Optimization
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/gopt20

Interior point filter method for semi-infinite programming problems

Ana I. Pereira [a], M. Fernanda P. Costa [b] & Edite M.G.P. Fernandes [c]

[a] Department of Mathematics - ESTiG, Polytechnic Institute of Bragança, 5301-857 Bragança, Portugal

[b] Department of Mathematics and Applications, University of Minho, 4800-058 Guimarães, Portugal

[c] Algoritmi R&D Center, University of Minho, 4710-057 Braga, Portugal

Available online: 04 Oct 2011

To cite this article: Ana I. Pereira, M. Fernanda P. Costa & Edite M.G.P. Fernandes (2011): Interior point filter method for semi-infinite programming problems, Optimization, 60:10-11, 1309-1338

To link to this article: http://dx.doi.org/10.1080/02331934.2011.616894

Optimization
Vol. 60, Nos. 10–11, October–November 2011, 1309–1338

Interior point filter method for semi-infinite programming problems

Ana I. Pereira[a]*, M. Fernanda P. Costa[b] and Edite M.G.P. Fernandes[c]

[a]*Department of Mathematics – ESTiG, Polytechnic Institute of Bragança, 5301-857 Bragança, Portugal;* [b]*Department of Mathematics and Applications, University of Minho, 4800-058 Guimarães, Portugal;* [c]*Algoritmi R&D Center, University of Minho, 4710-057 Braga, Portugal*

(*Received 31 August 2009; final version received 17 August 2011*)

Semi-infinite programming (SIP) problems can be efficiently solved by reduction-type methods. Here, we present a new reduction method for SIP, where the multi-local optimization is carried out with a stretched simulated annealing algorithm, the reduced (finite) problem is approximately solved by a Newton's primal–dual interior point method that uses a novel two-dimensional filter line search strategy to guarantee the convergence to a KKT point that is a minimizer, and the global convergence of the overall reduction method is promoted through the implementation of a classical two-dimensional filter line search. Numerical experiments with a set of well-known problems are shown.

Keywords: nonlinear optimization; semi-infinite programming; interior point; filter method; line search

AMS Subject Classifications: 90C30; 90C34; 90C51

1. Introduction

A reduction-type method based on a primal–dual interior point filter method for nonlinear semi-infinite programming (SIP) is proposed. To allow convergence from poor starting points, a backtracking line search filter strategy is implemented. The SIP problem is considered to be of the form

$$\min f(x) \text{ subject to } g(x,t) \leq 0, \quad \text{for every } t \in T, \tag{P}$$

where $T \subseteq \mathbb{R}^m$ is a nonempty set defined by $T = \{t \in \mathbb{R}^m : v(t) \leq 0\}$. Here, we assume that the set T does not depend on x. If the set T depends on x, the problem is called as generalized semi-infinite programming problem [31,32,38].

The nonlinear functions $f: \mathbb{R}^n \to \mathbb{R}$ and $g: \mathbb{R}^n \times T \to \mathbb{R}$ are twice continuously differentiable with respect to x, and g is continuously differentiable function with respect to t.

There are many problems in the engineering area that can be formulated as SIP problems. Approximation theory [15], optimal control [10], mechanical stress of

*Corresponding author. Email: apereira@ipb.pt

ISSN 0233–1934 print/ISSN 1029–4945 online

http://dx.doi.org/10.1080/02331934.2011.616894
http://www.tandfonline.com

materials and computer-aided design [43], air pollution control [37], robot trajectory planning [36], financial mathematics and computational biology and medicine [42] are some examples. For a review of other applications, the reader is referred to [7,15,25,28,38].

The numerical methods that are mostly used to solve SIP problems generate a sequence of finite problems. There are three main ways of generating the sequence: by discretization, exchange and reduction methods [10,25,36]. Methods that solve the SIP problem on the basis of the KKT system derived from the problem are emerging in the literature [12–14,23,24,29,30,43].

This work aims to describe a reduction method for SIP. Conceptually, the method is based on the local reduction theory. The focus of our proposal is on an interior point filter line search-based method, to compute an approximation to the finite optimization problem that emerges in a global reduction algorithm context, for solving a nonlinear SIP problem. The global convergence analysis of the interior point filter algorithm to a KKT point that is a minimizer is also included. This work comes in the sequence of a previous penalty-based reduction-type method presented in [21]. We have been observing that, when solving SIP problems with this type of methods, the performance depends strongly on the user-defined penalty parameters that are present in penalty functions when solving the reduced finite problem, and in merit functions when promoting the overall algorithm global convergence.

Here, we aim to simplify the solution method while solving the reduced problem, as well as to improve efficiency in terms of number of iterations required by the overall algorithm. In the new algorithm, the multi-local procedure uses the stretched simulated annealing method [21]. To solve the reduced finite optimization problem, we propose a Newton's primal–dual interior point method that uses a novel two-dimensional filter line search to guarantee the convergence to a KKT point, that is a minimizer. It is shown that every limit point of the sequence of iterates generated by the algorithm is feasible and satisfies the complementarity condition. We also show that there exists at least a limit point that is a KKT point of the problem. Finally, to promote convergence from any initial approximation, a two-dimensional filter methodology, as proposed in [6], is also incorporated into the reduction algorithm.

This article is organized as follows. In Section 2, we present the basic ideas behind the local reduction to finite problems. Section 3 is briefly devoted to the multi-local procedure and Section 4 contains a detailed description of the herein proposed primal–dual interior point filter line search method for solving the reduced optimization problem. The global convergence analysis of the algorithm is also included. Section 5 presents the filter methodology to promote global convergence of the reduction algorithm and Section 6 lists the conditions for its termination. Finally, Section 7 contains some numerical results and conclusions.

2. First-order optimality conditions and reduction method

In this section we present some definitions and the optimality conditions of problem (P). We denote the *feasible set* of problem (P) by X, where

$$X = \left\{x \in \mathbb{R}^n : g(x, t) \leq 0, \text{ for every } t \in T\right\}.$$

A feasible point $\bar{x} \in X$ is called a *strict local minimizer* of problem (P) if there exists a positive value ϵ such that

$$\forall x \in X : f(x) - f(\bar{x}) > 0 \wedge \|x - \bar{x}\| < \epsilon \wedge x \neq \bar{x},$$

where $\|\cdot\|$ represents the euclidean norm. For $\bar{x} \in X$, the *active index set*, $T_0(\bar{x})$, is defined by

$$T_0(\bar{x}) = \{t \in T : g(\bar{x}, t) = 0\}.$$

We first assume the following.

ASSUMPTION 2.1 *Let $\bar{x} \in X$. The linear independence constraint qualification* (*LICQ*) *holds at $\bar{x}$, i.e. $\{\nabla_x g(\bar{x}, t),\ t \in T_0(\bar{x})\}$ is a linearly independent set.*

Since LICQ implies the *Mangasarian–Fromovitz constraint qualification* (MFCQ) [15], we can conclude that for $\bar{x} \in X$ there exists a vector $d \in \mathbb{R}^n$ such that for every $t \in T_0(\bar{x})$ the condition $\nabla_x g(\bar{x}, t)^T d < 0$ is satisfied. A direction d that satisfies this condition is called a *strictly feasible direction*. Further, the vector $d \in \mathbb{R}^n$ is a *strictly feasible descent direction* if the following conditions hold:

$$\nabla f(\bar{x})^T d < 0, \quad \nabla_x g(\bar{x}, t)^T d < 0, \quad \text{for every } t \in T_0(\bar{x}). \tag{1}$$

If $\bar{x} \in X$ is a local minimizer of the problem (P) then there will not exist a strictly feasible descent direction $d \in \mathbb{R}^n \setminus \{0_n\}$, where 0_n represents the null vector of $\mathbb{R}^n$.

THEOREM 2.2 [15] *Let $\bar{x} \in X$. Suppose that there is no direction $d \in \mathbb{R}^n \setminus \{0_n\}$ satisfying*

$$\nabla f(\bar{x})^T d \leq 0 \quad \textit{and} \quad \nabla_x g(\bar{x}, t)^T d \leq 0, \quad \textit{for every } t \in T_0(\bar{x}).$$

Then $\bar{x}$ is a strict local minimizer of SIP.

Since Assumption 2.1 is verified, the set $T_0(\bar{x})$ is finite. Suppose that $T_0(\bar{x}) = \{t_1, \ldots, t_p\}$, then $p \leq n$. If $\bar{x}$ is a local minimizer of problem (P) and if the MFCQ holds at $\bar{x}$, then there exist nonnegative values λ_l for $l = 1, \ldots, p$ such that

$$\nabla f(\bar{x}) + \sum_{l=1}^{p} \lambda_l \nabla_x g(\bar{x}, t_l) = 0_n. \tag{2}$$

This is the Karush–Kuhn–Tucker (KKT) condition of problem (P).

Many papers exist in the literature devoted to the reduction theory [1,2,9,10,22,25,33]. The main idea is to describe, locally, the feasible set of the problem (P) by a finite set of constraints. Assume that $\bar{x}$ is a feasible point and that each $t_l \in \bar{T} \equiv T(\bar{x})$ is a local maximizer of the so-called *lower level problem*

$$\max_{t \in T} g(\bar{x}, t), \tag{3}$$

satisfying the following condition:

$$\left| g(\bar{x}, t_l) - g^* \right| \leq \delta^{ML}, \quad l = 1, \ldots, \bar{L}, \tag{4}$$

where $\bar{L} \geq p$ and represents the cardinality of $\bar{T}$, δ^{ML} is a positive constant and g^* is the global solution value of (3).

ASSUMPTION 2.3 *For any fixed $\bar{x} \in X$, each $t_l \in \bar{T}$ is an strict local maximizer, i.e.*

$$\exists \delta > 0, \quad \forall t \in T : g(\bar{x}, t_l) > g(\bar{x}, t) \wedge \|t - t_l\| < \delta \wedge t \neq t_l.$$

When the set T is compact, $\bar{x}$ is a feasible point and Assumption 2.3 holds, there exists a finite number of local maximizers of the problem (3) and the implicit function theorem can be applied, under some constraint qualifications [15]. So, it is possible to conclude that there exist open neighbourhoods $\bar{U}$, of $\bar{x}$, and V_l, of t_l, and implicit functions $t_1(x), \ldots, t_{\bar{L}}(x)$ defined as

(i) $t_l : \bar{U} \to V_l \cap T$, for $l = 1, \ldots, \bar{L}$;
(ii) $t_l(\bar{x}) = t_l$, for $l = 1, \ldots, \bar{L}$;
(iii) $\forall x \in \bar{U}$, $t_l(x)$ is a non-degenerate and strict local maximizer of the problem (3); so that

$$\{x \in \bar{U} : g(x, t) \leq 0, \text{ for every } t \in T\} \Leftrightarrow \{x \in \bar{U} : g(x, t_l(x)) \leq 0, l = 1, \ldots, \bar{L}\}.$$

So it is possible to replace the infinite set of constraints by a finite set that is locally sufficient to define the feasible region. Thus the problem (P) is locally equivalent to the so-called *reduced (finite) optimization problem*

$$\min_{x \in \bar{U}} f(x) \text{ subject to } g_l(x) \equiv g(x, t_l(x)) \leq 0, \quad l = 1, \ldots, \bar{L} \tag{5}$$

where $\bar{U}$ is an open neighbourhood of $\bar{x}$ and $t_l(x)$, $l = 1, \ldots, \bar{L}$, are implicitly defined functions satisfying (ii) and (iii) above.

A reduction method then emerges when any method for finite programming is applied to solve the locally reduced problem (5). Conceptually, the reduction method resumes to an iterative process. Thus, at each iteration, indexed by k, the Algorithm 2.1 shows the main procedures of the proposed reduction method.

Algorithm 2.1 (Global reduction algorithm)

For $k = 1, 2, \ldots$ an approximation x^k to the SIP problem is required:

(1) Based on x^k, compute the set T^k, solving problem (3), with condition (4).
(2) Based on the set T^k, implement at most $i^{\max}$ iterations to get an approximation $x^{k,i}$, by solving the reduced problem (5).
(3) Use a globalization technique to compute a new approximation x^{k+1} that improves significantly over x^k.
(4) Use termination criteria to decide if the iterative process should terminate.

The remaining part of this article presents our proposals for the four steps of the global reduction algorithm, Algorithm 2.1, for SIP. An algorithm to compute the set T^k is known in the literature as a multi-local procedure. In this article, a stretched simulated annealing algorithm that has been previously applied in other reduction-type methods is used (see [18–21] for details). To solve the reduced problem (5), a Newton's primal–dual interior point method is proposed. Further, a filter line search technique is incorporated into the interior point algorithm in order to promote global convergence, whatever may be the initial approximation. Each entry in the filter is composed by two components that measure the KKT error. As explained later in this article, the components come naturally from the KKT conditions for the reduced problem (5). The first component measures feasibility and complementarity, and the second measures optimality. Additionally to enforce progress towards a minimizer, and whenever a descent direction for the logarithmic barrier function (from the barrier problem associated with the reduced problem to be solved) is computed, a sufficient reduction is also imposed on that function by means of an Armijo condition. This is the main contribution of this article. As far as we know, a primal–dual interior point filter line search technique has not been applied in a nonlinear SIP context. Furthermore, the proposed two-dimensional filter line search strategy is a novelty in the field of interior point methods. The global convergence analysis of the interior point algorithm is also included. Finally, convergence of the overall reduction method to an SIP solution is encouraged by implementing a filter line search technique. The filter here aims to measure sufficient progress by using the constraint violation and the objective function value. This filter strategy has been shown to behave well for SIP problems when compared with merit function approaches [18,19].

3. The multi-local procedure

The multi-local procedure is used to compute the set T^k, i.e. the local solutions of the problem (3) that satisfy (4). Some procedures to find the local maximizers of the constraint function consist of two phases: first, a discretization of the set T is made and all maximizers are evaluated on that finite set; second, a local method is applied in order to increase the accuracy of the approximations found in the first phase (e.g. [2]). Our proposal combines the function stretching technique, proposed in [17], with a simulated annealing (SA)-type algorithm – the ASA variant of the SA in [11]. This is a stochastic point-to-point global search method that generates the elements of T^k sequentially.

