators  $T_{\mathcal{L}}$  and  $T_{\mathcal{L}}^{-1}$  are polyhedral variational inequalities, they satisfy condition (11), see [9, Sec. 3.4].



### 2.3 Practical Stopping Criteria for Inexactness

As stated in Item 2. of Theorem 2.1, in order to guarantee linear convergence, it is sufficient to impose algorithmically the condition in (10). In particular, using (7) or (9) and the fact that, in general, it holds

$$v \in N_D(x) \Leftrightarrow \Pi_D(x + v) = x,$$

see [9, Sec. 2.1], we can define the following *natural residuals* (used also in [7, 19]):

$$r(x, y) := \begin{bmatrix} x \\ y \end{bmatrix} - \Pi_D \left( \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} Hx + g - A^T y \\ Ax - b \end{bmatrix} \right), \tag{13}$$

and

$$r_k(x, y) := \begin{bmatrix} x \\ y \end{bmatrix} - \Pi_D \left( \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} Hx + g - A^T y + \rho(x - x_k) \\ Ax - b + \delta(y - y_k) \end{bmatrix} \right). \tag{14}$$

The following Lemma 2.2 summarizes some important properties of the natural residuals (13) and (14).

**Lemma 2.2** *The natural residuals satisfy the following properties:*

1.  $dist(0, T_{\mathcal{L}}^{-1}(x, y)) = O(\|r(x, y)\|)$ ;
2. *There exists a constant  $\tau_1 > 0$  s.t.*

$$\|(x, y) - \mathcal{P}(x_k, y_k)\| \leq \tau_1 \|r_k(x, y)\| \text{ for all } (x, y) \in \mathbb{R}^{d+m}.$$

**Proof** Part 1. follows using [24, Th. 18]. Part 2. follows using an analogous reasoning as that in the proof [19, Prop. 2, Item 3.] which, in turn, is based on [24, Th. 20] and the fact that  $D$  is convex. □

In Algorithm 1 we present the particular form of the inexact PPM considered in this work.

**Input:**  $tol > 0, \sigma_r \in (0, 1)$ .

**Initialization:** Iteration counter  $k = 0$ ; initial points  $x_0, y_0$

```

1 while  $\|r(x_k, y_k)\| > tol$  do
2   Find  $(x_{k+1}, y_{k+1})$  s.t.  $\|r_k(x_{k+1}, y_{k+1})\| < \frac{\sigma_r^k}{\tau_1} \min\{1, \|(x_{k+1}, y_{k+1}) - (x_k, y_k)\|\}$ 
3   Update the iteration counter:  $k := k + 1$ .
4 end
    
```

**Algorithm 1:** Inexact PPM for QP

### 3 Primal-Dual IPM for Proximal Point Evaluations

For problem (RP) let us introduce the Lagrangian

$$\begin{aligned} \mathcal{L}_k(\mathbf{x}, \mathbf{y}, \mathbf{s}) = & \frac{1}{2}[\mathbf{x}^T, \mathbf{y}^T] \begin{bmatrix} H + \rho I & 0 \\ 0 & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \\ & [\mathbf{g}^T - \rho \mathbf{x}_k^T, 0] \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} - \mathbf{y}^T (A\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b}) - \mathbf{s}^T \mathbf{x}_C, \end{aligned} \quad (\text{RL})$$

where  $\mathbf{s} \in \mathbb{R}^{|\mathcal{C}|}$  and  $\mathbf{s} \geq 0$ . Using (RL), we write the KKT conditions

$$\begin{aligned} \begin{bmatrix} H + \rho I & 0 \\ 0 & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + \begin{bmatrix} \mathbf{g} - \rho \mathbf{x}_k \\ 0 \end{bmatrix} - \begin{bmatrix} A^T \mathbf{y} \\ \delta \mathbf{y} + (A\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b}) \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{s} \\ 0 \end{bmatrix} = 0; \\ SX_C \mathbf{e} = 0; \\ \mathbf{x}_C \geq 0. \end{aligned}$$

We can then write the dual form of problem (RP) as

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{R}^d, \mathbf{y} \in \mathbb{R}^m, \mathbf{s} \in \mathbb{R}^{|\mathcal{C}|}} & \mathbf{y}^T \mathbf{b} - \frac{1}{2} \mathbf{x}^T H \mathbf{x} - \frac{\rho}{2} \|\mathbf{x}\|^2 - \frac{\delta}{2} \|\mathbf{y} - \mathbf{y}_k\|^2 \\ \text{s.t. } & (H + \rho I)\mathbf{x} + (\mathbf{g} - \rho \mathbf{x}_k) - A^T \mathbf{y} - \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} = 0 \\ & \mathbf{s} \geq 0, \end{aligned} \quad (\text{RD})$$

where we used the fact that  $A\mathbf{x} + \delta\mathbf{y} = \mathbf{b} + \delta\mathbf{y}_k$ .

In this work we consider an infeasible primal dual IPM for the solution of the problem (RP), see Algorithm 2. In particular, this is obtained considering the following Regularized Lagrangian Barrier function

$$\begin{aligned} \mathcal{L}_k(\mathbf{x}, \mathbf{y}) = & \frac{1}{2}[\mathbf{x}^T, \mathbf{y}^T] \begin{bmatrix} H + \rho I & 0 \\ 0 & \delta I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} + [\mathbf{g}^T - \rho \mathbf{x}_k^T, 0] \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \\ & - \mathbf{y}^T (A\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b}) - \mu \sum_{i \in \mathcal{C}} \ln(x_i) \end{aligned}$$

We write the corresponding KKT conditions

$$\begin{aligned} \nabla_{\mathbf{x}} \mathcal{L}_k(\mathbf{x}, \mathbf{y}) = & (H + \rho I)\mathbf{x} - A^T \mathbf{y} + \mathbf{g} - \rho \mathbf{x}_k - \begin{bmatrix} 0 \\ \frac{\mu}{x_{d+1}} \\ \vdots \\ \frac{\mu}{x_d} \end{bmatrix} = 0; \\ -\nabla_{\mathbf{y}} \mathcal{L}_k(\mathbf{x}, \mathbf{y}) = & (A\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b}) = 0. \end{aligned}$$

Setting  $s_i = \frac{\mu}{x_i}$  for  $i \in \mathcal{C}$ , we can then consider the following IPM map

$$F_k^{\mu,\sigma}(\mathbf{x}, \mathbf{y}, \mathbf{s}) := \begin{bmatrix} (H + \rho I)\mathbf{x} - A^T\mathbf{y} + \mathbf{g} - \rho\mathbf{x}_k - \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} \\ \mathbf{A}\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b} \\ SX_{\mathcal{C}}\mathbf{e} - \sigma\mu\mathbf{e} \end{bmatrix}, \tag{15}$$

where  $\sigma \in (0, 1)$  is the barrier reduction parameter. A primal-dual Interior-Point Method applied to the problems (RP)-(RD), is based on applying Newton iterations to solve a nonlinear problem of the form

$$F_k^{\mu,\sigma}(\mathbf{x}, \mathbf{y}, \mathbf{s}) = 0, \quad \mathbf{x}_{\mathcal{C}} > 0, \quad \mathbf{s} > 0.$$

A Newton step for (15) from the current iterate  $(\mathbf{x}, \mathbf{y}, \mathbf{s})$  is obtained by solving the system

$$\begin{bmatrix} H + \rho I & -A^T & \begin{bmatrix} 0 \\ -I \end{bmatrix} \\ A & \delta I & 0 \\ [0 \ S] & 0 & X_{\mathcal{C}} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{y} \\ \Delta\mathbf{s} \end{bmatrix} = -F_k^{\mu,\sigma}(\mathbf{x}, \mathbf{y}, \mathbf{s}) =: \begin{bmatrix} \xi_d \\ \xi_p \\ \xi_{\mu,\sigma} \end{bmatrix}.$$

Eliminating the variable  $\Delta\mathbf{s}$  we obtain the linear system

$$\underbrace{\begin{bmatrix} H + \rho I + \Theta^\dagger & -A^T \\ A & \delta I \end{bmatrix}}_{\mathcal{N}_{\rho,\delta,\Theta}} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\mathbf{y} \end{bmatrix} = \begin{bmatrix} \xi_d^1 \\ \xi_d^2 + X_{\mathcal{C}}^{-1}\xi_{\mu,\sigma} \\ \xi_p \end{bmatrix}, \tag{16}$$

where  $\Theta^\dagger = \text{diag}((0, \dots, 0); \mathbf{x}_{\mathcal{C}}^{-1}\mathbf{s})$ ,  $\xi_d^1 := [(\xi_d)_1, \dots, (\xi_d)_{\bar{d}}]^T$  and  $\xi_d^2 := [(\xi_d)_{\bar{d}+1}, \dots, (\xi_d)_d]^T$ .

In Algorithm 2 we report the IPM scheme for problem (RP). The method has a guaranteed polynomial convergence [31, Chap. 6] (cfr. also [1, 11, 12, 18]). For notational simplicity we consider the case  $\mathcal{C} = \{1, \dots, d\}$ . To this aim, we also define

$$\begin{aligned} \mathcal{N}_k(\bar{\gamma}, \underline{\gamma}, \gamma_p, \gamma_d) &:= \{(\mathbf{x}, \mathbf{y}, \mathbf{s}) : \bar{\gamma}\mathbf{x}^T\mathbf{s} \geq x_i s_i \geq \underline{\gamma}\mathbf{x}^T\mathbf{s} \text{ for } i = 1, \dots, d; \\ \gamma_p\mathbf{x}^T\mathbf{s} &\geq \|\mathbf{A}\mathbf{x} + \delta(\mathbf{y} - \mathbf{y}_k) - \mathbf{b}\|; \\ \gamma_d\mathbf{x}^T\mathbf{s} &\geq \|\mathbf{H}\mathbf{x} + \rho(\mathbf{x} - \mathbf{x}_k) - A^T\mathbf{y} - \mathbf{s}\| \} \text{ and} \\ \begin{bmatrix} \mathbf{x}_k^j(\alpha) \\ \mathbf{y}_k^j(\alpha) \\ \mathbf{s}_k^j(\alpha) \end{bmatrix} &:= \begin{bmatrix} \mathbf{x}_k^j \\ \mathbf{y}_k^j \\ \mathbf{s}_k^j \end{bmatrix} + \begin{bmatrix} \alpha\Delta\mathbf{x}_k^j \\ \alpha\Delta\mathbf{y}_k^j \\ \alpha\Delta\mathbf{s}_k^j \end{bmatrix}. \end{aligned}$$

**Input:**  $\sigma, \bar{\sigma} \in (0, 1)$  barrier reduction parameters s.t.  $\sigma < \bar{\sigma}$ ;

$\varepsilon_{p,k} > 0, \varepsilon_{d,k} > 0, \varepsilon_{c,k} > 0$  optimality tolerances;

**Initialization:**

Iteration counter  $j = 0$ ; primal-dual point  $\mathbf{x}_k^0 > 0, \mathbf{y}_k^0 > 0, \mathbf{s}_k^0 > 0$ ;

Compute  $\mu_k^0 := \mathbf{x}_k^{0T} \mathbf{s}_k^0 / d$  and  $\xi_{d,k}^0$ , and  $\xi_{p,k}^0$ .

1 **while** *Stopping Criterion False* **do**

2     Solve the KKT system (16) using  $[\xi_{p,k}^j, \xi_{d,k}^j, \xi_{\mu_k, \sigma}^j]^T$  to find  $[\Delta \mathbf{x}_k^j, \Delta \mathbf{y}_k^j, \Delta \mathbf{s}_k^j]^T$ ;

3     Find  $\alpha_k^j$  as the maximum for  $\alpha \in [0, 1]$  s.t.

$$(\mathbf{x}_k^j(\alpha), \mathbf{y}_k^j(\alpha), \mathbf{s}_k^j(\alpha)) \in \mathcal{N}_k(\bar{\gamma}, \underline{\gamma}, \gamma_p, \gamma_d)$$

and

$$\mathbf{x}_k^j(\alpha)^T \mathbf{s}_k^j(\alpha) \leq (1 - (1 - \bar{\sigma})\alpha) \mathbf{x}_k^{jT} \mathbf{s}_k^j;$$

4     Set  $\begin{bmatrix} \mathbf{x}_k^{j+1} \\ \mathbf{y}_k^{j+1} \\ \mathbf{s}_k^{j+1} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_k^j \\ \mathbf{y}_k^j \\ \mathbf{s}_k^j \end{bmatrix} + \begin{bmatrix} \alpha_k^j \Delta \mathbf{x}_k^j \\ \alpha_k^j \Delta \mathbf{y}_k^j \\ \alpha_k^j \Delta \mathbf{s}_k^j \end{bmatrix}$ ;

5     Compute the infeasibilities  $\xi_{d,k}^{j+1}, \xi_{p,k}^{j+1}$  and barrier parameter  $\mu_k^{j+1} := \mathbf{x}_k^{j+1T} \mathbf{s}_k^{j+1} / d$ ;

6     Update the iteration counter:  $j := j + 1$ .

7 **end**

**Algorithm 2:** Infeasible IPM for problem (RP)

### 3.1 The Proximal Stabilized-Interior Point Algorithm (PS-IPM)

In Algorithm 3 we present our proposal in full detail.

