Research Article

A Globally Convergent Line Search Filter SQP Method for Inequality Constrained Optimization

Zhong Jin

Department of Mathematics, Shanghai Maritime University, Shanghai 201306, China

Correspondence should be addressed to Zhong Jin; asdjz1234@163.com

A line search filter SQP method for inequality constrained optimization is presented. This method makes use of a backtracking line search procedure to generate step size and the efficiency of the filter technique to determine step acceptance. At each iteration, the subproblem is always consistent, and it only needs to solve one QP subproblem. Under some mild conditions, the global convergence property can be guaranteed. In the end, numerical experiments show that the method in this paper is effective.

1. Introduction

In this paper, we consider the following nonlinear inequality constrained optimization problem:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g_j(x) \le 0, \quad j \in I = \{1, 2, \dots, m\}, \end{aligned} \tag{P}$$

where $x \in R^n$, $f : R^n \rightarrow R$, and $g_j(j \in I) : R^n \rightarrow R^1$ are assumed to be continuously differentiable.

It is well known that the sequential quadratic programming (SQP) method is one of the most efficient methods to solve problem (P). Because of its superlinear convergence rate, it has been widely studied. See Boggs and Tolle [1] for an excellent literature survey. In SQP methods, at each iteration the search direction is generally obtained by solving the subproblem as follows:

$$\begin{aligned} \min \quad & \nabla f(x_k)^T d + \frac{1}{2} d^T B_k d \\ \text{s.t.} \quad & g_j(x_k) + \nabla g_j(x_k)^T d \le 0, \quad j \in I = \{1, 2, \dots, m\}, \end{aligned} \tag{1}$$

where $B_k \in R^{n \times n}$ is a symmetric positive definite matrix.

However, the previous QP subproblem has a serious shortcoming that constraints in (1) may be inconsistent. To overcome this disadvantage, much attention has been paid [2–7]. Burke and Han [2], Zhou [3], and J.-L. Zhang and X.-S. Zhang [4] modified the constraints in subproblem (1) to ensure that a revised QP subproblem is consistent, and their methods are globally convergent. Burke and Han's method is just a conceptual method and cannot be implementable practically, Zhou's method is based on the exact line search, and Zhang's method focuses on the inexact line search. Furthermore, Pantoja and Mayne [5] presented a modification of SQP algorithm by modifying both constraints and objective function in (1), and the search direction is obtained by solving a new QP subproblem which always has an optimal solution. In addition, Liu and Li [6] and Liu and Zeng [7] proposed SQP algorithms with cautious update criteria, which can be considered as modifications of the SQP algorithm given in [5].

In the previous methods [2–7], a penalty function is always used as a merit function to test the acceptability of the iterate points. However, as we all know, there are several difficulties associated with the use of penalty function and in particular with the choice of the penalty parameter. Too low a choice may result in the loss of an optimal solution; on the other hand, too large a choice damps out the effect of the objective function. Therefore, filter method was first introduced in trust region SQP method for constrained nonlinear optimization problems by Fletcher and Leyffer [8],

offering an alternative to merit functions, as a tool to guarantee global convergence. Subsequently, global convergences of filter SQP methods were established by Fletcher et al. [9, 10]. Furthermore, Wächter and Biegler [11, 12] presented line search filter methods for nonlinear equality constrained programming, and the global convergence and local convergence were given. The promising numerical results of filter methods have led to a growing interest in filter methods in recent years [13–26], in which trust region strategy is used in [15, 26], and line search technique is taken in others. It is noteworthy that, based on [2–4] and line search filter technique, Su and Che [13] and Su [14] presented modified SQP filter methods for inequality constrained optimization. In Su's methods, the subproblem is always consistent, and the global convergence property is obtained.

In this paper, derived from [5–7] and the filter technique, especially the filter line search technique provided by Wächter and Biegler [11, 12], we propose a line search filter SQP method for inequality constrained optimization. This method has the following merits: it avoids using the penalty function to guarantee global convergence and has no requirement on initial point; at each iterate point the subproblem is always consistent, and it only needs to solve one QP subproblem; under some mild conditions, we prove that the algorithm either terminates at a Karush-Kuhn-Tucker (KKT) point within finite steps or generates an infinite sequence whose at least one cluster is a KKT point; in the end, numerical experiments show that our method is effective.

This paper is organized as follows. In the next section, the structure of line search filter technique is discussed. The algorithm is put forward in Section 3, and the global convergence theory of the algorithm is presented in Section 4. The numerical results for some typical examples are listed in Section 5. Finally in Section 6, a brief discussion on the proposed algorithm is given.

2. The Structure of Line Search Filter Technique

In this section, we will present the structure of line search filter technique and some related strategies. Instead of combing the objective function and constraint violation into a single function, filter methods view nonlinear optimization as a biobjective optimization problem that minimizes the objective function and the constraint violation. Now we give the structure of line search filter technique.

In the line search filter technique, after a search direction d_k has been computed, a step size $\alpha_{k,l} \in (0, 1]$ is determined in order to obtain the trial iteration $x_k(\alpha_{k,l}) = x_k + \alpha_{k,l}d_k$. More precisely, for fixed constants $\gamma_f, \gamma_\theta \in (0, 1)$, a trial step size $\alpha_{k,l}$ provides sufficient reduction with respect to the current point x_k if

$$\theta(x_k(\alpha_{k,l})) \le (1 - \gamma_\theta)\theta(x_k)$$
$$\text{or } f(x_k(\alpha_{k,l})) \le f(x_k) - \gamma_f\theta(x_k), \tag{2}$$

where $\theta(x)$ is a constraint violation function. Similar to [18], let $\theta(x_k) = \max_{1 \le i \le m}[g_i(x_k)]_+$ and $[\cdot]_+ = \max\{\cdot, 0\}$.

For the sake of a simplified notation, the filter is defined in this paper not as a list but as a set $\mathscr{F}_k \subseteq [0, \infty] \times [-\infty, \infty]$ containing all (θ, f) pairs that are prohibited in iteration k. We say that a trial point $x_k(\alpha_{k,l})$ is acceptable to the filter if its (θ, f) pair does not lie in the taboo region, that is, if

$$(\theta(x_k(\alpha_{k,l})), f(x_k(\alpha_{k,l}))) \notin \mathscr{F}_k. \tag{3}$$

At the beginning of the algorithm, the filter is initialized by $\mathscr{F}_0 = \{(\theta, f) \in R^2 : \theta \ge u\}$, where u is a large positive number. Throughout the optimization the filter is then augmented in some iterations after the new iterate point x_{k+1} has been accepted. For this, the following updating formula is used:

$$\mathscr{F}_{k+1} = \mathscr{F}_k \bigcup \{(\theta, f) \in R^2 : \theta \ge (1 - \gamma_\theta)\theta(x_k),$$
$$f \ge f(x_k) - \gamma_f\theta(x_k)\}. \tag{4}$$

Similar to the traditional strategy of the filter method, to avoid the convergence to a feasible point but not an optimal solution, we consider the following f-type switching condition:

$$m_k(\alpha_{k,l}) < 0, \quad -m_k(\alpha_{k,l}) > \delta[\theta(x_k)]^{s_\theta}, \tag{5}$$

where $m_k(\alpha_{k,l}) = \alpha_{k,l}\nabla f(x_k)^T d_k$, $\delta > 0$, and $s_\theta \in (0, 1)$.

