Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2012, Article ID 273141, 15 pages
doi:10.1155/2012/273141

Research Article

An Improved Line Search Filter Method for the System of Nonlinear Equations

Zhong Jin[1] and Yuqing Wang[2]

[1] *Department of Mathematics, Shanghai Maritime University, Shanghai 201306, China*
[2] *Xianda College of Economics and Humanities, Shanghai International Studies University, Shanghai 200083, China*

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

Received 20 March 2012; Accepted 17 September 2012

Academic Editor: Hak-Keung Lam

Copyright © 2012 Z. Jin and Y. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An improved line search filter algorithm for the system of nonlinear equations is presented. We divide the equations into two groups, one contains the equations that are treated as equality constraints and the square of other equations is regarded as objective function. Two groups of equations are updated at every iteration in the works by Nie (2004, 2006, and 2006), by Nie et al. (2008), and by Gu (2011), while we just update them at the iterations when it is needed indeed. As a consequence, the scale of the calculation is decreased in a certain degree. Under some suitable conditions the global convergence can be induced. In the end, numerical experiments show that the method in this paper is effective.

1. Introduction

Many applied problems are reduced to solve the system of nonlinear equations, which is one of the most basic problems in mathematics. This task has applications in many scientific fields such as physics, chemistry, and economics. More formally, the problem to be solved is stated as follows

$$c_i(x) = 0, \quad i = 1, 2, \ldots, m, \tag{1.1}$$

where each $c_i : R^n \to R$ $(i = 1, 2, \ldots, m)$ is a smooth function.

A well-known method for solving nonlinear equations is the Newton method, an iterative scheme which is locally quadratical convergent only if the initial iteration is sufficiently close to the solution. To improve the global properties, some important algorithms [1] for nonlinear equations proceed by minimizing a least square problem:

$$\min h(x) = c(x)^T c(x), \tag{1.2}$$

which can be also handled by the Newton method, while Powell [2] gives a counterexample to show a dissatisfactory fact that the iterates generated by the above least square problem may converge to a nonstationary point of $h(x)$.

Traditionally, a penalty or augmented Lagrange function is always used as a merit function to test the acceptability of the iterates. However, as we all know, there are several difficulties associated with the use of penalty function, and in particular the choice of the penalty parameter. Too low a choice may result in an optimal solution that cannot be obtained, on the other hand, too large a choice damps out the effect of the objective function. Hence, filter method has been first introduced for constrained nonlinear optimization problems in a sequential quadratic programming (SQP) trust-region algorithm by Fletcher and Leyffer [3], offering an alternative to merit functions, as a tool to guarantee global convergence in algorithms for nonlinear programming (NLP). The computational results are very promising in [3]. Furthermore, Fletcher et al. [4, 5] give the global convergence of the trust-region filter-SQP method, then Ulbrich [6] gets its superlinear local convergence. Consequently, filter method has been actually applied in many optimization techniques [7–13], for instance the pattern search method [7], the SLP method [8], the interior method [9], the bundle approaches [10, 11], and so on. Also combined with the trust-region search technique, Gould et al. extended the filter method to the system of nonlinear equations (1.1) and nonlinear least squares in [14], and to the unconstrained optimization problem with multidimensional filter technique in [15]. In addition, Wächter and Biegler [16, 17] presented line search filter methods for nonlinear equality constrained programming and the global and local convergence were given.

Recently, some other ways were given to attack the problem (1.1) (see [18–22]). There are two common features in these papers, one is the filter approach that is utilized, and the other is that at every iteration the system of nonlinear equations is transformed into a constrained nonlinear programming problem and the equations are divided into two groups, some equations are treated as constraints and the others act as the objective function. For instance combined with the filter line search technique [16, 17], the system of nonlinear equations in [21] at the kth iteration x_k becomes the following optimization problem with equality constraints:

$$\begin{aligned} &\min \sum_{i\in S_1^k} c_i^2(x) \\ &\text{s.t.} \quad c_j(x) = 0, \quad j \in S_2^k, \end{aligned} \tag{1.3}$$

where the sets S_1^k and S_2^k are defined as $S_1^k = \{i_j \mid j \le n_0\}$ and $S_2^k = \{i_j \mid j \ge n_0 + 1\}$ for some positive constant $n_0 > 0$ such that $c_{i_1}^2(x_k) \ge c_{i_2}^2(x_k) \ge \cdots \ge c_{i_n}^2(x_k)$.

Motivated by the ideas and methods above, we propose an improved line search filter method for the system of nonlinear equations. We also divide the equations into two groups, one contains the equations that are treated as equality constraints and the square of other equations is regarded as objective function. Two groups of equations are updated at every iteration in those works [18–22], while we just update them at the iterations when it is needed indeed. Specifically, using similar transformed optimization problem (1.3), the difference between our method with [21] is that the sets S_1^k and S_2^k in (1.3) need not be updated at every iteration. As a consequence, the scale of the calculation is decreased in a certain degree. In our algorithm two groups of equations cannot be changed after a f-type iteration, thus in the case that $|\mathcal{A}| < \infty$, the two groups are fixed after finite number of iterations. In addition, the filter should not be updated after a f-type iteration, so naturally the global convergence is discussed, respectively, according to whether the number of the updated filter is infinite or not. And it is shown that every limit point of the sequence of iterates generated by the algorithm is the ϵ solution to (1.1) or a local infeasible point when $|\mathcal{A}| < \infty$. Furthermore, the global convergent property is induced under mild assumptions. In the end, numerical experiments show that the method in this paper is effective.