**Input:**  $tol > 0, \sigma_r \in (0, 1)$ .

**Initialization:** Iteration counter  $k = 0$ ; initial point  $(\mathbf{x}_0, \mathbf{y}_0)$

1 **while**  $\|r(\mathbf{x}_k, \mathbf{y}_k)\| > tol$  **do**

2     Use Algorithm 2 with starting point  $(\mathbf{x}_k^0, \mathbf{y}_k^0) = (\mathbf{x}_k, \mathbf{y}_k)$  to find  $(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$  s.t.

$$\|r_k(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})\| < \frac{\sigma_r^k}{\tau_1} \min\{1, \|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - (\mathbf{x}_k, \mathbf{y}_k)\|\} \tag{17}$$

3     Update the iteration counter:  $k := k + 1$ .

4 **end**

**Algorithm 3:** PS-IPM for QP

Two comments are in order at this stage.

1. It is important to observe that the *warm starting strategy* of starting Algorithm 2 from the previous PPM approximation  $(\mathbf{x}_k, \mathbf{y}_k)$  is justified by the fact that

$$\begin{aligned} & \| \mathcal{P}(\mathbf{x}_k, \mathbf{y}_k) - (\mathbf{x}_k, \mathbf{y}_k) \| \\ & \leq \| \mathcal{P}(\mathbf{x}_k, \mathbf{y}_k) - \mathcal{P}(\mathbf{x}_{k-1}, \mathbf{y}_{k-1}) \| + \| \mathcal{P}(\mathbf{x}_{k-1}, \mathbf{y}_{k-1}) - (\mathbf{x}_k, \mathbf{y}_k) \| \tag{18} \\ & \leq \eta \| (\mathbf{x}_k, \mathbf{y}_k) - (\mathbf{x}_{k-1}, \mathbf{y}_{k-1}) \| + \| \mathcal{P}(\mathbf{x}_{k-1}, \mathbf{y}_{k-1}) - (\mathbf{x}_k, \mathbf{y}_k) \|, \end{aligned}$$

where, in the second inequality,  $\eta$  is the Lipschitz constant of the proximal operator (see [19, Theorem 4]). Since the inexact PPM converges we have that

$$\|\mathcal{P}(\mathbf{x}_{k-1}, \mathbf{y}_{k-1}) - (\mathbf{x}_k, \mathbf{y}_k)\| \rightarrow 0 \text{ and } \|(\mathbf{x}_k, \mathbf{y}_k) - (\mathbf{x}_{k-1}, \mathbf{x}_{k-1})\| \rightarrow 0,$$

and hence  $\|\mathcal{P}(\mathbf{x}_k, \mathbf{y}_k) - (\mathbf{x}_k, \mathbf{y}_k)\| \rightarrow 0$ . Such observation suggests that, eventually,  $\|\mathcal{P}(\mathbf{x}_k, \mathbf{y}_k) - (\mathbf{x}_k, \mathbf{y}_k)\|$  will become sufficiently small so that the *fast final convergence of IPM* holds immediately and the IPM converges fast. As a result, we expect that each proximal subproblem will need a non-increasing number of IPM iterations to be solved. Indeed, we observe this behaviour in practice: typically after the first or second proximal iteration each subsequent proximal subproblem takes only two or one IPM iterations to converge (see Sect. 5.2).

2. The IPM Algorithm 2 uses (17) as a stopping condition.

### 4 Slack Formulation and Preconditioning

The presence of proximal point regularization brings several advantages to the IPM. One of them is bounding the spectrum of the matrices in Newton system [2, 3, 14, 23]. In this section, we will show how this may be combined with the replications of the variables involved in the inequality constraints to alleviate the inherent numerical instability which originates from the IPM scaling matrix  $\Theta$ .

For the sake of simplicity, in this section, we assume that in (1) all variables have nonnegative constraints, i.e.,  $\bar{d} = 0$ . Then, using variables replication, we get the following slack formulation [11, Sec. 6] of the original problem (1):

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^{d_1}} \quad & \frac{1}{2} \mathbf{x}^T H \mathbf{x} + \mathbf{g}^T \mathbf{x} \\ \text{s.t.} \quad & A \mathbf{x} = \mathbf{b}, \mathbf{x} - \mathbf{z} = 0 \\ & \mathbf{z} \geq 0. \end{aligned} \tag{19}$$

In this case the IPM map (15) can be written as

$$F_k^{\mu, \sigma}(\mathbf{x}, \mathbf{z}, \mathbf{y}_1, \mathbf{y}_2, \mathbf{s}) := \begin{bmatrix} \begin{bmatrix} H + \rho I & 0 \\ 0 & \rho I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} + \begin{bmatrix} \mathbf{g} - \rho \mathbf{x}_k \\ -\rho \mathbf{z}_k \end{bmatrix} - \begin{bmatrix} A^T & I \\ 0 & -I \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{s} \end{bmatrix} \\ \begin{bmatrix} A & 0 \\ I & -I \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \end{bmatrix} + \delta \begin{bmatrix} \mathbf{y}_1^1 - \mathbf{y}_k^1 \\ \mathbf{y}_2^2 - \mathbf{y}_k^2 \end{bmatrix} - \begin{bmatrix} \mathbf{b} \\ 0 \end{bmatrix} \\ S \mathbf{z} \mathbf{e} - \sigma \mu \mathbf{e} \end{bmatrix}. \tag{20}$$

Using (20), the corresponding Newton system (see also Eq. (16)), can be expressed as

$$\begin{bmatrix} H + \rho I & 0 & -A^T & -I \\ 0 & \Theta^{-1} + \rho I & 0 & I \\ A & 0 & \delta I & 0 \\ I & -I & 0 & \delta I \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{z} \\ \Delta \mathbf{y}_1 \\ \Delta \mathbf{y}_2 \end{bmatrix} = \begin{bmatrix} \xi_d^1 \\ \xi_d^2 + Z^{-1} \xi_{\mu, \sigma} \\ \xi_p^1 \\ \xi_p^2 \end{bmatrix}, \tag{21}$$

where  $\Theta = ZS^{-1}$ . For the convenience of the reader we also report, in the following, the explicit expressions of the IPM residuals: the natural PPM residual in (13) reads as

$$r(\mathbf{x}, \mathbf{z}, \mathbf{y}) := \begin{bmatrix} H\mathbf{x} + \mathbf{g} - [A^T \ I]\mathbf{y} \\ \mathbf{z} - \Pi_{\mathbb{R}_{\geq 0}}(\mathbf{z} - (-[0 \ -I]\mathbf{y})) \\ A\mathbf{x} - \mathbf{b} \\ \mathbf{x} - \mathbf{z} \end{bmatrix},$$

whereas the residual in (14) becomes

$$r_k(\mathbf{x}, \mathbf{z}, \mathbf{y}) := \begin{bmatrix} H\mathbf{x} + \mathbf{g} - [A^T \ I]\mathbf{y} + \rho(\mathbf{x} - \mathbf{x}_k) \\ \mathbf{z} - \Pi_{\mathbb{R}_{\geq 0}}(\mathbf{z} - (\rho(\mathbf{z} - \mathbf{z}_k) - [0 \ -I]\mathbf{y})) \\ A\mathbf{x} - \mathbf{b} + \delta(\mathbf{y}^1 - \mathbf{y}_k^1) \\ \mathbf{x} - \mathbf{z} + \delta(\mathbf{y}^2 - \mathbf{y}_k^2) \end{bmatrix}.$$

### 4.1 Solution of the Newton System

In this section we will study in details the solution of the linear system (21) when reordered and symmetrized as follows:

$$\underbrace{\begin{bmatrix} \Theta^{-1} + \rho I & -I & 0 & 0 \\ -I & -\delta I & I & 0 \\ 0 & I & H + \rho I & A^T \\ 0 & 0 & A & -\delta I \end{bmatrix}}_{=: \mathcal{N}(\Theta)} \begin{bmatrix} \Delta \mathbf{z} \\ -\Delta \mathbf{y}_2 \\ \Delta \mathbf{x} \\ -\Delta \mathbf{y}_1 \end{bmatrix} = \begin{bmatrix} \xi_d^2 + Z^{-1} \xi_{\mu, \sigma} \\ \xi_p^2 \\ \xi_p^1 \\ \xi_d^1 \end{bmatrix}. \tag{22}$$

To this aim, let us partition the matrix  $\mathcal{N}(\Theta)$  as

$$\mathcal{N}(\Theta) = \begin{bmatrix} N_{11}(\Theta) & N_{12} \\ N_{21} & N_{22} \end{bmatrix},$$

where

$$N_{11}(\Theta) := \begin{bmatrix} \Theta^{-1} + \rho I & -I \\ -I & -\delta I \end{bmatrix}, \quad N_{12} := \begin{bmatrix} 0 & 0 \\ I & 0 \end{bmatrix}, \quad N_{21} = N_{12}^T, \\ N_{22} := \begin{bmatrix} H + \rho I & A^T \\ A & -\delta I \end{bmatrix}.$$

Before continuing, let us observe that, under suitable hypothesis, the solution of a linear system of the form

$$\begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_x \\ \mathbf{b}_y \end{bmatrix},$$

can be obtained solving

$$\begin{aligned} (G_{22} - G_{21}G_{11}^{-1}G_{12})\mathbf{y} &= \mathbf{b}_y - G_{21}G_{11}^{-1}\mathbf{b}_x, \\ \mathbf{x} &= G_{11}^{-1}(\mathbf{b}_x - G_{12}\mathbf{y}). \end{aligned} \tag{23}$$

Since the linear systems involving  $N_{11}(\Theta)$  are easily solvable, using (23), the overall solution of the linear system (22) can be obtained from the following two ancillary ones:

$$\begin{aligned} S(\Theta) \begin{bmatrix} \Delta \mathbf{x} \\ -\Delta \mathbf{y}_1 \end{bmatrix} &= \begin{bmatrix} \xi_d^1 \\ \xi_p^1 \end{bmatrix} - N_{21}N_{11}(\Theta)^{-1} \begin{bmatrix} \xi_d^2 + Z^{-1}\xi_{\mu,\sigma} \\ \xi_p^2 \end{bmatrix} \\ N_{11}(\Theta) \begin{bmatrix} \Delta \mathbf{z} \\ -\Delta \mathbf{y}_2 \end{bmatrix} &= \begin{bmatrix} \xi_d^2 + Z^{-1}\xi_{\mu,\sigma} \\ \xi_p^2 \end{bmatrix} - N_{12} \begin{bmatrix} \Delta \mathbf{x} \\ -\Delta \mathbf{y}_1 \end{bmatrix}, \end{aligned} \tag{24}$$

where  $S(\Theta)$  is the Schur complement

$$S(\Theta) := \begin{bmatrix} H + \rho I + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} & A^T \\ A & -\delta I \end{bmatrix}. \tag{25}$$

It is important to observe that, at this stage, the reasons to go through the current reformulation of the problem are not completely apparent: we essentially doubled the dimension of the primal variables ending up with the necessity of solving linear systems involving a Schur complement, see equation (25), which has exactly the same sparsity pattern as the standard (symmetrized) Newton system

$$\mathcal{N}_C(\Theta) := \begin{bmatrix} H + \rho I + \Theta^{-1} & A^T \\ A & -\delta I \end{bmatrix}, \tag{26}$$

cfr. Equation (16).