When condition (5) holds, the step d_k is a descent direction for current objective function. Then, instead of insisting on (2), the Armijo-type reduction condition is employed as follows:

$$f(x_k(\alpha_{k,l})) \le f(x_k) + \eta m_k(\alpha_{k,l}), \tag{6}$$

where $\eta \in (0, (1/2))$ is a fixed constant. If (5) and (6) hold for the accepted trial step size, we may call it an f-type point, and accordingly this iteration is called an f-type iteration. An f-type point should be accepted as x_{k+1} without updating of the filter; that is, $\mathscr{F}_{k+1} = \mathscr{F}_k$, while if a trial point $x_k(\alpha_{k,l})$ does not satisfy the switching condition (5) but satisfies (2), we call it an h-type point (or accordingly an h-type iteration). An h-type point should be accepted as x_{k+1} with updating of the filter, and we denote the set of indices of those iterations where filter has been augmented by $\mathscr{A} \subset \mathbb{N}$.

In our method, when $\nabla f(x_k)^T d_k < 0$, the line search is performed. In the situation where no admissible step size can be found, the method switches to a feasibility restoration phrase, whose purpose is to find a new iterate point that satisfies (2) and is also acceptable to the current filter by trying to decrease the constraint violation. In order to detect the situation where no admissible step size can be found and the restoration phase has to be invoked, we approximate a minimum desired step size using linear models of involved functions. For this, we define that

$$\alpha_k^{\min} = \min = \left\{\gamma_\theta, -\frac{\gamma_f\theta(x_k)}{\nabla f(x_k)^T d_k}, -\frac{\delta[\theta(x_k)]^{s_\theta}}{\nabla f(x_k)^T d_k}\right\}. \tag{7}$$

3. Description of the Algorithm

We will now proceed to propose a line search filter SQP method for inequality constrained optimization and formally describe the algorithmic details in this section. The proposed algorithm consists of inner loops, and in the next section we show that there is no endless loop and the method is implementable and globally convergent.

In our algorithm, motivated by [5–7], the quadratic subproblem (1) is replaced by the following problem:

$$\begin{aligned} \min \quad & \nabla f(x_k)^T d + \frac{1}{2} d^T B_k d + b_k t \\ \text{s.t.} \quad & g_j(x_k) + \nabla g_j(x_k)^T d \le t \quad j \in I, \\ & 0 \le t \le \theta(x_k), \end{aligned} \tag{8}$$

where b_k is a positive parameter. Clearly, this subproblem is always consistent and is a convex programming if B_k is positive semidefinite.

Let $(d_k, t_k) \in (R^n, R^1)$ be the solution of subproblem (8). Then there exists a Lagrange multiplier $(\lambda_k, v_k, h_k) \in (R^m, R^1, R^1)$ satisfying the following KKT system:

$$\begin{aligned} & g_j(x_k) + \nabla g_j(x_k)^T d_k \le t_k \ (j \in I), \\ & 0 \le t_k \le \theta(x_k), \\ & \nabla f(x_k) + B_k d_k + \sum_{j \in I} \nabla g_j(x_k)(\lambda_k)_j = 0, \\ & b_k - \|\lambda_k\|_1 - v_k + h_k = 0, \\ & (\lambda_k)_j \left(g_j(x_k) + \nabla g_j(x_k)^T d_k - t_k \right) = 0 \ (j \in I), \\ & v_k t_k = 0, \qquad h_k(t_k - \theta(x_k)) = 0, \\ & \lambda_k \ge 0, \quad v_k \ge 0, \quad h_k \ge 0. \end{aligned} \tag{9}$$

Now, the algorithm for solving the problem (P) can be stated as follows.

Algorithm 1. Consider the following.

Step 1. Initialization: choose an initial point $x_1 \in R^n$, an initial filter $\mathscr{F}_1$, an initial parameter $b_1 > 0$, a symmetric and positive definite matrix $B_1 \in R^{n \times n}$, and $\delta > 0$, and $s_\theta \in (0, 1)$. Choose $\rho \in (0, (1/2))$, $\delta_1, \delta_2 > 0$, and $\gamma_f, \gamma_\theta \in (0, 1)$. Set $k = 1$.

Step 2. Solve subproblem (8) with x_k, B_k, and b_k to obtain the solution (d_k, t_k), and let (λ_k, v_k, h_k) be the Lagrange multiplier. Stop if $d_k = 0$ and $t_k = 0$.

Step 3. If $d_k = 0$ but $t_k \ne 0$, set $x_{k+1} = x_k$ and $\mathscr{F}_{k+1} = \mathscr{F}_k$; go to Step 6.

Step 4. If $d_k \ne 0$ and $\nabla f(x_k)^T d_k \ge 0$, go to Step 8. If $d_k \ne 0$ and $\nabla f(x_k)^T d_k < 0$, using backtracking line search consider the following.

> *Step 4.1.* Initial line search: set $l = 1$ and $\alpha_{k,l} = 1$.

Step 4.2. Compute a new trial point. If the trial step size $\alpha_{k,l} < \alpha_k^{\min}$, go to Step 8. Otherwise, compute new trial point $x_k(\alpha_{k,l}) = x_k + \alpha_{k,l} d_k$. Check acceptability to the filter; if $(\theta(x_k(\alpha_{k,l})), f(x_k(\alpha_{k,l}))) \in \mathscr{F}_k$, reject the trial step size and go to Step 4.4.

Step 4.3. Check sufficient decrease with respect to current iterate point.

> 4.3.1. Case I: condition (5) holds. If the Armijo condition (6) holds, accept the trial step (that is, an f-type iteration), and go to Step 4.5; otherwise, go to Step 4.4.
>
> 4.3.2. Case II: condition (5) does not hold. If (2) holds, accept the trial step (that is, an h-type iteration), and go to Step 4.5; otherwise, go to Step 4.4.

Step 4.4. Choose a new trial size $\alpha_{k,l+1} = \rho \alpha_{k,l}$. Set $l = l + 1$, and go to Step 4.2.

Step 4.5. Accept trial point. Set $\alpha_k = \alpha_{k,l}$ and $x_{k+1} = x_k(\alpha_k)$.

Step 5. Augment filter if it is necessary. If k is not an f-type iteration, augment the filter using (4); otherwise leave the filter unchanged; that is, set $\mathscr{F}_{k+1} = \mathscr{F}_k$.

Step 6. Update parameters. Compute a_k by

$$a_k = \min \left\{ \|d_k\|^{-1}, \|\lambda_k\|_1 + \delta_1 \right\}. \tag{10}$$

Set

$$b_{k+1} = \begin{cases} b_k, & \text{if } b_k \ge a_k, \\ b_k + \delta_2, & \text{otherwise.} \end{cases} \tag{11}$$

Step 7. Update B_k to B_{k+1}. Go to Step 2 with k replaced by $k + 1$.