The paper is outlined as follows. In Section 2, the line search filter method is developed and its key ingredients are described. In Section 3 we prove that under suitable conditions, the method is well defined and globally convergent. Finally, some numerical results are given in Section 4.

2. Description of the Algorithm

To solve the system of nonlinear Equation (1.1), we also transform it into the optimization problem (1.3) where equations are divided into two groups as the one in [21], then let $m_k(x) = \|c_{S_1^k}(x)\|_2^2 = \sum_{i\in S_1^k} c_i^2(x)$ and $\theta_k(x) = \|c_{S_2^k}(x)\|_2^2 = \sum_{i\in S_2^k} c_i^2(x)$. The linearization of the KKT condition of (1.3) at the kth iteration x_k is as follows:

$$\begin{pmatrix} B_k & A_{S_2^k}^k \\ \left(A_{S_2^k}^k\right)^T & 0 \end{pmatrix}\begin{pmatrix} s_k \\ \lambda_k^+ \end{pmatrix} = -\begin{pmatrix} g(x_k) \\ c_{S_2^k}(x_k) \end{pmatrix}, \tag{2.1}$$

where B_k is the Hessian or approximate Hessian matrix of $L(x,\lambda) = m_k(x) + \lambda^T c_{S_2^k}(x)$, $A_{S_2^k}^k = \nabla c_{S_2^k}(x_k)$ and $g(x_k) = \nabla m_k(x_k)$.

After a search direction s_k has been computed, a step size $\alpha_{k,l} \in (0,1]$ is determined in order to obtain the trial iteration

$$x_k(\alpha_{k,l}) = x_k + \alpha_{k,l} s_k. \tag{2.2}$$

More precisely, for fixed constants $\gamma_m, \gamma_\theta \in (0,1)$, we say that a trial step size $\alpha_{k,l}$ provides sufficient reduction with respect to the current x_k if

$$\theta_k(x_k(\alpha_{k,l})) \leq (1-\gamma_\theta)\theta_k(x_k) \quad \text{or} \quad m_k(x_k(\alpha_{k,l})) \leq m_k(x_k) - \gamma_m \theta_k(x_k). \tag{2.3}$$

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

$$(\theta(x_k(\alpha_{k,l})), m(x_k(\alpha_{k,l}))) \notin \mathcal{F}_k. \tag{2.4}$$

At the beginning of the optimization, the filter is initialized to be empty: $\mathcal{F}_0 = \emptyset$. Throughout the optimization the filter is then augmented in some iterations after the new iterate x_{k+1} has been accepted. For this, the following updating formula is used:

$$\mathcal{F}_{k+1} = \mathcal{F}_k \cup \left\{ (\theta, m) \in R^2 : \theta \geq (1 - \gamma_\theta)\theta_k(x_k), m \geq m_k(x_k) - \gamma_m \theta_k(x_k) \right\}. \tag{2.5}$$

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

$$u_k(\alpha_{k,l}) < 0, \qquad -u_k(\alpha_{k,l}) > \delta[\theta_k(x_k)]^{s_\theta}, \tag{2.6}$$

where $u_k(\alpha_{k,l}) = \alpha_{k,l} g_k^T s_k$, $\delta > 0$ and $s_\theta \in (0, 1)$.

When Condition (2.6) holds, the step s_k is a descent direction for current objective function. Then, instead of insisting on (2.3), the Armijo-type reduction condition is employed as follows:

$$m_k(x_k(\alpha_{k,l})) \leq m_k(x_k) + \tau_3 u_k(\alpha_{k,l}), \tag{2.7}$$

where $\tau_3 \in (0, 1/2)$ is a fixed constant.

If (2.6) and (2.7) 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} with no updating of the filter, that is,

$$\mathcal{F}_{k+1} = \mathcal{F}_k. \tag{2.8}$$

While if a trial point $x_k(\alpha_{k,l})$ does not satisfy the switching condition (2.6) but satisfies (2.3), 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.

In the situation, where no admissible step size can be found, the method switches to a feasibility restoration stage, whose purpose is to find a new iterate that satisfies (2.3) 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 define that

$$\alpha_k^{\min} = \begin{cases} \min\left\{ \gamma_\theta, \dfrac{\gamma_m [\theta_k(x_k)]^{s_\theta}}{-g_k^T s_k} \right\}, & \text{if } g_k^T s_k < 0, \\ \gamma_\theta, & \text{otherwise.} \end{cases} \tag{2.9}$$

We are now ready to formally state the overall algorithm for solving the the system of nonlinear Equation (1.1).

Algorithm 2.1. We have the following steps.

Step 1. Initialization: Choose an initial point $x_0 \in R^n$, $0 < \rho_1 < \rho_2 < 1$ and $\epsilon > 0$. Compute g_0, $c_i(x_0)$, S_1^0, S_2^0 and A_k for $i \in S_2^0$. Set $k = 0$ and $\mathcal{F}_0 = \emptyset$.

Step 2. If $\|c(x_k)\| \leq \epsilon$ then stop. Otherwise compute (2.1) to obtain s_k. If there exists no solution to (2.1), go to Step 6. If $\|s_k\| \leq \epsilon$ then stop.
Step 3. Let $\alpha_{k,l} = 1$ and $l = 0$. Compute $\alpha_k^{\min}$ by (2.9).

Step 3.1. If $\alpha_{k,l} < \alpha_k^{\min}$, go to Step 6. Otherwise compute $x_k(\alpha_{k,l}) = x_k + \alpha_{k,l}s_k$. If $x_k(\alpha_{k,l}) \in \mathcal{F}_k$, go to Step 3.2.