In the following Remarks 4.1 and 4.2 we highlight the advantages given by the current reformulation of the Newton system showing, in essence, that the formulation in (25) allows *better preconditioner re-usage* than in the standard formulation (26). To this aim, as it is customary in IPM methods, let us suppose that

$$\max_i \lambda_i(\Theta^{-1}) = O\left(\frac{1}{\mu}\right) \text{ and } \min_i \lambda_i(\Theta^{-1}) = O(\mu),$$

where  $\mu$  is the average complementarity product at any given IPM iteration. Using the above assumption, we obtain

$$\begin{aligned} \lim_{\mu \rightarrow 0} \min_i \lambda_i (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} &= \frac{\rho}{\delta\rho + 1} \\ \lim_{\mu \rightarrow 0} \max_i \lambda_i (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} &= \frac{1}{\delta}. \end{aligned} \quad (27)$$

From Eq. (27) the main advantage of dealing with the formulation (25) of the Schur complement becomes more apparent: whilst the elements of the diagonal IPM matrix appearing in  $\mathcal{N}_C(\Theta)$  are such that  $\Theta_{ii}^{-1} \in (0, +\infty)$  when  $\mu \rightarrow 0$ , the diagonal elements appearing in the Schur complement (25) belong to the interval  $(\frac{\rho}{\delta\rho+1}, \frac{1}{\delta})$  when  $\mu \rightarrow 0$ .

**Remark 4.1** When  $\mu \rightarrow 0$ , for the variables which have been identified as active or inactive by the PS-IPM, see Algorithm 3, we have that

$$(\delta I + (\Theta^{-1} + \rho I)^{-1})_{ii}^{-1} \approx \frac{1}{\delta} \quad \text{or} \quad (\delta I + (\Theta^{-1} + \rho I)^{-1})_{ii}^{-1} \approx \frac{\rho}{\delta\rho + 1},$$

and such values are expected to remain unchanged in the following PS-IPM steps. This suggests that, when close enough to convergence, any computed approximation of the matrix  $S(\Theta)$  may be used as an effective preconditioner also for subsequent PS-IPM steps.

In Lemma 4.1 we show that the regularization parameters  $(\rho, \delta)$  act as *attenuation coefficients*, see (29), for the variations  $\widehat{\Theta}_{ii}^{-1} - \Theta_{ii}^{-1}$  where  $\widehat{\Theta}^{-1}$  and  $\Theta^{-1}$  are two IPM matrices obtained, respectively, in two different IPM iterations.

**Lemma 4.1** *Define*

$$D_A := (\delta I + (\widehat{\Theta}^{-1} + \rho I)^{-1})^{-1} - (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1}$$

and

$$D_C := \widehat{\Theta}^{-1} - \Theta^{-1}.$$

Then,

$$\|S(\widehat{\Theta}) - S(\Theta)\|_2 = \|D_A\|_2 < \|D_C\|_2 = \|\mathcal{N}_C(\widehat{\Theta}) - \mathcal{N}_C(\Theta)\|_2. \quad (28)$$

**Proof** From direct computation we have

$$(D_A)_{ii} = \frac{\widehat{\Theta}_{ii}^{-1} - \Theta_{ii}^{-1}}{1 + \delta^2(\Theta_{ii}^{-1} + \rho)(\widehat{\Theta}_{ii}^{-1} + \rho) + \delta(\Theta_{ii}^{-1} + \rho) + \delta(\widehat{\Theta}_{ii}^{-1} + \rho)}. \quad (29)$$



Then,

$$|(D_A)_{ii}| < |(D_C)_{ii}|,$$

and the thesis follows using the definitions of  $S(\Theta)$  and  $\mathcal{N}_C(\Theta)$ . □

**Remark 4.2** Suppose we computed a preconditioner for  $S(\Theta)$ , e.g., an incomplete factorization. Equation (28) shows that any accurate preconditioner for  $S(\Theta)$  approximates  $S(\widehat{\Theta})$  better than an analogous preconditioner for  $\mathcal{N}_C(\Theta)$  would approximate  $\mathcal{N}_C(\widehat{\Theta})$ .

Moreover, from (29), we can observe that the variations  $(D_A)_{ii}$ , and hence the variation  $\|S(\widehat{\Theta}) - S(\Theta)\|_2$  of the overall Schur complement, are negatively correlated with the regularization parameters  $(\rho, \delta)$ . Then, it has to be expected that the computed preconditioner for  $S(\Theta)$  would be yet an effective preconditioner for the matrix  $S(\widehat{\Theta})$  if the regularization parameters  $(\rho, \delta)$  are sufficiently large.

On the other hand, according to (12), the rate of convergence of PPM correlates inversely with the regularization parameters  $(\rho, \delta)$ .

As a result of the above discussion, we are able to unveil a *precise interaction* between the computational footprint related to the necessity of re-computing preconditioners and the rate of convergence of the PS-IPM with the obvious benefit to allow a predictable tuning of such trade-off (see Sect. 5.3).

Finally, since  $\lim_{(\rho, \delta) \rightarrow (0, 0)} (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} = \Theta^{-1}$ , we have that

$$\lim_{(\rho, \delta) \rightarrow (0, 0)} S(\Theta) = \mathcal{N}_C(\Theta),$$

indicating the fact that the re-usability of preconditioners for  $\mathcal{N}_C(\Theta)$  can be assessed as a limit case of the re-usability of preconditioners for  $S(\Theta)$ .

To conclude this section, in Theorem 4.1, we analyse in more detail the eigenvalues of the matrix  $S(\Theta)^{-1}S(\widehat{\Theta})$ . Indeed, supposing we have computed an accurate preconditioner for  $S(\Theta)$ , then, the eigenvalues of  $S(\Theta)^{-1}S(\widehat{\Theta})$  may be considered as a measure of the effectiveness of such preconditioner when used as preconditioner for  $S(\widehat{\Theta})$ : the results there contained will further confirm that a high quality preconditioner is expected when the matrix  $D_A$  has small diagonal elements, see (30). In this case, indeed  $S(\Theta)^{-1}S(\widehat{\Theta})$  has a highly clustered spectrum.

**Theorem 4.1** *Let us define*

$$H_{A, \Theta, \delta, \rho} := H + \rho I + \frac{1}{\delta} A^T A + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1}.$$

*Then, the matrix  $S(\Theta)^{-1}S(\widehat{\Theta})$  has the eigenvalue  $\eta = 1$  with multiplicity at least  $m$  whereas, the other eigenvalues, are s.t.*

$$\eta \in \left( 1 + \frac{\min_i \lambda_i(D_A)}{\max_i \lambda_i(H_{A, \Theta, \delta, \rho})}, 1 + \frac{\max_i \lambda_i(D_A)}{\min_i \lambda_i(H_{A, \Theta, \delta, \rho})} \right). \tag{30}$$

**Proof** To analyse the eigenvalues we consider the problem

$$S(\widehat{\Theta})\mathbf{u} = \eta S(\Theta)\mathbf{u},$$

i.e.,

$$\begin{aligned} & \begin{bmatrix} H + \rho I + (\delta I + (\widehat{\Theta}^{-1} + \rho I)^{-1})^{-1} & A^T \\ A & -\delta I \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} = \\ & \eta \begin{bmatrix} H + \rho I + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} & A^T \\ A & -\delta I \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix}. \end{aligned} \quad (31)$$

If  $\eta = 1$ , we obtain that any vector of the form  $[0, \mathbf{u}_2]$  is a solution of (31) and hence the multiplicity of eigenvalue 1 is at least  $m$ . Let us suppose  $\eta \neq 1$ . Always from (31) we obtain  $A\mathbf{u}_1 = \delta\mathbf{u}_2$  and hence, using the equality

$$\begin{aligned} & (H + \rho I + (\delta I + (\widehat{\Theta}^{-1} + \rho I)^{-1})^{-1} + \frac{1}{\delta}A^T A)\mathbf{u}_1 \\ & = \eta(H + \rho I + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1} + \frac{1}{\delta}A^T A)\mathbf{u}_1, \end{aligned}$$

we obtain

$$\eta = 1 + \frac{\mathbf{u}_1^T((\delta I + (\widehat{\Theta}^{-1} + \rho I)^{-1})^{-1} - (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1})\mathbf{u}_1}{\mathbf{u}_1^T((H + \rho I + \frac{1}{\delta}A^T A + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1})\mathbf{u}_1)}. \quad (32)$$

This follows from (32) using the definition of  $D_A$ ,  $H_{A,\Theta,\delta,\rho}$  and the fact that both are symmetric matrices.  $\square$

#### 4.1.1 Further Schur Complement Reduction

In some particular cases, using once more (23), it might be computationally advantageous to further reduce the solution of the linear system in (24) to a smaller linear system involving its Schur complement. Among other situations, this is the case of IPM matrices coming from problems of the form (19) for which  $H$  is diagonal, e.g.,  $H = 0$  (LP case). In this section we prove a similar result to Lemma 4.1 when the involved matrices are the Schur complements of the linear systems (25) and (26). To this aim and for the sake of simplicity, we consider the case  $H = 0$  and define  $L_1(\Theta)$  as the Schur complement of (25), i.e.,

$$L_1(\Theta) := -(\delta I + A(\rho I + (\delta I + (\Theta^{-1} + \rho I)^{-1})^{-1})^{-1}A^T), \quad (33)$$

whereas we define  $L_2(\Theta)$  as the Schur complement of (26), i.e.,

$$L_2(\Theta) := -(\delta I + A(\rho I + \Theta^{-1})^{-1}A^T).$$

Moreover, considering diagonal scaling matrices  $\widehat{\Theta}^{-1}$  and  $\Theta^{-1}$  obtained, respectively in two different IPM iterations, we have

$$L_1(\widehat{\Theta}) - L_1(\Theta) = A[\underbrace{(\rho I + (\delta I + (\rho I + \Theta^{-1})^{-1})^{-1})^{-1} - (\rho I + (\delta I + (\rho I + \widehat{\Theta}^{-1})^{-1})^{-1})^{-1}}_{=: \Delta_{1, \widehat{\Theta}, \Theta}}]A^T$$

whereas

$$L_2(\widehat{\Theta}) - L_2(\Theta) = A[\underbrace{(\rho I + \Theta^{-1})^{-1} - (\rho I + \widehat{\Theta}^{-1})^{-1}}_{=: \Delta_{2, \widehat{\Theta}, \Theta}}]A^T.$$

We are ready to state Lemma 4.2, which guarantees that also when operating a further reduction to the Schur complement in (25), the regularization parameters  $(\rho, \delta)$  act as *damping factors* for the changes in the diagonal matrix  $|\Delta_{2, \widehat{\Theta}, \Theta}|$ .