The function stretching technique was initially proposed in [17], in a particle swarm optimization algorithm context, to provide a way to escape from a local solution, driving the search to a global one. When a local (non-global) solution, $\hat{t}$, is detected, this technique reduces by a certain amount the objective function values at all points t that verify $g(t) \leq g(\hat{t})$, and maintains the function values for all t such that $g(t) > g(\hat{t})$. The process is repeated until the global solution is encountered.

Since we need to compute global as well as local solutions, the inclusion of the function stretching technique aims to prevent the convergence of the ASA algorithm to an already detected solution. Let t_1 be the first computed global solution. The function stretching technique is applied, only locally, in order to transform $g(t)$ in a neighbourhood of t_1, say $V_{\varepsilon_1}(t_1)$ ($\varepsilon_1 > 0$), decreasing the function values on the region $V_{\varepsilon_1}(t_1)$, leaving all the other maxima unchanged. That particular maximum, $g(t_1)$, disappears although all the other maxima are left unchanged. The ASA algorithm is then applied to the modified objective function to detect a global solution of the new problem. This iterative process terminates if no other solution is found for a set of $\mathcal{K}^{ML}$ consecutive iterations (see [18,21] for details).

4. Finite optimization procedure

The sequential quadratic programming method is the mostly used finite programming procedure in reduction-type methods for solving SIP problems. Usually, they use the L_1 and L_∞ merit functions and rely on a trust region framework to ensure global convergence (see, e.g. [2,22,33]). Penalty methods with exponential and hyperbolic penalty functions have already been tested with some success [19,21]. However, to solve finite inequality constrained optimization problems, Newton's primal–dual interior point methods [5,26,27,34,35] and primal–dual barrier methods [5,39–41] have shown to be competitive and even more robust than sequential quadratic programming and penalty-type methods. This is the motivation of this work. Incorporating a primal–dual interior point method into a reduction-type method for nonlinear SIP is the main goal to improve the efficiency over previous reduction methods. The global convergence analysis of the proposed algorithm is also included.

For simplicity, we now consider the special case of SIP problems where $T = \{t \in \mathbb{R}^m : a \le t \le b\}$. To solve the reduced (finite) problem (5), we propose the use of an infeasible primal–dual interior point (P-DIP) method. We remark that this infeasible version of the method corresponds to the reformulation of the reduced problem (5) into an equivalent problem, where the unique inequality constraints are simple nonnegativity constraints. So, in this methodology, the first step is to introduce slack variables to replace all inequality constraints by equality constraints and simple nonnegativity constraints. Hence, adding nonnegative slack variables $w = (w_0, w_1, \ldots, w_{L^k+1})^T$ to the inequality constraints, the problem (5) is rewritten as follows:

$$\min_{x \in U^k \subset \mathbb{R}^n,\, w \in \mathbb{R}^{L^k+2}} f(x) \text{ subject to } g_l(x) + w_l = 0, \quad l = 0, \ldots, L^k + 1\,,\; w_l \ge 0, \tag{6}$$

where $g_0(x) = g(x, a)$ and $g_{L^k+1}(x) = g(x, b)$ correspond to the values of the constraint function $g(x, t)$ at the lower and upper limits of set T. The KKT conditions for problem (6) are

$$\begin{aligned} \nabla_x L(x, w, z, y) &= \nabla f(x) + \nabla g(x) z = 0 \\ \nabla_w L(x, w, z, y) &= z - y = 0 \\ \nabla_z L(x, w, z, y) &= g(x) + w = 0 \\ w^T y &= 0 \\ w \ge 0,&\ \ y \ge 0, \end{aligned} \tag{7}$$

where $L(x, w, z, y) = f(x) + (g(x) + w)^T z - w^T y$ is the Lagrangian function of the problem (6) and z and y are the Lagrange multiplier vectors associated with constraints $g(x) + w = 0$ and $w \ge 0$, respectively. From the second equation in (7), $z = y$, meaning that, at the solution, the multipliers associated with the equality constraints $g(x) + w = 0$ are the same as the multipliers associated with the constraints $w \ge 0$, and replacing yields

$$F(x, w, y) \equiv \begin{cases} \nabla f(x) + \nabla g(x)\, y = 0 \\ g(x) + w = 0 \\ WYe = 0 \end{cases} \tag{8}$$
$$w \ge 0,\ \ y \ge 0,$$

where $W = \text{diag}(w_0, \dots, w_{L^k+1})$ and $Y = \text{diag}(y_0, \dots, y_{L^k+1})$ are diagonal matrices and $e \in \mathbb{R}^{L^k+2}$ is a vector of ones.

In this method, the term 'infeasible' refers to the fact that dual and primal feasibilities (first and second equations in (8)) are not required at the beginning although they are enforced throughout the iterative process. The term 'primal–dual' refers to the fact that the Lagrange multipliers y are treated as independent variables in all calculations, as we do with the primal variables x and w. The term 'interior point' is used since the slack variables, w, and the dual variables, y, are required to satisfy the bounds in (8) strictly at the beginning and throughout the iterative process. By satisfying these bounds, the method avoids spurious solutions, i.e. points that satisfy the first three equations in (8) but not $w \geq 0$ and $y \geq 0$.

When Newton's method is applied to the system (8) to get the search directions Δx, Δw and Δy, it deals with the linearized complementarity equation

$$Y\Delta w + W\Delta y = -WYe$$

that may cause a serious problem. This equation forces the iterate to stick to the boundary of the feasible region once it approaches that boundary. This means that if the component l of the current iterate, w_l^i, becomes zero and $y_l^i > 0$, it will remain zero in all iterations after i. The same is true for the components of the vector y^i. This drawback is solved by modifying the Newton formulation so that zero variables become nonzero in subsequent iterations. This is done replacing the complementarity equation $WYe = 0$ by the perturbed complementarity $WYe = \widehat{\mu}e$, where $\widehat{\mu} > 0$. The following perturbed KKT conditions then appear:

$$F_{\hat{\mu}}(x, w, y) \equiv \begin{cases} \nabla f(x) + \nabla g(x)\, y = 0 \\ WYe - \widehat{\mu}e = 0 \\ g(x) + w = 0 \end{cases} \\ w \geq 0,\ \ y \geq 0. \tag{9}$$

We note that the perturbed KKT conditions for problem (6), given by (9), are equivalent to the KKT conditions of the barrier problem associated with problem (6), in the sense that they have the same solutions,

$$\begin{array}{ll} \min\limits_{x \in U^k \subset \mathbb{R}^n,\, w \in \mathbb{R}^{L^k+2}} & \varphi_{\hat{\mu}}(x, w) \equiv f(x) - \widehat{\mu} \sum\limits_{l=0}^{L^k+1} \ln(w_l) \\ \text{subject to} & g(x) + w = 0, \end{array} \tag{10}$$

where $\varphi_{\hat{\mu}}(x, w)$ is the logarithmic barrier function, for a fixed $\widehat{\mu} > 0$ [5].

Applying Newton's method to the system $F_{\hat{\mu}}(x, w, y) = 0$ in (9), we obtain a linear system to compute the search directions Δx, Δw, Δy

$$\begin{bmatrix} H(x, y) & 0 & \nabla g(x) \\ 0 & Y & W \\ \nabla g(x)^T & I & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta w \\ \Delta y \end{bmatrix} = -\begin{bmatrix} \nabla f(x) + \nabla g(x)\, y \\ WYe - \widehat{\mu}e \\ g(x) + w \end{bmatrix}, \tag{11}$$

where $H(x, y) = \nabla^2 f(x) + \sum_{l=0}^{L^k+1} y_l \nabla^2 g_l(x)$ is the Hessian of the Lagrangian. This system is not symmetric, but is easily symmetrized by multiplying the second equation by W^{-1}, yielding

$$\begin{bmatrix} H(x,y) & 0 & \nabla g(x) \\ 0 & W^{-1}Y & I \\ \nabla g(x)^T & I & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta w \\ \Delta y \end{bmatrix} = - \begin{bmatrix} \sigma \\ W^{-1}\gamma_{\hat{\mu}} \\ \rho \end{bmatrix}, \tag{12}$$

where we define

$$\begin{gathered} \sigma \equiv \sigma(x,y) = \nabla f(x) + \nabla g(x)\, y, \quad \gamma_{\hat{\mu}} \equiv \gamma_{\hat{\mu}}(w,y) = WYe - \widehat{\mu} e \\ \rho \equiv \rho(x,w) = g(x) + w. \end{gathered} \tag{13}$$

Let $N(x,w,y) = H(x,y) + \nabla g(x) W^{-1} Y \nabla g(x)^T$ denote the dual normal matrix.

THEOREM 4.1 *If N is nonsingular, then* (12) *has a unique solution. In particular,*

$$\begin{aligned} \Delta x &= -N^{-1}\nabla f(x) - \widehat{\mu} N^{-1}\nabla g(x) W^{-1} e - N^{-1}\nabla g(x) W^{-1} Y \rho \\ \Delta w &= \nabla g(x)^T N^{-1} \nabla f(x) + \widehat{\mu} \nabla g(x)^T N^{-1} \nabla g(x) W^{-1} e \\ &\quad - \left(I - \nabla g(x)^T N^{-1} \nabla g(x) W^{-1} Y\right)\rho. \end{aligned}$$

The proof of these results is left for the Appendix of this article.

It is known that if the initial approximation is close enough to the solution, methods based on the Newton's iteration converge quadratically under appropriate assumptions. For poor initial points, a backtracking line search can be implemented to promote convergence to the solution of problem (6) [16]. After the search directions have been computed, the idea is to choose $\alpha^i \in (0, \alpha^i_{\max}]$, at iteration i, so that $x^{k,i+1} = x^{k,i} + \alpha^i \Delta x^i$, $w^{k,i+1} = w^{k,i} + \alpha^i \Delta w^i$ and $y^{k,i+1} = y^{k,i} + \alpha^i \Delta y^i$ improve over a primal–dual estimate solution $(x^{k,i}, w^{k,i}, y^{k,i})$ for problem (6). The index i represents the iteration counter of this inner cycle. The parameter $\alpha^i_{\max}$ represents the longest step size that can be taken along the direction to maintain w^i and y^i strictly positive. Thus the maximal step size $\alpha^i_{\max} \in (0,1]$ is defined by

$$\alpha^i_{\max} = \max\{\alpha \in (0,1] : w^i + \alpha \Delta w^i \geq (1-\tau) w^i,\ y^i + \alpha \Delta y^i \geq (1-\tau)\, y^i\} \tag{14}$$

for a fixed parameter $\tau \in (0,1)$ (close to one).

The strategy to recover $\widehat{\mu}$ at each iteration considers a fraction of the average complementarity

$$\widehat{\mu} = \delta_c \mu, \quad \text{with } \mu = \frac{w^T y}{L^k + 2}, \tag{15}$$

where $\delta_c \in (0,1)$ is a centring parameter. This choice of $\widehat{\mu}$ allows a dynamic reduction of μ (see inequality (23) in Lemma 4.10).

To measure the progress, a penalty or merit function could be used. Although most penalty or merit functions are defined as a linear combination of the objective function and a measure of the constraint violation, there are others that also depend on the Lagrange multiplier vectors, and there is a particular choice that is defined as the squared l_2-norm of the residual vector $F(x,w,y)$ [5]. In the latter, the algorithm may be more likely to converge to stationary points that are not local minimizers. In general, penalty or merit functions depend on a penalty parameter. Unfortunately, a suitable value for the penalty parameter depends on the optimal solution of the problem. Thus, the choice of the penalty parameter is a difficult issue. On the other hand, Fletcher and Leyffer [6] proposed a filter method as an alternative to a merit function to guarantee global convergence in algorithms for nonlinear optimization. This technique incorporates the concept of nondominance to build a filter that is able to accept iterates if they improve either the objective function or the constraint violation, instead of a linear combination of those two measures. So the filter replaces the use of merit functions, avoiding the update of penalty parameters. The filter technique has already been adapted to interior point methods [3,4,27,34,39–41]. Different filters with two- and three-dimensional entries have been proposed in this context.

4.1. *A novel two-dimensional filter line search approach*

The herein proposed two-dimensional filter line search strategy aims to reduce the residual vector $F(x, w, y)$ and to enforce progress towards a KKT point that is a minimizer. According to the KKT conditions (8), each entry in the filter is herein defined by

$$\theta(x, w, y) = \|\rho\|_2 + \|\gamma_0\|_2 \quad \text{and} \quad \theta_o(x, y) = \|\sigma\|_2^2. \tag{16}$$

The first entry in the filter, $\theta(x, w, y)$, aims to measure feasibility and complementarity of each trial iterate, while the other, $\theta_o(x, y)$, measures optimality. Recall that $\gamma_0 \equiv WYe$, according to a previous definition (13). Convergence to KKT points that are minimizers will be guaranteed when sufficient reductions on θ_o as well as on the logarithmic barrier function, $\varphi_{\hat{\mu}}$, are also imposed, as shown later on in this section.

