Journal of Optimization Theory and Applications (2019) 181:197–215
https://doi.org/10.1007/s10957-018-1401-7

An Augmented Lagrangian Method for Equality Constrained Optimization with Rapid Infeasibility Detection Capabilities

Paul Armand[1] · Ngoc Nguyen Tran[1]

Received: 1 May 2018 / Accepted: 19 September 2018 / Published online: 29 September 2018

Abstract
We present a primal-dual augmented Lagrangian method for solving an equality constrained minimization problem, which is able to rapidly detect infeasibility. The method is based on a modification of the algorithm proposed in Armand and Omheni (Optim Methods Softw 32(1):1–21, 2017). A new parameter is introduced to scale the objective function and, in case of infeasibility, to force the convergence of the iterates to an infeasible stationary point. It is shown, under mild assumptions, that whenever the algorithm converges to an infeasible stationary point, the rate of convergence is quadratic. This is a new convergence result for the class of augmented Lagrangian methods. The global convergence of the algorithm is also analyzed. It is also proved that, when the algorithm converges to a stationary point, the properties of the original algorithm are preserved. The numerical experiments show that our new approach is as good as the original one when the algorithm converges to a local minimum, but much more efficient in case of infeasibility.

Keywords Nonlinear optimization · Augmented Lagrangian method · Infeasibility detection

1 Introduction

When solving an optimization problem, most of the time it is not known in advance if the problem is feasible or not. If care is not taken, the numerical solution of an

Communicated by Marco Antonio López-Cerdá.

✉ Paul Armand
paul.armand@unilim.fr

Ngoc Nguyen Tran
ngoc-nguyen.tran@unilim.fr

[1] Université de Limoges - Laboratoire XLIM, Limoges, France

infeasible nonlinear problem may lead to a long sequence of iterations until the algorithm stops because a failure is detected. Even if the problem is feasible, the sequence of iterates may sometimes converges to an infeasible stationary point. In that case, it would be convenient to quickly detect the infeasibility of the computed solution, in order to choose a new starting point and hope to converge to a feasible solution. A rapid infeasibility detection is particularly important when solving large sequences of nonlinear optimization problems that differ from each other by the variation of values of some parameters or by some slight modifications of the optimization model, as for example in branch-and-bound methods.

In this paper, we concentrate on the rapid detection of the infeasibility in the framework of the solution of a nonlinear optimization problem by means of a primal-dual augmented Lagrangian method. The augmented Lagrangian method was proposed independently by Hestenes [1] and Powell [2]. It is known to be very robust with respect to the degeneracy due to the linear dependence of the gradients of the constraints. This method is the basis of some efficient softwares like LANCELOT-A [3], ALGENCAN [4–6] and SPDOPT [7]. It is worth noting that the algorithm proposed in [7] departs from the standard augmented Lagrangian algorithm by its primal-dual nature. The advantage is to get a superlinear or quadratic rate of convergence of the iterates to an optimal solution under usual assumptions. In this paper, we propose a modification of this algorithm, which also guarantees a rapid rate of convergence when the sequence of iterates converges to an infeasible stationary point.

The infeasibility detection has been recently the focus of particular attention for improving the behavior of augmented Lagrangian algorithms. Martínez and Prudente [8] proposed an adaptive stopping criterion for the solution of the subproblems and showed that their new algorithm performs better than the original version of ALGENCAN. Birgin et al. [9,10] improved also an augmented Lagrangian algorithm within the framework of global optimization and show better performances than the initial implementation in [11]. Gonçalves et al. [12] extend the results of [9] to an entire class of penalty functions. Within the framework of sequential quadratic programming (SQP) methods, Byrd et al. [13] have proposed an algorithm to quickly detect infeasibility and have shown that their algorithm has fast local convergence properties. To our knowledge, this is the first work that analyzes the local convergence around an infeasible stationary point. More recently, Burke et al. [14] have improved the infeasibility detection for an SQP algorithm. They also proved the global and rapid local convergence properties of their algorithm.

The augmented Lagrangian method of Armand and Omheni [7] may detect infeasibility when the sequence of dual variables becomes unbounded and the penalty parameter is forced to zero. The main drawback is that the infeasibility can take long to detect. We then propose to introduce a new parameter, called the feasibility parameter, whose role is to control the progress of the iterates to the feasible set. This parameter scales the objective function relatively to the constraints until a nearly feasible point is detected. The level of feasibility detection is arbitrary and, for example, can be chosen of the same order as the overall stopping tolerance. Once this event arises, the algorithm switches to its normal behavior and continues the minimization process until convergence. From a formal point of view, the algorithm can be interpreted as the numerical solution of the Fritz John optimality conditions, but with a perturbation of the constraints due to the augmented Lagrangian parameters (Lagrange multiplier and quadratic penalty term). The feasibility parameter is updated dynamically. In particular, its value depends on the norm of the residual of a primal-dual system related to the minimization of the infeasibility measure. This leads to a superlinear or quadratic convergence of the sequence of iterates to an infeasible stationary point. To our knowledge, this is the first local convergence result in the infeasible case of an augmented Lagrangian method. The paper concentrates on the equality constraints case, to complete the capability of the algorithm [7] in detecting infeasibility. A possible extension to the general case with equalities and inequalities is discussed in Sect. 7.

The paper is organized as follows. In the next section, we summarize our notation and terminology which will be used. The algorithm is described in the next section. The global convergence and the asymptotic analysis are studied in Sects. 4 and 5. Some numerical experiments are reported in Sect. 6 to demonstrate the efficiency of new method. Section 7 ends the paper.

2 Notation

For two real vectors x and y of same lengths, $x^\top y$ is their Euclidean scalar product and $\|x\| = (x^\top x)^{1/2}$ is the associated norm. For a real matrix M, the induced matrix norm is $\|M\| = \max\{\|Md\| : \|d\| \leq 1\}$. The inertia of a real symmetric matrix M, denoted by $\mathrm{In}(M) := (\iota_+, \iota_-, \iota_0)$, is the numbers of positive, negative and null eigenvalues. For a function f and an iterate x_k, to simplify the notation we denote $f_k = f(x_k)$. Likewise, f^* stands for $f(x^*)$, g_k^+ stands for $g(x_k^+)$, and so on. The positive part of a real number r is defined by $r_+ = \max\{r, 0\}$. Let $\{a_k\}$ and $\{b_k\}$ be nonnegative scalar sequences. We write $a_k = O(b_k)$, or equivalently $b_k = \Omega(a_k)$, if there exists a constant $c > 0$ such that $a_k \leq c\, b_k$ for all $k \in \mathbb{N}$. The notation $a_k = \Theta(b_k)$ means that $a_k = O(b_k)$ and $a_k = \Omega(b_k)$. We also write $a_k = o(b_k)$, if $a_k = \epsilon_k b_k$ for all $k \in \mathbb{N}$, with $\lim \epsilon_k = 0$.

3 Algorithm

We consider the equality constrained optimization problem

$$\text{minimize } \rho f(x) \;\; \text{subject to } c(x) = 0, \tag{P$_\rho$}$$

where $f:\mathbb{R}^n \to \mathbb{R}$ and $c:\mathbb{R}^n \to \mathbb{R}^m$ are smooth, and where $\rho \geq 0$. For the value $\rho = 1$, the problem (P_1) is referred as the original problem, the one to be initially solved. For the value $\rho = 0$, any feasible solution is optimal for (P_0). The parameter ρ is then called as the feasibility parameter.

The main contribution of this paper is to propose an updating strategy of the feasibility parameter, in order to guarantee the global convergence of the minimization algorithm to a feasible or an infeasible stationary point of the original problem and also fast local convergence in both cases.