Step 3.1.1. Case 1. The switching condition (2.6) holds. If the reduction condition (2.7) holds, set $x_{k+1} = x_k(\alpha_{k,l})$, $\mathcal{F}_{k+1} = \mathcal{F}_k$, $S_1^{k+1} = S_1^k$, $S_2^{k+1} = S_2^k$ and go to Step 5. Otherwise, go to Step 3.2.

Step 3.1.2. Case 2. The switching condition (2.6) is not satisfied. If (2.3) holds, set $x_{k+1} = x_k(\alpha_{k,l})$, augment the filter using (2.5) and go to Step 4. Otherwise, go to Step 3.2.

Step 3.2. Choose $\alpha_{k,l+1} \in [\rho_1\alpha_{k,l}, \rho_2\alpha_{k,l}]$. Set $l = l + 1$ and go to Step 3.1.

Step 4. Compute S_1^{k+1} and S_2^{k+1} by (1.3). If $(\theta_{k+1}(x_{k+1}), m_{k+1}(x_{k+1})) \in \mathcal{F}_{k+1}$, set $S_1^{k+1} = S_1^k$ and $S_2^{k+1} = S_2^k$.
Step 5. Compute g_{k+1}, B_{k+1} and A_{k+1}. Go to Step 2 with k replaced by $k + 1$.
Step 6. (Feasibility Restoration Stage) Find $x_k^r = x_k + \alpha_k^r s_k^r$ such that x_k^r is accepted by the filter and the infeasibility θ is reduced. Go to Step 2.

3. Global Convergence of Algorithm

In the reminder of this paper we denote the set of indices of those iterations in which the filter has been augmented by $\mathcal{A} \subseteq \mathbb{N}$. Let us now state the assumptions necessary for the global convergence analysis.

Assumption 3.1. The sequence $\{x^k\}$ generated by Algorithm 2.1 is nonempty and bounded.

Assumption 3.2. The functions $c_i(x)$, $j = 1, 2, \ldots, m$ are all twice continuously differentiable on an open set containing X.

Assumption 3.3. There exist two constants $b \geq a > 0$ such that the matrices sequence $\{B_k\}$ satisfy $a\|s\|^2 \leq s^T B_k s \leq b\|s\|^2$ for all k and $s \in R^n$.

Assumption 3.4. $(A_{s_2^k}^k)^T$ has full column rank and $\|s_k\| \leq \gamma_s$ for all k with a positive constant γ_s.

By Assumption 3.1 we have $\Sigma_{i=1}^n c_i^2(x_k) \leq M^{\max}$. Since $0 \leq m_k(x_k) \leq m_k(x_k) + \theta_k(x_k) = \Sigma_{i=1}^n c_i^2(x_k)$ and $0 \leq \theta_k(x_k) \leq m_k(x_k) + \theta_k(x_k) = \Sigma_{i=1}^n c_i^2(x_k)$, then (θ, m) associated with the filter is restricted to

$$\mathcal{B} = [0, M^{\max}] \times [0, M^{\max}]. \tag{3.1}$$

Theorem 3.5. *If Algorithm 2.1 terminates at Step 2, then an ϵ solution to* (1.1) *is achieved or a local infeasibility point is obtained.*

Proof. The proof of this theorem can be found in [21]. □

Next we can assume that our algorithm does not terminate finitely and an infinite sequence of points is generated.

Lemma 3.6. *Under the above assumptions, there exists a solution to* (2.1) *with exact (or inexact) line search which satisfies the following descent conditions,*

$$|\theta_k(x_k + \alpha s_k) - (1 - 2\alpha)\theta_k(x_k)| \le \tau_1 \alpha^2 \|s_k\|^2, \tag{3.2}$$

$$|m_k(x_k + \alpha s_k) - m_k(x_k) - u_k(\alpha)| \le \tau_2 \alpha^2 \|s_k\|^2, \tag{3.3}$$

where $\alpha \in (0, 1)$, τ_1, *and* τ_2 *are all positive constants independent of* k.

Proof. By virtue of Taylor expansion of $c_i^2(x_k + \alpha s_k)$ with $i \in S_2$, we obtain

$$\begin{aligned} &\left| c_i^2(x_k + \alpha s_k) - c_i^2(x_k) - 2\alpha c_i(x_k) \nabla c_i(x_k)^T s_k \right| \\ &\quad = \left| c_i^2(x_k + \alpha s_k) - c_i^2(x_k) - 2 c_i(x_k) \nabla c_i(x_k)^T (\alpha s_k) \right| \\ &\quad = \left| \frac{1}{2} (\alpha s_k)^T \left[2 c_i(x_k + \zeta\alpha s_k) \nabla c_i^2(x_k + \zeta\alpha s_k) + 2 \nabla c_i(x_k + \zeta\alpha s_k) \nabla c_i(x_k + \zeta\alpha s_k)^T \right] (\alpha s_k) \right| \\ &\quad = \left| \alpha^2 s_k^T \left[c_i(x_k + \zeta\alpha s_k) \nabla c_i^2(x_k + \zeta\alpha s_k) + \nabla c_i(x_k + \zeta\alpha s_k) \nabla c_i(x_k + \zeta\alpha s_k)^T \right] s_k \right| \\ &\quad \le \frac{1}{m} \tau_1 \alpha^2 \|s_k\|^2, \end{aligned} \tag{3.4}$$

where the last inequality follows the above assumptions and $\zeta \in [0, 1]$. Also, from (2.1) we immediately get that $c_i(x_k) + \nabla c_i(x_k)^T s_k = 0$, that is, $-2\alpha c_i^2(x_k) - 2\alpha c_i(x_k) \nabla c_i(x_k)^T s_k = 0$. With $|S_2| \le m$, thereby,