Step 8. Obtain a new point x_{k+1} from a feasible restoration phrase. Set $k = k + 1$, and go to Step 2.

Remark 2. The mechanisms of the filter could ensure that $(\theta(x_k), f(x_k)) \notin \mathscr{F}_k$.

Remark 3. The feasibility restoration phrase in Step 8 could be any iterative algorithm with the goal of finding a less infeasible point; for example, a nonlinear optimization algorithm is applied to minimise θ such as Algorithm B in [14].

4. Global Convergence of Algorithm

In this section, we will show the proposed algorithm is well defined and globally convergent under some mild conditions. Then throughout this paper, we always assume that the following assumptions hold.

Assumptions

(A1) The sequence $\{x_k\}$ remains in compacted subset $X \in R^n$.

(A2) The functions f and g_j, $j = 1, 2, \ldots m$, are twice continuously differentiable.

(A3) For any $x \in X$, the vectors $\{\nabla g_j(x), j \in I(x)\}$ are linearly independent, where $I(x) = \{j \in I \mid g_j(x) = \theta(x)\}$.

(A4) There exist two constants $q \geq p > 0$ such that the matrices sequence $\{B_k\}$ satisfy $p\|d\|^2 \leq d^T B_k d \leq q\|d\|^2$ for all k and $d \in R^n$.

The first lemma shows that there is no cycle between Step 3 and Step 6 in Algorithm 1.

Lemma 4. *Let $\{x_k\}$ and $\{d_k\}$ be generated by Algorithm 1. If $d_k = 0$ and $t_k > 0$, then there exists a finite positive integer r such that $d_{k+r} \neq 0$.*

Proof. Suppose by contradiction that $d_{k+r} = 0$ and $t_{k+r} > 0$ for all $r \geq 1$ and some $k \in \mathbb{N}$. Then, we have $x_{k+r} = x_k$, and hence $t_{k+r} = \theta(x_{k+r}) = \theta(x_k) = t_k > 0$ and $v_{k+r} = 0$. It follows from (9) and (10) that

$$b_{k+r} = \|\lambda_{k+r}\|_1 - h_{k+r} < \|\lambda_{k+r}\|_1 + \delta_1 = a_{k+r}. \qquad (12)$$

This together with (11) implies that $b_{k+r+1} = b_{k+r} + \delta_2$ for all $r \geq 1$. As a result, we have

$$b_{k+r} \longrightarrow +\infty,$$
$$\|\lambda_{k+r}\|_1 = b_{k+r} + h_{k+r} \longrightarrow +\infty \quad \text{as } r \longrightarrow +\infty. \qquad (13)$$

Because $x_{k+r} = x_k$ for all $r \geq 1$, we denote $x_{k+r} = \overline{x}$. So, we get

$$0 < t_k = \lim_{r \to +\infty} t_{k+r} = \lim_{r \to +\infty} \theta(x_{k+r}) = \theta(\overline{x}),$$
$$\lim_{r \to +\infty} \left(g_i(x_{k+r}) + \nabla g_i(x_{k+r})^T d_{k+r} \right)$$
$$= \lim_{r \to +\infty} g_i(x_{k+r}) = g_i(\overline{x}) < \theta(\overline{x}) = t_k,$$
$$\forall i \notin I(\overline{x}). \qquad (14)$$

Therefore, we claim that for all $r \geq 1$,

$$g_i(x_{k+r}) + \nabla g_i(x_{k+r})^T d_{k+r} < t_{k+r}, \quad \forall i \notin I(\overline{x}). \qquad (15)$$

We also have by (9) and (15) that for all $r \geq 1$, $(\lambda_{k+r})_i = 0$, for all $i \notin I(\overline{x})$. This yields

$$\nabla f(x_{k+r}) + \sum_{j \in I(\overline{x})} \nabla g_j(x_{k+r})(\lambda_{k+r})_j = 0. \qquad (16)$$

Without loss of generality, we assume that $\lim_{r \to +\infty} \lambda_{k+r} / \|\lambda_{k+r}\|_1 = \widehat{\lambda}$. Dividing by $\|\lambda_{k+r}\|_1$ in both sides of the last equality and taking limits as $r \to +\infty$, we obtain

$$\sum_{j \in I(\overline{x})} \widehat{\lambda}_j \nabla g_j(\overline{x}) = 0, \quad \widehat{\lambda}_j \geq 0, \ j \in I(\overline{x}). \qquad (17)$$

Moreover, there exists at least one $j \in I(\overline{x})$ satisfying $\widehat{\lambda}_j > 0$. This contradicts Assumption (A3). $\qquad \square$

The next lemma shows that the value of the parameter b_k increased only a finite number of times.

Lemma 5. *Let $\{x_k\}$ be an infinite sequence generated by Algorithm 1; then there exists an integer k_0 such that $b_k = b_{\max}$ for all $k \geq k_0$. In addition, $b_{\max} = \max\{b_k\}$.*

Proof. The proof is by contradiction. Suppose that $b_k \to +\infty$, as $k \to +\infty$. The step 6 of Algorithm 1 implies that inequality

$$b_k < a_k = \min \left\{ \|d_k\|^{-1}, \|\lambda_k\|_1 + \delta_1 \right\} \qquad (18)$$

holds for infinitely many k. In addition, there exists a subsequence satisfying

$$\lim_{k_i \to \infty} \|d_{k_i}\| = 0, \qquad \lim_{k_i \to \infty} \|\lambda_{k_i}\|_1 = +\infty. \qquad (19)$$

Without loss of generality, using Assumption (A1), we suppose that $x_{k_i} \to \overline{x}$ as $k_i \to \infty$. In a way similar to the proof of Lemma 4, we can derive, for all sufficiently large k_i,

$$g_j(x_{k_i}) + \nabla g_j(x_{k_i})^T d_{k_i} < t_{k_i}, \quad \forall j \notin I(\overline{x}). \qquad (20)$$

By (9) we have that for all k_i sufficiently large, $(\lambda_{k_i})_j = 0$, for all $j \notin I(\overline{x})$, which yields

$$\nabla f(x_{k_i}) + B_{k_i} d_{k_i} + \sum_{j \in I(\overline{x})} \nabla g_j(x_{k_i})(\lambda_{k_i})_j = 0. \qquad (21)$$

We can also assume that $\lim_{k_i \to +\infty} \lambda_{k_i} / \|\lambda_{k_i}\|_1 = \widehat{\lambda}$. Dividing by $\|\overline{\lambda_{k_i}}\|_1$ in both sides of the last equality and taking limits as $k_i \to +\infty$, then

$$\sum_{j \in I(\overline{x})} \widehat{\lambda}_j \nabla g_j(\overline{x}) = 0, \quad \widehat{\lambda}_j \geq 0, \ j \in I(\overline{x}). \qquad (22)$$

Furthermore, there exists at least one $j \in I(\overline{x})$ satisfying $\widehat{\lambda}_j > 0$. This contradicts Assumption (A3).