Let us recall some definitions about stationary points. A point $x \in \mathbb{R}^n$ is called a Fritz John (FJ) point of problem (P$_1$) if there exists $(z, y) \in \mathbb{R} \times \mathbb{R}^m$ with $(z, y) \neq 0$ such that

$$c(x) = 0 \quad \text{and} \quad zg(x) + A(x)y = 0,$$

where $g(x) = \nabla f(x)$ denotes the gradient of f at x and $A(x) = \nabla c(x)$ is the transpose of the Jacobian matrix of the constraints at x. A FJ point x is a Karush–Kuhn–Tucker (KKT) point whenever $z \neq 0$. In that case, y/z is the vector of Lagrange multipliers related to problem (P$_1$). A FJ point x for which $z = 0$ is called a singular stationary point. In other words, a singular stationary point is a feasible point at which the linear independence constraint qualification (LICQ) does not hold. A point $x \in \mathbb{R}^n$ is called an infeasible stationary point of problem (P$_1$) if

$$c(x) \neq 0 \quad \text{and} \quad A(x)c(x) = 0.$$

In other words, an infeasible stationary point is not feasible for the problem (P$_1$) and is a stationary point for the feasibility problem

$$\text{minimize}_{x \in \mathbb{R}^n} \ \tfrac{1}{2}\|c(x)\|^2. \tag{1}$$

The augmented Lagrangian associated with (P$_\rho$) is defined as

$$\mathcal{L}_{\rho,\sigma}(x, \lambda) := \rho f(x) + \lambda^\top c(x) + \tfrac{1}{2\sigma}\|c(x)\|^2, \tag{2}$$

where $\lambda \in \mathbb{R}^m$ is an estimate of the vector of Lagrange multipliers associated with the equality constraints and $\sigma > 0$ is a quadratic penalty parameter. Recall that when x^* is a KKT point for (P$_\rho$), with an associated vector of Lagrange multipliers λ^*, if the sufficient second-order optimality conditions hold at x^*, then x^* is a strict local minimum of $\mathcal{L}_{\rho,\sigma}(\cdot, \lambda^*)$ provided that σ is small enough, see e.g., [15, Proposition 1.26]. This result serves as a basis of augmented Lagrangian methods, in which the augmented Lagrangian is minimized while the parameters λ and σ are updated in an appropriate manner, see e.g., [16, Chapter 17].

The first-order optimality conditions for minimizing $\mathcal{L}_{\rho,\sigma}(\cdot, \lambda)$ are

$$\rho g(x) + A(x)\left(\lambda + \tfrac{1}{\sigma}c(x)\right) = 0.$$

By introducing the dual variable $y \in \mathbb{R}^m$ and the notation $w := (x, y)$, these optimality conditions can be reformulated as

$$\Phi(w, \lambda, \rho, \sigma) := \begin{pmatrix} \rho g(x) + A(x)y \\ c(x) + \sigma(\lambda - y) \end{pmatrix} = 0.$$

This formulation of the optimality conditions for minimizing (2) serves as a basis of our algorithm. Note that by setting $\lambda = y$, we retrieve the optimality conditions of problem (P$_\rho$).

Let us define the regularized Jacobian matrix of the function Φ with respect to w by

$$J_{\rho,\sigma,\theta}(w) = \begin{pmatrix} H_{\rho,\theta}(w) & A(x) \\ A(x)^\top & -\sigma I \end{pmatrix},$$

where $\theta \geq 0$ is a regularization parameter and where

$$H_{\rho,\theta}(w) = \rho \nabla^2 f(x) + \sum_{i=1}^{m} y_i \nabla^2 c_i(x) + \theta I$$

is the regularized Hessian of the Lagrangian associated with (P_ρ). During the iterations, the regularization parameter is chosen to control the inertia of regularized Jacobian matrix of Φ. It is well known that $\mathrm{In}(J_{\rho,\sigma,\theta}(w)) = (n, m, 0)$ if and only if the matrix

$$K_{\rho,\sigma,\theta}(w) := H_{\rho,\theta}(w) + \tfrac{1}{\sigma} A(x) A(x)^\top$$

is positive definite (see e.g., [17, Lemma A.16]). A link with the augmented Lagrangian is given by the following formula:

$$K_{\rho,\sigma,\theta}(w) = \nabla^2_{xx} \mathcal{L}_{\rho,\sigma}(x, y - \tfrac{1}{\sigma} c(x)) + \theta I.$$

The algorithm is a Newton-type method for the solution of the optimality system $\Phi = 0$, and it follows the one proposed in [7]. The globalization scheme of the algorithm uses two kinds of iteration. At a main iteration, called outer iteration, all the parameters λ, ρ and σ are updated and a full Newton step for the solution of $\Phi = 0$ is performed. If the norm of the residual $\|\Phi\|$ at the trial iterate is deemed sufficiently small, then the new iterate is updated and a new outer iteration is called, otherwise the parameters are fixed to their current values and a sequence of inner iterations is applied in order to reduce sufficiently $\|\Phi\|$. The inner iteration algorithm is a backtracking line search applied to a primal-dual merit function, whose first-order optimality conditions correspond to $\Phi = 0$.

Algorithm 1 is quite similar to [7, Algorithm 1], except for the first four steps which are related to the updating of the parameters.

Initially, a primal-dual starting point is defined and the values of the parameters are chosen. A flag is used to indicate if the algorithm is in the feasibility detection phase ($\mathtt{F} = \mathtt{true}$) or not ($\mathtt{F} = \mathtt{false}$).

At the first step, the algorithm tests if a nearly feasible point, with regards to a feasibility tolerance $\epsilon > 0$, has been detected. If it is the case, the algorithm switches into the normal operating mode of [7, Algorithm 1]. This means in particular that the feasibility parameter ρ_k will remain constant for all further iterations. This switching mechanism is necessary to avoid the undesirable situation where the feasibility measure goes to zero very slowly, while the condition (3) is alternatively satisfied and not satisfied an infinite number of times, leading to decreasing the feasibility parameter to zero. Moreover, in this situation, it would be impossible to make the distinction between the satisfaction of the KKT conditions and the regularity of the constraints.

Algorithm 1

```
0. Initialize $w_0 = (x_0, y_0) \in \mathbb{R}^{n+m}$ and set $\lambda_0 = y_0$. Choose parameters $\rho_0 > 0$, $\sigma_0 > 0$, $\epsilon > 0$, $a \in ]0, 1[$, $\ell \in \mathbb{N}$ and $\tau \in ]0, 1[$. Set $k = 0$ and $i_0 = 0$. Set $\mathtt{F} = \mathtt{true}$.
1. If $\|c_k\| \leq \epsilon$, then set $\mathtt{F} = \mathtt{false}$.
2. Choose $\zeta_k > 0$ such that $\{\zeta_k\} \to 0$. If $k = 0$ or
       $$\|c_k\| \leq a \max\{\|c_{i_j}\| : (k - \ell)_+ \leq j \leq k\} + \zeta_k \qquad (3)$$
   then set $i_{k+1} = k$ and go to Step 4, otherwise set $i_{k+1} = i_k$.
3. If $\mathtt{F} = \mathtt{true}$, then choose $0 < \rho_{k+1} \leq \tau\rho_k$ and set $\sigma_{k+1} = \sigma_k$, else choose $0 < \sigma_{k+1} \leq \tau\sigma_k$ and set $\rho_{k+1} = \rho_k$. Set $\lambda_{k+1} = \frac{\rho_{k+1}}{\rho_k}\lambda_k$ and go to Step 5.
4. Choose $0 < \sigma_{k+1} \leq \sigma_k$. Set $\rho_{k+1} = \rho_k$ and $\lambda_{k+1} = y_k$.
5. Choose $\theta_k \geq 0$, such that $\mathrm{In}(J_k) = (n, m, 0)$, where $J_k = J_{\rho_{k+1},\sigma_{k+1},\theta_k}(w_k)$. Compute $w_k^+$ by solving the linear system
       $$J_k(w_k^+ - w_k) = -\Phi(w_k, \lambda_{k+1}, \rho_{k+1}, \sigma_{k+1}).$$
6. Choose $\varepsilon_k > 0$ such that $\{\varepsilon_k\} \to 0$. If
       $$\|\Phi(w_k^+, \lambda_{k+1}, \rho_{k+1}, \sigma_{k+1})\| \leq \varepsilon_k, \qquad (4)$$
   then set $w_{k+1} = w_k^+$. Otherwise, apply a sequence of inner iterations to find $w_{k+1}$ such that
       $$\|\Phi(w_{k+1}, \lambda_{k+1}, \rho_{k+1}, \sigma_{k+1})\| \leq \varepsilon_k. \qquad (5)$$
7. If termination criteria hold for ($P_\rho$) then stop, else increment $k$ by 1 and go to Step 1.
```

At the second step, the algorithm tests if a sufficient reduction in the feasibility measure has been obtained. If it is the case, the feasibility parameter is kept constant, the Lagrange multiplier estimate is set to the current value of the dual variable and a new value of the quadratic penalty parameter is chosen. For $k \geq 1$, the index i_k is the number of the last iteration prior to k at which inequality (3) holds. Note that, at Step 4, the quadratic penalty parameter is chosen in such a way that it could remain constant all along the iterations. But in that case, the convergence to a KKT point is only linear and the numerical experiments in [7] have shown that, in practice, it is better to force the convergence of σ_k to zero.