$$\begin{aligned} |\theta_k(x_k + \alpha s_k) - (1 - 2\alpha)\theta_k(x_k)| &= \left| \sum_{i \in S_2} \left(c_i^2(x_k + \alpha s_k) - (1 - 2\alpha) c_i^2(x_k) \right) \right| \\ &\le \sum_{i \in S_2} \left| c_i^2(x_k + \alpha s_k) - (1 - 2\alpha) c_i^2(x_k) \right| \\ &= \sum_{i \in S_2} \left| c_i^2(x_k + \alpha s_k) - c_i^2(x_k) - 2\alpha c_i(x_k) \nabla c_i(x_k)^T s_k \right| \\ &\le m \cdot \frac{1}{m} \tau_1 \alpha^2 \|s_k\|^2 \\ &\le \tau_1 \alpha^2 \|s_k\|^2, \end{aligned} \tag{3.5}$$

then (3.2) consequently holds.

According to Taylor expansion of $\Sigma_{i \in S_1}(c_i^2(x_k + \alpha s_k))$ (i.e., $m_k(x_k + \alpha s_k)$), we then have

$$\begin{aligned} \left| \sum_{i \in S_1} \left(c_i^2(x_k + \alpha s_k) \right) - \sum_{i \in S_1} \left(c_i^2(x_k) \right) - \alpha g_k^T s_k \right| &= \left| \frac{1}{2} \alpha^2 (s_k)^T \nabla^2 \sum_{i \in S_1} \left(c_i^2(x_k + \rho\alpha s_k) \right) s_k \right| \\ &\le \tau_2 \alpha^2 \|s_k\|^2, \end{aligned} \tag{3.6}$$

where the last inequality follows the first two assumptions and $\rho \in [0,1]$. That is to say,

$$|m_k(x_k + \alpha s_k) - m_k(x_k) - u_k(\alpha)| \le \tau_2 \alpha^2 \|s_k\|^2, \tag{3.7}$$

which is just (3.3). □

Theorem 3.7. *If there are only finite or infinite number of iterates entering the filter, then*

$$\lim_{k \to \infty} \theta(x_k) = 0. \tag{3.8}$$

Proof. Consider the following.

Case 1 (if $|\mathcal{A}| = \infty$). The proof is by contraction. Suppose that there exists an infinite subsequence $\{k_i\}$ of $\mathcal{A}$ such that

$$\theta_{k_i}(x_{k_i}) \ge \varepsilon, \tag{3.9}$$

for some $\varepsilon > 0$. At each iteration k_i, $(\theta_{k_i}(x_{k_i}), m_{k_i}(x_{k_i}))$ is added to the filter which means that no other (θ, m) can be added to the filter at a later stage within the area

$$\left[\theta_{k_i}(x_{k_i}) - \gamma_\theta \theta_{k_i}(x_{k_i}), \theta_{k_i}(x_{k_i})\right] \times \left[m_{k_i}(x_{k_i}) - \gamma_m \theta_{k_i}(x_{k_i}), m_{k_i}\right], \tag{3.10}$$

and the area of each of these squares is at least $\gamma_\theta \gamma_m \varepsilon^2$.

Thus the $\mathcal{B}$ is completely covered by at most a finite number of such areas in contraction to the infinite subsequence $\{k_i\}$ satisfying (3.9). This means that (3.8) is true.

Case 2 (if $|\mathcal{A}| < \infty$). From $|\mathcal{A}| < \infty$, we know the filter updates in a finite number, then there exists $K \in \mathbb{N}$, for $k > K$ the filter does not update. As h-type iteration and restoration algorithm all need the updating of the filter, then for $k > K$ our algorithm only executes the f-type iterations.

Because S_1 and S_2 do not change in the f-type iteration, then for $k \ge K + 1$ we have $\theta_k(x) = \theta_K(x)$ and $m_k(x) = m_K(x)$. Thus we obtain

$$\sum_{k=K+1}^{\infty} (m_k(x_k) - m_k(x_{k+1})) = \sum_{k=K+1}^{\infty} (m_K(x_k) - m_K(x_{k+1})). \tag{3.11}$$

From the switching condition (2.6) we know that

$$\tau_3 u_k(\alpha_{k,l}) = \tau_3 \alpha_{k,l} g_k^T s_k < -\tau_3 \delta [\theta_k(x_k)]^{\gamma_\theta}, \tag{3.12}$$

together with the reduction condition (2.7) we get

$$m_k(x_k(\alpha_{k,l})) \le m_k(x_k) + \tau_3 u_k(\alpha_{k,l}) \le m_k(x_k) - \tau_3 \delta [\theta_k(x_k)]^{\gamma_\theta}, \tag{3.13}$$

and since $x_{k+1} = x_k(\alpha_{k,l})$ for $k \geq K+1$, this reduces to

$$m_k(x_{k+1}) = m_k(x_k(\alpha_{k,l})) \leq m_k(x_k) - \tau_3\delta[\theta_k(x_k)]^{\gamma_\theta}. \tag{3.14}$$