**Lemma 4.2** *With the notation introduced above, we have*

$$|(\Delta_{1, \widehat{\Theta}, \Theta})_{ii}| < |(\Delta_{2, \widehat{\Theta}, \Theta})_{ii}| \tag{34}$$

and

$$\|A|\Delta_{1, \widehat{\Theta}, \Theta}|A^T\|_2 < \|A|\Delta_{2, \widehat{\Theta}, \Theta}|A^T\|_2.$$

**Proof** From direct computation, we have that

$$(\Delta_{1, \widehat{\Theta}, \Theta})_{ii} = \frac{\widehat{\Theta}_{ii}^{-1} - \Theta_{ii}^{-1}}{(\rho(\delta + (\widehat{\Theta}^{-1} + \rho I)_{ii}^{-1}) + 1)(\rho(\delta + (\Theta^{-1} + \rho I)_{ii}^{-1}) + 1)(\widehat{\Theta}_{ii}^{-1} + \rho)(\Theta_{ii}^{-1} + \rho)},$$

from which, observing that  $(\Delta_{2, \widehat{\Theta}, \Theta})_{ii} = \frac{\widehat{\Theta}_{ii}^{-1} - \Theta_{ii}^{-1}}{(\widehat{\Theta}_{ii}^{-1} + \rho)(\Theta_{ii}^{-1} + \rho)}$ , (34) follows. The second part of the statement follows observing that

$$\begin{aligned} \|A|\Delta_{1, \widehat{\Theta}, \Theta}|A^T\|_2 &= \max_{\mathbf{x}} \frac{\mathbf{x}^T A|\Delta_{1, \widehat{\Theta}, \Theta}|A^T \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \\ &= \max_{\mathbf{x}} \frac{\mathbf{x}^T A|\Delta_{1, \widehat{\Theta}, \Theta}|A^T \mathbf{x}}{\mathbf{x}^T A A^T \mathbf{x}} \frac{\mathbf{x}^T A A^T \mathbf{x}}{\mathbf{x}^T \mathbf{x}} < \\ &= \max_{\mathbf{x}} \frac{\mathbf{x}^T A|\Delta_{2, \widehat{\Theta}, \Theta}|A^T \mathbf{x}}{\mathbf{x}^T A A^T \mathbf{x}} \frac{\mathbf{x}^T A A^T \mathbf{x}}{\mathbf{x}^T \mathbf{x}} = \|A|\Delta_{2, \widehat{\Theta}, \Theta}|A^T\|_2, \end{aligned}$$

where in the last inequality we used (34). □

## 5 Numerical Results

Aim of this section is to present the computational results obtained using Algorithm 3 (PS-IPM) when solving a set of small to large scale linear and convex quadratic problems. All the numerical results presented in this section are performed using a Dell PowerEdge R740 running Scientific Linux 7 with  $4 \times$  Intel Gold 6234 3.3 G, 8C/16T, 10.4GT/s, 24.75 M Cache, Turbo, HT (130W) DDR4-2933. Our implementation closely follows the one from [25] and is written in Matlab<sup>®</sup> R2022a (available at the website <https://github.com/StefanoCipolla/PS-IPM>). Concerning the choice of the initial guess, we use the same initial point as in [25, Sec. 5.1.3], which, in turn, is based on the developments in [22]. Concerning instead the choice of the parameters in Algorithm 3, we set  $\sigma_r = 0.7$ . Moreover, to prevent wasting time on finding excessively accurate solutions in the early PPM sub-problems, we replace (17) with

$$\|r_k(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})\| < 10^4 \sigma_r^k \min\{1, \|(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - (\mathbf{x}_k, \mathbf{y}_k)\|\}.$$

Indeed, in our computational experience, we have found that driving the IPM solver to a high accuracy in the initial PPM iterations is unnecessary and, usually, leads to a significant deterioration of the overall performance. Finally, we set as regularization parameters  $\delta = \rho$ , where

$$\rho = \max \left\{ \frac{\text{tol}}{\max\{\|A\|_\infty, \|H\|_\infty\}}, 10^{-10} \right\}, \quad (35)$$

see [25]. The remainder of this section is organized as follows.

In Sect. 5.1 we present the comparison of PS-IPM with Matlab<sup>®</sup>'s `linprog` ('Algorithm': 'interior-point') and `quadprog` ('Algorithm': 'interior-point-convex'). For the fairness of comparison and to be sure that all the compared algorithms are working on the same dataset, we use the option 'Preprocess': 'none' in `linprog` (the same option is not available for `quadprog`, which always performs a *Preprocess* phase). Moreover, both solvers use a multiple centrality corrections strategy to find the search direction, see [15]. For this reason and for the fairness of comparison, we use the same strategy in our PS-IPM implementation. This issue represents the main point where practical implementation deviates from the theory in order to gain computational efficiency. Concerning the stopping criteria, always for the fairness of the comparison, instead of using the natural residual (13), we stop the iterations of Algorithm 3 using the same stopping criteria of `linprog` & `quadprog`, i.e.,

$$\|\mathbf{g} - A^T \mathbf{y} + H \mathbf{x} - \mathbf{s}\|_\infty \leq R \cdot \text{tol} \wedge \|\mathbf{b} - A \mathbf{x}\|_1 \leq R \cdot \text{tol} \wedge c_{\mathbf{x}, \mathbf{s}} \leq \text{tol},$$

where

$$\text{tol} = 10^{-6}, \quad R := \max\{\|A\|, \|H\|, \|\mathbf{b}\|, \|\mathbf{c}\|\},$$

and

$$c_{\mathbf{x},\mathbf{s}} := \max_i \{\min\{|\langle \mathbf{x}_i, \mathbf{s}_i \rangle|, |\mathbf{x}_i|, |\mathbf{s}_i|\}\}.$$

In Sect. 5.2 we present the comparison of PS-IPM with IP-PMM [25]. In our PS-IPM implementation, analogously to [25], in order to find the search direction, we employ a widely used predictor–corrector method [22]. Also in this case, this issue represents the main point where practical implementation deviates from the theory in order to gain computational efficiency and is here used for the fairness of comparison with IP-PMM. Concerning the stopping criterion, always for the fairness of the comparison with IP-PMM, instead of using the natural residual (13), we stop the iterations of Algorithm 3 when

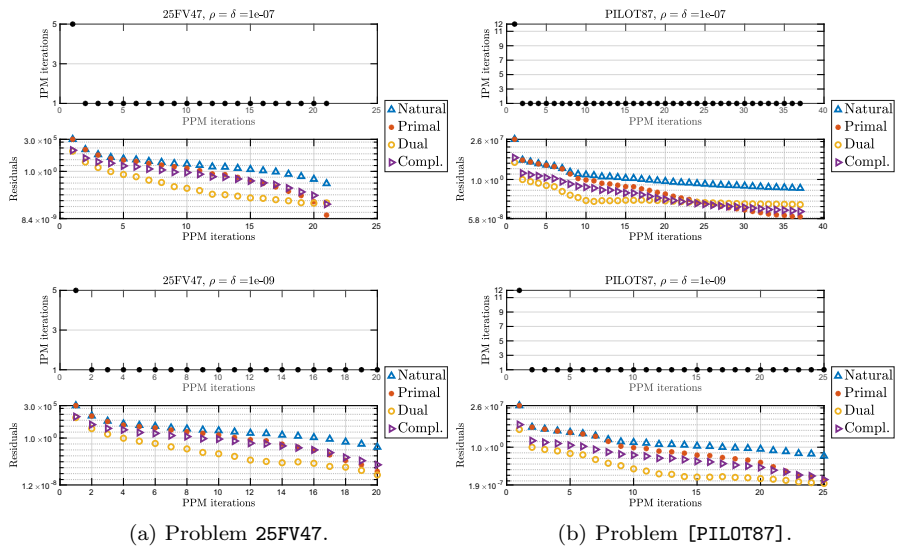
$$\frac{\|\mathbf{g} - A^T \mathbf{y} + H \mathbf{x} - \mathbf{s}\|}{\max\{\|\mathbf{g}\|, 1\}} \leq \text{tol} \wedge \frac{\|\mathbf{b} - A \mathbf{x}\|}{\max\{\|\mathbf{b}\|, 1\}} \leq \text{tol} \wedge \mu \leq \text{tol}, \quad (36)$$

and when  $\text{tol} = 10^{-6}$ .

As highlighted in the above discussion, the choice of presenting separate comparison is due essentially to the fact that `linprog` & `quadprog` use different stopping criteria and different strategies for generating the search directions w.r.t. IP-PMM and, for the sake of presenting trustworthy comparative results, we replicated such features in our PS-IPM implementation.

In both, Sects. 5.1 and 5.2, we solve the (symmetrized) Newton linear systems (16), using Matlab<sup>®</sup>'s `ldl` factorization. The factorization threshold parameter is set equal to the regularization parameter [see (35)] and is incremented by a factor 10 if numerical instabilities are detected in the obtained solution of the given linear system. Finally, it is important to note that the reported computational times in this work are just indicative of the relative performance rather than the absolute ones. Indeed, each call of the Matlab<sup>®</sup>'s `ldl` (which uses MA57 [10]) requires an *Analysis Phase* [10, Sec. 6.2] which, in an optimal implementation, can be performed just once in the pre-processing phase since the sparsity pattern of the Newton matrices does not change during the IPM iterations. Moreover, from the practical point of view, it is important to note that, if on the one hand choosing the threshold parameter equal to the regularization accelerates the `ldl` routine for the Newton systems arising in Algorithm 2, on the other, the presence of such regularization terms stabilizes it. For this reason, we expect for Algorithm 3 similar stability and robustness properties when compared to IP-PMM, which, in turn, shows improved robustness capabilities with respect to the non regularized version (see [11, 25]).

Finally, in Sect. 5.3, we present the numerical results relative to Sect. 4. In particular, we present here a series of numerical results aiming to showcase to which extent the theory developed in Sect. 4 would allow the reusability of a given factorization as preconditioner for Krylov iterative solvers [30]. To this aim, in Sect. 5.3.1, we consider the case when the linear system involving  $S(\Theta)$  in (24) is solved without further reduction to Schur complement, whereas, in Sect. 5.3.2, we present the numerical results when such further reduction is considered, see Sect. 4.1.1 for the theoretical details.



**Fig. 1** Theory showcase. Upper Panels: PPM Iterations and IPM Iterations. Lower Panels: Behaviour of residuals

For the sake of readability, we postpone the detailed description of the computational frameworks used to the particular sections. Moreover, we point out that in Sect. 5.3 we use the same stopping criterion, see (36), and the same predictor-corrector strategy as the one used in Sect. 5.2.

*Dataset* For the small/medium scale linear programming problems, the test set consists of 98 linear programming problems from the Netlib collection. For the small/medium scale convex quadratic problems, the test set consists of 122 problems from the Maros-Mészáros collection [21]. For the large scale linear programming problems, the test set consists of 19 problems coming from Netlib, Qaplib and Mittelmann’s collection.

For the sake of completeness, in Table 1, we report the dimensions and the number of non-zeros of the largest instances considered in this work [when the problems are formulated as in (1)].

Before showing the detailed comparison results, we start by briefly showcasing the theory developed in Sects. 2 and 3. In particular, in Fig. 1a, b, we report the details of the run of Algorithm 3 on the problems 25FV47 and PILOT87 from the Netlib collection. As the figures show, accordingly to (12) in Theorem 2.1, the rate of convergence of PPM decreases when the regularization parameter is increased (lower panels of Fig. 1a, b). Moreover, accordingly to (18), the number of IPM iterations needed to solve the PPM sub-problems is non-increasing when the PPM iterations proceed (somehow our choice of the parameters amplifies this feature since in the majority of PPM iterations just one IPM sweep is enough to meet the inexactness criterion, see upper panels in Fig. 1a, b).

**Table 1** Details of some of the largest and/or denser instances present in the test set

Problem	$n$	$m$	$nnz$ (A)	$nnz$ (H)	Ineq. constr.
<i>Netlib</i>					
25FV47	1876	821	10,705	0	1876
80BAU3B	15,545	5746	30,232	0	15,545
D6CUBE	6184	415	37,704	0	6184
FIT2D	21,024	10,525	150,042	0	21,024
FIT2P	21,025	10,500	65,284	0	21,025
PILOT87	8478	3828	78,545	0	8478
QAP15	22,275	6330	94,950	0	22,275
<i>Maros-Mészáros</i>					
CVXQP1	30,000	25,000	54,998	69,968	20,000
LISWET1	20,002	10,000	40,000	10,002	10,000
LISWET10	20,002	10,000	40,000	10,002	10,000
POWELL20	20,000	10,000	30,000	10,000	10,000
SHIP12L	5533	1151	16,276	122,433	5533
STCQP1	12,291	10,246	29,726	49,109	8194
<i>Large scale</i>					
Mittelmann/fome21	286,071	137,469	604,736	0	286,071
LPnetlib/lp_cre_b	77,137	9648	260,785	0	77,137
LPnetlib/lp_cre_d	73,948	8926	246,614	0	73,948
LPnetlib/lp_ken_18	309,398	259,826	667,569	0	309,398
Qaplib/lp_nug20	72,600	15,240	304,800	0	72,600
LPnetlib/lp_osa_30	104,374	4350	604,488	0	104,374
LPnetlib/lp_osa_60	243,246	10,280	1,408,073	0	243,246
LPnetlib/lp_pds_10	66,080	32,706	139,901	0	66,080
LPnetlib/lp_pds_20	143,063	68,762	302,423	0	143,063
LPnetlib/lp_stocfor3	23,541	16,675	72,721	0	23,541
Mittelmann/pds-100	724,224	365,890	1,515,296	0	724,224
Mittelmann/pds-30	212,001	103,456	447,659	0	212,001
Mittelmann/pds-40	292,934	142,247	617,606	0	292,934
Mittelmann/pds-50	374,331	181,577	787,867	0	374,331
Mittelmann/pds-60	459,275	222,285	965,265	0	459,275
Mittelmann/pds-70	536,575	261,514	1,126,605	0	536,575
Mittelmann/rail2586	923,269	2586	8,011,362	0	923,269
Mittelmann/rail4284	1,096,894	4284	11,284,032	0	1,096,894
Mittelmann/rail582	56,097	582	402,290	0	56,097

## 5.1 PS-IPM Versus Matlab®'s Solvers

In this section we compare the performance of Algorithm 3 (PS-IPM) with that of Matlab®'s IPM-based solvers. To this aim, we present in Fig. 2a–c the performance profiles of our proposal, PS-IPM, when compared to Matlab®'s `linprog` and `quadprog`. Moreover, in Table 2 we report the overall statistics of the performance of two methods.