If the algorithm detects that the constraints have not decreased sufficiently, because condition (3) is not satisfied, then there are two situations. If $\mathtt{F} = \mathtt{true}$, then the algorithm is still in the feasibility detection phase. In that case, the feasibility parameter is sufficiently decreased, the quadratic penalty parameter is kept constant and the Lagrange multiplier estimate is rescaled. This scaling is important to force the convergence to zero of $\{\lambda_k\}$ when this step is always executed from some iteration (see Lemma 3.1-(ii)), ensuring that the sequence of iterates approaches stationarity of the feasibility problem (see Theorem 4.1-(ii)). The second situation is when $\mathtt{F} = \mathtt{false}$. In that case, the algorithm has left the feasibility detection phase. Then, the feasibility parameter is kept constant, but the quadratic penalty parameter is decreased to penalize the constraints and the Lagrange multiplier estimate is kept constant as in a classical augmented Lagrangian algorithm.

The following lemma summarizes the behavior of the algorithm regarding the feasibility detection strategy and the update rules of the parameters.

Lemma 3.1 *Assume that Algorithm 1 generates an infinite sequence $\{w_k\}$. Let $\mathcal{K} \subset \mathbb{N}$ be the set of iteration indices for which the condition (3) is satisfied.*

(i) *If $\mathcal{K}$ is infinite, then the subsequence $\{c_k\}_{k\in\mathcal{K}}$ converges to zero and $\{\rho_k\}$ is eventually constant.*
(ii) *If $\mathcal{K}$ is finite, then* $\liminf \|c_k\| > 0$ *and both sequences $\{\sigma_k\rho_k\}$ and $\{\sigma_k\lambda_k\}$ converge to zero.*

Proof For $k \in \mathbb{N}$, set $\beta_k = \|c_{i_k}\|$. We then have for all $k \in \mathcal{K}$,

$$\beta_{k+1} \leq a \max\{\beta_j : (k-\ell)_+ \leq j \leq k\} + \zeta_k$$

and for all $k \notin \mathcal{K}$, $\beta_{k+1} = \beta_k$. It has been shown in [7, Lemma 3.1] that such a sequence converges to zero. This proves the first conclusion of assertion (i). Since $\{c_k\}_{k\in\mathcal{K}}$ converges to zero, then there exists $k_0 \in \mathcal{K}$ such that $\|c_{k_0}\| \leq \epsilon$ and thus $\mathtt{F} = \mathtt{false}$ for all further iterations. The update rules of the feasibility parameter at Step 3 and Step 4 imply that $\rho_k = \rho_{k_0}$ for all $k \geq k_0$, which proves the second conclusion of assertion (i).

To prove conclusion (ii), suppose that $\mathcal{K}$ is finite and let $k_0 = \max \mathcal{K}$. For all $k \geq k_0 + 1$, $i_k = k_0$ and Step 3 is executed. It follows that for all $k \geq k_0 + \ell$, we have $\|c_k\| > a\|c_{k_0}\|$, and therefore $\liminf \|c_k\| > 0$. We consider two cases. If at some iteration k, $\|c_k\| \leq \epsilon$, then $\mathtt{F} = \mathtt{false}$ for all further iterations. The update of the parameters at Step 3 implies that both sequences $\{\rho_k\}$ and $\{\lambda_k\}$ are eventually constant and $\{\sigma_k\}$ tends to zero. It follows that $\{\sigma_k\rho_k\}$ and $\{\sigma_k\lambda_k\}$ tend to zero. The second case is when $\|c_k\| > \epsilon$ for all $k \in \mathbb{N}$, which implies that $\mathtt{F} = \mathtt{true}$ at each iteration. In that case, for all $k \geq k_0 + 1$, $\rho_{k+1} \leq \tau\rho_k$, $\sigma_{k+1} = \sigma_k$ and $\lambda_{k+1} = \frac{\rho_{k+1}}{\rho_{k_0}} y_{k_0}$. We deduce that $\{\rho_k\}$ goes to zero, $\{\sigma_k\}$ is eventually constant, and $\{\lambda_k\}$ goes to zero, which implies that both sequences $\{\sigma_k\rho_k\}$ and $\{\sigma_k\lambda_k\}$ tend to zero. □

At Step 5 of Algorithm 1, the parameter θ_k is selected to control the inertia of the matrix J_k. This issue is important to avoid that the outer iterates converge to a stationary point, which is not a local minimum, see [18].

At Step 6, a tolerance $\varepsilon_k > 0$ is chosen to check if a sufficient reduction in the norm of the optimality conditions at the candidate iterate w_k^+ has been obtained. An example of choice of ε_k is detailed in Sect. 6. If the residual norm is not smaller than this tolerance, then a sequence of inner iterations is called to compute the new iterate.

The inner iteration algorithm consists of a minimization procedure of the primal-dual merit function defined by

$$\varphi_{\lambda,\rho,\sigma,\nu}(w) = \mathcal{L}_{\rho,\sigma}(x,\lambda) + \tfrac{\nu}{2\sigma}\|c(x) + \sigma(\lambda - y)\|^2,$$

where $\nu > 0$ is a scaling parameter. It is easy to see that w is a stationary point of this function if and only if $\Phi(w,\lambda,\rho,\sigma) = 0$. The minimization procedure is a backtracking line search algorithm quite similar to [7, Algorithm 2], and we refer the interested reader to this paper for a full description of this algorithm. The only difference is that in our description of Algorithm 1, the quadratic parameter σ_{k+1} is kept constant during the inner iterations, while in [7] it can be increased. This choice has no impact from a theoretical point of view and simplifies the presentation of the algorithm. In our numerical experiments, the value of the quadratic penalty parameter is also kept constant during the inner iterations.

4 Global Convergence Analysis

The global convergence of the inner iteration algorithm has been studied in [7, Theorem 2.3]. It has been shown that if the function f is bounded from below and if the gradient of the constraints and the regularized Hessian of the Lagrangian stay bounded during the inner iterations, then the iterate w_{k+1} can be computed in a finite number of inner iterations. In view of this result, it will be assumed that the inner iteration algorithm succeeds in a finite number of iterations in finding w_{k+1} each time it is called at Step 6 of Algorithm 1.

Theorem 4.1 *Assume that Algorithm 1 generates an infinite sequence $\{w_k\}$. Assume also that the sequence $\{(g_k, A_k)\}$ is bounded. In any case, the iterates approach feasible or infeasible stationarity of problem (P_1). More precisely, let $\mathcal{K} \subset \mathbb{N}$ be the set of iteration indices for which the condition (3) is satisfied. Then, at least one of the following situations occurs.*

(i) *If $\mathcal{K}$ is infinite, then the subsequence $\{c_k\}_{\mathcal{K}}$ tends to zero. In addition, if $\{y_k\}_{\mathcal{K}}$ is bounded, then the sequence $\{(g_k, A_k)\}$ has a limit point $(\bar{g}, \bar{A})$ such that $\bar{g}+\bar{A}\bar{y} = 0$ for some $\bar{y} \in \mathbb{R}^m$. If $\{y_k\}_{\mathcal{K}}$ is unbounded, then $\{A_k\}$ has a limit point $\bar{A}$ which is rank deficient.*

(ii) *If $\mathcal{K}$ is finite, then $\{\|c_k\|\}$ is bounded away from zero and $\{A_k c_k\}$ tends to zero.*

Proof First note that the convergence to zero of the sequence $\{\rho_k g_k + A_k y_k\}$ is a direct consequence of Step 6 of Algorithm 1.