Because b_k is increased only finitely many times, it is easy to see that $b_{\max} = \max\{b_k\}$. $\qquad \square$

Lemma 6. *The line search is actually performed; that is, our method can generate d_k such that $\nabla f(x_k)^T d_k < 0$ at some iteration.*

Proof. The proof is by contradiction. Suppose that d_k satisfying $\nabla f(x_k)^T d_k < 0$ cannot be generated and the current iterate point cannot be changed as Step 3; then the iterate points cannot be escaped from the feasibility restoration phrase. Therefore, all points of $\{x_k\}$ are generated by the the feasibility restoration phrase, which follows $|\mathscr{A}| = \infty$. Now we first prove that

$$\lim_{k \to \infty} \theta(x_k) = 0. \qquad (23)$$

Consider an infinite subsequence $\{k_i\}$ of $\mathscr{A}$ with $\theta(x_{k_i}) \geq \varepsilon$ for some $\varepsilon > 0$. At each iteration k_i, $(\theta(x_{k_i}), f(x_{k_i}))$ is added to

the filter which means that no other (θ, f) can be added to the filter at a later stage within the area $[\theta(x_{k_i}) - \gamma_\theta \theta(x_{k_i}), \theta(x_{k_i})] \times [f(x_{k_i}) - \gamma_f \theta(x_{k_i}), f(x_{k_i})]$, and the area of each square is at least $\gamma_\theta \gamma_f \varepsilon^2$. By Assumption (A1), we have $0 \le \theta(x) \le \theta^{\max}$ and $f^{\min} \le f(x) \le f^{\max}$, and then (θ, f) associated with the filter are restricted to $\mathscr{B} = [0, \theta^{\max}] \times [f^{\min}, f^{\max}]$. Hence, $\mathscr{B}$ is completely covered by at most a finite number of such areas, which forms contradiction with the infinite subsequence $\{k_i\}$ satisfying (3). This means that (23) is true.

We can see that $(0, \ldots, 0, \theta(x_k))^T \in R^{n+1}$ is a feasible point of QP subproblem (8); then we get

$$\nabla f(x_k)^T d_k + \frac{1}{2}(d_k)^T B_k d_k + b_k t_k < b_k \theta(x_k). \tag{24}$$

Because our method does not stop at Step 2, it holds that $(d_k, t_k) \nrightarrow (0, 0)$. By $0 \le t_k \le \theta(x_k)$, (23) implies that $\lim_{k \to \infty} t_k = 0$. With the facts $(d_k, t_k) \nrightarrow (0, 0)$ and $\lim_{k \to \infty} t_k = 0$, we can assume that $\|d_k\| > \epsilon_0$ for some $\epsilon_0 > 0$. Let $\epsilon_1 = p\epsilon_0^2/4b_{\max}$; (23) implies that for k sufficiently large enough it holds that $\theta(x_k) < \epsilon_1$, and then $b_k \theta(x_k) \le (pb_k \epsilon_0^2/4b_{\max}) \le (1/4)(d_k)^T B_k d_k$; thus,

$$\nabla f(x_k)^T d_k < -\frac{1}{2}(d_k)^T B_k d_k - b_k t_k + b_k \theta(x_k)$$

$$< -\frac{1}{2}(d_k)^T B_k d_k + b_k \theta(x_k) \tag{25}$$

$$\le -\frac{1}{4}(d_k)^T B_k d_k \le -\frac{p\epsilon_0^2}{4} < 0,$$

which is a contradiction. □

Lemma 7. *If $\theta(x_k) = 0$, then $m_k(\alpha) < 0$ for all $\alpha \in (0, 1]$. In addition, $\Theta_k = \min\{\theta : (\theta, f) \in \mathscr{F}_k\} > 0$ for all k.*

Proof. If $\theta(x_k) = 0$, then $(0, \ldots, 0)^T \in R^{n+1}$ is a feasible point of subproblem (8), so we get

$$\frac{m_k(\alpha)}{\alpha} = \nabla f(x_k)^T d_k \le -\frac{1}{2}(d_k)^T B_k d_k - b_k t_k < 0. \tag{26}$$

For proof of second statement please see [11, Lemma 4]. □

Theorem 8. *Suppose all stated assumptions hold. Then*

$$\lim_{k \to \infty} \theta(x_k) = 0. \tag{27}$$

Proof. Please see [11, Theorem 1]. □

Lemma 9. *If Algorithm 1 generates an infinite sequence $\{x_k\}$, then the sequence $\{d_k\}$ is bounded; that is to say, there exists a constant $M_d > 0$, such that for all $k \ge 0$,*

$$\|d_k\| \le M_d. \tag{28}$$

Proof. We can see that $(0, \ldots, 0, \theta(x_k))^T \in R^{n+1}$ is a feasible point of QP subproblem (8), so we get

$$\nabla f(x_k)^T d_k + \frac{1}{2}(d_k)^T B_k d_k + b_k t_k \le b_k \theta(x_k). \tag{29}$$

From Theorem 8, there exists a constant M_θ such that $\theta(x_k) \le M_\theta$; thus,

$$-\|\nabla f(x_k)\| \cdot \|d_k\| + \frac{1}{2} p \|d_k\|^2$$

$$\le \nabla f(x_k)^T d_k + \frac{1}{2}(d_k)^T B_k d_k \tag{30}$$

$$+ b_k t_k \le b_k \theta(x_k) \le b_{\max} M_\theta,$$

which implies that $\{d_k\}$ is bounded; then, there exists a constant $M_d > 0$ such that

$$\|d_k\| \le M_d. \tag{31}$$

□

Lemma 10. *If $\{x_{k_i}\}$ is a subsequence of iterate points for which $\|d_{k_i}\| \ge \epsilon$ with some constant $\epsilon > 0$ independent of i, then there exist constants $\epsilon_1, \epsilon_2 > 0$, such that*

$$\theta(x_{k_i}) \le \epsilon_1 \implies m_{k_i}(\alpha) \le -\epsilon_2 \alpha \tag{32}$$

for all i and $\alpha \in (0, 1]$.