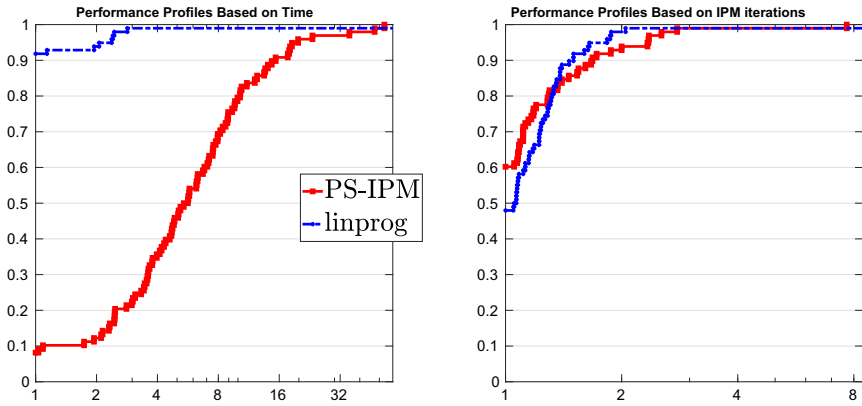
As already mentioned at the beginning of this section, we compare the two methods without using the pre-solved version of the problem collection (i.e. allowing rank-deficient matrices). Our proposal reaches the required accuracy on all the problems whereas Matlab®'s solvers do not solve a total of six problems. Hence, as expected, one of the benefits of the PPM framework becomes immediately obvious: the introduction of the regularization alleviates the rank deficiency of the constraint matrix while guaranteeing convergence. Moreover, in the small/medium setting, despite PS-IPM might not be the best performer on particular instances, it always outperforms `linprog` and `quadprog` in terms of overall execution time, see Fig. 2a, b and Table 2. Instead, as Fig. 2c clearly shows, PS-IPM is undoubtedly the best performer in terms of robustness and efficiency on the large scale dataset. All the obtained objective values from the methods are comparable.

## 5.2 PS-IPM Versus IP-PMM [25]

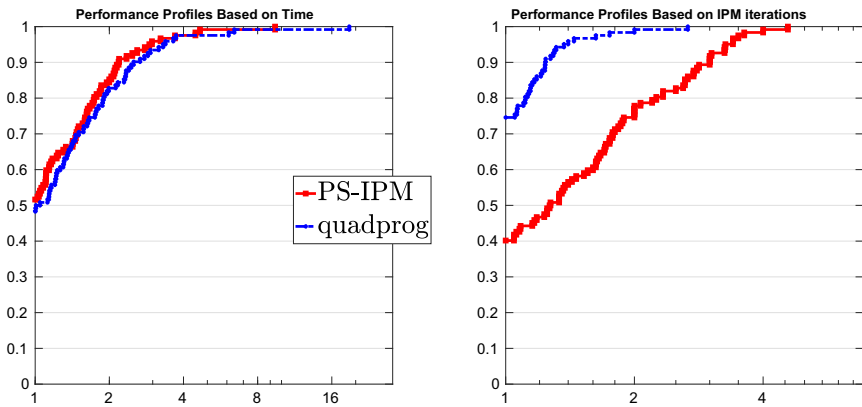
In this section we compare the performance of Algorithm 3 (PS-IPM) with that of IP-PMM [25]. To this aim, we report in Fig. 3a–c the performance profiles of PS-IPM when compared IP-PMM. Moreover, in Table 3, we report the overall statistics of the performance of two methods on the considered datasets. Our proposal reaches the required accuracy on all the problems whereas IP-PMM does not solve a total of two problems. Hence, as expected, our proposal has the same robustness capabilities as IP-PMM. On the other hand, as revealed from the Fig. 3a–c, the framework here proposed outperforms consistently IP-PMM in terms of IPM iterations (right panels). This fact is indeed always reflected in a reduction of the execution time (left panels and see also Table 3). All the obtained objective values from the two methods are comparable also in this case.

Finally, in order to give a measure of the number of IPM sweeps performed per PPM iteration when solving large scale problems, we report in Fig. 4 the ratio “Interior Point Iterations/Proximal Point Iterations” ( $\frac{\text{IPM It.}}{\text{PPM It.}}$ ). Clearly, lower ratios are indicative of the fact that, in average, fewer IPM iterations have been performed per PPM step. The figure further confirms the fact that thanks to the *warm starting* in Algorithm 3 and the property (18), the average number of IPM sweeps per PPM iteration remains bounded by the worst case factor of four also for larger problems than those corresponding to Fig. 1a, b (see Table 4 for details of the performance of PS-IPM on the large scale LPs considered).

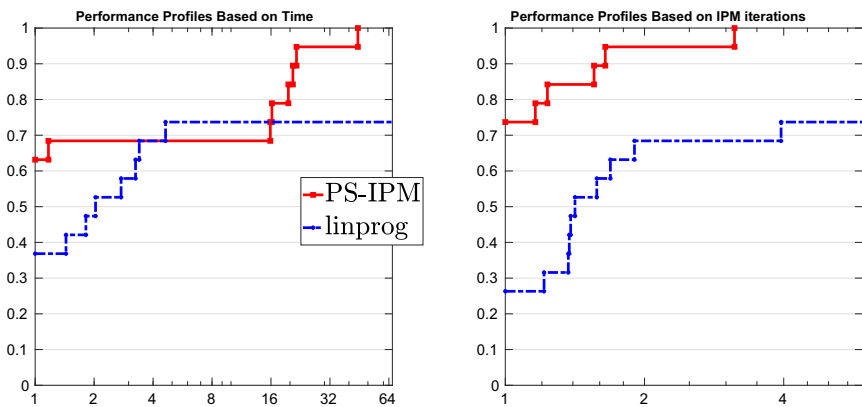




(a) Performance Profiles for Netlib's LP problems.



(b) Performance Profiles for Maros–Mészáros test set.

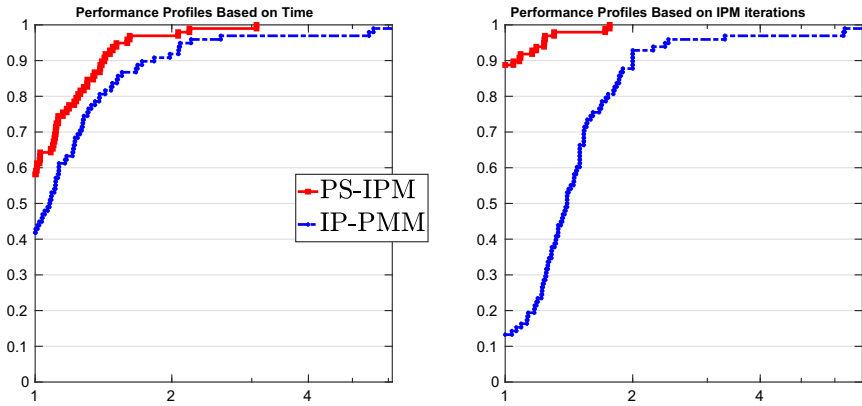


(c) Performance Profiles for Large Scale Problems.

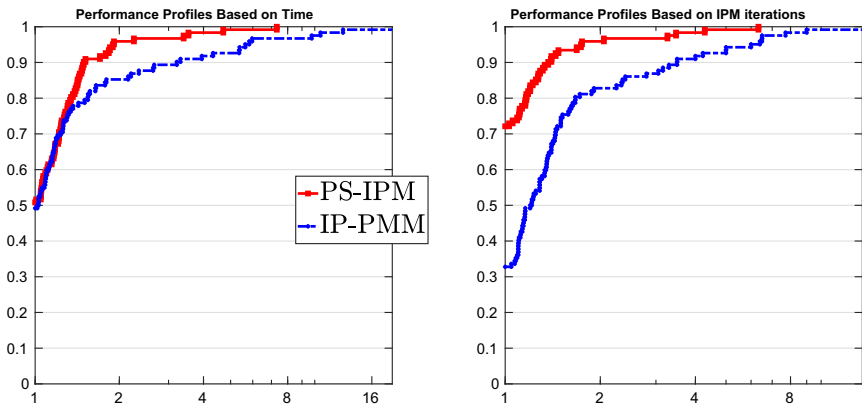
Fig. 2 PS-IPM versus Matlab®'s solvers: overall statistics

**Table 2** PS-IPM versus Matlab®'s solvers: performance profiles

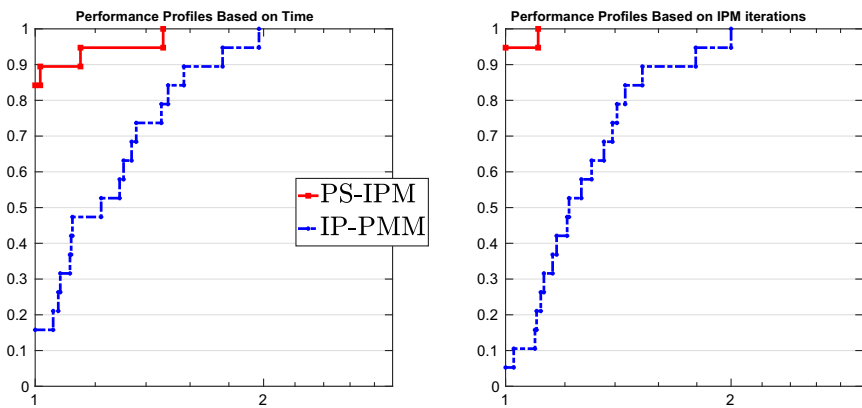
Test set/method	PS-IPM		Matlab®'s solvers				
	IPM	PMM	Optimized	Tot time (s)	IPM	Optimized	Tot time (s)
Netlib	2006	1423	98	1.84e+02	1909	97	2.11e+02
Maros–Mészáros	2192	1675	122	1.25e+02	1583	122	1.84e+02
Large scale	703	419	19	1.53e+04	1195	14	2.53e+04



(a) Performance Profiles for Netlib's LP problems.



(b) Performance Profiles for Maros–Mészáros test set.



(c) Performance Profiles for Large Scale Problems.