Let us prove outcome (i). Lemma 3.1-(i) implies that $\lim_{\mathcal{K}} c_k = 0$ and $\{\rho_k\}$ is eventually constant. If $\{y_k\}_{\mathcal{K}}$ is bounded, then the assumptions imply that the whole sequence $\{(g_k, A_k, y_k/\rho_k)\}_{\mathcal{K}}$ is bounded and so has a limit point $(\bar{g}, \bar{A}, \bar{y})$ such that $\bar{g} + \bar{A}\bar{y} = 0$, which proves the first part of outcome (i). Suppose now that $\{y_k\}_{\mathcal{K}}$ is unbounded. There exists $\mathcal{K}' \subset \mathcal{K}$ such that $y_k \neq 0$ for all $k \in \mathcal{K}'$ and $\lim_{\mathcal{K}'} \|y_k\| = \infty$. For $k \in \mathcal{K}'$, we have

$$\|A_k u_k\| \leq \tfrac{1}{\|y_k\|}\|\rho_k g_k + A_k y_k\| + \tfrac{\rho_k}{\|y_k\|}\|g_k\|,$$

where $u_k = y_k/\|y_k\|$. Since $\{(A_k, u_k)\}_{\mathcal{K}'}$ is bounded, it has a limit point $(\bar{A}, \bar{u})$, with $\bar{u} \neq 0$. By taking the limit in the previous inequality, knowing that the two terms of the right-hand side tend to zero, we deduce that $\bar{A}\bar{u} = 0$, which proves the second part of outcome (i).

For outcome (ii), suppose that $\mathcal{K}$ is finite. Lemma 3.1-(ii) implies that $\{\|c_k\|\}$ is bounded away from zero and $\{\sigma_k \rho_k, \sigma_k \lambda_k\}$ tends to zero. For all $k \in \mathbb{N}$, we have

$$A_k c_k = A_k(c_k + \sigma_k(\lambda_k - y_k)) - \sigma_k A_k \lambda_k + \sigma_k(\rho_k g_k + A_k y_k) - \sigma_k \rho_k g_k.$$

By taking the norm on both sides, for all k we have

$$\begin{aligned} &\|A_k c_k\| \\ &\quad \leq \|A_k\| \|c_k + \sigma_k(\lambda_k - y_k)\| + \sigma_k \|A_k\| \|\lambda_k\| + \sigma_k \|\rho_k g_k + A_k y_k\| + \sigma_k \rho_k \|g_k\| \\ &\quad \leq \max\{\|A_k\|, \sigma_k, \|g_k\|\}(2\|\Phi(w_k, \lambda_k, y_k, \sigma_k)\| + \sigma_k \|\lambda_k\| + \sigma_k \rho_k). \end{aligned}$$

Since the first term of the right-hand side of this inequality is bounded above and all the terms in the parentheses tend to zero, we have $\lim A_k c_k = 0$, which concludes the proof. □

To sum up, the next result shows the behavior of the algorithm when the sequence of primal iterates remains bounded, a usual and mild assumption in a global convergence analysis.

Theorem 4.2 *Assume that Algorithm 1 generates an infinite sequence $\{w_k\}$ such that the sequence $\{x_k\}$ lies in a compact set.*

(i) *Any feasible limit point of the sequence $\{x_k\}$ is a Fritz John point of problem (P_1).*

(ii) *If the sequence $\{x_k\}$ has no feasible limit point, then any limit point is an infeasible stationary point of problem (P_1).*

Proof The compactness assumption implies that the sequences $\{g_k\}$ and $\{A_k\}$ are bounded and so Theorem 4.1 applies.

Let $\bar{x}$ be a limit point of $\{x_k\}$ such that $\bar{c} = 0$. From Lemma 3.1-(ii), we have that the condition (3) is satisfied an infinite number of times. It follows from Lemma 3.1-(i) that there exists $k_0 \in \mathbb{N}$ such that for all $k \geq k_0$, $\rho_k = \rho_{k_0}$. Let $\mathcal{J} \subset \mathbb{N}$ such that the subsequence $\{x_k\}_{\mathcal{J}}$ tends to $\bar{x}$. Step 6 of Algorithm 1 implies that the sequence $\{\rho_{k_0} g_k + A_k y_k\}$ tends to zero. Dividing by $\|(\rho_{k_0}, y_k)\|$ and because $\rho_{k_0} \neq 0$, we have

$$\lim_{\substack{k \to \infty \\ k \in \mathcal{J}}} \frac{\rho_{k_0} g_k + A_k y_k}{\|(\rho_{k_0}, y_k)\|} = 0.$$

By compactness, the sequence $\{(\rho_{k_0}, y_k)/\|(\rho_{k_0}, y_k)\|\}_{\mathcal{J}}$ has a limit point $(\bar{\rho}, \bar{y})$, such that $\|(\bar{\rho}, \bar{y})\| = 1$ and $\bar{\rho}\bar{g} + \bar{A}\bar{y} = 0$, which proves assertion (i).

Suppose now that $\{x_k\}$ has no feasible limit point. From Lemma 3.1-(i), we have that the condition (3) is only satisfied a finite number of times. Theorem 4.1-(ii) implies that $\bar{A}\bar{c} = 0$ for any limit point $\bar{x}$ of $\{x_k\}$, which proves assertion (ii). □

5 Asymptotic Analysis

We have to distinguish two cases for the asymptotic analysis. The first one is when the sequence $\{w_k\}$ converges to a primal-dual solution of the problem. In this case, because $\{c_k\}$ converges to zero, the feasibility parameter becomes constant after a finite number of iterations and the algorithm is reduced to Algorithm 1 in [7] applied to the solution of problem (P_ρ) with a fixed value of ρ. It has been shown that under the classical assumptions of linear independence constraint qualification and second-order sufficient conditions at the optimal limit point, a suitable choice of the parameters allows to get a superlinear or quadratic rate of convergence of $\{w_k\}$, see [7, Theorems 4.4 and 4.5]. The second case to analyze is when the sequence $\{x_k\}$ converges to an infeasible stationary point. This is what we will develop in detail in this section.

The first assumption is that the sequence of iterates converges to an infeasible stationary point.

Assumption 1 *Algorithm 1 generates an infinite sequence $\{w_k\}$ which converges to $w^* = (x^*, y^*) \in \mathbb{R}^{n+m}$, where x^* is an infeasible stationary point of problem (P_1).*

This assumption is very usual for the analysis of the rate of convergence of a numerical optimization algorithm. Note that it is equivalent to assume that $\{x_k\}$ converges to an infeasible stationary point x^* and that the algorithm always stays in the feasibility detection phase, i.e., $\mathtt{F} = \mathtt{true}$ for all the iterations. Indeed, by Lemma 3.1-(i), the convergence of $\{x_k\}$ to an infeasible stationary point x^* implies that the condition (3) is satisfied for a finite number of iterations. In that case, by Lemma 3.1-(ii), the sequence $\{\sigma_k \lambda_k\}$ tends to zero. Step 3 of Algorithm 1 implies that if $\mathtt{F} = \mathtt{true}$ at any iteration, then $\{\sigma_k\}$ is eventually constant and on the contrary, if $\mathtt{F} = \mathtt{false}$ at some iteration, then $\{\sigma_k\} \to 0$. Since $\{c_k + \sigma_k(\lambda_k - y_k)\}$ tends to zero, the sequence $\{\sigma_k y_k\}$ tends to $c^* \neq 0$ and thus $\{y_k\}$ has a finite limit if and only if $\mathtt{F}$ keeps the value $\mathtt{true}$ at all the iterations. This indicates that the choice of the value of the feasibility tolerance ϵ is an important issue relatively to the behavior of the algorithm. In practice, ϵ is chosen equal to, or smaller than, the stopping tolerance of the overall algorithm.