Proof. We can see that $(0, \ldots, 0, \theta(x_{k_i}))^T \in R^{n+1}$ is a feasible point of QP subproblem (8); then we get

$$\nabla f(x_{k_i})^T d_{k_i} + \frac{1}{2}(d_{k_i})^T B_{k_i} d_{k_i} + b_{k_i} t_{k_i} < b_{k_i} \theta(x_{k_i}). \tag{33}$$

Let $\epsilon_1 = p\epsilon^2/4b_{\max}$; then $b_{k_i} \theta(x_{k_i}) \le (pb_{k_i} \epsilon^2/4b_{\max}) \le (1/4)(d_{k_i})^T B_{k_i} d_{k_i}$; thus,

$$\frac{m_{k_i}(\alpha)}{\alpha} = \nabla f(x_{k_i})^T d_{k_i}$$

$$< -\frac{1}{2}(d_{k_i})^T B_{k_i} d_{k_i} - b_{k_i} t_{k_i} + b_{k_i} \theta(x_{k_i})$$

$$< -\frac{1}{2}(d_{k_i})^T B_{k_i} d_{k_i} + b_{k_i} \theta(x_{k_i}) \tag{34}$$

$$\le -\frac{1}{4}(d_{k_i})^T B_{k_i} d_{k_i} \le -\frac{p\epsilon^2}{4}.$$

Define $\epsilon_2 = p\epsilon^2/4$; then it implies that $m_{k_i}(\alpha) \le -\epsilon_2 \alpha$. □

Lemma 11. *Suppose that the filter is augmented only a finite number of times; that is, $|\mathscr{A}| < \infty$. Then*

$$\lim_{k \to \infty} d_k = 0. \tag{35}$$

Proof. The proof of this lemma can be found in [11, Lemma 8]. There the proof is stated for slightly different circumstances, but it is easy to verify that it is still valid in our context. □

Lemma 12. *There exists some constant $C_f > 0$, such that*

$$f(x_k + \alpha d_k) - f(x_k) - m_k(\alpha) \le C_f \alpha^2 \|d_k\|^2, \tag{36}$$

for $\alpha \in (0, 1]$.

Proof. The inequality follows directly from the second order Taylor expansion. $\qquad\square$

Lemma 13. *Let $\{x_{k_i}\}$ be a subsequence with $m_{k_i}(\alpha) \leq -\alpha\epsilon_2$ for some constant $\epsilon_2 > 0$ independent of k_i and for all $\alpha \in (0,1]$. Then there exists some constant $\overline{\alpha} > 0$ such that*

$$f\left(x_{k_i} + \alpha d_{k_i}\right) - f\left(x_{k_i}\right) \leq \eta m_{k_i}(\alpha), \tag{37}$$

for all k_i and $\alpha \leq \overline{\alpha}$.

Proof. Let M_d and C_f be the constants from Lemmas 9 and 12. It then follows from Lemma 12 for all $\alpha \leq \overline{\alpha}$ with $\overline{\alpha} = (1-\eta)\epsilon_2/C_f M_d^2$ that

$$\begin{aligned}
f\left(x_{k_i} + \alpha d_{k_i}\right) &- f\left(x_{k_i}\right) - m_{k_i}(\alpha) \\
&\leq C_f \alpha^2 \left\|d_{k_i}\right\|^2 \leq \alpha(1-\eta)\epsilon_2 \\
&\leq -(1-\eta)m_{k_i}(\alpha).
\end{aligned} \tag{38}$$

That is,

$$f\left(x_{k_i} + \alpha d_{k_i}\right) - f\left(x_{k_i}\right) \leq \eta m_{k_i}(\alpha). \tag{39}$$

$\qquad\square$

Lemma 14. *Let $\{x_{k_i}\}$ be a subsequence with $m_{k_i}(\alpha) \leq -\alpha\epsilon_2$ for some constant $\epsilon_2 > 0$ independent of k_i and for all $\alpha \in (0,1]$. Then there exist constants $C_1, C(x_{k_i}) > 0$ so that*

$$\left(\theta\left(x_{k_i} + \alpha d_{k_i}\right), f\left(x_{k_i} + \alpha d_{k_i}\right)\right) \notin \mathscr{F}_{k_i}, \tag{40}$$

for all k_i and $\alpha \leq \min\{C_1, C(x_{k_i})\}$.

Proof. Let M_d, C_f be the constants from Lemmas 9 and 12. The mechanisms of Algorithm 1 ensure that

$$\left(\theta\left(x_{k_i}\right), f\left(x_{k_i}\right)\right) \notin \mathscr{F}_{k_i}. \tag{41}$$

Define $\mathscr{F}_{k_i}^{(1)} = \{(\theta, f) \in \mathscr{F}_{k_i} \mid f > f(x_{k_i})\}$, $\mathscr{F}_{k_i}^{(2)} = \mathscr{F}_{k_i} \setminus \mathscr{F}_{k_i}^{(1)}$, and $C_1 = \min\{1, \epsilon_2/(M_d^2 C_f)\}$. For $\alpha \leq C_1$ we have $\alpha^2 \leq (\alpha\epsilon_2/M_d^2 C_f) \leq (-m_{k_i}(\alpha)/C_f \|d_{k_i}\|^2)$, or equivalently

$$m_{k_i}(\alpha) + C_f \alpha^2 \left\|d_{k_i}\right\| \leq 0, \tag{42}$$

and it follows with (36) that

$$f\left(x_{k_i} + \alpha d_{k_i}\right) \leq f\left(x_{k_i}\right). \tag{43}$$

So for $\alpha \leq C_1$ we have $(\theta(x_{k_i} + \alpha d_{k_i}), f(x_{k_i} + \alpha d_{k_i})) \notin \mathscr{F}_{k_i}^{(1)}$.

Let $\theta_{\min} = \min\{\theta \mid \theta \in \mathscr{F}_{k_i}^{(2)}\}$; from (41) we know that $\theta(x_{k_i}) < \theta_{\min}$. Therefore, there exists some constant $C(x_{k_i})$ such that for $\alpha \leq C(x_{k_i})$, it holds that $\theta(x_{k_i} + \alpha d_{k_i}) < \theta_{\min}$. So for $\alpha \leq C(x_{k_i})$ we have $(\theta(x_{k_i} + \alpha d_{k_i}), f(x_{k_i} + \alpha d_{k_i})) \notin \mathscr{F}_{k_i}^{(2)}$. Thereby for $\alpha < \min\{C_1, C(x_{k_i})\}$, we have

$$\left(\theta\left(x_{k_i} + \alpha d_{k_i}\right), f\left(x_{k_i} + \alpha d_{k_i}\right)\right) \notin \mathscr{F}_{k_i}^{(1)} \cup \mathscr{F}_{k_i}^{(2)} = \mathscr{F}_{k_i}. \tag{44}$$

$\qquad\square$

Lemma 15. *Let $\{x_{k_i}\}$ be a subsequence with $\|d_{k_i}\| \geq \epsilon$ for some constant $\epsilon > 0$ independent of k_i. Then there exists $K \in \mathbb{N}$ such that the filter is not augmented in iteration K; that is, $K \notin \mathscr{A}$.*

Proof. Since we have $\lim_{i\to\infty} \theta(x_{k_i}) = 0$ by Theorem 8, it follows from Lemma 10 that there exist constants $\epsilon_1, \epsilon_2 > 0$ such that

$$\theta\left(x_{k_i}\right) \leq \epsilon_1, \qquad m_{k_i}(\alpha) \leq -\alpha\epsilon_2, \tag{45}$$

for k_i sufficiently large and $\alpha \in (0,1]$. Without loss of generality we can assume that (45) is valid for all k_i.