Borrowing the ideas presented in [39–41], our filter is defined as a set $\overline{F}_i$ that contains values of θ and θ_o that are prohibited for a successful iterate in iteration i. At the beginning of the iterative process, the filter is initialized to

$$\overline{F}_0 \subseteq \left\{(\theta, \theta_o) \in \mathbb{R}^2 : \theta \geq \theta^{\max} > 0,\ \theta_o \geq \theta_o^{\max} > 0\right\}.$$

After computing the search directions $\Delta^i = (\Delta x^i, \Delta w^i, \Delta y^i)$ by (12), a new iterate $u^{k,i+1} = u^{k,i} + \alpha^i \Delta^i$, where $u = (x, w, y)$, might be acceptable if the following condition, known as 'acceptance condition', holds:

$$\theta^{i+1} \leq (1 - \gamma_1)\,\theta^i \quad \text{or} \quad \theta_o^{i+1} \leq \theta_o^i - \gamma_2\,\theta^i, \tag{AC-r}$$

where $\gamma_1, \gamma_2 \in (0, 1)$. For simplicity, the following notation is used $\theta^{i+1} = \theta(u^{k,i+1})$ and $\theta_o^{i+1} = \theta_o(x^{k,i+1}, y^{k,i+1})$. To prevent the convergence to a point that is feasible and satisfies the complementarity condition but is nonoptimal, another set of conditions must be satisfied for a trial point $u^{k,i+1}$ to be acceptable. If $\theta^i \leq \theta^{\min}$ (for $\theta^{\min} > 0$) and the 'switching conditions'

$$\begin{array}{c} m_1^i(\alpha) < 0, \quad m_2^i(\alpha) < 0 \quad \text{and} \\ -m_1^i(\alpha) > \delta\left[\theta^i\right]^r \quad \text{and} \quad -m_2^i(\alpha) > \delta\left[\theta^i\right]^r \end{array} \tag{SC-r}$$

hold, where

$$\begin{array}{ll} m_1^i(\alpha) = \alpha^i\, \nabla\varphi_{\hat{\mu}}(x^i, w^i)^T \Delta^{1,i}, & \text{with } \Delta^{1,i} = (\Delta x^i, \Delta w^i), \\ m_2^i(\alpha) = \alpha^i\, \nabla\theta_o(x^i, y^i)^T \Delta^{2,i}, & \text{with } \Delta^{2,i} = (\Delta x^i, \Delta y^i) \end{array} \tag{17}$$

and $\varphi_{\hat{\mu}}(x, w)$ is the barrier function (10), then $u^{k,i+1}$ is acceptable if sufficient reductions on $\varphi_{\hat{\mu}}$ and θ_o, according to the 'Armijo condition',

$$\varphi_{\hat{\mu}}^{i+1} \leq \varphi_{\hat{\mu}}^{i} + \eta\, m_1^i(\alpha) \quad \text{and} \quad \theta_o^{i+1} \leq \theta_o^i + \eta\, m_2^i(\alpha) \tag{A-r}$$

are verified, for $\delta > 0$, $r > 1$, $\eta \in (0, 0.5)$. Besides requiring sufficient decrease with respect to the current iterate, the trial point $u^{k,i+1}$ is accepted only if it is acceptable by the current filter.

We remark that the point $u^{k,i+1}$ – or the corresponding pair $\left(\theta^{i+1}, \theta_o^{i+1}\right)$ – is acceptable to the filter, i.e.

$$\left(\theta^{i+1}, \theta_o^{i+1}\right) \notin \overline{F}_i$$

only if $\theta^{i+1} < \theta$ or $\theta_o^{i+1} < \theta_o$ for all (θ, θ_o) in the current filter $\overline{F}_i$. This means that the point is not dominated by any other point in the filter.

We note that using conditions (A-r) progress towards a KKT point that is a minimizer would be guaranteed. Under mild conditions, as stated later on in Theorem 4.2, the direction $\Delta^{1,i}$ is a descent direction for the logarithmic barrier function $\varphi_{\hat{\mu}}$, at a feasible point, and $\Delta^{2,i}$ is a descent direction for θ_o. The algorithm updates the filter using the update formula

$$\overline{F}_{i+1} \quad = \overline{F}_i \cup \left\{(\theta, \theta_o) \in \mathbb{R}^2 : \theta \geq (1 - \gamma_1)\theta^i \ \text{ and } \ \theta_o \geq \theta_o^i - \gamma_2 \theta^i\right\} \tag{18}$$

if the acceptable iterate satisfies (AC-r), and maintains the filter unchanged if the acceptable iterate satisfies (SC-r) and (A-r).

If the described backtracking line search cannot find an acceptable $\alpha^i \geq \alpha_{\min} > 0$, a restoration phase is implemented in order to find a new iterate $u^{k,i+1}$ that is acceptable by the filter, by imposing a sufficient reduction in both measures (see (13)): feasibility $\theta_f = \frac{1}{2}\|\rho\|_2^2$ and centrality $\theta_c = \frac{1}{2}\left\|\gamma_{\hat{\mu}}\right\|_2^2$,

$$\theta_f^{i+1} \leq \theta_f^i + \eta\,\alpha^i\,(\nabla\theta_f^i)^T \Delta^{1,i} \quad \text{and} \quad \theta_c^{i+1} \leq \theta_c^i + \eta\,\alpha^i\,(\nabla\theta_c^i)^T \Delta^{3,i},$$

respectively, where $\Delta^{3,i} = (\Delta w^i, \Delta y^i)$. We remark that the search directions Δ^1 and Δ^3, computed from (12), are descent directions for θ_f and θ_c, respectively (see Theorem 4.2). Algorithm 4.1 shows the main steps of the proposed interior point filter line search technique.

Algorithm 4.1 (Primal-dual interior point filter line search algorithm)

Given $x^{k,0}$, $w^{k,0} > 0$, $y^{k,0} > 0$, $\gamma_1, \gamma_2, \delta, r, \eta, \theta^{\max}, \theta_o^{\max}, \theta^{\min}, \alpha_{\min}$, δ_c, τ, δ_1^{IP}, δ_2^{IP}, τ_s, $i^{\max}$.

(1) Initialize the filter and set $i = 0$.
(2) Stop if termination criteria are met.
(3) Based on current point u^i, compute $\widehat{\mu}^i$, compute search direction Δ^i and $\alpha_{\max}^i$.
(4) Set $\alpha^i = \alpha_{\max}^i$. Compute trial point $u^{i+1} = u^i + \alpha^i\Delta^i$ using backtracking line search:

(4.1) If $\alpha^i < \alpha_{\min}$ go to Step 7.
(4.2) If $u^i + \alpha^i\Delta^i \in \overline{F}_i$ go to Step 4.5.

(4.3) If (SC-r) and (A-r) hold, go to Step 6.
(4.4) If (AC-r) holds, go to Step 5.
(4.5) Set $\alpha^i = \alpha^i/2$ and go to Step 4.1.

(5) Augment the filter using (18).
(6) Set $i = i + 1$ and go back to Step 2.
(7) Use the restoration phase to produce a point u^{i+1} that is acceptable to the filter. Augment the filter and continue to Step 6.

The following theorem shows that the search direction Δ^1 defined by (12) is a descent direction for the barrier function $\varphi_{\hat{\mu}}$ whenever the problem is strictly convex. Furthermore, Δ^2 is a descent direction for θ_o, Δ^1 is a descent direction for θ_f and Δ^3 is a descent direction for θ_c.

THEOREM 4.2 *The search directions generated by Algorithm* 4.1 *satisfy*:

(i) *If the matrix* N *is positive definite and* $\rho = 0$, *then*

$$(\nabla \varphi_{\hat{\mu}})^T \Delta^1 \leq 0.$$

(ii) *Furthermore*

$$(\nabla \theta_o)^T \Delta^2 \leq 0, \quad (\nabla \theta_f)^T \Delta^1 \leq 0, \quad (\nabla \theta_c)^T \Delta^3 \leq 0.$$

In all cases, equality holds if and only if (x, w) *satisfies* (12) *for some* y.

The proof of these results is left for the Appendix of this article.

4.2. *Algorithm implementation details*

4.2.1. *Termination criteria*

This iterative process terminates at iteration i if

$$\left(\max\left\{ \|\rho^i\|_\infty, \frac{\|\gamma_0^i\|_\infty}{s^i} \right\} \leq \delta_1^{IP} \text{ and } \frac{\|\sigma^i\|_\infty}{s^i} \leq \delta_2^{IP} \right) \quad \text{or} \quad i > i^{\max} \tag{19}$$

is verified, where $s^i = \max\{1, \tau_s \|y^i\|_1/(L^k + 2)\}$, $0 < \tau_s < 1$, for some small positive constants $\delta_1^{IP}, \delta_2^{IP}$ and $i^{\max} > 0$.

4.2.2. *Local adaptation procedure*

The classical definition of a reduction-type method considers $i^{\max} = 1$ to guarantee that the optimal set T^k does not change. When $i^{\max} > 1$, the values of the maximizers might change as $x^{k,i}$ changes along this inner iterative process, even if L^k does not change. Practical implementation of reduction-type methods has shown that efficiency may be improved if $i^{\max} > 1$ and a local adaptation procedure is used [8,21]. Our local adaptation algorithm is very simple and is described below as Algorithm 4.2. This procedure aims to correct the maximizers, if necessary, each time a new approximation is computed, $x^{k,i}$.

Algorithm 4.2 (Local adaptation algorithm)

For $l = 1, \ldots, L^k$

(1) Compute $5m$ random points in the neighbourhood of t_l

$$\widetilde{t}_j = t_l + p\,, j = 1, \ldots, 5m \quad \text{where } p_i \sim U[-0.5, 0.5],\ i = 1, \ldots, m.$$

(2) Identify $\widetilde{t}_l$ (the one with largest g value).
(3) If $g(x^{k,i}, \widetilde{t}_l) > g(x^{k,i}, t_l)$ then replace t_l by $\widetilde{t}_l$.

4.3. *Global convergence of the algorithm to a KKT point*

This part of the section aims at providing a global convergence analysis of the Algorithm 4.1 to a KKT point that is a minimizer. In what follows, we denote the set of indices of those iterations in which the filter has been augmented by $\mathcal{A} \subseteq \mathbb{N}$ and the set of indices of those iterations in which the restoration phase is called by $\mathcal{R} \subseteq \mathbb{N}$. From Step 7 of Algorithm 4.1 we have $\mathcal{R} \subseteq \mathcal{A}$.

We now state the assumptions that are needed to show global convergence to stationary points for this interior point filter line search algorithm. These assumptions are similar to common assumptions on interior point line search methods [39].

Given a starting point $x^{k,0}$, and $w^{k,0} > 0$, $y^{k,0} > 0$, let $\{u^i\}$ (where $u^i \equiv u^{k,i}$ for simplicity and $u = (x, w, y)$) be the sequence generated by Algorithm 4.1, where we assume that the restoration phase in Step 7 always terminates successfully and the algorithm does not stop in Step 2 at a KKT point.

Assumption 4.3 *There exists an open set $\mathcal{C} \subseteq \mathbb{R}^{\overline{n}}$ with $[u^i, u^i + \alpha^i_{\max}\Delta^i] \subseteq \mathcal{C}$ for all i so that $\gamma_{\hat{\mu}}$ is differentiable on $\mathcal{C}$, and f, g, ρ, σ are differentiable on $\mathcal{C}$, and their function values, as well as their first derivatives, are bounded and Lipschitz continuous over $\mathcal{C}$, with $\overline{n} = n + 2(L^k + 2)$.*

Assumption 4.4 *The matrices B^i that approximate the Hessian of the Lagrangian used in* (12) *are uniformly bounded for all i.*

Assumption 4.5 *The matrices $N^i = B^i + \nabla g(x^i)(W^i)^{-1} Y^i \nabla g(x^i)^T$ are uniformly positive definite on the null space of the Jacobian of the active constraints $\nabla g(x^i)_a^T$.*

Assumption 4.6 *There exists a constant $m_g > 0$ so that for all i, $\sigma_{\min}(\nabla g(x^i)_a) \geq m_g$, where $\sigma_{\min}$ denotes the smallest singular value.*

Assumption 4.7 *The sequence $\{u^i\}$ is bounded.*

Assumption 4.8 *At all feasible limit points $(\overline{x}, \overline{w})$ of $\{(x^i, w^i)\}$, the gradients of the active constraints*

$$\nabla g_j(\overline{x}) \quad \text{for } j \in \{l : g_l(\overline{x}) + \overline{w}_l = 0\} \text{ and } e_j \text{ for } j \in \{l : \overline{w}_l = 0\}, \tag{20}$$

are linearly independent.

Assumption 4.9 *There exist constants $\widetilde{\delta}_\theta, \widetilde{\delta}_w > 0$ so that whenever the restoration phase is called in Step 7 in an iteration $i \in \mathcal{R}$ with $\|\theta^i\| \leq \widetilde{\delta}_\theta$, it returns a new iterate with $w_j^{i+1} \geq w_j^i$ for all components satisfying $w_j^i \leq \widetilde{\delta}_w$.*

4.3.1. *Preliminary results*

We now state some preliminary results. The global convergence properties of our algorithm have some similarities with those of [34] and [39]. The first lemma measures the decrease on the complementarity obtained by the new iterate $u^{i+1} = u^i + \alpha^i \Delta^i$ and is needed to prove Lemma 4.15.

LEMMA 4.10 *For* $\alpha \in (0, 1]$ *and all* $l = 0, \ldots, L^k + 1$ *it holds*

$$w_l^{i+1} y_l^{i+1} \leq (1-\alpha) w_l^i y_l^i + \alpha \delta_c \mu^i + (\alpha)^2 \left\| \Delta^{3,i} \right\|^2, \tag{21}$$

$$w_l^{i+1} y_l^{i+1} \geq (1-\alpha) w_l^i y_l^i + \alpha \delta_c \mu^i - (\alpha)^2 \left\| \Delta^{3,i} \right\|^2, \tag{22}$$

$$\mu^{i+1} \leq (1 - \alpha(1-\delta_c)) \mu^i + (\alpha)^2 \left\| \Delta^{3,i} \right\|^2, \tag{23}$$

$$\mu^{i+1} \geq (1 - \alpha(1-\delta_c)) \mu^i - (\alpha)^2 \left\| \Delta^{3,i} \right\|^2. \tag{24}$$

The proof of these results is left for the Appendix of this article.

We now remark that there are two important issues, related to the barrier function, that should be addressed when using this type of interior point method:

(i) $\varphi_{\hat{\mu}}$ is defined only for positive components of w;
(ii) $\varphi_{\hat{\mu}}$ and its derivatives become unbounded if any component of w approaches the boundary.

To address the issue (i) we use (14), and (ii) is addressed by Theorem 4.12 which states that the iterates w^i generated by Algorithm 4.1 are bounded away from the boundary defined by the bound constraints. The following lemma is required for Theorem 4.12.

LEMMA 4.11 *Suppose Assumptions* 4.3–4.9 *hold. Then, for a given subset of indices* $S \subseteq \{0, \ldots, L^k + 1\}$ *and a constant* $\delta_l > 0$, *there exist* $\delta_s, \delta_\rho > 0$ *so that* $\Delta w_l^i > 0$ *for* $l \in S$ *whenever* $i \notin \mathcal{R}$ *and*

$$u^i \in L = \left\{ w^i \geq 0 : \; w_l^i \leq \delta_s \text{ for } l \in S, \; w_l^i \geq \delta_l \text{ for } l \notin S, \; \|\rho^i\| \leq \delta_\rho \right\},$$

i.e. at sufficiently feasible points, the search direction Δw^i *points away from almost active bounds.*

The proof of this lemma is based on the solution of the system (12). Since the perturbed KKT conditions, given by (9), for problem (6), are equivalent to the KKT conditions of the barrier problem associated with problem (6), a proof similar to that of Lemma 11 in [39] applies. We remark that in [39] a barrier interior point method is proposed.