Suppose to the contrary that there exists an infinite subsequence $\{k_i\}$ of $\mathcal{A}$ such that

$$\theta_{k_i}(x_{k_i}) \geq \varepsilon, \tag{3.15}$$

for some $\varepsilon > 0$. Choose an arbitrary subsequence $\{\overline{k}_i\}$ of $\{k_i\}$ satisfying $\overline{k}_i > K$, we find that

$$+\infty > \sum_{k=K+1}^{\infty}(m_K(x_k) - m_K(x_{k+1})) \geq \sum_{i=1}^{\infty}\left(m_K\left(x_{\overline{k}_i}\right) - m_K\left(x_{\overline{k}_i+1}\right)\right) \geq \sum_{i=1}^{\infty}\tau_3\delta\left[\theta_{\overline{k}_i}\left(x_{\overline{k}_i}\right)\right]^{\gamma_\theta} \longrightarrow +\infty, \tag{3.16}$$

which is a contraction. □

Lemma 3.8. *Let $\{x_{k_i}\}$ be a subsequence of iterates for which* (2.6) *hold and have the same S_1 and S_2. Then there exists a $\widehat{\alpha} \in (0,1]$ such that*

$$m_{k_i}(x_{k_i} + \widehat{\alpha}s_{k_i}) \leq m_{k_i}(x_{k_i}) + \widehat{\alpha}\tau_3 g_{k_i}^T s_{k_i}. \tag{3.17}$$

Proof. Because $\{x_{k_i}\}$ have the same S_1 and S_2, $m_{k_i}(x)$ are fixed. And d_{k_i} is a decent direction due to (2.6). Hence there exists a $\widehat{\alpha} \in (0,1]$ satisfying (3.17). □

Lemma 3.9. *If there exists $\overline{K} \in \mathbb{N}$ such that for $k > \overline{K}$ it holds $\|s_k\| \geq (1/2)\varepsilon_1$, where $\varepsilon_1 > 0$, then there exist $K_1 \in \mathbb{N}$ and a constant $\varepsilon_0 > 0$, for $k > K_1$ such that*

$$g_k^T s_k \leq -\varepsilon_0. \tag{3.18}$$

Proof. From (2.1) we obtain $g_k + A_{S_2}^k \lambda_k^+ + B_k s_k = 0$ and $c_{S_2}^k + (A_{S_2}^k)^T s_k = 0$, so

$$g_k^T s_k = -s_k^T B_k s_k - (\lambda_k^+)^T c_{S_2}^k \leq -a\|s_k\|^2 - (\lambda_k^+)^T c_{S_2}^k \leq -a\|s_k\|^2 + c_1\left\|c_{S_2}^k\right\|. \tag{3.19}$$

As $\lim_{k\to\infty}\theta(x_k) = 0$ implies $\lim_{k\to\infty}\|c_{S_2}^k\| = 0$, then there exists K_1 $(\in N) \geq \overline{K}$, for $k > K_1$ such that $\|c_{S_2}^k\| \leq (a\varepsilon_1^2)/(8c_1)$. Thus

$$g_k^T s_k \leq -a\|s_k\|^2 + c_1\left\|c_{S_2}^k\right\| \leq -a\left(\frac{1}{2}\varepsilon_1\right)^2 + c_1\left(\frac{a\varepsilon_1^2}{8c_1}\right) = -\varepsilon_0, \tag{3.20}$$

with $\varepsilon_0 = (a\varepsilon_1^2)/(8c_1)$. □

Theorem 3.10. *Suppose that* $\{x_k\}$ *is an infinite sequence generated by Algorithm 2.1 and* $|\mathcal{A}| < \infty$*, one has*

$$\lim_{k\to\infty} \left\| c^k_{S_2^k} \right\| + \|s_k\| = 0. \tag{3.21}$$

Namely, every limit point is the ϵ *solution to* (1.1) *or a local infeasible point. If the gradients of* $c_i(x_k)$ *are linear independent for all* k *and* $i = 1, 2, \dots, m$*, then the solution to* (1.1) *is obtained.*

Proof. As was noted in the proof of the case 2 in Theorem 3.7, from $|\mathcal{A}| < \infty$, the algorithm only executes the f-type iterations. Now the proof is by contraction. If the result is false, there should have been a constant $\varepsilon_1 > 0$ and a subsequence $\{x_{k_i}\}$ of $\{x_k\}$ such that

$$\left\| c^{k_i}_{S_2^{k_i}} \right\| + \|s_{k_i}\| > \varepsilon_1, \tag{3.22}$$

for all k_i. Without loss of generality, we can suppose there are only f-type iterations for $k \geq k_1$. Moreover, there exist the following results for sufficiently large k_i.

As $\lim_{k\to\infty}\theta(x_k) = 0$, it is apparent that $\|c^{k_i}_{S_2^{k_i}}\| \leq (1/2)\varepsilon_1$ hold for large enough k_i. It is reasonable that we have $\|s_{k_i}\| \geq (1/2)\varepsilon_1$ from (3.22), so similar to the Lemma 3.9, we can get $g_{k_i}^T s_{k_i} \leq -\varepsilon_0$ for large enough k_i. Without loss of generality, suppose $g_{k_i}^T s_{k_i} \leq -\varepsilon_0$ for all k_i.

Because S_1 and S_2 do not change in the f-type iteration, then for all k_i we have $\theta_{k_i}(x) = \theta_{k_1}(x)$ and $m_{k_i}(x) = m_{k_1}(x)$. Since the f-type point satisfies the reduction condition (2.7), that is, $m_k(x_k) - m_k(x_{k+1}) > -\tau_3\alpha g_k^T s_k$, then it holds that

$$\sum_{i=1}^{\infty}\left(m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i+1})\right) = \sum_{i=1}^{\infty}\left(m_{k_1}(x_{k_i}) - m_{k_1}(x_{k_i+1})\right) > \sum_{i=1}^{\infty}\left(-\tau_3\alpha g_{k_i}^T s_{k_i}\right). \tag{3.23}$$

By $m_k(x_k)$ is decreasing for $k \geq k_1$ and Lemma 3.8, we get

$$\begin{aligned} +\infty > \sum_{k=k_1}^{\infty}\left(m_k(x_k) - m_k(x_{k+1})\right) > \sum_{i=1}^{\infty}\left(m_{k_i}(x_{k_i}) - m_{k_i}(x_{k_i+1})\right) &> \sum_{i=1}^{\infty}\left(-\widehat{\alpha}\tau_3 g_{k_i}^T s_{k_i}\right) \\ &> \sum_{i=1}^{\infty}\widehat{\alpha}\tau_3\varepsilon_0 \longrightarrow +\infty, \end{aligned} \tag{3.24}$$

which is a contraction.