Fig. 3 PS-IPM versus PMM: performance profiles

**Table 3** PS-IPM versus PMM: overall statistics

Test set/method	PS-IPM		PMM		PPM [25]		Tot time (s)	Optimized	Tot time (s)
	IPM		IPM		IPM				
Netlib	2223		1555		3141		1.05e+02	97	1.27e+02
Maros-Mészáros	2515		1749		2969		9.12e+01	121	1.33e+02
Large scale	1072		425		1358		1.80e+04	19	2.33e+04

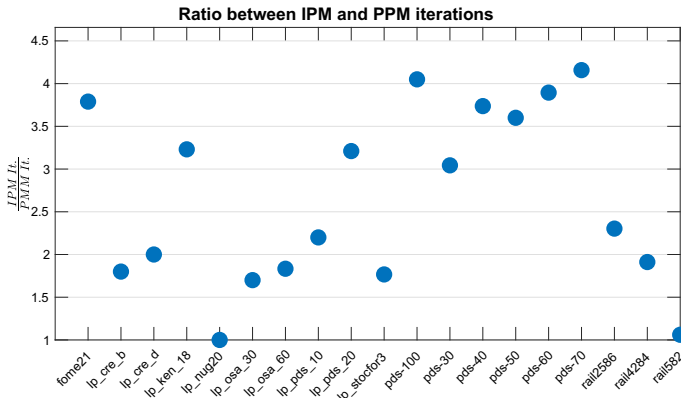


Fig. 4 Average IPM sweeps per PPM iteration

### 5.3 PS-IPM and Krylov Solvers

In this section we present the computational results obtained using Algorithm 3 when the problem is reformulated as in (19) and the corresponding linear systems arising from the PS-IPM subproblems are solved using (24) in conjunction with Krylov iterative solvers [30]. More in details, all the computational results presented here are devoted to showcase the theory developed in Sect. 4.1 and, in particular, to show that PS-IPM requires, in practice, a number of factorizations (short-handed as *Fact.* in the following) equal to a fraction of the IPM iterations, delivering hence considerable savings of computational time for problems where the factorization footprint is dominant. It is important to note that, in this context, a given factorization is not used any more as back-solver for a given Newton system but, rather, as preconditioner for suitable Krylov methods. In the following discussions and through all this section, we will use the ratio “Interior Point Iterations/ Factorizations” ( $\frac{IPM\ It.}{Fact.}$ ), as a measure of the frequency at which the preconditioner is recomputed: higher ratios are indicative of the fact that the same factorization has been used successfully by a larger number of IPM iterations. Moreover, we will use the ratio “Krylov Iterations/Factorizations” ( $\frac{Krylov\ It.}{Fact.}$ ) as a measure of how much the Krylov iterative solver is able to successfully exploit a given preconditioner: higher ratios are indicative of the fact that the same factorization has been used successfully to solve a larger number of linear systems.

#### 5.3.1 Strategy 1: Solution of the Linear System (24) Without Reduction to Schur Complement

*Computational framework’s details* In this section, for the solution of the linear system (24), we use GMRES [29] without restart and  $MaxIt = 100$ . Moreover, as suggested in the discussion in Sect. 4.1, as preconditioner of a given Schur complement  $S(\hat{\Theta})$ , we use the  $ldl$  decomposition of  $S(\Theta)$  computed in a previous PS-IPM iteration with factorization threshold parameter set equal to the regularization parameter, see

**Table 4** Details of PS-IPM performance for large scale problems

Problem	PPM iter	IPM iter	Time (s)	Obj val	Reg. par.	Status
Mittelmann/fome21	19	72	462.67	47346318912.00	5.42e-09	Opt
LPnetlib/lp_cre_b	25	45	17.97	23129639.89	5.00e-09	Opt
LPnetlib/lp_cre_d	24	48	18.68	24454969.76	5.00e-09	Opt
LPnetlib/lp_ken_18	13	42	61.06	-52217025287.38	5.00e-09	Opt
Qaplib/lp_nug20	17	17	270.18	2181.63	1.25e-07	Opt
LPnetlib/lp_osa_30	20	34	14.39	2142139.89	5.00e-09	Opt
LPnetlib/lp_osa_60	18	33	43.45	4044072.58	5.00e-09	Opt
LPnetlib/lp_pds_10	20	44	24.42	26727094976.00	5.42e-09	Opt
LPnetlib/lp_pds_20	19	61	211.31	23821658640.00	5.42e-09	Opt
LPnetlib/lp_stocfor3	30	53	5.48	-39976.78	5.00e-09	Opt
Mittelmann/pds-100	20	81	4322.40	10928229968.05	5.00e-09	Opt
Mittelmann/pds-30	23	70	472.93	21385445736.00	5.42e-09	Opt
Mittelmann/pds-40	19	71	1082.27	18855198824.11	5.42e-09	Opt
Mittelmann/pds-50	20	72	1323.81	16603525724.00	5.42e-09	Opt
Mittelmann/pds-60	19	74	1811.99	14265904407.18	5.42e-09	Opt
Mittelmann/pds-70	19	79	2726.32	12241162812.00	5.42e-09	Opt
Mittelmann/rail2586	33	76	2381.71	936.66	5.00e-09	Opt
Mittelmann/rail4284	34	65	2734.07	1054.51	5.00e-09	Opt
Mittelmann/rail582	33	35	71.41	209.72	5.00e-09	Opt

(35). It is important to note that when GMRES is applied to a non-normal matrix its convergence behaviour is not fully determined by the spectral distribution of the matrix, or better, this spectral distribution is completely irrelevant [16]. Nevertheless, when  $\|S(\Theta) - S(\widehat{\Theta})\|_2$  is small, we expect the matrix  $S(\Theta)^{-1}S(\widehat{\Theta})$  to be close to the identity. In this case, and in general for symmetric matrices, the spectral distribution is fully descriptive of GMRES behaviour [30, Cor. 6.33] and hence we expect GMRES to behave accordingly to the spectrum of the preconditioned matrix as in Theorem 4.1. As previously mentioned, we factorize the Schur complement in (25) using Matlab<sup>®</sup>'s `ldl` routine and, in our experiments, this factorization is recomputed if in the current PS-IPM step, GMRES has performed more than the 51% of the maximum allowed iterations in the solution of at least one of the two predictor-corrector systems. The stopping (absolute) tolerance for GMRES is set as  $\min\{10^{-1}, 0.8\mu\}$  where  $\mu$  is the current duality gap (the interested reader can see [8] for a recent analysis of inexact predictor-corrector schemes).

*Obtained numerical results* In Tables 5 and 6 we report the details for the largest instances of the medium-size LPs and QPs already considered in Sects. 5.1 and 5.2, see Table 1, when varying the stopping tolerance (tol) and when the regularization parameters have been suitably increased w.r.t. the ones used in the aforementioned sections in order to better showcase the developed theory.

The results obtained confirm that the ratio  $\frac{\text{IPM It.}}{\text{Fact.}}$  remains roughly in the interval (2.5, 6) for all the considered problems, see also Fig. 5, confirming, in general, that our proposal allows a small number of preconditioner re-computations. Moreover, as it becomes apparent from Fig. 5, when switching from  $\text{tol} = 10^{-5}$  to  $\text{tol} = 10^{-7}$  or  $\text{tol} = 10^{-8}$  the above mentioned ratio tends to increase for the problems PILOT87, CVXQP1, LISWET1, LISWET10, POWELL20, SHIP12L (the same happens for the ratio  $\frac{\text{Krylov It.}}{\text{Fact.}}$ ) essentially indicating that the computed preconditioners have been used to solve successfully a larger number of linear systems during the optimization procedure.

Indeed, this is in accordance with the observation carried out in Remark 4.1 of Sect. 4.1 regarding the fact that, when close enough to convergence, less *re-factorizations* are needed due to the convergence behaviour of the IPM contribution  $\Theta^{-1}$  to the matrix  $S(\Theta)$  [see Eq. (25)].

In Tables 7 and 8 we report the details for the instances of large size considered in Sect. 5.2 when the regularization is increased, respectively, by a factor  $f = 10$  and  $f = 500$  if compared to the regularization parameters used in Table 4.

In this case, the ratio  $\frac{\text{IPM It.}}{\text{Fact.}}$  remains bounded from below by a factor strictly greater than two, clearly indicating that, also in this case, the number of necessary factorizations to optimize successfully a given problem is just a fraction of the total IPM iterations. This fact could lead to reduced computational times for instances in which the effort related to the factorization is dominant, when compared to approaches where the `ldl` factorization is recomputed at each IPM iteration in order to solve the Newton system (see, e.g., the first part of this work).

As it becomes apparent from Fig. 6, when increasing the regularization parameters, the ratios  $\frac{\text{IPM It.}}{\text{Fact.}}$  and  $\frac{\text{Krylov It.}}{\text{Fact.}}$  tend to increase for the majority of the problems, indicating that the number of computed factorizations can be further reduced. Indeed,

**Table 5** Medium size LPs and QPs results,  $\text{tol} = 10^{-5}$

LP problems	$\text{tol} = 10^{-5}$							Status
	PPM It	IPM It	Krylov It	Fact	Time (s)	Obj val	Reg	
25FV47	26	26	1331	8	3.26	5501.85	7.00e-08	Opt
80BAU3B	28	38	2447	12	11.61	987224.23	7.00e-08	Opt
D6CUBE	16	16	983	5	2.45	315.50	7.00e-08	Opt
FIT2D	23	23	1274	7	10.84	-68464.27	7.00e-08	Opt
FIT2P	16	18	1002	5	5.81	68464.32	7.00e-08	Opt
PILOT87	33	39	2467	12	30.13	301.94	7.00e-08	Opt
QAP15	15	15	1045	5	46.36	1041.11	4.00e-06	Opt
QP problems	$\text{tol} = 10^{-7}$							Status
PPM It	IPM It	Krylov It	Fact	Time (s)	Obj val	Reg		
CVXQP1	13	18	1161	5	23.70	108704555.75	1.00e-10	Opt
LISWET1	29	31	1946	9	12.64	25.07	1.00e-09	Opt
LISWET10	31	31	1863	12	12.18	25.01	1.00e-09	Opt
POWELL20	5	28	1828	10	12.85	52089582811.44	1.00e-09	Opt
SHIP12L	10	13	918	5	3.45	3018876.58	1.27e-09	Opt
STCQP1	9	9	410	3	2.22	155143.55	7.36e-07	Opt



**Table 6** Medium size LPs and QPs results,  $tol = 10^{-7}$  or  $tol = 10^{-8}$

LP problems	$tol = 10^{-8}$								Status
	PPM It	IPM It	Krylov It	Fact	Time (s)	Obj val	Reg		
25FV47	27	27	1332	9	3.43	5501.85	7.00e-08		Opt
80BAU3B	32	42	2646	14	14.70	987224.19	7.00e-08		Opt
D6CUBE	17	17	986	6	3.00	315.49	7.00e-08		Opt
FIT2D	26	26	1302	9	13.36	-68464.29	7.00e-08		Opt
FIT2P	19	21	1184	6	7.30	68464.29	7.00e-08		Opt
PILOT87	42	78	4870	21	62.73	301.80	7.00e-08		Opt
QAP15	22	22	1478	8	68.24	1040.99	4.00e-06		Opt
QP problems	$tol = 10^{-7}$								Status
	PPM It	IPM It	Krylov It	Fact	Time (s)	Obj val	Reg		
CVXQP1	14	19	1192	5	23.52	108704648.71	1.00e-10		Opt
LISWET1	44	58	3584	11	22.79	25.12	1.00e-09		Opt
LISWET10	33	33	2049	12	13.47	25.01	1.00e-09		Opt
POWELL20	6	29	1832	10	12.89	52089582812.49	1.00e-09		Opt
SHIP12L	28	33	1934	6	6.75	3018876.58	1.27e-09		Opt
STCQP1	10	10	412	4	2.23	155143.55	7.36e-07		Opt

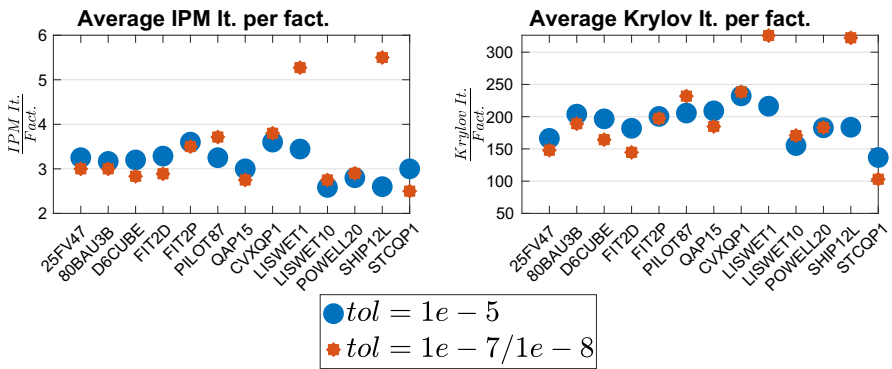


Fig. 5 Average IPM and Krylov It. per factorization

this is in accordance with the observation carried out in Remark 4.2 of Sect. 4.1 regarding the fact that the diagonal variations  $(D_A)_{ii}$  of the Schur complements  $S(\Theta)$  are inversely proportional to the regularization parameters  $(\rho, \delta)$  [see Eq. (29)].

Moreover, to further assess the robustness of our proposal on large scale problems, we complement the dataset used until now with additional large scale LP instances. We report in Table 9 the corresponding details.