THEOREM 4.12 *Suppose Assumptions* 4.3–4.9 *hold. Then there exists a constant* $\varepsilon_w > 0$ *so that* $w^i \geq \varepsilon_w e$ *for all i.*

Here a proof similar to that of Theorem 3 in [39] applies.

The following lemma proves that the solution of the system (12) is uniformly bounded.

LEMMA 4.13 *Suppose Assumptions* 4.3–4.7 *hold. Then there exists a constant* $M_\Delta > 0$, *such that for all i and* $\alpha \in (0, 1]$

$$\left\|\Delta^i\right\| \leq M_\Delta.$$

Proof From Assumptions 4.3 and 4.7 we have that ρ, σ and $\gamma_{\hat{\mu}}$ are uniformly bounded. Assumptions 4.4–4.6 also guarantee that the inverse of the matrix in (12) exists and is uniformly bounded for all i. Consequently, the solution Δ^i is uniformly bounded. ■

Theorem 4.12 and Lemma 4.13, together with (14), establish that the starting step size in the backtracking line search $\alpha^i_{\max}$ is uniformly bounded away from zero.

The following result shows that the search direction is a direction of sufficient descent for the barrier function at points that are sufficiently close to feasibility and complementarity, but are nonoptimal.

LEMMA 4.14 *Suppose Assumptions* 4.3–4.7 *hold. If* $\{u^{i_j}\}$ *is a subsequence of iterates for which* $\|\sigma^{i_j}\|_2 \geq \epsilon$ *with a constant* $\epsilon > 0$ *independent of j, then there exist constants* ε_1, ε_2, $\varepsilon_3 > 0$, *such that for all j and* $\alpha \in (0, 1]$

$$\theta^{i_j} \leq \varepsilon_1 \Rightarrow m_1^{i_j}(\alpha) \leq -\alpha\varepsilon_2 \tag{25}$$

and

$$m_2^{i_j}(\alpha) \leq -\alpha\varepsilon_3. \tag{26}$$

The proof of this result is given in the Appendix of this article.

The following lemma provides upper bounds for $\|\sigma\|^2$, $\|\rho\|$, $\|\gamma_0\|$ and θ at the new iterate u^{i+1} in terms of $\alpha^i\Delta^i$ and their corresponding values at the current iterate u^i.

LEMMA 4.15 *Suppose Assumptions* 4.3 *and* 4.7 *hold. There exist positive constants* M_{θ_o}, M_ρ, M_θ *depending on the Lipschitz constants of* $\nabla\sigma$ *and* $\nabla\rho$, *and the constants* M_{γ_0}, $M_{\varphi_{\hat{\mu}}} > 0$, *such that, for* $\alpha \in (0, 1]$

$$\theta_o^{i+1} \leq (1-\alpha)\theta_o^i + M_{\theta_o}\alpha^2\left\|\Delta^{2,i}\right\|^2, \tag{27}$$

$$\|\rho^{i+1}\| \leq (1-\alpha)\|\rho^i\| + M_\rho\alpha^2\left\|\Delta^{1,i}\right\|^2, \tag{28}$$

$$\|\gamma_0^{i+1}\| \leq (1-\alpha)\|\gamma_0^i\| + M_{\gamma_0}\alpha^2\left\|\Delta^{3,i}\right\|^2, \tag{29}$$

$$\varphi_{\hat{\mu}}^{i+1} \leq \varphi_{\hat{\mu}}^i + m_1^i(\alpha) + M_{\varphi_{\hat{\mu}}}\alpha^2\left\|\Delta^{1,i}\right\|^2, \tag{30}$$

$$\theta^{i+1} \leq (1-\alpha)\theta^i + M_\theta\alpha^2\left\|\Delta^i\right\|^2. \tag{31}$$

The proof of these results are given in the Appendix of this article.

The following two lemmas are a direct consequence of the structure of the Algorithm 4.1. Lemma 4.16 states that all entries in the filter satisfy $\theta > 0$ and shows that no pair (θ, θ_o) corresponding to a feasible point that satisfies the complementarity condition is ever included in the filter.

LEMMA 4.16 *Suppose Assumptions* 4.3–4.7 *hold. For all i and* $\alpha \in (0, 1]$

$$\theta(u^i) = 0 \Rightarrow m_1^i(\alpha) < 0 \tag{32}$$

and

$$\Theta^i = \min\{\theta : (\theta, \theta_o) \in \overline{F}_i\} > 0. \tag{33}$$

Proof If $\theta(u^i) = 0$ (and $\|\rho^i\| = 0$), we have that $\|\sigma\| > 0$, otherwise Algorithm 4.1 would have been terminated in Step 2. Using (45), it follows that

$$m_1^i(\alpha)/\alpha = (\nabla\varphi_{\hat{\mu}}^i)^T \Delta^{1,i} \leq -c_1 \|\sigma^i\|_2^2 = -c_1 \theta_o^i < 0$$

and (32) holds. The proof of (33) is by induction. From Step 1 of Algorithm 4.1 it follows that the claim is valid for $i = 0$ since $\theta^{\max} > 0$. Suppose the claim is true for *i*. Then, if $\theta^i > 0$ and the filter is augmented at iteration *i*, the update rule (18) implies that $\Theta^{i+1} > 0$, since $\gamma_1 \in (0, 1)$. On the other hand, if $\theta^i = 0$, we have from (32) that $m_1^i(\alpha) < 0$ for all $\alpha \in (0, 1]$ so that the switching conditions (SC-r) are true for all trial step sizes (since $m_2^i(\alpha) < 0$ also holds). Thus, the accepted α^i should satisfy the Armijo conditions in (A-r). In this case, the filter is not augmented and $\Theta^{i+1} = \Theta^i > 0$. ■

The following lemma shows that new iterates are always acceptable to the filter.

LEMMA 4.17 *In all iterations* $i \geq 0$, *the current iterate* u^i *is acceptable to the filter.*

Proof The proof is by induction. Since $\overline{F}$ is empty in iteration $i = 0$, the initial iterate u^0 is acceptable to the filter. Consider $i > 0$, and assume that u^i is acceptable to the filter. According to Algorithm 4.1, u^{i+1} is either generated in Step 4 or in Step 7. If acceptance condition (AC-r) holds, then the filter is augmented and u^{i+1} is added to the filter. When conditions (SC-r) and (A-r) hold, the filter remains unchanged, and u^{i+1} is not added to the filter. By the induction hypothesis, u^i is acceptable to the filter. Since the filter is not changed, the same holds true for u^{i+1}. Finally, if u^{i+1} is obtained in Step 7, then u^{i+1} is added to the filter and the restoration phase returns a point u^{i+1} that is acceptable to the filter. ■

LEMMA 4.18 *From the moment that* u^i *is added to the filter, the filter always contains an entry that dominates* u^i.

Here a proof similar to that of Lemma 9 in [34] applies.

4.3.2. *Feasibility and Complementarity*

In this subsection we show that under Assumptions 4.3–4.7 the sequence $\{\theta^i\}$ converges to zero, i.e. all limit points of $\{u^i\}$ are feasible and satisfy the complementarity condition (Theorem 4.21). First, we consider the case where the filter is augmented only a finite number of times.

LEMMA 4.19 *Suppose that Assumptions* 4.3–4.7 *hold and that the filter is augmented only a finite number of times, i.e.* $|\mathcal{A}| < \infty$. *Then*

$$\lim_{i \to \infty} \theta^i = 0. \tag{34}$$

The proof of this result is left to the Appendix of this article.

We now consider the case where infinitely many iterates are added to the filter. Thus we consider a subsequence $\{u^{i_j}\}$ with $i_j \in \mathcal{A}$ for all j.

LEMMA 4.20 *Let* $\{u^{i_j}\}$ *be a subsequence of iterates generated by Algorithm* 4.1 *so that the filter is augmented in iteration* i_j ($i_j \in \mathcal{A}$ *for all* j). *It then follows that*

$$\lim_{j\to\infty} \theta^{i_j} = 0. \tag{35}$$

The proof of this result is given in the Appendix of this article.

Both Lemmas 4.19 and 4.20 are needed for the proof of the following theorem.

THEOREM 4.21 *Suppose Assumptions* 4.3–4.6 *hold. Then*

$$\lim_{i\to\infty} \theta^{i} = 0. \tag{36}$$

Proof We have to consider three cases. When the filter is augmented only a finite number of times, then Lemma 4.19 proves the claim. However, if there exists some $I \in \mathbb{N}$ so that the filter is updated by (18) in all iteration $i \geq I$, then Lemma 4.20 proves the claim.

We now consider the case where for all $I \in \mathbb{N}$ there exist $i_1, i_2 \geq I$ with $i_1 \in \mathcal{A}$ and $i_2 \notin \mathcal{A}$. The proof is by contradiction. We will use a reasoning similar to that of Theorem 1 in [39]. Suppose $\limsup_i \theta^i = M$, with $M > 0$. Next, two subsequences u^{i_j} and u^{l_j} of u^i are defined as follows:

(i) Set $j = 0$ and $i_{-1} = -1$.
(ii) Pick $i_j > i_{j-1}$ with

$$\theta^{i_j} \geq M/2 \tag{37}$$

and $i_j \notin \mathcal{A}$. We remark that Lemma 4.20 ensures the existence of $i_j \notin \mathcal{A}$ since otherwise $\theta^{i_j} \to 0$.
(iii) Choose $l_j \in \mathcal{A}$ as the first iteration after i_j, i.e. $l_j > i_j$, in which the filter is augmented.
(iv) Set $j = j + 1$ and go back to (ii).

Therefore, every u^{i_j} satisfies (37), and for each u^{i_j} the iterate u^{l_j} is the first iterate after u^{i_j} for which $(\theta^{l_j}, \theta_o^{l_j})$ is included in the filter.

Since (46) and (47) hold, for all $i = i_j, \ldots, l_j - 1 \notin \mathcal{A}$, we obtain for all j

$$\varphi_{\hat{\mu}}^{i_j+1} < \varphi_{\hat{\mu}}^{i_j} - \widetilde{c}\,[M/2]^r \quad \text{and} \quad \theta_o^{l_j} \leq \theta_o^{i_j+1} < \theta_o^{i_j} - \widetilde{c}\,[M/2]^r. \tag{38}$$

This ensures that for $I \in \mathbb{N}$ there exists some $j \geq I$ with $\theta_o^{i_{(j+1)}} \geq \theta_o^{l_j}$ because otherwise (38) would imply

$$\varphi_{\hat{\mu}}^{i_{(j+1)}} < \varphi_{\hat{\mu}}^{i_j} - \widetilde{c}[M/2]^r \quad \text{and} \quad \theta_o^{i_{(j+1)}} < \theta_o^{l_j} < \theta_o^{i_j} - \widetilde{c}[M/2]^r$$

for all j and consequently $\lim_j \varphi_{\hat{\mu}}^{i_j} = -\infty$. This contradicts the fact that $\varphi_{\hat{\mu}}^{i}$ is bounded below. We remark that θ_o^i is bounded. Thus, there exists a subsequence $\{j_p\}$ of $\{j\}$ such that $\theta_o^{i_{(j_p+1)}} \geq \theta_o^{l_{j_p}}$. This inequality and the filter update rule (18) imply that

$$\theta^{i_{(j_p+1)}} \leq (1 - \gamma_1)\theta^{l_{j_p}}, \tag{39}$$

since $u^{i_{j_p+1}} \notin \overline{F}_{i_{(j_p+1)}} \supseteq F_{l_{j_p}}$ and $l_{j_p} \in \mathcal{A}$ for all p. Then Lemma 4.20 yields $\lim_p \theta^{l_{j_p}} = 0$ so that from (39) $\lim_p \theta^{i_{j_p}} = 0$, in contradiction to (37). ■

4.3.3. *Optimality*

Here we show that Assumptions 4.3–4.7 guarantee that the optimality measure θ_o^i is not bounded away from zero, i.e. there exists at least one limit point that is a KKT point for the problem (6) (Theorem 4.26).

The following lemma establishes conditions that guarantee that there exists a step size bounded away from zero so that the Armijo conditions (A-r) are satisfied.

LEMMA 4.22 *Suppose Assumptions* 4.3–4.7 *hold. Let* $\{u^{i_j}\}$ *be a subsequence and* $m_1^{i_j}(\alpha) \leq -\alpha\varepsilon_2$ *and* $m_2^{i_j}(\alpha) \leq -\alpha\varepsilon_3$ *for constant* ε_2 *and* ε_3 *independent of* i_j *and for all* $\alpha \in (0, 1]$. *Then there exists some constant* $\overline{\alpha} > 0$ *so that for all* i_j *and* $\alpha \leq \overline{\alpha}$

$$\varphi_{\hat{\mu}}^{i_j+1} - \varphi_{\hat{\mu}}^{i_j} \leq \eta\, m_1^{i_j}(\alpha) \quad \textit{and} \quad \theta_o^{i_j+1} - \theta_o^{i_j} \leq \eta\, m_2^{i_j}(\alpha). \tag{40}$$

The proof of these results is given in the Appendix of this article.

To show that the sequence $\{\theta_o^i\}$ converges to zero, we first consider the case where the filter is augmented only a finite number of times.

LEMMA 4.23 *Suppose that Assumptions* 4.3–4.7 *hold and that the filter is augmented only a finite number of times, i.e.* $|\mathcal{A}| < \infty$. *Then*

$$\lim_{i\to\infty} \theta_o^i = 0.$$

The proof of this result is left to the Appendix of this article.

For the case where infinitely many iterates are added to the filter, Lemma 4.24 establishes conditions under which a step size can be found that is acceptable to the current filter.

LEMMA 4.24 *Suppose Assumptions* 4.3–4.7 *hold. Let* $\{u^{i_j}\}$ *be a subsequence and* $m_1^{i_j}(\alpha) < -\alpha\varepsilon_2$ *for a constant* $\varepsilon_2 > 0$ *independent of* i_j *and for all* $\alpha \in (0, 1]$. *Then there exist constants* $c_1, c_2, c_3 > 0$ *so that*

$$\left(\theta(u^{i_j} + \alpha\Delta^{i_j}), \theta_o(u_2^{i_j} + \alpha\Delta u^{2,i_j})\right) \notin \overline{F}_{i_j}$$

for all i_j *and* $\alpha \leq \min\{c_1, c_2, c_3\theta(u^{i_j})\}$.