We can now apply Lemmas 13 and 14 to obtain the constants $\overline{\alpha}, C_1, C(x_{k_i}) > 0$. As $\lim_{i\to\infty} \theta(x_{k_i}) = 0$, choose a constant C_2 and a sufficiently large $K \in \{k_i\}$ such that $C(x_K) > C_2\theta(x_K) > 0$ and

$$\theta\left(x_K\right) < \min\left\{\frac{\overline{\alpha}}{C_2}, \frac{C_1}{C_2}, \left[\frac{\rho C_2 \epsilon_2}{\delta}\right]^{1/(s_\theta - 1)}\right\}. \tag{46}$$

We note that (46) implies that

$$\frac{\delta[\theta(x_K)]^{s_\theta}}{\epsilon_2} < \rho C_2 \theta(x_K), \tag{47}$$

as well as

$$C_2 \theta(x_K) < \min\{\overline{\alpha}, C_1\}. \tag{48}$$

Now define

$$\beta_K = C_2 \theta(x_K) = \min\{\overline{\alpha}, C_1, C_2 \theta(x_K)\}. \tag{49}$$

Lemmas 13 and 14 then imply that a trial step size $\alpha_{K,l} \leq \beta_K$ satisfies

$$\begin{aligned}
f\left(x_K(\alpha_{K,l})\right) &\leq f(x_K) + \eta m_K(\alpha_{K,l}), \\
\left(\theta\left(x_K(\alpha_{K,l})\right), f\left(x_K(\alpha_{K,l})\right)\right) &\notin \mathscr{F}_K.
\end{aligned} \tag{50}$$

If we now denote with $\alpha_{K,L}$ the first step size satisfying (50), the backtracking line search procedure in Step 4 then implies that for $\alpha \geq \alpha_{K,l}$

$$\alpha \geq \rho\beta_K = \rho C_2 \theta(x_K) > \frac{\delta[\theta(x_K)]^{s_\theta}}{\epsilon_2}, \tag{51}$$

and therefore for $\alpha \geq \alpha_{K,l}$

$$\delta[\theta(x_K)]^{s_\theta} < \alpha\epsilon_2 \leq -m_K(\alpha). \tag{52}$$

This means that $\alpha_{K,l}$ and all previous trial step sizes are f-step sizes. Hence the method does not switch to feasibility restoration phrase in Step 4 for those trial step sizes. Therefore, $\alpha_{K,l}$ is the accepted step size α_K indeed. Since it satisfies both the f-type switching conditions (5) and (6), the filter is not augmented in iteration K. $\qquad\square$

Lemma 16. *Suppose that the filter is augmented infinite number of times; that is, $|\mathscr{A}| = \infty$. Then*

$$\lim_{k\to\infty, k\in\mathscr{A}} d_k = 0. \tag{53}$$

TABLE 1: The detail information of the numerical results for feasible initial point.

No.	n	m	NIT	NOF	NOG	FV
hs001	2	1	35	88	65	$2.394473e - 014$
hs003	2	1	4	12	10	$-4.440892e - 016$
hs004	2	2	2	2	2	$2.666667e + 000$
hs005	2	4	6	14	12	$-1.913222e + 000$
hs012	2	1	5	12	11	$-3.000000e + 001$
hs033	3	6	3	4	3	$-4.585786e + 000$
hs035	3	4	7	15	13	$1.111111e - 001$
hs037	3	8	9	23	20	$-3.456000e + 003$
hs043	4	3	10	14	13	$-4.399999e + 001$
hs044	4	10	5	10	10	$-1.500000e + 001$
hs076	4	7	6	7	7	$-4.681818e + 000$
hs086	5	15	4	10	9	$-3.234868e + 001$
hs100	7	4	6	22	20	$6.806301e + 002$
hs110	10	20	5	12	12	$-4.577847e + 001$
hs113	10	8	12	16	15	$2.430621e + 001$
hs117	15	20	17	35	25	$3.234868e + 001$
hs118	15	59	21	42	36	$6.648204e + 002$

TABLE 2: The detail information of the numerical results for infeasible initial point.

No.	n	m	NIT	NOF	NOG	FV
hs002	2	1	9	34	30	$5.042618e - 002$
hs010	2	1	4	14	12	$-1.000000e + 000$
hs011	2	1	12	17	16	$-8.498464e + 000$
hs013	2	3	16	30	26	$1.000021e + 000$
hs015	2	3	3	15	13	$3.065000e + 002$
hs016	2	5	12	27	24	$2.500000e - 001$
hs017	2	5	11	26	23	$1.000000e + 000$
hs021	2	5	3	6	6	$-9.995999e + 001$
hs022	2	2	4	4	4	$9.999999e - 001$
hs023	2	9	6	12	11	$2.000000e + 000$
hs059	2	7	14	34	32	$-6.749505e + 000$
hs065	3	7	8	12	10	$9.535282e - 001$
hs096	6	16	34	113	104	$1.561953e - 002$
hs108	9	14	12	26	22	$-8.660256e + 001$

Proof. Suppose that $\lim_{k \to \infty, k \in \mathscr{A}} \sup \|d_k\| > 0$. Then there exist a subsequence $\{x_{k_j}\}$ of $\{x_k, k \in \mathscr{A}\}$ and a constant $\epsilon > 0$, such that $\lim_{j \to \infty} \theta(x_{k_j}) = 0$ and $\|d_{k_j}\| \geq \epsilon$ for all k_j. Applying Lemma 15 to $\{x_{k_j}\}$, we see that there is an iteration k_j, in which the filter is not augmented; that is to say, $k_j \notin \mathscr{A}$. This contradicts the choice of $\{x_{k_j}\}$, and then the claim follows. □

Although $0 \leq t_k \leq \theta(x_k)$ and Theorem 8 can imply that $\lim_{k \to \infty} t_k = 0$, the next lemma shows better result.

Lemma 17. *Let $\{x_{k_i}\}$ be a subsequence with $d_{k_i} \to 0$; then $t_{k_i} = 0$ for k_i sufficiently large.*

Proof. Consider that $d_{k_i} \to 0$ together with Lemma 5 implies for k sufficiently large

$$b_{k_i} \geq a_{k_i} = \min \left\{ \|d_{k_i}\|^{-1}, \|\lambda_{k_i}\|_1 + \delta_1 \right\} = \|\lambda_{k_i}\|_1 + \delta_1. \quad (54)$$

On the other hand, we have from (9) that

$$b_{k_i} = \|\lambda_{k_i}\|_1 + v_{k_i} - h_{k_i}. \quad (55)$$

Thereby, it holds that $v_{k_i} > h_{k_i} + \delta_1 > 0$ for k_i sufficiently large. So we get $t_{k_i} = 0$ for k_i sufficiently large. □

Theorem 18. *Suppose that all stated assumptions hold; the outcome of applying Algorithm 1 is one of the following.*

(A) *A KKT point of problem* (P) *is found (Algorithm 1 stops at Step 2).*

Table 3: Comparison of our algorithm with two methods in [5, 7].