If the gradients of $c_i(x_k)$ are linear independent for all k and $i = 1, 2, \dots, m$, then the solution to (1.1) is obtained by virtue of the result in [23]. Thus, the result is established. □

Lemma 3.11. *If* $g_k^T s_k \leq -\varepsilon_0$ *for a positive constant* ε_0 *independent of* k*, then there exists a constant* $\overline{\alpha} > 0$*, for all* k *and* $\alpha \leq \overline{\alpha}$ *such that*

$$m_k(x_k + \alpha s_k) - m_k(x_k) \leq \tau_3 u_k(\alpha). \tag{3.25}$$

Proof. Let $\overline{\alpha} = ((1-\tau_3)\varepsilon_0)/\tau_2\gamma_s^2$. From (3.3), $\|s_k\| \leq \gamma_s$ and $\alpha \leq \overline{\alpha}$, we obtain

$$\begin{aligned} m_k(x_k+\alpha s_k) - m_k(x_k) - u_k(\alpha) \leq \tau_2\alpha^2\|s_k\|^2 \leq \tau_2\alpha\overline{\alpha}\gamma_s^2 = (1-\tau_3)\alpha\varepsilon_0 \leq -(1-\tau_3)\alpha g_k^T s_k \\ = -(1-\tau_3)u_k(\alpha), \end{aligned} \tag{3.26}$$

which shows that (3.25) is true. □

Lemma 3.12. *Under all existing assumptions, if $u_k(\alpha) \leq -\alpha\varepsilon_0$ for a positive constant ε_0 independent of k and $\alpha \geq \alpha_{k,l}^{\min}$ for all $\alpha \in (0,1]$ with $(\theta_k(x_k), m_k(x_k)) \notin \mathcal{F}_k$, then there exist $\gamma_1, \gamma_2 > 0$ such that $(\theta_k(x_k+\alpha s_k), m_k(x_k+\alpha s_k)) \notin \mathcal{F}_k$ for all k and $\alpha \leq \min\{\gamma_1, \gamma_2\theta_k(x_k)\}$.*

Proof. Choose $\gamma_1 = \varepsilon_0/\tau_2\gamma_s^2$, then $\alpha \leq \gamma_1$ implies that $-\alpha\varepsilon_0 + \tau_2\alpha^2\gamma_s^2 \leq 0$. So from (3.3), we obtain

$$\begin{aligned} m_k(x_k+\alpha s_k) &\leq m_k(x_k) + u_k(\alpha) + \tau_2\alpha^2\|s_k\|^2 \\ &\leq m_k(x_k) - \alpha\varepsilon_0 + \tau_2\alpha^2\gamma_s^2 \\ &\leq m_k(x_k). \end{aligned} \tag{3.27}$$

And choose $\gamma_2 = 2/\tau_1\gamma_s^2$, then $\alpha \leq \gamma_2\theta_k(x_k)$ implies that $-2\alpha\theta_k(x_k) + \tau_1\alpha^2\gamma_s^2 \leq 0$. It follows from (3.2) that

$$\begin{aligned} \theta_k(x_k+\alpha s_k) &\leq \theta_k(x_k) - 2\alpha\theta_k(x_k) + \tau_2\alpha^2\|s_k\|^2 \\ &\leq \theta_k(x_k) - 2\alpha\theta_k(x_k) + \tau_1\alpha^2\gamma_s^2 \\ &\leq \theta_k(x_k). \end{aligned} \tag{3.28}$$

We further point out a fact according to the definition of filter. If $(\overline{\theta}, \overline{m}) \notin \mathcal{F}_k$ and $\theta \leq \overline{\theta}$, $m \leq \overline{m}$, we note that $(\theta, m) \notin \mathcal{F}_k$. So from $(\theta_k(x_k), m_k(x_k)) \notin \mathcal{F}_k$, $m_k(x_k+\alpha s_k) \leq m_k(x_k)$ and $\theta_k(x_k+\alpha s_k) \leq \theta_k(x_k)$, we get $(\theta_k(x_k+\alpha s_k), m_k(x_k+\alpha s_k)) \notin \mathcal{F}_k$. □

Lemma 3.13. *If there exists $\overline{K} \in \mathbb{N}$ such that for $k > \overline{K}$ it holds $\|s_k\| \geq (1/2)\varepsilon_1$ for a constant $\varepsilon_1 > 0$, then there exists $K \in \mathbb{N}$ so that for all $k > K$ the f-type iterate is always satisfied.*

Proof. By Lemma 3.9, there exists $K_1 \in \mathbb{N}$ and a constant $\varepsilon_0 > 0$, for $k > K_1$ such that $g_k^T s_k \leq -\varepsilon_0$. From the definition of $\alpha_k^{\min}$, when $g_k^T s_k \leq 0$, we have $\alpha_k^{\min} = \min\{\gamma_\theta, (\delta[\theta_k(x_k)]^{s_\theta})/(-g_k^T s_k)\}$. As $\lim_{k\to\infty}\theta(x_k) = 0$, we have $\lim_{k\to\infty}(\delta[\theta_k(x_k)]^{s_\theta})/\varepsilon_0 = 0$. So there exists $K_2\ (> K_1) \in \mathbb{N}$ such that for $k > K_2$ it holds

$$\frac{\delta[\theta_k(x_k)]^{s_\theta}}{-g_k^T s_k} < \frac{\delta[\theta_k(x_k)]^{s_\theta}}{\varepsilon_0} < \gamma_\theta, \tag{3.29}$$

which implies for $k > K_1$ we have $\alpha_k^{\min} = \min\{\gamma_\theta, (\delta[\theta_k(x_k)]^{s_\theta})/(-g_k^T s_k)\} = (\gamma_m[\theta_k(x_k)]^{s_\theta})/(-g_k^T s_k)$.