The proof of this result is given in the Appendix of this article.

Lemma 4.25 shows that in iterations corresponding to a subsequence with only nonoptimal limit points the filter is eventually not augmented. This result is used in the proof of the main global convergence theorem (Theorem 4.26).

LEMMA 4.25 *Suppose Assumptions* 4.3–4.7 *hold. Let* $\{u^{i_j}\}$ *be a subsequence with* $\theta_o^{i_j} > \epsilon$ *for a constant* $\epsilon > 0$ *independent of* i_j. Then there exists $I \in \mathbb{N}$ *so that for all* $i_j \geq I$ *the filter is not augmented in iteration* i_j ($i_j \notin \mathcal{A}$).

The proof of this result is given in the Appendix of this article.

We state now the main global convergence result.

THEOREM 4.26 *Suppose Assumptions* 4.3–4.7 *hold. Then*

$$\lim_{i\to\infty} \theta^i = 0, \tag{41}$$

$$\lim_{i\to\infty} \inf \theta_o^i = 0. \tag{42}$$

In other words, all limit points are feasible and satisfy the complementarity condition, and there exists a limit point u^* of $\{u^i\}$ *which is a KKT point for the problem* (6).

Proof Equation (41) follows from Theorem 4.21. To prove (42) we consider the two cases. When the filter is augmented only a finite number of times, then Lemma 4.23 proves the claim. The proof of the case where infinitely many iterates are added to the filter is by contradiction. Suppose that there exists a subsequence $\{u^{i_j}\}$ so that $i_j \in \mathcal{A}$ for all j. If we assume that $\limsup_i \theta_o^i > 0$, then there exist a subsequence $\{u^{i_{j_k}}\}$ of $\{u^{i_j}\}$, and a constant $\epsilon > 0$, so that $\lim_k \theta^{i_{j_k}} = 0$ and $\theta_o^{i_{j_k}} > \epsilon$ for all i_{j_k}. By applying Lemma 4.25 to $\{u^{i_{j_k}}\}$, there exists an iteration i_{j_k} in which the filter is not augmented ($i_{j_k} \notin \mathcal{A}$). This contradicts the choice of $\{u^{i_j}\}$ so that $\lim_j \theta_o^{i_j} = 0$, which proves (42). ■

5. Globalization procedure

To achieve convergence to the solution within a local framework, line search methods use, in general, penalty or merit functions. As previously explained, a backtracking line search method based on a filter approach, as a tool to guarantee global convergence in algorithms for nonlinear constrained finite optimization [6,39], avoids the use of a merit function. This new technique has been combined with a variety of optimization methods to solve different types of optimization problems. Its use to promote global convergence to the solution of an SIP problem was originally presented in [18,19]. Here, we also extend its use to the new proposed interior point reduction method. Its practical competitiveness with other methods in the literature suggests that this research is worth pursuing and the theoretical convergence analysis should be carried out in a near future.

To define the next approximation to the SIP problem, a two-dimensional filter line search method is implemented. Each entry in the filter has two components, one measures SIP-feasibility, $\Theta(x) = \|\max_{t\in T}(0, g(x,t))\|_2$, and the other SIP-optimality, f (the objective function). First, we assume that $d^k = x^{k,i} - x^k$, where i is the iteration index that satisfies the termination criteria in (19). Based on d^k, the below-described filter line search methodology computes the trial point $x^{k+1} = x^k + d^k$ and tests if it is acceptable by the filter. However, if this trial point is rejected, the algorithm recovers the direction of the first iteration, $d^k = x^{k,1} - x^k$, and tries to compute a trial step size α^k such that $x^{k+1} = x^k + \alpha^k d^k$ satisfies one of the below acceptance conditions and it is acceptable by the filter.

Here, a trial step size α^k is acceptable if a sufficient progress towards either the SIP-feasibility or the SIP-optimality is verified, i.e. if

$$\Theta^{k+1} \le (1-\gamma)\Theta^k \quad \text{or} \quad f^{k+1} \le f^k - \gamma\Theta^k \tag{AC-sip}$$

holds, for a fixed $\gamma \in (0,1)$. Θ^{k+1} is the simplified notation of $\Theta(x^{k+1})$. On the other hand, if

$$\Theta^k \le \Theta^{\min}, \quad \alpha^k(\nabla f^k)^T d^k < 0 \quad \text{and} \quad -\alpha^k(\nabla f^k)^T d^k > \delta\left[\Theta^k\right]^r, \tag{SW-sip}$$

are satisfied, for fixed positive constants $\Theta^{\min}$, δ and r, then the trial approximation x^{k+1} is acceptable only if a sufficient decrease in f is verified

$$f^{k+1} \leq f^k + \eta\alpha^k(\nabla f^k)^T d^k \tag{A-sip}$$

for $\eta \in (0, 0.5)$. The filter is initialized with pairs (Θ, f) that have $\Theta \geq \Theta^{\max} > 0$. If the acceptable approximation satisfies the condition (AC-sip), the filter is updated; and if the conditions (SW-sip) and (A-sip) hold, then the filter remains unchanged. The reader is referred to [18] for more details concerning the implementation of this filter strategy in the SIP context.

6. Termination criteria

As far as the termination criteria are concerned, our reduction algorithm stops at a point x^{k+1} if the following conditions hold:

$$\left\{\max\{g_l(x^{k+1}), l = 0, \ldots, L^{k+1} + 1\} < \epsilon_g \text{ and } \frac{|f^{k+1} - f^k|}{1 + |f^{k+1}|} < \epsilon_f\right\} \quad \text{or} \quad k > k^{\max},$$

for small positive constants ϵ_g and ϵ_f.

7. Numerical results and conclusions

The proposed reduction method was implemented in the C programming language on a Intel Core2, Duo T8300 2.4 GHz with 4 GB of RAM. For the computational experiences we consider eight test problems from the literature [2,14,23,24,43]. Different initial points were tested with some problems so that a comparison with other results is possible [2,43]. In the multi-local procedure we fix the following constants: $\mathcal{K}^{ML} = 3$, $\delta^{ML} = 1.0$, $\varepsilon_1 = 0.25$. In the primal–dual interior point method we define the constants $\delta_1^{IP} = \delta_2^{IP} = 10^{-6}$, $\tau_s = 0.01$ and $i^{\max} = 10$. The Hessian $H(x, y)$ is approximated by a BFGS quasi-Newton update, with a guaranteed positive definite initial approximation. Other parameters are defined as follows [41]: $\theta^{\max} = 10^4 \max\{1, \theta^0\}$, $\theta^{\min} = 10^{-4} \max\{1, \theta^0\}$, $\theta_o^{\max} = 10^4 \max\{1, \theta_o^0\}$, $\Theta^{\max} = 10^4 \max\{1, \Theta^0\}$, $\Theta^{\min} = 10^{-4} \max\{1, \Theta^0\}$, $\gamma = \gamma_1 = \gamma_2 = 10^{-5}$, $\eta = 10^{-4}$, $\delta = 1$, $r = 1.1$, $\tau = 0.95$, $\delta_c = 0.1$. In the termination criteria we fix the following constants: $k^{\max} = 100$, $\epsilon_g = \epsilon_f = 10^{-5}$.

In Table 1, $P\#$ refers to the problem number as reported in [2], $|T^*|$ represents the number of maximizers satisfying (4) at the final iterate, f^* is the objective function value at the final iterate, N_{rm}, N_{ML} and N_{IP} give the number of iterations needed by the reduction method, the number of the multi-local optimization calls and the average number of iterations needed in the primal–dual interior point method, respectively. We used the initial approximations proposed in the above-cited paper. Problems 2, 3, 6 and 14 were solved with the initial approximation proposed in [43] as well. They are identified in Table 1 with $^{(2)}$. Some problems were also solved using the initial approximation 0_n (see $^{(1)}$ in the table).

We also include Tables 2 and 3 displaying results from the literature, so that a comparison between the herein proposed reduction method and a selection of other well-known methods is possible. The compared results are taken from the

Table 1. Computational results.

$P\#$	n	$\|T^*\|$	f^*	N_{rm}	N_{ML}	N_{IP}
1	2	1	$-2.51381E-001$	5	6	9
$1^{(1)}$		1	$-2.51383E-001$	6	7	9
2	2	1	$2.61803E+000$	2	3	9
$2^{(1)}$		1	$1.94466E-001$	3	4	8
$2^{(2)}$		1	$1.94466E-001$	2	3	8
3	3	1	$5.33469E+000$	3	4	9
$3^{(1)}$		1	$5.33469E+000$	3	4	9
$3^{(2)}$		1	$5.33469E+000$	3	4	9
4	3	1	$6.49049E-001$	6	7	9
5	3	1	$4.30118E+000$	15	16	7
$6^{(1)}$	2	1	$1.07422E+002$	3	4	9
$6^{(2)}$		1	$1.07422E+002$	2	3	9
7	3	2	$1.00001E+000$	3	4	9
$7^{(1)}$		2	$1.00001E+000$	3	4	9
14	2	1	$2.20000E+000$	3	4	9
$14^{(2)}$		1	$2.20000E+000$	2	3	9

Table 2. Results from other methods.

			in [12]		in [13]		in [24]	in [23]		in [43]	
$P\#$	n	m	N_{it}	N_{feval}	N_{it}	CPU	N_{it}	N_{it}	CPU	N_{it}	N_{feval}
2	2	2	2	22	10	0.13	7	7	0.13	7	20
3	3	1	5	39	9	0.17	4	9	0.17	3	36
6	2	1	3	26	20	0.28	6	5	0.11	5	25
14	2	1	–	–	7	0.05	3	6	0.09	3	19

cited papers. In Table 2, N_{it} denotes the number of iterations, N_{feval} denotes the number of function evaluations, CPU represents the total cost time in seconds for solving the problem, and '–' means that the problem is not in test set of the article. Although there are some differences between the work required at each iteration for the methods in comparison, the reported results give important information concerning the efficiency of each method. The results show that the herein proposed reduction method for nonlinear SIP performs well for these test problems.

The selected results for Table 3 are from papers based on reduction-type methods. We observe that the use of P-DIP, when solving the reduced optimization problem, significantly reduces N_{rm} and N_{ML}, especially when compared with our previous reduction-type methods based on penalty techniques. The comparison with other reduction-type methods is also favourable to our proposal. The use of the trial point 'acceptance condition' (AC-sip) in the filter methodology results in a significant reduction of multi-local procedure calls. Further, the use of a filter method allows the full Newton step to be taken more often. Another advantage related with the proposed P-DIP filter line search method is that no parameters are required to be updated during both iterative cycles – the inner cycle, aiming to obtain

Table 3. Results from other reduction methods.

			in [21]		in [18]		in [22]		in [33]		in [2]
P#	n	m	N_{rm}	N_{ML}	N_{rm}	N_{ML}	N_{rm}	N_{ML}	N_{rm}	N_{ML}	N_{rm}
1	2	1	48	60	47	81	17	21	17	19	16
2	2	2	3	38	4	5	8	10	5	11	7
3	3	1	3	13	21	105	11	23	9	12	10
4	3	1	11	126	–	–	10	11	5	15	5
5	3	1	41	1000	–	–	8	14	4	9	4
6	2	1	7	8	8	9	27	87	16	19	9
7	3	2	8	9	7	8	9	14	2	4	3
14	2	1	10	95	–	–	6	7	5	8	5

an approximate solution of the reduced finite problem, and the outer cycle, to find the SIP solution.

Future developments will address the use of the sparse symmetric indefinite linear solver MA27 from Harwell Subroutine Library to solve the system (12) in order to improve efficiency. A dynamic updating of the tolerances δ_1^{IP} and δ_2^{IP} (depending on the iteration counter k) is now under investigation.

Acknowledgements

The authors wish to thank two anonymous referees for their careful reading of the manuscript and their valuable comments and suggestions.

References

[1] A. Ben-Tal, M. Teboule, and J. Zowe, *Second order necessary optimality conditions for semi-infinite programming problems*, Lecture Notes Control Inform. Sci. 15 (1979), pp. 17–30.

[2] I.D. Coope and G.A. Watson, *A projected Lagrangian algorithm for semi-infinite programming*, Math. Progr. 32 (1985), pp. 337–356.

[3] M.F.P. Costa and E.M.G.P. Fernandes, *Practical implementation of an interior point nonmonotone line search filter method*, Int. J. Comput. Math. 85 (2008), pp. 397–409.

[4] M.F.P. Costa and E.M.G.P. Fernandes, *A three-D filter line search method within an interior point framework*, Proceedings of 2008 International Computational and Mathematical Methods in Science and Engineering, Salamanca, Spain, ISBN: 978-84-612-1982-7, 2008, pp. 173–187.

[5] A.S. El-Bakry, R.A. Tapia, T. Tsuchiya, and Y. Zhang, *On the formulation and theory of the Newton interior-point method for nonlinear programming*, J. Optim. Theory Appl. 89 (1996), pp. 507–541.

[6] R. Fletcher and S. Leyffer, *Nonlinear programming without a penalty function*, Math. Progr. 91 (2002), pp. 239–269.

[7] M.A. Goberna and M.A. López (Eds.), *Semi-Infinite Programming. Recent Advances*, Nonconvex Optimization and Its Applications, Vol. 57, Springer-Verlag, Kluwer Academics Publishers, Dordrecht, The Netherlands, 2001.