No.	Pro.	(n, m)	FilterSQP1 (NIT; NOF)	CSQP (NIT; NOF)	JSQP (NIT; NOF)
hs001	PBRT11	(2, 1)	(35; 88)	(332; 442)	(627; 700)
hs002	PBRT12	(2, 1)	(9; 34)	(11; 119)	F
hs003	QBRT11	(2, 1)	(4; 12)	(3; 3)	(3; 3)
hs004	PBRT13	(2, 2)	(2; 2)	(2; 2)	(2; 2)
hs005	GBRT11	(2, 4)	(6; 14)	(15; 32)	(17; 35)
hs010	LQRT11	(2, 1)	(4; 14)	(2; 2)	(2; 2)
hs011	QQRT12	(2, 1)	(12; 17)	F	F
hs012	QQRT13	(2, 1)	(5; 12)	(2; 9)	(3; 338)
hs013	QPRT11	(2, 3)	(16; 30)	(20; 20)	F
hs015	PQRT11	(2, 3)	(3; 15)	(3; 33)	(7; 498)
hs022	QQRT16	(2, 2)	(3; 4)	(7; 8)	(7; 8)
hs100	PPRP17	(7, 4)	(6; 22)	(7; 120)	(7; 7128)
hs108	QQRP16	(9, 14)	(12; 26)	(12; 12)	(12; 28)

(B) *All limit points are feasible, and there exists at least one limit point x^* of $\{x_k\}$ which is a KKT point for the inequality constrained NLP (1); that is to say,*

$$\lim_{k \to \infty} \theta(x_k) = 0, \qquad (56)$$

and there exists at least one subsequence $\mathcal{K}$ such that

$$\lim_{k \to \infty, k \in \mathcal{K}} d_k = \lim_{k \to \infty, k \in \mathcal{K}} t_k = 0. \qquad (57)$$

Proof. (A) It is obviously true.

(B) Equation (56) follows from Theorem 8. In order to show (57), we now distinguish it into two cases:

(i) if the filter is augmented only a finite number of times, from Lemmas 11 and 17 we know every limit point is a KKT point for the inequality constrained NLP (1), so the claim is true,

(ii) if the filter is augmented infinite number of times, from Lemmas 16 and 17 it holds that

$$\lim_{k \to \infty, k \in \mathscr{A}} d_k = \lim_{k \to \infty, k \in \mathscr{A}} t_k = 0. \qquad (58)$$

Let $\mathcal{K} = \mathscr{A}$; the claim is also true.

In a word, if Algorithm 1 generates an infinite sequence of iterate $\{x_k\}$, all limit points are feasible, and there exists at least one limit point x^* of $\{x_k\}$ which is a KKT point for the inequality constrained NLP (1). □

5. Numerical Experiments

In this section, we carry out some typical numerical experiments based on Algorithm 1. In the whole process, the program is coded in MATLAB 7.0, and we use its optimization toolbox to solve the QP subproblem (8).

(1) During the numerical experiments, updating of B_k is done by the following damped BFGS formula (see [27, Chapter 18]):

$$B_{k+1} = B_k - \frac{B_k s_k (s_k)^T B_k}{(s_k)^T B_k s_k} + \frac{r_k (r_k)^T}{(s_k)^T r_k}, \qquad (59)$$

where $s_k = x_{k+1} - x_k$, $r_k = \widehat{\theta}_k y_k + (1 - \widehat{\theta}_k) B_k s_k$, $y_k = \nabla_x L(x_{k+1}, \lambda_k) - \nabla_x L(x_k, \lambda_k)$ and

$$\widehat{\theta}_k = \begin{cases} 1 & \text{if } (s_k)^T y_k \geq 0.2 \, (s_k)^T B_k s_k, \\ \dfrac{0.8(s_k)^T B_k s_k}{(s_k)^T B_k s_k - (s_k)^T y_k} & \text{otherwise.} \end{cases} \qquad (60)$$

(2) The algorithm parameters are set as follows:

$$\begin{aligned} & B_1 = E_n, && b_1 = 111, && s_\theta = 0.75, \\ & \delta = 1, && \delta_1 = 10, && \delta_2 = 11, \\ & \eta = 0.25, && \rho = 0.5, && \gamma_f = \gamma_\theta = 0.01, \\ & && u = 1000. && \end{aligned} \qquad (61)$$

(3) The stop criteria are $\|d_0^k\|$ sufficiently small, where $d_0^k = (d_k^T, t_k)^T$. In particular, the stop criteria of Step 2 are changed to

$$\text{if } \left\| d_0^k \right\| \leq 10^{-7}, \quad \text{stop.} \qquad (62)$$

(4) Now we report the numerical results over a set of problems from [28], where no equality constraints are contained and an initial point is provided for each problem. In addition, we tested our algorithm in two cases: a feasible initial point is provided for each problem or an infeasible initial point is given. These results are listed in Tables 1 and 2, respectively. Because the quadratic subproblem (1) is modified but the filter technique is not used in [5, 7] and the filter technique is used but the quadratic subproblem (1) is not modified in [18, 23], for the sake of comparing equally, under the same initial points, we select examples tested both in [5, 7] and examples tested both in [18, 23] to do the comparisons. The comparison numerical results among our algorithm and two groups of methods are listed in Tables 3 and 4, respectively.

In Tables 1, 2, 3, and 4, the notations mean as follows:

(i) No.: the number of problems in [28],

(ii) Pro.: the name of problems in [28],

(iii) n: the number of variables,

(iv) m: the number of inequality constraints,

(v) NIT: the number of iterations,

TABLE 4: Comparison of our algorithm with two methods in [18, 23].

No.	The initial point	Method	NIT	NOF	NOG
		FilterSQP1	5	12	11
hs012	(0, 0)	FilterSQP2	7	10	8
		FilterSQP3	8	9	9
		FilterSQP1	3	4	3
hs033	(0, 0, 3)	FilterSQP2	3	4	4
		FilterSQP3	4	5	5
		FilterSQP1	10	14	13
hs043	(0, 0, 0, 0)	FilterSQP2	11	19	12
		FilterSQP3	13	15	14
		FilterSQP1	8	12	10
hs065	(−5, 5, 0)	FilterSQP2	15	11	10
		FilterSQP3	9	10	10
		FilterSQP1	6	7	7
hs076	(0.5, 0.5, 0.5, 0.5)	FilterSQP2	6	7	7
		FilterSQP3	6	7	7
		FilterSQP1	12	16	15
hs113	(2, 3, 5, 5, 1, 2, 7, 3, 6, 10)	FilterSQP2	14	25	15
		FilterSQP3	16	18	17

(vi) NOF: the number of evaluations of objective functions,

(vii) NOG: the number of evaluations of objective function gradient,

(viii) FV: the final value of the objective function,

(ix) F: it indicates the algorithm's failure on the problem,

(x) FilterSQP1: our algorithm in this paper,

(xi) JSQP: the algorithm proposed in [5],

(xii) CSQP: the algorithm proposed in [7],

(xiii) FilterSQP2: the algorithm proposed in [18],

(xiv) FilterSQP3: the algorithm proposed in [23].