Lemma 5.1 *Under Assumption 1, the inequality (3) is satisfied a finite number of times, the sequence $\{\rho_k\}$ converges to zero, $\{\sigma_k\}$ is eventually constant, and $\|\lambda_k\| = O(\rho_k)$.*

Proof Assumption 1 implies that $\{c_k\}$ converges to a nonzero value. Therefore, by virtue of Lemma 3.1-(i), the inequality (3) is satisfied only a finite number of times. It follows that Step 3 of Algorithm 1 is always executed for k sufficiently large and that $\mathtt{F} = \mathtt{true}$ for all iterations. Indeed, for all $k \in \mathbb{N}$ we have

$$\|c_k\| \leq \|c_k + \sigma_k(\lambda_k - y_k)\| + \sigma_k\|\lambda_k\| + \sigma_k\|y_k\|.$$

Step 6 and Lemma 3.1-(ii) imply that the first two terms of the right-hand side of the inequality tend to zero. Because $\{y_k\}$ is supposed to be convergent, we deduce that the sequence $\{\sigma_k\}$ does not converge to zero, which implies that $\mathtt{F} = \mathtt{true}$ for all iterations. Therefore, there exists $k_0 \in \mathbb{N}$ such that for all $k \geq k_0$, $\rho_k \leq \tau^{k-k_0}\rho_{k_0}$, $\sigma_k = \sigma_{k_0}$ and $\lambda_k/\rho_k = \lambda_{k_0}/\rho_{k_0}$, the conclusion follows. □

Let $\sigma > 0$ be the limit value of $\{\sigma_k\}$. For $w = (x, y) \in \mathbb{R}^{n+m}$, let us define

$$F(w) = \begin{pmatrix} A(x)y \\ c(x) - \sigma y \end{pmatrix}. \tag{6}$$

We have $\lim \Phi(w_k, \lambda_k, \rho_k, \sigma_k) = \Phi(w^*, 0, 0, \sigma) = F(w^*)$, and therefore $y^* = \frac{1}{\sigma} c^*$.

Assumption 2 *The functions f and c are twice continuously differentiable and their second derivatives are Lipschitz continuous over an open neighborhood of x^*.*

The Hessian matrix of the function $\frac{1}{2}\|c\|^2$ is defined as

$$C := \textstyle\sum_i c_i \nabla^2 c_i + AA^\top.$$

Assumption 3 *The sufficient second-order optimality conditions hold at x^* for the feasibility problem (1), i.e., the matrix C^* is positive definite.*

The following lemma is a direct consequence of these assumptions.

Lemma 5.2 *Under Assumptions 2 and 3, there exist a neighborhood W of w^*, $M > 0$, $L > 0$ and $0 < a_1 \le a_2$ such that for all $w, w' \in W$ we have*

(i) $||F'(w)^{-1}|| \le M$,
(ii) $||F(w') - F(w) - F'(w)(w' - w)\| \le \frac{L}{2}\|w - w'\|^2$,
(iii) $a_1\|w - w'\| \le \|F(w) - F(w')\| \le a_2\|w - w'\|$.

Proof To prove (i), it suffices to show that $F'(w^*)$ is nonsingular. By using the fact that $y^* = \frac{1}{\sigma} c^*$, we have

$$F'(w^*) = \begin{pmatrix} \frac{1}{\sigma}\sum_i c_i^* \nabla^2 c_i^* & A^* \\ A^{*\top} & -\sigma I \end{pmatrix}.$$

It is well known that $F'(w^*)$ is nonsingular if and only if the matrix $\frac{1}{\sigma} C^*$, the Schur complement of $-\sigma I$ of the matrix $F'(w^*)$, is positive definite (see e.g., [19, Lemma 4.1]). Thus, item (i) follows from Assumption 3. Assumption 2 implies that F' is Lipschitz continuous on W with the Lipschitz constant L. Property (ii) then follows from the Lipschitz continuity of F' and from [20, Lemma 4.1.12]. The assertion (iii) follows from [20, Lemma 4.1.16]. □

The next lemma shows that the matrix J_k used at Step 5 of Algorithm 1 is a good approximation of the Jacobian matrix of F at w_k when the feasibility parameter goes to zero.

Lemma 5.3 *Under Assumptions 1–3, there exists $\beta > 0$ such that for all $k \in \mathbb{N}$ large enough,*

$$\|J_k - F_k'\| \le \beta\rho_{k+1} \quad \textit{and} \quad \|J_k^{-1}\| \le 2M,$$

where M is defined by Lemma 5.2.

Proof From the definition of J_k, for all $k \in \mathbb{N}$ we have

$$\|J_k - F_k'\| = \|\rho_{k+1}\nabla^2 f_k + \theta_k I\|.$$

Since $\{x_k\}$ converges to x^* and f is assumed to be twice continuously differentiable in a neighborhood of x^*, the first inequality will be proved if we show that $\theta_k = 0$ for k large enough. This is the case if $\mathrm{In}(J_k) = (n, m, 0)$ or, equivalently, if $K_{\rho_{k+1},\sigma,0}(w_k)$ is positive definite for k large enough. For all $k \in \mathbb{N}$, we have

$$\begin{aligned} K_{\rho_{k+1},\sigma,0}(w_k) &= H_{\rho_{k+1},0}(w_k) + \tfrac{1}{\sigma} A_k A_k^\top \\ &= \tfrac{1}{\sigma} C^* + \tfrac{1}{\sigma}(C_k - C^*) + H_{\rho_{k+1},0}(x_k, y_k - \tfrac{1}{\sigma} c_k). \end{aligned}$$

By assumption, C^* is positive definite and the two other matrices tend to zero when k tends to infinity. It follows that $K_{\rho_{k+1},\sigma,0}(w_k)$ is positive definite for k large enough, first inequality.

Using Lemma 5.2-(i), the inequality just proved and the fact that $\{\rho_k\}$ tends to zero, for k large enough we have

$$\begin{aligned} \|F_k'^{-1}(J_k - F_k')\| &\leq \|F_k'^{-1}\| \|J_k - F_k'\| \\ &\leq M\beta\rho_{k+1} \\ &\leq \tfrac{1}{2}. \end{aligned}$$

It then suffices to apply [20, Theorem 3.1.4] to prove the second inequality.

The last lemma gives an estimate of the distance of the Newton iterate w_k^+ to the solution w^*.

Lemma 5.4 *Assume that Assumptions 1–3 hold. The sequence of iterates generated by Algorithm 1 satisfies*

$$\|w_k^+ - w^*\| = O(\|w_k - w^*\|^2) + O(\rho_{k+1}).$$

Proof Let $k \in \mathbb{N}$. From the definition of the trial iterate w_k^+ at Step 5 of Algorithm 1, we have

$$\begin{aligned} w_k^+ - w^* &= w_k - w^* - J_k^{-1}\Phi(w_k, \lambda_{k+1}, \rho_{k+1}, \sigma) \\ &= J_k^{-1}\big((J_k - F_k')(w_k - w^*) + F_k'(w_k - w^*) - F_k \\ &\quad + F_k - \Phi(w_k, \lambda_{k+1}, \rho_{k+1}, \sigma)\big). \end{aligned}$$

By using $F^* = 0$, by taking the norm on both sides, then by applying Lemma 5.3, Lemma 5.2-(ii), finally by using the convergence of $\{w_k\}$ to w^*, the boundedness of $\{g_k\}$ and $\|\lambda_k\| = O(\rho_k)$ from Lemma 5.1, we obtain

$$\begin{aligned} &\|w_k^+ - w^*\| \\ &\leq \|J_k^{-1}\|\big(\|J_k - F_k'\|\|w_k - w^*\| + \|F^* - F_k - F_k'(w^* - w_k)\| \\ &\quad + \|F_k - \Phi(w_k, \lambda_{k+1}, \rho_{k+1}, \sigma)\|\big) \\ &\leq 2M\big(\beta\rho_{k+1}\|w_k - w^*\| + \tfrac{L}{2}\|w_k - w^*\|^2 + \rho_{k+1}\|g_k\| + \sigma\|\lambda_{k+1}\|\big) \\ &= O(\rho_{k+1}) + O(\|w_k - w^*\|^2), \end{aligned}$$

which concludes the proof. □

We now state the main result of this section. This theorem shows the rapid rate of convergence of Algorithm 1 in the infeasible case. In addition, under a suitable choice of the parameters, there is no need of inner iterations for k large enough. In this case, the cost of the one iteration of the algorithm is reduced to the solution of the linear system at Step 5, which can be done with $O((n+m)^3)$ arithmetic operations.