[8] G. Gramlich, R. Hettich, and E.W. Sachs, *Local convergence of SQP methods in semi-infinite programming*, SIAM J. Optim. 5 (1995), pp. 641–658.
[9] R. Hettich and H.Th. Jongen, *Semi-infinite programming: Conditions of optimality and applications*, in *Lectures Notes in Control and Information Science – Optimization Techniques*, Vol. 7, J. Stoer, ed., Springer-Verlag, Berlin, 1978, pp. 1–11.
[10] R. Hettich and K.O. Kortanek, *Semi-infinite programming: Theory, methods and applications*, SIAM Rev. 35 (1993), pp. 380–429.
[11] L. Ingber, *Very fast simulated re-annealing*, Math. Comput. Model. 12 (1989), pp. 967–973.
[12] D.-H. Li, L. Qi, J. Tam, and S.-Y. Wu, *A smoothing Newton method for semi-infinite programming*, J. Global Optim. 30 (2004), pp. 169–194.
[13] C. Ling, Q. Ni, L. Qi, and S.-Y. Wu, *A new smoothing Newton-type algorithm for semi-infinite programming*, J. Global Optim. 47 (2010), pp. 133–159.
[14] G.-X. Liu, *A homotopy interior point method for semi-infinite programming problems*, J. Global Optim. 37 (2007), pp. 631–646.
[15] M. López and G. Still, *Semi-infinite programming*, Eur. J. Oper. Res. 180 (2007), pp. 491–518.
[16] J. Nocedal and S.J. Wright, *Numerical Optimization*, Springer-Verlag, New York, 1999.
[17] K. Parsopoulos, V. Plagianakos, G. Magoulas, and M. Vrahatis, *Objective function stretching to alleviate convergence to local minima*, Nonlinear Anal. 47 (2001), pp. 3419–3424.
[18] A.I. Pereira and E.M.G.P. Fernandes, *On a reduction line search filter method for nonlinear semi-infinite programming problems*, in *Euro Mini Conference on Continuous Optimization and Knowledge-Based Technologies*, L. Sakalauskas, G.W. Weber, and E.K. Zavadskas, eds., Neringa, Lithuania, 2008, pp. 174–179.
[19] A.I. Pereira and E.M.G.P. Fernandes, *An hyperbolic penalty filter method for semi-infinite programming*, in *Numerical Analysis and Applied Mathematics*, Vol. 1048, T.E. Simos, G. Psihoyios, and Ch. Tsitouras, eds., AIP Conference Proceedings, Springer-Verlag, Kos, Greece, 2008, pp. 269–273.
[20] A.I. Pereira and E.M.G.P. Fernandes, *Numerical experiments with a continuous L_2-exponential merit function for semi-infinite programming*, in *International Electronic Conference on Computer Science*, Vol. 1060, T.E. Simos and G. Psihoyios, eds., AIP Conference Proceedings, Springer-Verlag, 2008, pp. 11–14. Available at: http://link.aip.org/link/?APC/1060/11/1
[21] A.I. Pereira and E.M.G.P. Fernandes, *A reduction method for semi-infinite programming by means of a global stochastic approach*, Optimization 58 (2009), pp. 713–726.
[22] C.J. Price and I.D. Coope, *Numerical experiments in semi-infinite programming*, Comput. Optim. Appl. 6 (1996), pp. 169–189.
[23] L. Qi, C. Ling, X. Tong, and G. Zhou, *A smoothing projected Newton-type algorithm for semi-infinite programming*, Comput. Optim. Appl. 42 (2009), pp. 1–30.
[24] L. Qi, W.S.-Y. Wu, and G. Zhou, *Semismooth Newton methods for solving semi-infinite programming problems*, J. Global Optim. 27 (2003), pp. 215–232.
[25] R. Reemtsen and J.-J. Rückmann, *Semi-infinite Programming*, Nonconvex Optimization and Its Applications, Vol. 25, Kluwer Academic Publishers, Boston, 1998.
[26] D.F. Shanno and R.J. Vanderbei, *Interior-point methods for nonconvex nonlinear programming: Orderings and higher-order methods*, Math. Program. Ser. B 87 (2000), pp. 303–316.
[27] R. Silva, M. Ulbrich, S. Ulbrich, and L.N. Vicente, *A globally convergent primal–dual interior-point filter method for nonlinear programming: New filter optimality measures and computational results*, Preprint 08-49, Department of Mathematics, U. Coimbra, 2008.

[28] O. Stein and G. Still, *Solving semi-infinite optimization problems with interior point techniques*, SIAM J. Contr. Optim. 42 (2003), pp. 769–788.
[29] O. Stein and A. Tezel, *The semismooth approach for semi-infinite programming under the reduction ansatz*, J. Global Optim. 41 (2008), pp. 245–266.
[30] O. Stein and A. Tezel, *The semismooth approach for semi-infinite programming without strict complementarity*, SIAM J. Optim. 20 (2009), pp. 1052–1072.
[31] O. Stein and A. Winterfeld, *Feasible method for generalized semi-infinite programming*, J. Optim. Theory Appl. 146 (2010), pp. 419–443.
[32] G. Still, *Generalized semi-infinite programming: Theory and methods*, Eur. J. Oper. Res. 119 (1999), pp. 301–313.
[33] Y. Tanaka, M. Fukushima, and T. Ibaraki, *A comparative study of several semi-infinite nonlinear programmnig algorithms*, Eur. J. Oper. Res. 36 (1988), pp. 92–100.
[34] M. Ulbrich, S. Ulbrich, and L.N. Vicente, *A globally convergent primal–dual interior-point filter method for nonlinear programming*, Math. Program. 100 (2004), pp. 379–410.
[35] R.J. Vanderbei and D.F. Shanno, *An interior-point algorithm for nonconvex nonlinear programming*, Comput. Optim. Appl. 13 (1999), pp. 231–252.
[36] A.I.F. Vaz, E.M.G.P. Fernandes, and M.P.S.F. Gomes, *Robot trajectory planning with semi-infinite programming*, Eur. J. Oper. Res. 153 (2004), pp. 607–617.
[37] A.I.F. Vaz and E.C. Ferreira, *Air pollution control with semi-infinite programming*, Appl. Math. Model. 33 (2009), pp. 1957–1969.
[38] F.G. Vázquez, J.-J. Rückmann, O. Stein, and G. Still, *Generalized semi-infinite programming: A tutorial*, J. Comput. Appl. Math. 217 (2008), pp. 394–419.
[39] A. Wächter and L.T. Biegler, *Line search filter methods for nonlinear programming: Motivation and global convergence*, SIAM J. Optim. 16 (2005), pp. 1–31.
[40] A. Wächter and L.T. Biegler, *Line search filter methods for nonlinear programming: Local convergence*, SIAM J. Optim. 16 (2005), pp. 32–48.
[41] A. Wächter and L.T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Math. Program. 106 (2006), pp. 25–57.
[42] G.-W. Weber and A. Tezel, *On generalized semi-infinite optimization of genetic network*, TOP 15 (2007), pp. 65–77.
[43] O. Yi-gui, *A filter trust region method for solving semi-infinite programming problems*, J. Appl. Math. Comput. 29 (2009), pp. 311–324.

Appendix

Proof of Theorem 4.1

Proof From the second block of equations in (12) we obtain

$$\Delta w = -Y^{-1}\gamma_{\hat{\mu}} - Y^{-1}W\Delta y,$$

which substituting in the third block of equations, one yields the reduced KKT system

$$\begin{bmatrix} H & \nabla g \\ \nabla g^T & -Y^{-1}W \end{bmatrix}\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = -\begin{bmatrix} \sigma \\ \rho - Y^{-1}\gamma_{\hat{\mu}} \end{bmatrix}. \tag{43}$$

Now, solving the second block of equations in (43) for Δy we obtain

$$\Delta y = W^{-1}Y\nabla g^T\Delta x + W^{-1}Y\rho - W^{-1}\gamma_{\hat{\mu}},$$

and eliminating Δy from the first block of equations in (12) yields a system involving only Δx whose solution is

$$\begin{aligned}
\Delta x &= N^{-1}\big(-\sigma - \nabla g W^{-1} Y\rho + \nabla g W^{-1}\gamma_{\hat{\mu}}\big)\\
&= N^{-1}\big(-\nabla f - \nabla g\, y - \nabla g(W^{-1} Y\rho + \widehat{\mu} W^{-1} e - y)\big)\\
&= -N^{-1}\nabla f - N^{-1}\nabla g W^{-1} Y\rho - \widehat{\mu} N^{-1}\nabla g W^{-1} e,
\end{aligned}$$

where we used the definitions of σ and $\gamma_{\hat{\mu}}$. Using this formula of Δx, we can then obtain Δy and then Δw:

$$\begin{aligned}
\Delta w &= -Y^{-1}\gamma_{\hat{\mu}} - Y^{-1} W\big(W^{-1} Y\nabla g^T\Delta x + W^{-1} Y\rho - W^{-1}\gamma_{\hat{\mu}}\big),\\
&= -\nabla g^T\Delta x - \rho,\\
&= -\nabla g^T\big(-N^{-1}\nabla f - N^{-1}\nabla g W^{-1} Y\rho - \widehat{\mu} N^{-1}\nabla g W^{-1} e\big) - \rho\\
&= \nabla g^T N^{-1}\nabla f + \widehat{\mu}\nabla g^T N^{-1}\nabla g W^{-1} e - \big(I - \nabla g^T N^{-1}\nabla g W^{-1} Y\big)\rho.
\end{aligned}$$

■

Proof of Theorem 4.2

Proof We first consider (i). Note that $\nabla_x\varphi_{\hat{\mu}} = \nabla f$ and $\nabla_w\varphi_{\hat{\mu}} = -\widehat{\mu} W^{-1} e$. Let $y = \widehat{\mu} W^{-1} e$ and put $\sigma = \nabla f + \nabla g y$. Using the expressions for Δx and Δw given in Theorem 4.1 and assuming that $\rho = 0$, we get

$$\begin{aligned}
\begin{bmatrix}\nabla f\\ -y\end{bmatrix}^T\begin{bmatrix}\Delta x\\ \Delta w\end{bmatrix} &= \nabla f^T\Delta x - y^T\Delta w\\
&= \nabla f^T\big(-N^{-1}\nabla f - \widehat{\mu} N^{-1}\nabla g W^{-1} e\big)\\
&\quad - y^T\big(\nabla g^T N^{-1}\nabla f + \widehat{\mu}\nabla g^T N^{-1}\nabla g W^{-1} e\big)\\
&= \nabla f^T\big(-N^{-1}(\nabla f + \nabla g y)\big) - y^T\nabla g^T\big(N^{-1}(\nabla f + \nabla g\, y)\big)\\
&= \nabla f^T\big(-N^{-1}\sigma\big) + y^T\nabla g^T\big(-N^{-1}\sigma\big)\\
&= \big(\nabla f^T + y^T\nabla g^T\big)\big(-N^{-1}\sigma\big)\\
&= -\sigma^T N^{-1}\sigma \le 0,
\end{aligned}$$

which completes the proof. For (ii), we start with the measure θ_o. It is easy to see that $\nabla_x\theta_o = 2H\sigma$ and $\nabla_y\theta_o = 2\nabla g^T\sigma$, and from (11) we get

$$\begin{aligned}
\begin{bmatrix}\nabla_x\theta_o\\ \nabla_y\theta_o\end{bmatrix}^T\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix} &= 2\begin{bmatrix}H\sigma\\ \nabla g^T\sigma\end{bmatrix}^T\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}\\
&= 2(\sigma^T H\Delta x + \sigma^T\nabla g\Delta y)\\
&= 2(\sigma^T(H\Delta x + \nabla g\Delta y)) = -2\sigma^T\sigma = -2\theta_o \le 0.
\end{aligned} \tag{44}$$

We now address θ_f. Since $\nabla_x\theta_f = \nabla g\rho$ and $\nabla_w\theta_f = \rho$, from (11) we get

$$\begin{aligned}
\begin{bmatrix}\nabla_x\theta_f\\ \nabla_w\theta_f\end{bmatrix}^T\begin{bmatrix}\Delta x\\ \Delta w\end{bmatrix} &= \begin{bmatrix}\nabla g\rho\\ \rho\end{bmatrix}^T\begin{bmatrix}\Delta x\\ \Delta w\end{bmatrix}\\
&= \big(\rho^T\nabla g^T\big)\Delta x + \big(\rho^T\big)\Delta w\\
&= \rho^T\big(\nabla g^T\Delta x + \Delta w\big) = -\rho^T\rho \le 0.
\end{aligned}$$

We now address the centrality measure θ_c. Since $\nabla_w\theta_c = Y\gamma_{\hat{\mu}}$ and $\nabla_y\theta_c = W\gamma_{\hat{\mu}}$, from (11) we get

$$\begin{aligned}\begin{bmatrix}\nabla_w\theta_c\\ \nabla_y\theta_c\end{bmatrix}^T\begin{bmatrix}\Delta w\\ \Delta y\end{bmatrix} &= \begin{bmatrix}Y\gamma_{\hat{\mu}}\\ W\gamma_{\hat{\mu}}\end{bmatrix}^T\begin{bmatrix}\Delta w\\ \Delta y\end{bmatrix}\\ &= \gamma_{\hat{\mu}}^T(Y\Delta w + W\Delta y) = -\gamma_{\hat{\mu}}^T\gamma_{\hat{\mu}} \le 0.\end{aligned}$$

So, the perturbed Newton system indeed gives descent directions for the measures θ_o, θ_f and θ_c. ∎

Proof of Lemma 4.10

Proof For each $l=0,\dots,L^k+1$

$$\begin{aligned}w_l^{i+1}y_l^{i+1} &= (w_l^i+\alpha\Delta w_l^i)(y_l^i+\alpha\Delta y_l^i)\\ &= w_l^iy_l^i+\alpha(y_l^i\Delta w_l^i+w_l^i\Delta y_l^i)+\alpha\Delta w_l^i\alpha\Delta y_l^i\\ &= w_l^iy_l^i+\alpha(\widehat{\mu}^i-w_l^iy_l^i)+\big(w_l^{i+1}-w_l^i\big)\big(y_l^{i+1}-y_l^i\big)\\ &= (1-\alpha)w_l^iy_l^i+\alpha\delta_c\mu^i+\big(w_l^{i+1}-w_l^i\big)\big(y_l^{i+1}-y_l^i\big).\end{aligned}$$

So, inequalities (21) and (22) follow from this derivation and

$$\left|w_l^{i+1}-w_l^i\right|\left|y_l^{i+1}-y_l^i\right| \le (\alpha)^2\left|\Delta w_l^i\right|\,\left|\Delta y_l^i\right| \le (\alpha)^2\left\|\Delta^{3,i}\right\|^2.$$

Summing (21) and (22) over all l, we obtain

$$\begin{aligned}(w^{i+1})^Ty^{i+1} &\le (w^i)^Ty^i-\alpha(w^i)^Ty^i+(L^k+2)\alpha\delta_c\mu^i+(L^k+2)(\alpha)^2\left\|\Delta^{3,i}\right\|^2\\ (w^{i+1})^Ty^{i+1} &\ge (w^i)^Ty^i-\alpha(w^i)^Ty^i+(L^k+2)\alpha\delta_c\mu^i-(L^k+2)(\alpha)^2\left\|\Delta^{3,i}\right\|^2.\end{aligned}$$