### 5.3.2 Strategy 2: Solution of the Linear System (24) with Reduction to Schur Complement

*Computational framework’s details* When a further reduction to Schur complement is considered for the solution of the linear system (33), see Sect. 4.1.1, we propose to use the Preconditioned Conjugate Gradient (PCG) with  $MaxIt = 200$ . Moreover, as suggested in the discussion carried out in Sect. 4.1.1, as preconditioner of a given  $L_1(\hat{\Theta})$ , we use the Cholesky decomposition of  $L_1(\Theta)$  computed in a previous PS-IPM iteration. Analogously of what has been done in Sect. 5.3.1, we factorize a given  $L_1(\Theta)$  as (33) using Matlab®’s `chol` routine and, in our experiments, this factorization is recomputed if in the current PS-IPM step, PCG has performed more than 51% of the maximum allowed iterations in the solution of at least one of the two predictor-corrector systems. The stopping (absolute) tolerance for PCG is set as  $10^{-1} \cdot tol$  (this choice does not guarantee in general the best performance, see [32] for a recent analysis, but it is a robust one).

*Obtained computational results* Aim of this section is to show that also in the current approach the number of necessary factorizations is equal to a fraction of the total number of IPM iterations and that a further reduction to Schur complement might improve computational times when precise criteria are met. For this reason and for the sake of brevity, we present the obtained numerical results only on a selected subset of problems for which such reduced computational times are observed when compared to those presented in Sect. 5.3.1.

In Table 10 we summarize the statistics of the runs of our proposal when Newton linear systems are solved with PCG+chol.

**Table 7** Large scale problems  $f = 10$

Problem	PPM It.	IPM It.	Krylov It.	Fact.	Time (s)	Obj val	Reg. par.	Status
Mittelman/fome21	20	75	5057	23	700.63	47346318912.00	5.43e-09	Opt
LPnetlib/lp_cre_b	23	48	3760	16	81.57	23129639.89	5.00e-09	Opt
LPnetlib/lp_cre_d	22	46	3084	18	59.21	24454969.78	5.00e-09	Opt
LPnetlib/lp_ken_18	14	38	2241	14	215.48	-52217025287.38	5.00e-09	Opt
Qaplib/lp_nug20	17	17	1056	8	310.74	2181.64	1.25e-07	Opt
LPnetlib/lp_osa_30	19	29	1548	10	42.96	2142139.87	5.00e-09	Opt
LPnetlib/lp_osa_60	17	36	1992	11	121.06	4044072.51	5.00e-09	Opt
LPnetlib/lp_pds_10	19	46	3239	14	80.81	26727094976.01	5.43e-09	Opt
LPnetlib/lp_pds_20	19	60	4125	19	339.66	23821658640.00	5.43e-09	Opt
LPnetlib/lp_stocfor3	32	35	1808	11	19.82	-39976.78	5.00e-09	Opt
Mittelman/pds-100	20	85	5971	29	5638.99	10928229968.00	5.00e-09	Opt
Mittelman/pds-30	22	77	5087	23	709.16	21385445736.00	5.43e-09	Opt
Mittelman/pds-40	20	75	4953	23	1265.16	18855198824.08	5.43e-09	Opt
Mittelman/pds-50	19	78	5188	25	1666.61	16603525724.02	5.43e-09	Opt
Mittelman/pds-60	19	82	5909	26	2655.46	14265904407.03	5.43e-09	Opt
Mittelman/pds-70	20	80	5763	26	3511.44	12241162812.00	5.43e-09	Opt
Mittelman/rail2586	34	84	5734	33	2412.17	936.55	5.00e-09	Opt
Mittelman/rail4284	35	76	5353	27	2892.35	1054.89	5.00e-09	Opt
Mittelman/rail582	35	35	2461	11	56.05	209.75	5.00e-09	Opt

**Table 8** Large scale problems  $f = 500$

Problem	PPM It.	IPM It.	Krylov It.	Fact.	Time (s)	Obj val	Reg. par.	Status
Mittelman/fome21	19	71	4757	22	656.32	47346318912.12	2.71e-07	Opt
LPnetlib/lp_cre_b	24	50	3522	18	84.55	23129639.89	2.50e-07	Opt
LPnetlib/lp_cre_d	21	45	3076	17	62.66	24454969.77	2.50e-07	Opt
LNetlib/lp_ken_18	15	38	2332	12	212.58	-52217025287.40	2.50e-07	Opt
Qaplib/lp_nug20	17	17	1116	7	302.52	2181.63	6.25e-06	Opt
LNetlib/lp_osa_30	21	27	1581	10	45.61	2142139.87	2.50e-07	Opt
LNetlib/lp_osa_60	21	34	1978	10	115.57	4044072.57	2.50e-07	Opt
LNetlib/lp_pds_10	20	46	3139	14	81.69	26727094976.05	2.71e-07	Opt
LNetlib/lp_pds_20	19	62	4430	18	350.32	23821658639.93	2.71e-07	Opt
LNetlib/lp_stocfor3	36	39	2107	10	28.49	-39976.77	2.50e-07	Opt
Mittelman/pds-100	20	86	6004	26	5614.32	10928229968.01	2.50e-07	Opt
Mittelman/pds-30	22	77	5192	23	743.52	21385445736.02	2.71e-07	Opt
Mittelman/pds-40	20	75	4823	23	1216.00	18855198824.05	2.71e-07	Opt
Mittelman/pds-50	20	77	5368	24	1656.96	16603525724.00	2.71e-07	Opt
Mittelman/pds-60	19	76	5207	25	2317.59	14265904407.01	2.71e-07	Opt
Mittelman/pds-70	19	78	5284	27	3192.52	12241162812.00	2.71e-07	Opt
Mittelman/rail2586	34	82	5509	32	2389.66	936.54	2.50e-07	Opt
Mittelman/rail4284	35	77	5203	28	2799.40	1054.90	2.50e-07	Opt
Mittelman/rail582	36	36	2538	12	59.42	209.73	2.50e-07	Opt

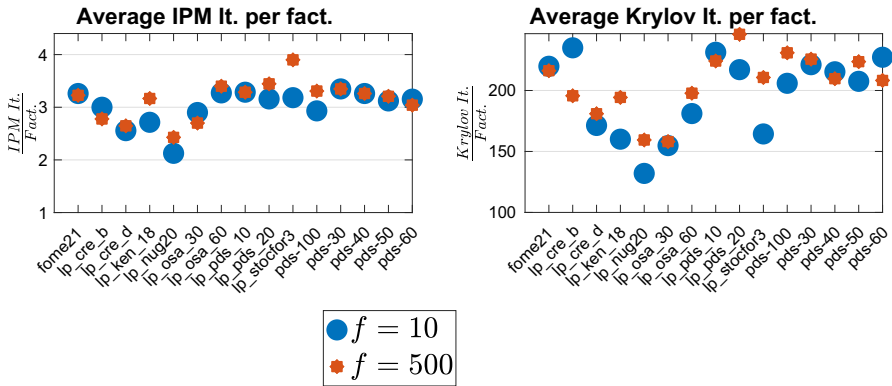


Fig. 6 Average IPM and Krylov It. per factorization

As the results presented in Table 10 confirm, in LP problems for which the pattern of the matrix  $A^T A + \delta I$  is particularly sparse and/or such matrix is of small dimension, the approach considered in this section leads to improved computational times while performing a limited number of Cholesky factorizations. Indeed, as confirmed in Fig. 7, the Cholesky factors of  $A^T A + \delta I$  for which improved computational times are observed when compared to those presented in Sect. 5.3.1, have a consistently smaller number of non-zeros entries when compared to the LDL factors of the saddle matrix  $\begin{bmatrix} \rho I & A^T \\ A & -\delta I \end{bmatrix}$ .

### 6 Conclusions

In this work we have clarified certain nuances of the convergence theory of primal-dual regularized Interior Point Methods (IPMs) using the inexact Proximal Point Method (PPM) framework: if on one hand this closes an existing literature gap, on the other, it sheds further light on their optimal implementation especially in the (nearly) rank deficient case of the linear constraints and/or in the large scale setting. Indeed, our convergence analysis does not require any linear independence assumption on the linear constraints nor the positive definiteness of the quadratic term. Moreover, when a direct solver can be used for the solution of the Newton system, we showed experimentally in Sect. 5.1, that the framework here proposed is competitive with Matlab®’s IPM-based solvers and, in Sect. 5.2, that a fixed but small regularization parameter is preferred to strategies in which the regularization is driven to zero. The second part of this work has been devoted to the study of the interactions between the regularization parameters and the computational complexity of the linear algebra solvers used in IPM. In Sect. 4.1 we proposed a new preconditioning technique able to exploit regularization as a tool to reduce the number of preconditioner re-computations when an iterative solver is needed for the solution of the IPMs Newton systems. Indeed, we were able to devise general purpose preconditioners which require an *update frequency* inversely proportional to the magnitude of the regularization parameters. Finally, building the

**Table 9** Large scale problems, additional dataset

Problem	$n$	$m$	$nmz$ (A)	$nmz$ (H)	Ineq. constr.
Mittelman/neos	515,905	479,119	1,526,794	0	515,905
Mittelman/neos3	518,832	512,209	2,055,024	0	518,832
Mittelman/hug08-3rd	29,856	19,728	148,416	0	29,856
Mittelman/stormG2_1000	1,377,306	528,185	3,459,881	0	1,377,306
Meszaros/tp-6	1,014,301	142,752	11,537,419	0	1,014,301
Mittelman/watson_1	409,997	224,160	1,101,103	0	409,997
Mittelman/watson_2	677,224	352,013	1,846,391	0	677,224

Problem	PPM It.	IPM It.	Krylov It.	Fact.	Time (s)	Obj val	Reg. par.	Status
Mittelman/neos	28	86	5118	22	1055.82	225425492.22	5.95e-08	Opt
Mittelman/neos3	9	14	437	3	119.47	27777.78	5.00e-09	Opt
Mittelman/hug08-3rd	8	8	613	3	322.88	214.00	7.81e-07	Opt
Mittelman/stormG2_1000	20	94	5897	19	10048.66	15802591.43	5.00e-09	Opt
Meszaros/tp-6	27	34	2376	9	966.40	-13194651.26	5.00e-09	Opt
Mittelman/watson_1	31	90	5803	21	834.08	-1693.39	5.00e-09	Opt
Mittelman/watson_2	31	104	6974	28	2067.68	-7544.06	5.00e-09	Opt

**Table 10** Results obtained for PCG + cho1

Problem	PPM It.	IPM It.	Krylow It.	Fact.	Time (s)	Obj val	Reg. par.	Status
<i>Compare with Table 5 (tol = 10<sup>-5</sup>)</i>								
25FV47	25	25	3110	8	0.66	5501.85	7.00e-08	Opt
80BAU3B	35	40	5356	13	3.39	987224.24	7.00e-08	Opt
D6CUBE	16	16	1871	5	0.65	315.51	7.00e-08	Opt
FIT2D	23	23	2576	5	3.23	-68464.26	7.00e-08	Opt
PILOT87	39	41	5498	13	7.14	301.95	7.00e-08	Opt
<i>Compare with Table 6 (tol = 10<sup>-8</sup>)</i>								
25FV47	28	28	3374	10	0.76	5501.85	7.00e-08	Opt
80BAU3B	39	44	5599	15	4.04	987224.19	7.00e-08	Opt
D6CUBE	18	18	1878	7	0.77	315.49	7.00e-08	Opt
FIT2D	26	26	2773	6	3.58	-68464.29	7.00e-08	Opt
PILOT87	49	95	11,780	20	14.96	301.79	7.00e-08	Opt
<i>Compare with Table 7 (tol = 10<sup>-6</sup>, f = 10)</i>								
LNetlib/lp_osa_30	27	27	3242	7	13.95	2142139.88	5.00e-09	Opt
LNetlib/lp_osa_60	26	34	4042	8	48.25	4044072.51	5.00e-09	Opt
LNetlib/lp_pds_10	25	47	5601	16	61.05	26727095000.48	5.43e-09	Opt