Theorem 5.1 *Assume that Assumptions 1–3 hold. Let $t \in]0,2]$. If the feasibility parameter of Algorithm 1 is chosen so that $\rho_{k+1} = O(\|F_k\|^t)$, then*

$$\|w_{k+1} - w^*\| = O(\|w_k - w^*\|^t). \tag{7}$$

In addition, if $\rho_{k+1} = \Theta(\|F_k\|^t)$ and if $\varepsilon_k = \Omega(\rho_k^{t'})$ for $0 < t' < t$, then for k large enough there is no inner iterations, i.e., $w_{k+1} = w_k^+$.

Proof The assumption on the value of ρ_{k+1} and the Lipschitz property of F from Lemma 5.2-(iii) imply that

$$\rho_{k+1} = O(\|w_k - w^*\|^t). \tag{8}$$

Using this estimate in Lemma 5.4, we deduce that

$$\|w_k^+ - w^*\| = O(\|w_k - w^*\|^t). \tag{9}$$

At Step 6 of Algorithm 1, we have either $w_{k+1} = w_k^+$ or w_{k+1} is computed by means of the inner iterations. In the first case, it is clear that (7) follows from (9). Suppose now that the second case holds, i.e., the inequality (4) is not satisfied at iteration k. We then have

$$\|\Phi(w_{k+1}, \lambda_{k+1}, \rho_{k+1}, \sigma)\| \le \varepsilon_k < \|\Phi(w_k^+, \lambda_{k+1}, \rho_{k+1}, \sigma)\|. \tag{10}$$

From (9), the sequence $\{w_k^+\}$ tends to w^*, and therefore $\{g_k^+\}$ is bounded. Using the second inequality of Lemma 5.2-(iii) and Lemma 5.1, then (8) and (9), we deduce that

$$\begin{aligned}\|\Phi(w_k^+, \lambda_{k+1}, \rho_{k+1}, \sigma)\| &\le \|F_k^+ - F^*\| + \rho_{k+1}\|g_k^+\| + \sigma\|\lambda_{k+1}\| \\ &= O(\|w_k^+ - w^*\|) + O(\rho_{k+1}) \\ &= O(\|w_k - w^*\|^t). \end{aligned} \tag{11}$$

Combining (10) and(11), we obtain

$$\|\Phi(w_{k+1}, \lambda_{k+1}, \rho_{k+1}, \sigma)\| = O(\|w_k - w^*\|^t).$$

Finally, from the first inequality of Lemma 5.2-(iii), the last estimate, the boundedness of $\{g_k\}$, Lemma 5.1 and the estimate (8), we have

$$\begin{aligned} a_1\|w_{k+1}-w^*\| &\le \|F_{k+1}-F^*\| \\ &= \|F_{k+1}\| \\ &\le \|\Phi(w_{k+1},\lambda_{k+1},\rho_{k+1},\sigma)\| + \rho_{k+1}\|g_{k+1}\| + \sigma\|\lambda_{k+1}\| \\ &= O(\|w_k-w^*\|^t) + O(\rho_{k+1}) \\ &= O(\|w_k-w^*\|^t), \end{aligned}$$

which proves (7).

Let us now prove the second assertion of the theorem. On the one hand, Lemma 5.2-(iii) and (7) imply that $\|F_{k+1}\| = O(\|F_k\|^t)$. By assumption, we have $\rho_{k+1} = \Theta(\|F_k\|^t)$, and thus $\rho_{k+1} = O(\rho_k^t)$. Since $t' < t$, we then have $\rho_{k+1} = o(\rho_k^{t'})$. On the other hand, the estimate (11) implies that

$$\|\Phi(w_k^+,\lambda_{k+1},\rho_{k+1},\sigma)\| = O(\|F_k\|^t) = O(\rho_{k+1}).$$

By assumption, $\varepsilon_k = \Omega(\rho_k^{t'})$; therefore, for k large enough, the inequality (4) is satisfied. □

6 Numerical Experiments

Our algorithm is called SPDOPT-ID (strongly primal-dual optimization with infeasibility detection) and has been implemented in C. The performances of SPDOPT-ID are compared with those of SPDOPT-AL [7] on a set of 130 *standard* problems from the CUTEr collection [21]. The selected problems are those with equality constraints and a solution time less than 300 s. To create a second set of 130 *infeasible* problems, the constraint $c_1^2 + 1 = 0$, where c_1 is the first component of c, has been added to each problem. Note that the addition of this new constraint leads to a twofold difficulty. Indeed, not only the constraints are infeasible, but their gradients are linearly dependent.

We also compare the condition used to update the parameters in Step 2 of Algorithm 1, with the one used in [7, Algorithm 1]. The algorithm called SPDOPT-IDOld is Algorithm 1, but with the inequality (3) which is replaced by

$$\|c_k\| \le a\max\{\|c_{i_j}\| + \zeta_{i_j} : (k-\ell)_+ \le j \le k\}. \tag{12}$$

We will show that this modification is of importance when solving an infeasible problem and that the use of (3) in place of (12) leads to better numerical performances.

The feasibility parameter is initially set to $\rho_0 = 1$. When $\texttt{F} = \texttt{true}$, the feasibility parameter in Step 3 is updated by the formula

$$\rho_{k+1} = \min\{0.2\rho_k, 0.2\|F_k\|^2, 1/(k+1)\}.$$

The assumption on ρ_{k+1} in the statements of Theorem 5.1 is satisfied with $t = 2$. The rate of convergence of $\{w_k\}$ to w^* is then quadratic. A lower bound of 10^{-16} is applied on this parameter.

The parameters σ_k and θ_k are updated at Step 4 and Step 5 as in [7, Algorithm 1].

To be able to solve a quadratic problem in only one iteration, we adopt the same procedure as in [22] for the choice of the starting point. Let $\bar{w} = (\bar{x}, \bar{y})$, where $\bar{x}$ is the default starting point and $\bar{y} = (1, \ldots, 1)^\top$. Initially, the following linear system $J_{1,0,0}(\bar{w})d = -\Phi(\bar{w}, \bar{y}, 1, 0)$ is solved. If the inequality $\|\Phi(\bar{w} + d, 0, 1, 0)\|_\infty \leq \|\Phi(\bar{w}, 0, 1, 0)\|_\infty$ is satisfied, then $w_0 = \bar{w} + d$, otherwise $w_0 = \bar{w}$.

The algorithm is terminated and an optimal solution is declared to be found if $\|(g_k + A_k y_k/\rho_k, c_k)\|_\infty \leq \varepsilon_{\text{tol}}$ with $\varepsilon_{\text{tol}} = 10^{-8}$. Otherwise, the algorithm returns a notification that an infeasible stationary point has been found if $\rho_k \leq \varepsilon_{\text{tol}}$, $\|c_k\| > \varepsilon_{\text{tol}}$ and $\|\Phi(w_k, 0, 0, \sigma_k)\|_\infty \leq \varepsilon_{\text{tol}}$. For SPDOPT-AL, we also add the stopping condition $\|c_k\| > \varepsilon_{\text{tol}}$, $\|A_k c_k\| \leq \varepsilon_{\text{tol}}$ and $\sigma_k \leq \varepsilon_{\text{tol}}$ to terminate this algorithm at an infeasible stationary point.

As mentioned in Sect. 5, the feasibility tolerance at Step 1 is set to $\epsilon = \varepsilon_{\text{tol}}$, to get a fast local convergence when the algorithm converges to an infeasible stationary point.

At Step 2 of Algorithm 1, we choose $a = 0.9$, $\ell = 2$ and $\zeta_k = 10\sigma_k\rho_k$ for all iteration k.

The sequence of tolerance $\{\varepsilon_k\}$ in Step 6 is defined by the following formula:

$$\varepsilon_k = 0.9 \max\{\|\Phi(w_i, \lambda_i, \rho_i, \sigma_i)\| : (k-4)_+ \leq i \leq k\} + \zeta_k.$$

The convergence to zero of the sequence $\{\varepsilon_k\}$ is a consequence of [22, Proposition 1]. This choice meets the requirements to get a fast convergence both in the feasible case, i.e., $\varepsilon_k = \Omega(\sigma_{k+1})$, and in the infeasible case, i.e., $\varepsilon_k = \Omega(\rho_k^{t'})$, with $t' = 1$.