Dividing these results by L^k+2, and using information from (15), we obtain

$$\begin{aligned}\mu^{i+1} &\le \mu^i-\alpha\mu^i+\alpha\delta_c\mu^i+(\alpha)^2\left\|\Delta^{3,i}\right\|^2 = (1-\alpha(1-\delta_c))\mu^i+(\alpha)^2\left\|\Delta^{3,i}\right\|^2\\ \mu^{i+1} &\ge \mu^i-\alpha\mu^i+\alpha\delta_c\mu^i-(\alpha)^2\left\|\Delta^{3,i}\right\|^2 = (1-\alpha(1-\delta_c))\mu^i-(\alpha)^2\left\|\Delta^{3,i}\right\|^2\end{aligned}$$

which proves (23) and (24). ∎

Proof of Lemma 4.14

Proof Consider (25). Recall (17) and the results of Theorem 4.1

$$\begin{aligned}m_1^{i_j}(\alpha)/\alpha &= (\nabla\varphi_{\hat{\mu}}^{i_j})^T\Delta^{1,i_j}\\ &= -(\sigma^{i_j})^T(N^{i_j})^{-1}\sigma^{i_j}+(\nabla f^{i_j})^T\big(-(N^{i_j})^{-1}\nabla g^{i_j}(W^{i_j})^{-1}Y^{i_j}\rho^{i_j}\big)\\ &\quad+(y^{i_j})^T(\nabla g^{i_j})^T\big(-(N^{i_j})^{-1}\nabla g^{i_j}(W^{i_j})^{-1}Y^{i_j}\rho^{i_j}\big)+(y^{i_j})^T\rho^{i_j}\\ &= -(\sigma^{i_j})^T(N^{i_j})^{-1}\sigma^{i_j}-(\sigma^{i_j})^T(N^{i_j})^{-1}\nabla g^{i_j}(W^{i_j})^{-1}Y^{i_j}\rho^{i_j}+(y^{i_j})^T\rho^{i_j}\\ &\le -c_1\|\sigma^{i_j}\|_2^2+c_2\|\sigma^{i_j}\|_2\|\rho^{i_j}\|_2+c_3\|\rho^{i_j}\|_2\\ &\le \|\sigma^{i_j}\|_2\Big(-\epsilon c_1+c_2\|\rho^{i_j}\|_2+\frac{c_3}{\epsilon}\|\rho^{i_j}\|_2\Big)\end{aligned}\tag{45}$$

for some constants $c_1, c_2, c_3 > 0$, where we used $\|\sigma^{i_j}\|_2 \ge \epsilon$ in the last inequality. If we now define $\varepsilon_1 = \frac{\epsilon^2c_1}{2(c_2\epsilon+c_3)}$, it follows for all u^{i_j} with $\theta^{i_j} \le \varepsilon_1$ (and thus $\|\rho^{i_j}\|_2 \le \varepsilon_1$) that

$$m_1^{i_j}(\alpha) \le -\alpha\frac{\epsilon c_1}{2}\|\sigma^{i_j}\|_2 \le -\alpha\frac{\epsilon^2c_1}{2} = -\alpha\varepsilon_2.$$

Now, we consider (26). By (44) (in the proof of Theorem 4.2) we have

$$m_2^{i_j}(\alpha) = -2\alpha(\sigma^{i_j})^T\sigma^{i_j} = -2\alpha\left\|\sigma^{i_j}\right\|_2^2 \le -2\epsilon^2\alpha = -\alpha\varepsilon_3.$$ ∎

Proof of Lemma 4.15

Proof Let $C_\sigma, C_\rho > 1$ be the Lipschitz constants for $\nabla\sigma$, $\nabla\rho$ respectively, and $\alpha \in (0,1]$. The positive constants defined in Lemma 4.15 are: $M_{\theta_o} = \frac{C_\sigma^2}{4} M_\Delta^2$, $M_\rho = \frac{C_\rho}{2}$, $M_{\gamma_0} = 2\sqrt{(L^k+2)}$, $M_{\varphi_{\hat{\mu}}} > 0$ and $M_\theta = \frac{C_\rho}{2} + 2\sqrt{(L^k+2)}$.

Let $s_{u_2}(\alpha) = u_2^i(\alpha) - u_2^i = \alpha\Delta^{2,i}$, where we use the simplified notation $u_2^i = (x^i, y^i)$. For simplicity, we also define $\sigma^{i+1} = \sigma(u_2^i(\alpha))$ and $\sigma^i = \sigma(u_2^i)$, and noting that $\nabla_x\sigma = H$, $\nabla_y\sigma = \nabla g^T$ and using (11)

$$\nabla\sigma(u_2)^T s_{u_2}(\alpha) = \alpha \begin{bmatrix} H \\ \nabla g^T \end{bmatrix}^T \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \alpha(H\Delta x + \nabla g \Delta y) = -\alpha\sigma(u_2),$$

we get ($\theta_o^{i+1} = \|\sigma^{i+1}\|^2$):

$$\begin{aligned}
\theta_o^{i+1} &= \|\sigma^{i+1} - \sigma^i + \sigma^i\|^2 \\
&= \left\| \sigma^i + \int_0^1 \nabla\sigma\big(u_2^i + t(u_2^i(\alpha) - u_2^i)\big)^T (u_2^i(\alpha) - u_2^i)\mathrm{d}t \right\|^2 \\
&= \left\| \sigma^i + \int_0^1 \nabla\sigma\big(u_2^i\big)^T s_{u_2}(\alpha)\mathrm{d}t + \int_0^1 \big(\nabla\sigma\big(u_2^i + ts_{u_2}(\alpha)\big) - \nabla\sigma\big(u_2^i\big)\big)^T s_{u_2}(\alpha)\mathrm{d}t \right\|^2 \\
&\le \left\| \sigma^i + \nabla\sigma\big(u_2^i\big)^T s_{u_2}(\alpha) \int_0^1 \mathrm{d}t \right\|^2 + C_\sigma^2 \|s_{u_2}(\alpha)\|^4 \left(\int_0^1 t\,\mathrm{d}t \right)^2 \\
&= (1-\alpha)^2 \big\|\sigma\big(u_2^i\big)\big\|^2 + \frac{1}{4} C_\sigma^2 \alpha^4 \|\Delta^{2,i}\|^4 \\
&\le (1-\alpha)^2 \theta_o^i + \frac{1}{4} C_\sigma^2 M_\Delta^2 \alpha^2 \|\Delta^{2,i}\|^2 \le (1-\alpha)\theta_o^i + \frac{1}{4} C_\sigma^2 M_\Delta^2 \alpha^2 \|\Delta^{2,i}\|^2,
\end{aligned}$$

which proves (27).

Similarly, let $s_{u_1}(\alpha) = u_1^i(\alpha) - u_1^i = \alpha\Delta^{1,i}$, where we use the simplified notation $u_1^i = (x^i, w^i)$. Further, we define $\rho^{i+1} = \rho(u_1^i(\alpha))$ and $\rho^i = \rho(u_1^i)$. Noting that $\nabla_x\rho = \nabla g$, $\nabla_w\rho = I$ and using (11)

$$\nabla\rho(u_1)^T s_{u_1}(\alpha) = \alpha \begin{bmatrix} \nabla g \\ I \end{bmatrix}^T \begin{bmatrix} \Delta x \\ \Delta w \end{bmatrix} = \alpha\big(\nabla g^T \Delta x + \Delta w\big) = -\alpha\rho(u_1),$$

thus

$$\begin{aligned}
\|\rho^{i+1}\| &= \|\rho^{i+1} - \rho^i + \rho^i\| \\
&= \left\| \rho^i + \int_0^1 \nabla\rho\big(u_1^i + t(u_1^i(\alpha) - u_1^i)\big)^T (u_1^i(\alpha) - u_1^i)\mathrm{d}t \right\| \\
&= \left\| \rho^i + \int_0^1 \nabla\rho(u_1^i)^T s_{u_1}(\alpha)\mathrm{d}t + \int_0^1 \big(\nabla\rho\big(u_1^i + ts_{u_1}(\alpha)\big) - \nabla\rho(u_1^i)\big)^T s_{u_1}(\alpha)\mathrm{d}t \right\| \\
&\le \left\| \rho^i + (\nabla\rho^i)^T s_{u_1}(\alpha) \int_0^1 \mathrm{d}t \right\| + C_\rho \|s_{u_1}(\alpha)\|^2 \int_0^1 t\,\mathrm{d}t \\
&= (1-\alpha)\|\rho^i\| + \frac{1}{2} C_\rho \alpha^2 \|\Delta^{1,i}\|^2,
\end{aligned}$$

which proves (28).

The estimate (29) follows from Lemma 4.10, where for each component $l = 0, \ldots, L^k+1$ we have

$$\pm w_l^i(\alpha)\, y_l^i(\alpha) \le \pm\big((1-\alpha)w_l^i y_l^i + \alpha\widehat{\mu}^i\big) + 2\alpha^2 \|\Delta^{3,i}\|^2,$$

and since $\widehat{\mu}=0$ at γ_0, using the definition $\gamma_0=WYe$, and taking the norm, one gets

$$\|\gamma_0^{i+1}\| \le (1-\alpha)\|\gamma_0^i\| + 2\sqrt{(L^k+2)}\alpha^2\|\Delta^{3,i}\|^2,$$

which proves (29).

Inequality (30) is derived from the second-order Taylor expansion, and noting that $\alpha(\nabla\varphi_{\hat{\mu}})^T\Delta^1 = m_1^i(\alpha)$, we obtain

$$\begin{aligned}\varphi_{\hat{\mu}}^{i+1} &= \varphi_{\hat{\mu}}^i + \alpha(\nabla\varphi_{\hat{\mu}}^i)^T\Delta^{1,i} + \frac{1}{2}\alpha^2(\Delta^{1,i})^T\nabla^2\varphi_{\hat{\mu}}^i\Delta^{1,i} + \cdots \\ &\le \varphi_{\hat{\mu}}^i + m_1^i(\alpha) + M_{\varphi_{\hat{\mu}}}\alpha^2\left\|\Delta^{1,i}\right\|^2\end{aligned}$$

for some $M_{\varphi_{\hat{\mu}}} > 0$, which proves (30).

Inequality (31) is derived by applying the previously obtained inequalities (28) and (29):

$$\begin{aligned}\theta^{i+1} &= \|\rho^{i+1}\|_2 + \|\gamma_0^{i+1}\|_2 \\ &\le (1-\alpha)\|\rho^i\|_2 + \frac{1}{2}C_\rho\alpha^2\left\|\Delta^{1,i}\right\|^2 + (1-\alpha)\|\gamma_0^i\|_2 + 2\sqrt{(L^k+2)}\alpha^2\left\|\Delta^{3,i}\right\|^2 \\ &\le (1-\alpha)\big(\|\rho^i\|_2 + \|\gamma_0^i\|_2\big) + \left(\frac{1}{2}C_\rho + 2\sqrt{(L^k+2)}\right)\alpha^2\left\|\Delta^i\right\|^2 \\ &= (1-\alpha)\theta^i + \left(\frac{1}{2}C_\rho + 2\sqrt{(L^k+2)}\right)\alpha^2\left\|\Delta^i\right\|^2,\end{aligned}$$

which proves (31).

Proof of Lemma 4.19

Proof Choose I so that for all iterations $i \ge I$ the filter is not augmented in iteration i. This means that in Step 4.3 of Algorithm 4.1 we have that for all $i \ge I$ both conditions (SC-r) and (A-r) are satisfied for α^i.

From (SC-r) we have $\delta\left[\theta^i\right]^r < -m_1^i(\alpha)$ so that from (A-r) we obtain $\varphi_{\hat{\mu}}^{i+1} - \varphi_{\hat{\mu}}^i \le \eta m_1^i(\alpha) < -\eta\delta\left[\theta^i\right]^r$. Thus for all $i \notin \mathcal{A}$

$$\varphi_{\hat{\mu}}^{i+1} - \varphi_{\hat{\mu}}^i < -\widetilde{c}\left[\theta^i\right]^r \tag{46}$$

holds for some $\widetilde{c} > 0$. Hence, for all $j = 1, 2, \ldots,$

$$\begin{aligned}\varphi_{\hat{\mu}}^{I+j} &= \varphi_{\hat{\mu}}^I + \sum_{i=I}^{I+j-1}\left(\varphi_{\hat{\mu}}^{i+1} - \varphi_{\hat{\mu}}^i\right) \\ &< \varphi_{\hat{\mu}}^I - \widetilde{c}\sum_{i=I}^{I+j-1}\left[\theta^i\right]^r.\end{aligned}$$

Since $\varphi_{\hat{\mu}}^{I+j}$ is bounded below (Assumption 4.7) as $i \to \infty$, the series on the right-hand side is bounded, and (34) follows immediately.

Similarly, from (SC-r) we have $\delta\left[\theta^i\right]^r < -m_2^i(\alpha)$ so that from (A-r) we obtain $\theta_o^{i+1} - \theta_o^i \le \eta m_2^i(\alpha) < -\eta\delta\left[\theta^i\right]^r$. Thus for all $i \notin \mathcal{A}$, and some $\widetilde{c} > 0$, it also holds:

$$\theta_o^{i+1} - \theta_o^i < -\widetilde{c}\left[\theta^i\right]^r. \tag{47}$$

Hence, the above reasoning could also be used here.

Proof of Lemma 4.20

Proof We prove by contradiction. Assume that the assertion is wrong. Since by Lemma 4.16, $\theta^{i_j} > 0$ holds for all $i_j \in \mathcal{A}$, then we can find $\varepsilon > 0$ with $\theta^{i_j} \ge \varepsilon$ for all $i_j \in \mathcal{A}$. Thus for $i_j \in \mathcal{A}$, define the region

$$\overline{S}_{i_j} = [\theta^{i_j} - \gamma_1\varepsilon, \theta^{i_j}] \times [\theta_o^{i_j} - \gamma_2\varepsilon, \theta_o^{i_j}].$$

We prove for all $i_j, i_l \in \mathcal{A}$, with $i_j > i_l$, that

$$\overline{S}_{i_j} \cap \overline{S}_{i_l} = \varnothing \tag{48}$$

holds. When u^{i_j} is added to the filter, the filter contains an entry u^{j_l} that dominates u^{i_l} according to Lemma 4.18. By Lemma 4.17, u^{i_j} is acceptable to the filter so that at least one of the following inequalities holds:

$$\theta^{i_j} \le \theta^{j_l} - \gamma_1\theta^{j_l} \le \theta^{i_l} - \gamma_1\varepsilon \quad \text{or} \quad \theta_o^{i_j} \le \theta_o^{j_l} - \gamma_2\theta^{j_l} \le \theta_o^{i_l} - \gamma_2\varepsilon,$$

which implies (48). Thus all the regions $\overline{S}_{i_j}$, $i_j \in \mathcal{A}$, are disjoint.