Because $\lim_{k\to\infty}\theta(x_k) = 0$, there exists $K\ (> K_2) \in \mathbb{N}$ such that for $k > K$ it follows

$$\theta_k(x_k) < \min\left\{\frac{\overline{\alpha}}{\gamma_2}, \frac{\gamma_1}{\gamma_2}, \left[\frac{\rho_1\gamma_2\varepsilon_0}{\delta}\right]^{1/(s_\theta-1)}\right\}, \tag{3.30}$$

which shows that

$$\begin{gathered}\gamma_2\theta_k(x_k) \leq \min\{\overline{\alpha}, \gamma_1, \gamma_2\theta_k(x_k)\}, \\ \frac{\delta[\theta_k(x_k)]^{s_\theta}}{\varepsilon_0} < \rho_1\gamma_2\theta_k(x_k).\end{gathered} \tag{3.31}$$

Moreover, $K > K_2 > K_1$ implies $g_k^T s_k \leq -\varepsilon_0$ for all $k > K$.

Let $\beta_k = \gamma_2\theta_k(x_k)$, by Lemmas 3.11 and 3.12, for $\alpha \leq \beta_k$ we obtain

$$\begin{gathered}m_k(x_k + \alpha s_k) - m_k(x_k) \leq \tau_3 u_k(\alpha), \\ (\theta_k(x_k + \alpha s_k), m_k(x_k + \alpha s_k)) \notin \mathcal{F}_k.\end{gathered} \tag{3.32}$$

If we now denote with $\alpha_{k,L}$ the first trial size satisfying (3.32), the backtracking line search procedure then implies that for $\alpha \geq \alpha_{k,L}$

$$\alpha \geq \rho_1\beta_k = \rho_1\gamma_2\theta_k(x_k) > \frac{\delta[\theta_k(x_k)]^{s_\theta}}{\varepsilon_0}, \tag{3.33}$$

from which it follows that

$$\begin{gathered}\alpha > \frac{\delta[\theta_k(x_k)]^{s_\theta}}{\varepsilon_0} > \frac{\delta[\theta_k(x_k)]^{s_\theta}}{-g_k^T s_k} > \min\left\{\gamma_\theta, \frac{\delta[\theta_k(x_k)]^{s_\theta}}{\varepsilon_0}\right\} = \alpha_k^{\min}, \\ -u_k(\alpha) > \alpha\varepsilon_0 > \delta[\theta_k(x_k)]^{s_\theta}.\end{gathered} \tag{3.34}$$

This means that $\alpha_{k,L}$ and all previous trial step sizes are f-step sizes and larger than $\alpha_k^{\min}$. Therefore, $\alpha_{k,L}$ is the accepted step size α_k indeed. Since the trial point $x_k(\alpha_{k,L}) \notin \mathcal{F}_k$ satisfies the switching condition (2.6) and the reduction condition (2.7), the iteration is a f-type iteration for each $k > K$. The claim is true. □

Theorem 3.14. *Suppose that $\{x_k\}$ is an infinite sequence generated by Algorithm 2.1 and $|\mathcal{A}| = \infty$, then*

$$\lim_{k\to\infty}\inf\left[\left\|c^k_{S_2^k}\right\| + \|s_k\|\right] = 0. \tag{3.35}$$

Namely, it has an accumulation which is the ϵ solution to (1.1) *or a local infeasible point. If the gradients of $c_i(x_k)$ are linear independent for all k and $i = 1, 2, \ldots, m$, then the solution to* (1.1) *is obtained.*

Table 1: Numerical results of Example 4.1.

Starting point	NIT	NOF	NOG
(3, 1)	6	12	10
(6, 2)	9	17	14
(9, 3)	12	24	21

Proof. Suppose by contraction that there should have been a constant $\varepsilon_1 > 0$ such that

$$\left\| c^k_{S^k_2} \right\| + \|s_k\| > \varepsilon_1, \tag{3.36}$$

for all k. Furthermore, there exists the following results for sufficiently large k.

As $\lim_{k\to\infty}\theta(x_k) = 0$, it is apparent that $\|c^k_{S^k_2}\| \le (1/2)\varepsilon_1$ holds for large enough k. It is reasonable that we have $\|s_k\| \ge (1/2)\varepsilon_1$ for large enough k from (3.36). Therefore there exists $\overline{K} \in \mathbb{N}$ such that for $k > \overline{K}$ it holds $\|s_k\| \ge (1/2)\varepsilon_1$ for a constant $\varepsilon_1 > 0$, then by Lemma 3.13, we know that there exists $K \in \mathbb{N}$ such that for all $k > K$ the f-type iterate is always satisfied, which is a contraction to $|\mathcal{A}| = \infty$.