The symmetric indefinite factorization code MA57 [23] is used to factorize and regularize the matrix J_k. Since this factorization reveals the inertia of the matrix, the correction parameter θ_k, initially set to zero, is increased and a new factorization is performed until the inertia of J_k has the correct value.

The maximum number of iterations, counting both the inner and the outer iterations, is limited to 3000.

For the standard problems, only 129 problems solved by at least one of three algorithms are selected for the comparison purpose (problem `dixchlng` has not been solved). Figure 1 shows the performance profiles of Dolan and Moré [24] on the numbers of function and gradient evaluations. For $\tau \geq 0$, $\rho_s(\tau)$ is the fraction of test problems for which the performance of the solver s is within a factor 2^τ of the best one. These profiles show that the performances of the three algorithms are very similar, the difference is not significant. In terms of robustness, the three algorithms solve successfully the same number of problems (128 problems). We can conclude that the infeasibility detection does not reduce the performances of the algorithm for solving standard problems.

Figure 2 shows the performances of these algorithms in terms of numbers of function and gradient evaluations on a set of 126 infeasible problems (the problems `gilbert, hager3, porous1, porous2` have been eliminated since three algorithms can-

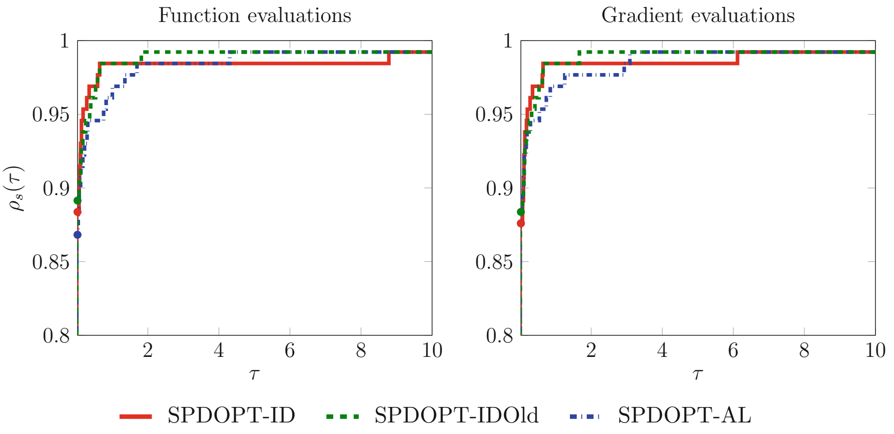

Fig. 1 Performance profiles comparing the three algorithms on the set of standard problems

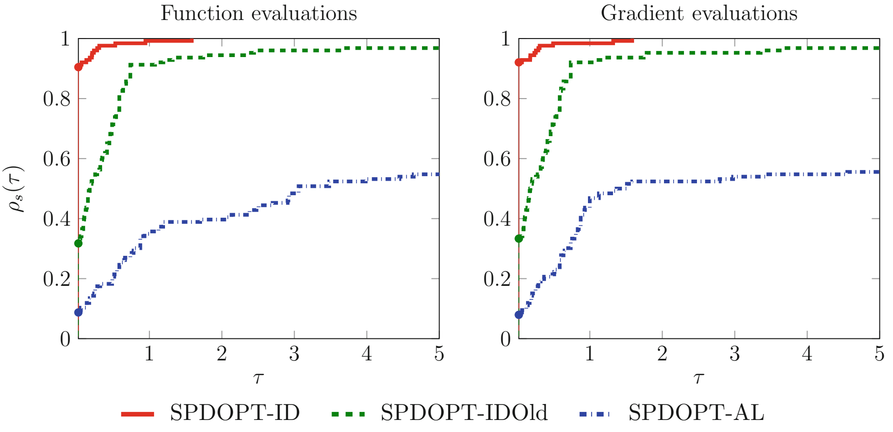

Fig. 2 Performance profiles comparing the three algorithms on the set of infeasible problems

not detect the infeasibility). We observe that SPDOPT-ID is the most efficient algorithm for detecting infeasible problems, with an efficiency rate of approximately 90%. In any case, the efficiency of SPDOPT-ID and SPDOPT-IDOld is very significant comparing to SPDOPT-AL. In terms of robustness, our two algorithms are more robust than SPDOPT-AL since they can detect more than 95% of problems, whereas SPDOPT-AL only detects less than 60%. This figure also shows that SPDOPT-ID is better comparing to SPDOPT-IDOld, justifying the choice of new criterion for updating parameters.

We conclude this section by a comparison of a numerical estimate of the rate of convergence of the new algorithm SPDOPT-ID and of the original one SPDOPT-AL, when the sequence of iterates converges to an infeasible stationary point. We used a graphical representation inspired by [14]. We selected a set of 58 problems among the collection of infeasible problems, for which both algorithms generate a sequence converging to an infeasible stationary point. Figure 3 shows the last ten values of $\|A_k c_k\|$ for SPDOPT-AL and of $\|F_k\|$ for SPDOPT-ID. We cannot plot

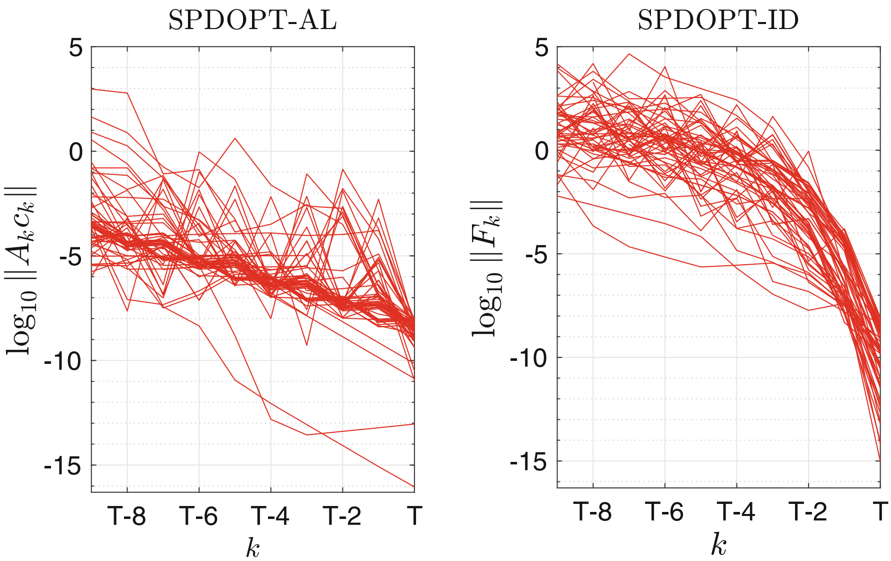

Fig. 3 Values of $\log_{10}\|A_k c_k\|$ and $\log_{10}\|F_k\|$ for the last ten iterations of SPDOPT-AL and SPDOPT-ID. T represents the index of the stopping iteration for each run

the values $\|F_k\|$ for SPDOPT-AL, because when the sequence of iterates converges to an infeasible stationary point, $\{\sigma_k\}$ goes to zero and $\{y_k\}$ becomes unbounded. Under some regularity assumptions, we obviously have $\|A_k c_k\| = \Theta(\|x_k - x^*\|)$ and $\|F_k\| = \Theta(\|w_k - w^*\|)$. These curves empirically show that there is a true improvement in the rate of convergence of the algorithm, from linear to quadratic.

For the solution of these infeasible problems, we observed that for the last four outer iterations, there is no inner iterations (i.e., $w_{k+1} = w_k^+$) for 90% of the problems. This percentage is more than 93% if one considers the last three outer iterations. For the infeasible problems for which SPDOPT-ID uses inner iterations at the last outer iterations, either the algorithm does not terminate successfully or the quadratic convergence is not observed. These observations confirm the asymptotic property of our algorithm.