Since the sequence $\{\theta^{i_j}, \theta_o^{i_j}\}$ is bounded, we obtain the desired contradiction. ■

Proof of Lemma 4.22

Proof Let M_Δ, $M_{\varphi_{\hat{\mu}}}$ and M_{θ_o} be the constants from Lemmas 4.13 and 4.15. It follows that for all $\alpha \le \overline{\alpha}$ with $\overline{\alpha} = \min\left\{\frac{(1-\eta)\varepsilon_2}{M_{\varphi_{\hat{\mu}}} M_\Delta^2}, \frac{(\frac{1}{2}-\eta)\varepsilon_3}{M_{\theta_o} M_\Delta^2}\right\}$

$$\begin{aligned}\varphi_{\hat{\mu}}^{i_j+1} - \varphi_{\hat{\mu}}^{i_j} - m_1^{i_j}(\alpha) &\le M_{\varphi_{\hat{\mu}}}\alpha^2\left\|\Delta^{1,i_j}\right\|^2 \\ &\le \alpha(1-\eta)\varepsilon_2 \le -(1-\eta)m_1^{i_j}(\alpha)\end{aligned}$$

and recalling (17) and using (44) we have

$$\begin{aligned}\theta_o^{i_j+1} - (1-\alpha)\theta_o^{i_j} &\le M_{\theta_o}\alpha^2\left\|\Delta^{2,i_j}\right\|^2 \\ \theta_o^{i_j+1} - \theta_o^{i_j} - \frac{m_2^{i_j}(\alpha)}{2} &\le M_{\theta_o}\alpha^2\left\|\Delta^{2,i_j}\right\|^2 \\ &\le \alpha\left(\frac{1}{2}-\eta\right)\varepsilon_3 \le -\left(\frac{1}{2}-\eta\right)m_2^{i_j}(\alpha),\end{aligned}$$

which implies (40). ■

Proof of Lemma 4.23

Proof Since $|\mathcal{A}| < \infty$, there exists $I \in \mathbb{N}$ so that $i \notin \mathcal{A}$ for all $i \ge I$. The proof is by contradiction. Suppose the claim is not true. Then there exist a subsequence $\{u^{i_j}\}$ and a constant $\epsilon > 0$ so that $\theta_o^{i_j} > \epsilon$ for all j. From Lemma 4.14 there exist ε_1, ε_2, $\varepsilon_3 > 0$ and $\widetilde{I} \ge I$ so that for all $i_j \ge \widetilde{I}$ we have $\theta^{i_j} \le \varepsilon_1$ and

$$m_1^{i_j}(\alpha) \le -\alpha\varepsilon_2 \quad \text{and} \quad m_2^{i_j}(\alpha) \le -\alpha\varepsilon_3 \quad \text{for all } \alpha \in (0, 1]. \tag{49}$$

Then from (A-r) and for $i_j \ge \widetilde{I}$

$$\varphi_{\hat{\mu}}^{i_j+1} - \varphi_{\hat{\mu}}^{i_j} \le \eta m_1^{i_j}(\alpha^{i_j}) \le -\eta\alpha^{i_j}\varepsilon_2 \quad \text{and} \quad \theta_o^{i_j+1} - \theta_o^{i_j} \le \eta m_2^{i_j}(\alpha^{i_j}) \le -\eta\alpha^{i_j}\varepsilon_3.$$

Using an argument similar to that in the proof of Lemma 4.19, and since $\varphi_{\hat{\mu}}^{i_j}$ and $\theta_o^{i_j}$ are bounded below, and $\varphi_{\hat{\mu}}^{i_j}$ and $\theta_o^{i_j}$ are monotonically decreasing for all $i_j \ge \widetilde{I}$ (from (46) and (47)), we conclude that $\lim_j \alpha^{i_j} = 0$.

Now assume that $\widetilde{I}$ is sufficiently large so that $\alpha^{i_j} < \alpha_{\max}^i$. This means that for $i_j \ge \widetilde{I}$ the first trial step $\alpha^{i_j,0} = \alpha_{\max}^{i_j}$ has not been accepted, and the last rejected trial step size during the backtracking line search procedure is

$$\alpha_R \equiv \alpha^{i_j, l_j} = 2\alpha^{i_j}, \tag{50}$$

which satisfies (SC-r) since $i_j \notin \mathcal{A}$. This rejected trial step size either satisfies

$$\varphi_{\hat{\mu}}(u_1^{i_j} + \alpha_R\Delta^{1,i_j}) - \varphi_{\hat{\mu}}(u_1^{i_j}) > \eta m_1^{i_j}(\alpha_R) \quad \text{or} \quad \theta_o(u_2^{i_j} + \alpha_R\Delta^{2,i_j}) - \theta_o(u_2^{i_j}) > \eta m_2^{i_j}(\alpha_R) \tag{51}$$

(because (A-r) is violated), or it is not acceptable to the current filter

$$\left(\theta(u^{i_j}(\alpha_R)), \theta_o(u_2^{i_j}(\alpha_R))\right) \in \overline{F}_{i_j} = \overline{F}_I. \tag{52}$$

We show that neither (51) nor (52) can be true for sufficiently large i_j.

Since $\lim_j \alpha^{i_j} = 0$, we also have $\lim_j \alpha^{i_j,l_j} = 0$ (see (50)). In particular, for sufficiently large i_j we have $\alpha^{i_j,l_j} \leq \overline{\alpha}$ with $\overline{\alpha}$ from Lemma 4.22. Thus (51) cannot be satisfied for those i_j.

Now we consider (52). Let $\Theta^I = \min\{\theta : (\theta, \theta_o) \in \overline{F}_I\} > 0$, as defined in Lemma 4.16. Using Lemmas 4.13 and 4.15, we see that

$$\theta(u^{i_j}(\alpha^{i_j,l_j})) \leq \left(1 - \alpha^{i_j,l_j}\right)\theta(u^{i_j}) + M_\theta M_\Delta^2\left(\alpha^{i_j,l_j}\right)^2.$$

Since $\lim_j \alpha^{i_j,l_j} = 0$ and from Theorem 4.21 we also have $\lim_j \theta(u^{i_j}) = 0$, it follows that for i_j sufficiently large $\theta(u^{i_j}(\alpha^{i_j,l_j})) < \Theta^I$, which contradicts (52). ■

Proof of Lemma 4.24

Proof Let M_Δ, M_{θ_o}, $M_{\varphi_{\hat{\mu}}}$ and M_θ be the constants from Lemmas 4.13 and 4.15 and define the following constants:

$$c_1 = \min\left\{1, \frac{\varepsilon_2}{M_{\varphi_{\hat{\mu}}} M_\Delta^2}\right\}, \quad c_2 = \min\left\{1, \frac{\frac{1}{2}\varepsilon_3}{M_{\theta_o} M_\Delta^2}\right\} \quad \text{and} \quad c_3 = \frac{1}{M_\theta M_\Delta^2}.$$

Let u^{i_j} be an iterate. The Algorithm 4.1 ensures that

$$\left(\theta^{i_j}, \theta_o^{i_j}\right) \notin \overline{F}_{i_j}. \tag{53}$$

For $\alpha \leq c_1$ we have $\alpha^2 \leq \frac{\alpha\varepsilon_2}{M_{\varphi_{\hat{\mu}}} M_\Delta^2} \leq \frac{-m_1^{i_j}(\alpha)}{M_{\varphi_{\hat{\mu}}}\left\|\Delta^{1,i_j}\right\|^2}$, or equivalently $m_1^{i_j}(\alpha) + M_{\varphi_{\hat{\mu}}}\alpha^2\left\|\Delta^{1,i_j}\right\|^2 \leq 0$, and from (30) it follows that

$$\varphi_{\hat{\mu}}^{i_j+1} \leq \varphi_{\hat{\mu}}^{i_j}. \tag{54}$$

For $\alpha \leq c_2$ we have $\alpha^2 \leq \frac{\alpha\frac{\varepsilon_3}{2}}{M_{\theta_o} M_\Delta^2} \leq \frac{-\frac{m_2^{i_j}(\alpha)}{2}}{M_{\theta_o}\left\|\Delta^{2,i_j}\right\|^2}$, or

$$\frac{m_2^{i_j}(\alpha)}{2} + M_{\theta_o}\alpha^2\left\|\Delta^{2,i_j}\right\|^2 \leq 0,$$
$$-\alpha\theta_o^{i_j} + M_{\theta_o}\alpha^2\left\|\Delta^{2,i_j}\right\|^2 \leq 0$$

and from (27) it follows that

$$\theta_o^{i_j+1} \leq \theta_o^{i_j}. \tag{55}$$

Now, for $\alpha \leq c_3\theta^{i_j}$, we have $\alpha^2 \leq \frac{\alpha\theta^{i_j}}{M_\theta\left\|\Delta^{i_j}\right\|^2}$, or $-\alpha\theta^{i_j} + \alpha^2 M_\theta\left\|\Delta^{i_j}\right\|^2 \leq 0$ and from (31)

$$\theta^{i_j+1} \leq \theta^{i_j}. \tag{56}$$

By the filter definition, the claim then follows from (53), (55) and (56). ■

Proof of Lemma 4.25

Proof Recall that $\lim_{i\to\infty}\theta^i = 0$ by Theorem 4.21. From Lemma 4.14 there exist constants $\varepsilon_1, \varepsilon_2, \varepsilon_3 > 0$ so that

$$\theta^{i_j} \leq \varepsilon_1 \quad \text{and} \quad m_1^{i_j}(\alpha) \leq -\varepsilon_2\alpha \quad \text{and} \quad m_2^{i_j}(\alpha) \leq -\varepsilon_3\alpha \tag{57}$$

for i_j sufficiently large and $\alpha \in (0, 1]$. (Without loss of generality we can assume that (57) is valid for all i_j.) We use Lemmas 4.22 and 4.24 to obtain the constants $\overline{\alpha}$, $c_1, c_2, c_3 > 0$. We now choose $I \in \mathbb{N}$ so that for all $i_j \geq I$

$$\theta^{i_j} < \min\left\{\frac{\overline{\alpha}}{c_3}, \frac{c_1}{c_3}, \frac{c_2}{c_3}, \left[\frac{c_3\varepsilon_2}{\delta}\right]^{\frac{1}{r-1}}, \left[\frac{c_3\varepsilon_3}{\delta}\right]^{\frac{1}{r-1}}\right\}. \tag{58}$$

Using an argument similar to that in the proof of Lemma 4.16, if $\theta^{i_j} = 0$ for all $i_j \geq I$, we have that both (SC-r) and (A-r) hold in iteration i_j so that $i_j \notin \mathcal{A}$.

On the other hand, if $\theta^{i_j} > 0$ for the remaining iterations $i_j \geq I$ then (58) implies that

$$\frac{\delta\left[\theta^{i_j}\right]^r}{\varepsilon_2} < \frac{\delta\,\theta^{i_j}}{\varepsilon_2}\frac{c_3\varepsilon_2}{\delta} = c_3\theta^{i_j} \quad \text{and} \quad \frac{\delta\left[\theta^{i_j}\right]^r}{\varepsilon_3} < \frac{\delta\,\theta^{i_j}}{\varepsilon_3}\frac{c_3\varepsilon_3}{\delta} = c_3\theta^{i_j} \tag{59}$$

(since $r > 1$), as well as $c_3\theta^{i_j} < \min\{\overline{\alpha}, c_1, c_2\}$. Choosing an arbitrary $i_j \geq I$ with $\theta^{i_j} > 0$ and defining

$$\beta^{i_j} = c_3\theta^{i_j} = \min\{\overline{\alpha}, c_1, c_2, c_3\theta^{i_j}\}, \tag{60}$$

Lemmas 4.22 and 4.24 then imply that a trial step size $\alpha^{i_j,l} \leq \beta^{i_j}$ satisfies both

$$\varphi_{\hat{\mu}}(u_1^{i_j}(\alpha^{i_j,l})) \leq \varphi_{\hat{\mu}}(u_1^{i_j}) + \eta m_1^{i_j}(\alpha^{i_j,l}) \quad \text{and} \quad \theta_o(u_2^{i_j}(\alpha^{i_j,l})) \leq \theta_o(u_2^{i_j}) + \eta m_2^{i_j}(\alpha^{i_j,l}) \tag{61}$$

and

$$\left(\theta(u^{i_j}(\alpha^{i_j,l})), \theta_o(u_2^{i_j}(\alpha^{i_j,l}))\right) \notin \overline{F}_{i_j}. \tag{62}$$

Assuming that $\alpha^{i_j,L}$ is the first trial step size that satisfies both (61) and (62), then the backtracking line search implies that $\alpha \geq \alpha^{i_j,L}$ (using (59) and (60))

$$\alpha \geq \beta^{i_j} = c_3\theta^{i_j} > \min\left\{\frac{\delta\left[\theta^{i_j}\right]^r}{\varepsilon_2}, \frac{\delta\left[\theta^{i_j}\right]^r}{\varepsilon_3}\right\}$$

and therefore for $\alpha \geq \alpha^{i_j,L}$ (using (57))

$$\delta\left[\theta^{i_j}\right]^r < \alpha\varepsilon_2 \leq -m_1^{i_j}(\alpha) \quad \text{and} \quad \delta\left[\theta^{i_j}\right]^r < \alpha\varepsilon_3 \leq -m_2^{i_j}(\alpha).$$

Thus, for all trial step sizes $\alpha^{i_j,l} \geq \alpha^{i_j,L}$, conditions (SC-r) and (A-r) hold and by definition, we have $\alpha^{i_j,L} \geq \alpha_{\min}$, i.e. the restoration phase is not implemented. Hence $\alpha^{i_j,L}$ is the accepted step size α^{i_j}. Since it satisfies both (SC-r) and (61), the filter is not augmented in iteration i_j. ■