Table 10 continued

Problem	PPM It.	IPM It.	Kry/ov It.	Fáct.	Time (s)	Obj val	Reg. par.	Status
LPnetlib/tp_stocfor3	34	34	4447	11	7.48	-39976.78	5.00e-09	Opt
Mittelmann/rail2586	40	90	10,308	20	576.67	936.58	5.00e-09	Opt
Mittelmann/rail4284	41	86	9717	20	872.56	1054.62	5.00e-09	Opt
Mittelmann/rail582	36	36	5003	9	10.03	209.76	5.00e-09	Opt
<i>Compare with Table 8 (tol = 10<sup>-6</sup>, f = 500)</i>								
LPnetlib/tp_osa_30	28	28	3014	8	15.79	2142139.87	2.50e-07	Opt
LPnetlib/tp_osa_60	27	42	4894	9	62.70	4044072.53	2.50e-07	Opt
LPnetlib/tp_pds_10	25	47	6544	15	62.05	26727094975.92	2.71e-07	Opt
LPnetlib/tp_stocfor3	41	41	5300	12	8.29	-39976.77	2.50e-07	Opt
Mittelmann/rail2586	40	94	11,200	21	619.89	936.60	2.50e-07	Opt
Mittelmann/rail4284	41	79	9752	19	898.94	1054.80	2.50e-07	Opt
Mittelmann/rail582	37	37	4836	10	10.09	209.75	2.50e-07	Opt
<i>Compare with Table 9 (tol = 10<sup>-6</sup>)</i>								
Meszaros/tp-6	31	34	4325	9	758.84	-13194651.17	5.00e-09	Opt
Mittelmann/watson_1	35	84	10,786	24	331.30	-1693.28	5.00e-09	Opt
Mittelmann/watson_2	36	80	9055	21	481.80	-7543.88	5.00e-09	Opt



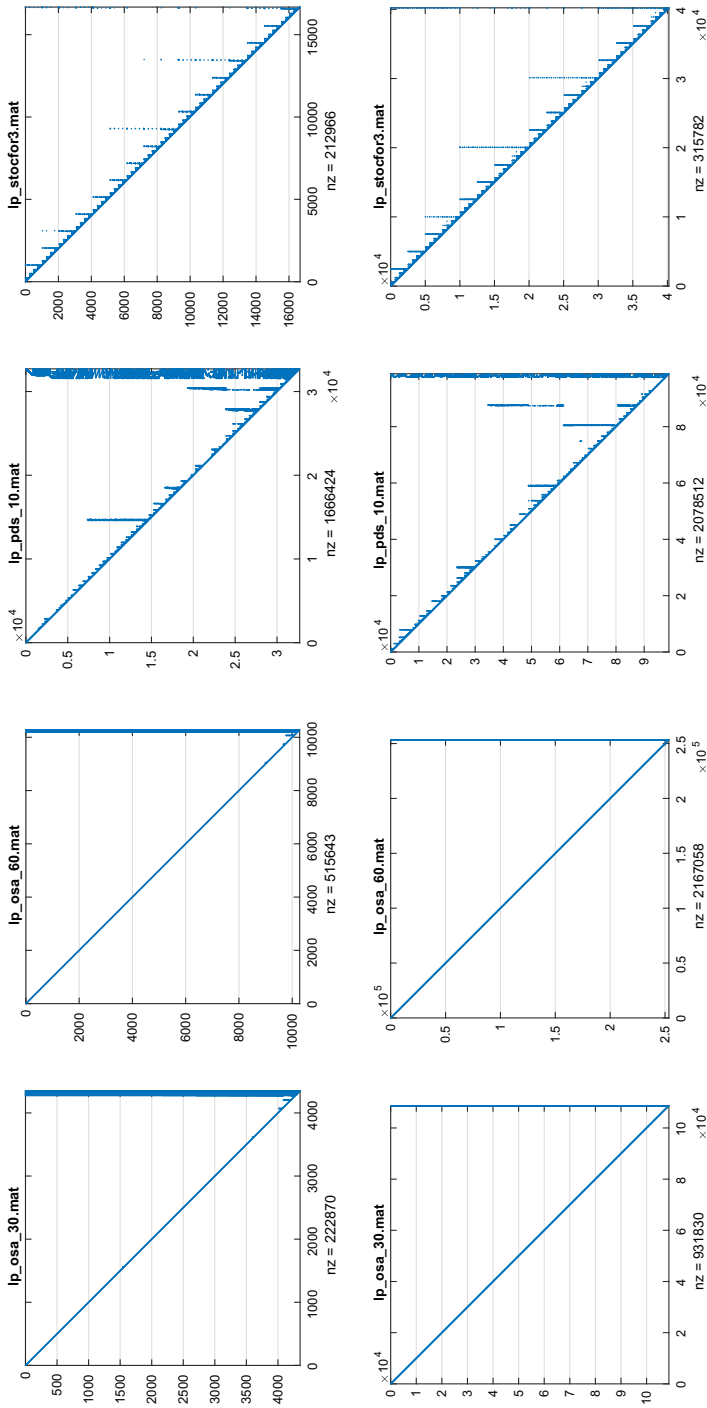


Fig. 7 Sparsity patterns of the Cholesky factors for normal equations (top) versus sparsity of LDL factors for saddle matrices (bottom)

momentum from the correct interpretation of the primal-dual regularization parameters in connection with the overall rate of convergence of the PPM and the proposed preconditioning strategy, we were able to show the robustness and efficiency of our proposal on a class of medium and large scale LPs and QPs.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Al-Jeiroudi, G., Gondzio, J.: Convergence analysis of the inexact infeasible interior point method for linear optimization. *J. Optim. Theory Appl.* **141**(2), 231–247 (2009). <https://doi.org/10.1007/s10957-008-9500-5>
2. Altman, A., Gondzio, J.: Regularized symmetric indefinite systems in interior-point methods for linear and quadratic optimization. In: *Optim. Methods Softw.* 11/12.1-4 (1999), pp. 275–302. <https://doi.org/10.1080/10556789908805754>
3. Armand, P., Benoist, J.: Uniform boundedness of the inverse of a Jacobian matrix arising in regularized interior-point methods. In: *Math. Program.* 137.1-2, Ser. A (2013), pp. 587–592. <https://doi.org/10.1007/s10107-011-0498-3>
4. Clason, C., Valkonen, T.: Introduction to nonsmooth analysis and optimization. In: *arXiv* (2020). <https://doi.org/10.48550/ARXIV.2001.00216>
5. Colombo, M., Gondzio, J.: Further development of multiple centrality correctors for interior point methods. *Comput. Optim. Appl.* **41.3**, 277–305 (2008). <https://doi.org/10.1007/s10589-007-9106-0>
6. D'Apuzzo, M., De Simone, V., di Serafino, D.: On mutual impact of numerical linear algebra and large-scale optimization with focus on interior point methods. *Comput. Optim. Appl.* **45.2**, 283–310 (2010). <https://doi.org/10.1007/s10589-008-9226-1>
7. De Marchi, A.: On a primal-dual Newton proximal method for convex quadratic programs. *Comput. Optim. Appl.* **81**, 369–395 (2022). <https://doi.org/10.1007/s10589-021-00342-y>
8. Dexter, G., Chowdhury, A., Avron, H., Drineas, P.: On the convergence of inexact predictor-corrector methods for linear programming. In: *arXiv* (2022). <https://doi.org/10.48550/ARXIV.2202.01756>
9. Dontchev, A.L., Rockafellar, R.T.: *Implicit Functions and Solution Mappings*. Springer Monographs in Mathematics, p. xii+375. Springer, Dordrecht (2009). <https://doi.org/10.1007/978-0-387-87821-8>
10. Duff, I.S.: MA57—a code for the solution of sparse symmetric definite and indefinite systems. *ACM Trans. Math. Softw.* **30.2**, 118–144 (2004). <https://doi.org/10.1145/992200.992202>
11. Friedlander, M.P., Orban, D.: A primal-dual regularized interior point method for convex quadratic programs. *Math. Program. Comput.* **4.1**, 71–107 (2012). <https://doi.org/10.1007/s12532-012-0035-2>
12. Gondzio, J.: Convergence analysis of an inexact feasible interior-point method for convex quadratic programming. *SIAM J. Optim.* **23.3**, 1510–1527 (2013). <https://doi.org/10.1137/120886017>
13. Gondzio, J.: Interior point methods 25 years later. *Eur. J. Oper. Res.* **218.3**, 587–601 (2012). <https://doi.org/10.1016/j.ejor.2011.09.017>
14. Gondzio, J.: Matrix-free interior point methods. *Comput. Optim. Appl.* **51.2**, 457–480 (2012). <https://doi.org/10.1007/s10589-010-9361-3>
15. Gondzio, J.: Multiple centrality corrections in a primal-dual method for linear programming. *Comput. Optim. Appl.* **6.2**, 137–156 (1996). <https://doi.org/10.1007/BF00249643>
16. Greenbaum, A., Pták, V., Strakoš, Z.: Any nonincreasing convergence curve is possible for GMRES. *SIAM J. Matrix Anal. Appl.* **17.3**, 465–469 (1996). <https://doi.org/10.1137/S0895479894275030>

17. Hestenes, M.R.: Multiplier and gradient methods. *J. Optim. Theory Appl.* **4**, 303–320 (1969). <https://doi.org/10.1007/BF00927673>
18. Kojima, M., Megiddo, N., Mizuno, S.: A primal-dual infeasible interior point algorithm for linear programming. In: *Math. Programming* 61.3, Ser. A (1993), pp. 263–280. <https://doi.org/10.1007/BF01582151>
19. Liao-McPherson, D., Kolmanovsky, I.: FBstab: a proximally stabilized semismooth algorithm for convex quadratic programming. In: *Automatica J. IFAC* 113 (2020). <https://doi.org/10.1016/j.automatica.2019.108801>
20. Luque, F.J.: Asymptotic convergence analysis of the proximal point algorithm. *SIAM J. Control Optim.* **22.2**, 277–293 (1984). <https://doi.org/10.1137/0322019>
21. Maros, I., Mészáros, C.: A repository of convex quadratic programming problems. In: *Optim. Methods Softw.* 11/12.1-4 (1999), pp. 671–681. <https://doi.org/10.1080/10556789908805768>
22. Mehrotra, S.: On the implementation of a primal-dual interior-point methods. *SIAM J. Optim.* **2.4**, 575–601 (1992). <https://doi.org/10.1137/0802028>
23. Morini, B., Simoncini, V., Tani, M.: A comparison of reduced and unreduced KKT systems arising from interior point methods. *Comput. Optim. Appl.* **68.1**, 1–27 (2017). <https://doi.org/10.1007/s10589-017-9907-8>
24. Pang, J.-S.: Error bounds in mathematical programming. In: *Math. Program.* 79.1-3, Ser. B (1997), pp. 299–332. [https://doi.org/10.1016/S0025-5610\(97\)00042-7](https://doi.org/10.1016/S0025-5610(97)00042-7)
25. Pougkakiotis, S., Gondzio, J.: An interior point-proximal method of multipliers for convex quadratic programming. *Comput. Optim. Appl.* **78.2**, 307–351 (2021). <https://doi.org/10.1007/s10589-020-00240-9>
26. Rockafellar, R.T.: Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Math. Oper. Res.* **1.2**, 97–116 (1976). <https://doi.org/10.1287/moor.1.2.97>
27. Rockafellar, R.T.: Monotone operators associated with saddle-functions and minimax problems. In: *Nonlinear Functional Analysis (Proc. Sympos. Pure Math., Vol. XVIII, Part 1, Chicago, Ill., 1968)*. Amer. Math. Soc., Providence, R.I., (1970), pp. 241–250
28. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. *SIAM J. Control Optim.* **14.5**, 877–898 (1976). <https://doi.org/10.1137/0314056>
29. Saad, Y., Schultz, M.H.: GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **7.3**, 856–869 (1986). <https://doi.org/10.1137/0907058>
30. Saad, Y.: *Iterative Methods for Sparse Linear Systems*, 2nd edn., p. xviii+528. Society for Industrial and Applied Mathematics, Philadelphia (2003). <https://doi.org/10.1137/1.9780898718003>
31. Wright, S.J.: *Primal-Dual Interior-Point Methods*, p. xx+289. Society for Industrial and Applied Mathematics, Philadelphia, PA (1997). <https://doi.org/10.1137/1.9781611971453>
32. Zanetti, F., Gondzio, J.: A new stopping criterion for Krylov solvers applied in interior point methods. In: *SIAM J. on Sci. Comp.* (Accepted)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.