7 Conclusions

During the solution of an optimization problem, a rapid detection of the infeasibility is a difficult task. In the framework of an augmented Lagrangian method, we have proposed to add a new parameter, which has to scale down the objective function when the infeasibility measure is not sufficiently reduced. The global and local convergence analyses, as well as the numerical results, show that this method is reliable for both solving a feasible optimization problem and quickly detecting the infeasibility.

The original algorithm SPDOPT-AL [7] uses only one penalty parameter σ_k to balance between optimality and feasibility as in a classical augmented Lagrangian algorithm. According to [7, Theorem 3.4], when the problem is infeasible, any limit point of $\{x_k\}$ is an infeasible stationary point, but the sequence $\{y_k\}$ becomes unbounded. For this classical scheme, we did not find a way to update the penalty parameter to get a fast rate of convergence to an infeasible stationary point. In practice, we only observed a linear rate of convergence, as shown in Fig. 3. The introduction of the feasibility parameter ρ_k allows to balance between the solution of the first-order optimality conditions of the original problem (P_1) and the ones of the feasibility problem (1). By this way, we see that taking $\sigma_k \to 0$ in the classical scheme is not equivalent to taking $\rho_k \to 0$ and keeping σ_k constant in the new scheme.

Another advantage of our new scheme is that in case of convergence to an infeasible stationary point, when the quadratic penalty parameter σ_k is kept constant, it becomes a regularization term for the matrix $F'(w)$. For this reason, the asymptotic analysis near an infeasible stationary point does not require a constraint qualification like some other works in the literature, see e.g., [13, Assumption 3.2-(b)] and [14, Assumption 4.19-(a)].

The quadratic convergence to an infeasible stationary point is an original result for an augmented Lagrangian algorithm. However, the fast local convergence is only guaranteed under the assumption that the infeasible stationary point, the limit point of the sequence of iterates, is sufficiently infeasible, i.e., $\|c^*\| > \epsilon$, for some $\epsilon > 0$. On some examples, we observed that the algorithm converges with a linear rate to an infeasible stationary point, because the norm of the constraints evaluated at this point is very close to the stopping tolerance. Hence, an open question is to design an algorithm with rapid convergence in the infeasible case, regardless the norm of the constraints at the limit point.

Another natural question is the extension of this approach to the solution of general optimization problem with equality and inequality constraints. One possibility is to introduce slack variables to inequality constraints and apply augmented Lagrangian method in the case of simple bounds as in [4,25]. On the other hand, we note that infeasibility detection has been used in the framework of interior point methods [26, 27]. However, these works did not report complete global and local convergence analyses of their methods, despite good numerical results. To our knowledge, there is no local convergence result for nonlinear interior points methods in the infeasible case. To extend our approach, a possibility is to combine an augmented Lagrangian method and a log-barrier penalty to handle inequalities as in [28]. But the introduction of a feasibility parameter makes the convergence analysis quite different. Indeed, in that case the feasibility parameter becomes a scaling factor between the log-barrier function and the quadratic penalty term; therefore, when the sequence of iterates converges to an infeasible stationary point, this sequence follows a path of solutions parameterized by the log-barrier penalty parameter, see [29].

References

1. Hestenes, M.R.: Multiplier and gradient methods. J. Optim. Theory Appl. **4**, 303–320 (1969)
2. Powell, M.J.D.: A method for nonlinear constraints in minimization problems. In: Optimization (Symposium of University Keele, Keele, 1968), pp. 283–298. Academic Press, London (1969)
3. Conn, A.R., Gould, N.I.M., Toint, P.L.: LANCELOT, Springer Series in Computational Mathematics, vol. 17. Springer, Berlin (1992). A Fortran package for large-scale nonlinear optimization (release A)

4. Andreani, R., Birgin, E.G., Martínez, J.M., Schuverdt, M.L.: On augmented Lagrangian methods with general lower-level constraints. SIAM J. Optim. **18**(4), 1286–1309 (2007)
5. Andreani, R., Birgin, E.G., Martínez, J.M., Schuverdt, M.L.: Augmented Lagrangian methods under the constant positive linear dependence constraint qualification. Math. Program. **111**(1–2), 5–32 (2008)
6. Birgin, E.G., Martínez, J.M.: Practical augmented Lagrangian methods for constrained optimization, Fundamentals of Algorithms, vol. 10. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2014)
7. Armand, P., Omheni, R.: A globally and quadratically convergent primal-dual augmented Lagrangian algorithm for equality constrained optimization. Optim. Methods Softw. **32**(1), 1–21 (2017)
8. Martínez, J.M., Prudente, L.D.F.: Handling infeasibility in a large-scale nonlinear optimization algorithm. Numer. Algorithms **60**(2), 263–277 (2012)
9. Birgin, E.G., Martínez, J.M., Prudente, L.F.: Augmented Lagrangians with possible infeasibility and finite termination for global nonlinear programming. J. Glob. Optim. **58**(2), 207–242 (2014)
10. Birgin, E.G., Martínez, J.M., Prudente, L.F.: Optimality properties of an augmented Lagrangian method on infeasible problems. Comput. Optim. Appl. **60**(3), 609–631 (2015)
11. Birgin, E.G., Floudas, C.A., Martínez, J.M.: Global minimization using an augmented Lagrangian method with variable lower-level constraints. Math. Program. **125**(1), 139–162 (2010)
12. Gonçalves, M.L.N., Melo, J.G., Prudente, L.F.: Augmented Lagrangian methods for nonlinear programming with possible infeasibility. J. Glob. Optim. **63**(2), 297–318 (2015)
13. Byrd, R.H., Curtis, F.E., Nocedal, J.: Infeasibility detection and SQP methods for nonlinear optimization. SIAM J. Optim. **20**(5), 2281–2299 (2010)
14. Burke, J.V., Curtis, F.E., Wang, H.: A sequential quadratic optimization algorithm with rapid infeasibility detection. SIAM J. Optim. **24**(2), 839–872 (2014)
15. Bertsekas, D.P.: Constrained optimization and Lagrange multiplier methods. In: Computer Science and Applied Mathematics. Academic Press, Inc. [Harcourt Brace Jovanovich, Publishers], New York (1982)
16. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer Series in Operations Research and Financial Engineering. Springer, New York (2006)
17. Forsgren, A., Gill, P.E., Wright, M.H.: Interior methods for nonlinear optimization. SIAM Rev. **44**(4), 525–597 (2002). (2003)
18. Armand, P., Benoist, J., Orban, D.: From global to local convergence of interior methods for nonlinear optimization. Optim. Methods Softw. **28**(5), 1051–1080 (2013)
19. Forsgren, A., Gill, P.E.: Primal-dual interior methods for nonconvex nonlinear programming. SIAM J. Optim. **8**(4), 1132–1152 (1998)
20. Dennis Jr., J.E., Schnabel, R.B.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Classics in Applied Mathematics, vol. 16. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1996)
21. Gould, N.I.M., Orban, D., Toint, P.L.: Cuter and sifdec: a constrained and unconstrained testing environment, revisited. ACM Trans. Math. Softw. **29**(4), 373–394 (2003)
22. Armand, P., Benoist, J., Omheni, R., Pateloup, V.: Study of a primal-dual algorithm for equality constrained minimization. Comput. Optim. Appl. **59**(3), 405–433 (2014)
23. 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)
24. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program. **91**(2), 201–213 (2002)
25. Birgin, E.G., Bueno, L.F., Martínez, J.M.: Sequential equality-constrained optimization for nonlinear programming. Comput. Optim. Appl. **65**(3), 699–721 (2016)
26. Curtis, F.E.: A penalty-interior-point algorithm for nonlinear constrained optimization. Math. Program. Comput. **4**(2), 181–209 (2012)
27. Nocedal, J., Öztoprak, F., Waltz, R.A.: An interior point method for nonlinear programming with infeasibility detection capabilities. Optim. Methods Softw. **29**(4), 837–854 (2014)
28. Armand, P., Omheni, R.: A mixed logarithmic barrier-augmented Lagrangian method for nonlinear optimization. J. Optim. Theory Appl. **173**(2), 523–547 (2017)
29. Armand, P., Tran, N.N.: Rapid infeasibility detection in a mixed logarithmic barrier augmented Lagrangian method for nonlinear optimization. Optim. Methods Softw. (2018). https://doi.org/10.1080/10556788.2018.1528250