From Tables 1 and 2, we can see that our algorithm executes well for these problems taken from [28]. Whenever the initial point is feasible or not, the results are promising. From the computation efficiency in Tables 3 and 4, we should point out that our algorithm is competitive with some existed SQP methods in terms of the number of iterations, for example, [5, 7, 18, 23].

According to the forms of examples listed in these two groups of papers, we show the numerical results by two different representations in Tables 3 and 4. From Table 3, our algorithm is competitive with the first group of methods, that is, two methods in [5, 7]. Especially utilizing the algorithm in this paper, the number of iterations for problem hs001 (i.e., PBRT11) is much smaller than ones in other two algorithms. And the problem hs011 (i.e., QQRT12) converges to a good approximate optimal solution with small iterations, while algorithms CSQP and JSQP both fail on it. From Table 4, our algorithm is also competitive with the second group of methods, that is, two methods in [18, 23]. Although the filter technique is taken in these three algorithms, the quadratic subproblem (1) in [18, 23] may be inconsistent, while the subproblem is always consistent, and it only needs to solve one QP subproblem at each iteration in our algorithm, which is simpler and can be applied much more conveniently.

All results summarized in Tables 1–4 show that our algorithm is practical and effective.

6. Conclusion

In this paper, combining a modification strategy of QP subproblem and the filter technique, we present a line search filter SQP method for inequality constrained optimization. This method can start with any arbitrary initial point rather than a feasible initial point; it makes use of a backtracking line search procedure to generate step size and the efficiency of the filter technique to determine step acceptance; it only needs to solve one QP subproblem at each iteration, and the subproblem is always consistent. Under some mild conditions, the method is well defined, and the global convergence property is obtained. Many numerical experiments in Section 5 show that our algorithm is effective.

Acknowledgments

References

[1] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," in *Acta Numerica*, pp. 1–51, Cambridge University Press, Cambridge, UK, 1995.

[2] J. V. Burke and S.-P. Han, "A robust sequential quadratic programming method," *Mathematical Programming*, vol. 43, no. 3, pp. 277–303, 1989.

[3] G. Zhou, "A modified SQP method and its global convergence," *Journal of Global Optimization*, vol. 11, no. 2, pp. 193–205, 1997.

[4] J.-L. Zhang and X.-S. Zhang, "A modified SQP method with nonmonotone linesearch technique," *Journal of Global Optimization*, vol. 21, no. 2, pp. 201–218, 2001.

[5] J. F. A. D. O. Pantoja and D. Q. Mayne, "Exact penalty function algorithm with simple updating of the penalty parameter," *Journal of Optimization Theory and Applications*, vol. 69, no. 3, pp. 441–467, 1991.

[6] T. Liu and D. Li, "Convergence of the BFGS-SQP method for degenerate problems," *Numerical Functional Analysis and Optimization*, vol. 28, no. 7-8, pp. 927–944, 2007.

[7] T.-w. Liu and J.-p. Zeng, "An SQP algorithm with cautious updating criteria for nonlinear degenerate problems," *Acta Mathematicae Applicatae Sinica*, vol. 25, no. 1, pp. 33–42, 2009.

[8] R. Fletcher and S. Leyffer, "Nonlinear programming without a penalty function," *Mathematical Programming*, vol. 91, no. 2, pp. 239–269, 2002.

[9] R. Fletcher, S. Leyffer, and P. L. Toint, "On the global convergence of a filter-SQP algorithm," *SIAM Journal on Optimization*, vol. 13, no. 1, pp. 44–59, 2002.

[10] R. Fletcher, N. I. M. Gould, S. Leyffer, P. L. Toint, and A. Wächter, "Global convergence of a trust-region SQP-filter algorithm for general nonlinear programming," *SIAM Journal on Optimization*, vol. 13, no. 3, pp. 635–660, 2002.

[11] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: motivation and global convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 1–31, 2005.

[12] A. Wächter and L. T. Biegler, "Line search filter methods for nonlinear programming: local convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 32–48, 2005.

[13] K. Su and J. Che, "A modified SQP-filter method and its global convergence," *Applied Mathematics and Computation*, vol. 194, no. 1, pp. 92–101, 2007.

[14] K. Su, "A globally and superlinearly convergent modified SQP-filter method," *Journal of Global Optimization*, vol. 41, no. 2, pp. 203–217, 2008.

[15] C. Shen, W. Xue, and X. Chen, "Global convergence of a robust filter SQP algorithm," *European Journal of Operational Research*, vol. 206, no. 1, pp. 34–45, 2010.

[16] C. Shen, W. Xue, and D. Pu, "Global convergence of a tri-dimensional filter SQP algorithm based on the line search method," *Applied Numerical Mathematics*, vol. 59, no. 2, pp. 235–250, 2009.

[17] W. Xue, C. Shen, and D. Pu, "A penalty-function-free line search SQP method for nonlinear programming," *Journal of Computational and Applied Mathematics*, vol. 228, no. 1, pp. 313–325, 2009.

[18] X. Wang, Z. Zhu, S. Zuo, and Q. Huang, "An SQP-filter method for inequality constrained optimization and its global convergence," *Applied Mathematics and Computation*, vol. 217, no. 24, pp. 10224–10230, 2011.

[19] C. Gu, "Nonmonotone filter DQMM method for the system of nonlinear equations," *Applied Mathematics and Computation*, vol. 217, no. 22, pp. 9351–9357, 2011.

[20] C. Gu and D. Zhu, "A secant algorithm with line search filter method for nonlinear optimization," *Applied Mathematical Modelling*, vol. 35, no. 2, pp. 879–894, 2011.

[21] C. Gu and D. Zhu, "A filter algorithm for nonlinear systems of equalities and inequalities," *Applied Mathematics and Computation*, vol. 218, no. 20, pp. 10289–10298, 2012.

[22] Z. Wang and D. Zhu, "A reduced Hessian algorithm with line search filter method for nonlinear programming," *Applied Mathematics and Computation*, vol. 217, no. 19, pp. 7679–7691, 2011.

[23] X. Zhu and D. Pu, "Sequential quadratic programming with a flexible step acceptance strategy," *Applied Mathematical Modelling*, vol. 36, no. 9, pp. 3992–4002, 2012.

[24] X. Zhu and D. Pu, "A line search filter algorithm with inexact step computations for equality constrained optimization," *Applied Numerical Mathematics*, vol. 62, no. 3, pp. 212–223, 2012.

[25] X. Zhu and D. Pu, "A restoration-free filter SQP algorithm for equality constrained optimization," *Applied Mathematics and Computation*, vol. 219, no. 11, pp. 6016–6029, 2013.

[26] X. Zhang, Z. Liu, and S. Liu, "A trust region SQP-filter method for nonlinear second-order cone programming," *Computers & Mathematics with Applications*, vol. 63, no. 12, pp. 1569–1576, 2012.

[27] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer Series in Operations Research, Springer, New York, NY, USA, 1999.

[28] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, vol. 187 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, Berlin, Germany, 1981.