Similar to Theorem 3.10, if the gradients of $c_i(x_k)$ are linear independent for all k and $i = 1, 2, \ldots, m$, then the solution to (1.1) is also obtained. The whole proof is completed. □

4. Numerical Examples

In this section, we develop the implementation of Algorithm 2.1 in order to observe its performance on some illustrative examples. In the whole process, the program is coded in MatLab with exact line search. The first example is from [2], which converges to a nonstationary point if the least squares approach is employed. The second example comes from [24], which Newton method fails to solve, and the final problem is given from [25]. In the tables, the notations NIT, NOF, and NOG mean the number of iterates, number of functions and number of gradients, respectively.

Example 4.1. Find a solution of the nonlinear equations system:

$$\begin{pmatrix} x \\ \dfrac{10x}{x+0.1} + 2y^2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{4.1}$$

The unique solution is $(x^*, y^*) = (0, 0)$. It has been proved in [2] that, under initial point $(x_0, y_0) = (3, 1)$, the iterates converge to the point $z = (1.8016, 0.0000)$, which is not a stationary point. Utilizing our algorithm, a sequence of points converging to (x^*, y^*) is obtained. We assume the error tolerance ϵ in this paper is always $1.0e-5$. The detailed numerical results for Example 4.1 are listed in Table 1.

Example 4.2. Find a solution of the nonlinear equations system as follows:

$$\begin{pmatrix} x + 3y^2 \\ (x - 1.0)y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \tag{4.2}$$

Table 2: Numerical results of Example 4.2.

Starting point	NIT	NOF	NOG
(1, 0)	2	4	8
(1, 2)	11	18	15

Table 3: Numerical results of Example 4.3.

Starting point	(0.5, 0.5)	(−0.5, 0.5)	(0.5, −0.5)
NIT	5	9	7
NOF	10	12	14
NOG	9	15	10
Solution	(1, 1)	(−1, 1)	(1, −1)

Table 4: Numerical results of Example 4.4.

	$N = 5$	$N = 10$	$N = 15$	$N = 30$	$N = 50$
NIT	6	8	14	19	36
NOF	8	10	16	21	40
NOG	7	12	15	20	38

The only solution of Example 4.2 is $(x^*, y^*) = (0, 0)$. Define the line $\Gamma = \{(1, y) : y \in \mathbb{R}\}$. If the starting point $(x_0, y_0) \in \Gamma$, the Newton method [24] is confined to Γ. We choose two starting points which are belong to Γ in the experiments and then the (x^*, y^*) is obtained. The numerical results of Example 4.2 are given in Table 2.

Example 4.3. Consider the following system of nonlinear equations:

$$\begin{aligned} f_1(x) &= x_1^2 + x_1x_2 + 2x_2^2 - x_1 - x_2 - 2, \\ f_2(x) &= 2x_1^2 + x_1x_2 + 3x_2^2 - x_1 - x_2 - 4. \end{aligned} \tag{4.3}$$

There are three solutions of above example, $(1, 1)^T$, $(-1, 1)^T$, $(1, -1)^T$. The numerical results of Example 4.3 are given in Table 3.

Example 4.4. Consider the system of nonlinear equations:

$$\begin{aligned} f_i(x) &= -(N+1) + 2x_i + \sum_{j=1, j\neq i}^{N} x_j, \quad i = 1, 2, \ldots, N-1, \\ f_N(x) &= -1 + \prod_{j=1}^{N} x_j, \end{aligned} \tag{4.4}$$

with the with the initial point $x_i^{(0)} = 0.5$, $i = 1, 2, \ldots, N$. The solution to Example 4.4 is $x^* = (1, 1, \ldots, 1)^T$. The numerical results of Example 4.4 are given in Table 4.

All results summarized show that our proposed algorithm is practical and effective.

Acknowledgments

The authors would like to thank the anonymous referees for the careful reading and helpful comments, which led to an improved version of the original paper. The authors gratefully acknowledge the National Natural Science Foundation of China (no. 11126060) and the Science and Technology Program of Shanghai Maritime University (no. 20120060).

References

[1] J. Nocedal and S. J. Wright, *Numerical Optimization*, Springer, New York, NY, USA, 1999.

[2] M. J. D. Powell, “A hybrid method for nonlinear equations,” in *Numerical Methods for Nonlinear Algebraic Equations*, P. Rabinowitz, Ed., pp. 87–114, Gordon and Breach, London, UK, 1970.

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

[4] 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.

[5] 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. 635–659, 2002.

[6] S. Ulbrich, “On the superlinear local convergence of a filter-SQP method,” *Mathematical Programming*, vol. 100, no. 1, pp. 217–245, 2004.

[7] C. Audet and J. E. Dennis, Jr., “A pattern search filter method for nonlinear programming without derivatives,” *SIAM Journal on Optimization*, vol. 14, no. 4, pp. 980–1010, 2004.

[8] C. M. Chin and R. Fletcher, “On the global convergence of an SLP-filter algorithm that takes EQP steps,” *Mathematical Programming*, vol. 96, no. 1, pp. 161–177, 2003.

[9] M. Ulbrich, S. Ulbrich, and L. N. Vicente, “A globally convergent primal-dual interior filter method for nonconvex nonlinear programming,” *Mathematical Programming*, vol. 100, pp. 379–410, 2003.

[10] R. Fletcher and S. Leyffer, “A bundle filter method for nonsmooth nonlinear optimization,” Tech. Rep. NA/195, Department of Mathematics, University of Dundee, Scotland, UK, 1999.

[11] E. Karas, A. Ribeiro, C. Sagastizábal, and M. Solodov, “A bundle-filter method for nonsmooth convex constrained optimization,” *Mathematical Programming*, vol. 116, no. 1-2, pp. 297–320, 2009.

[12] Y. Zhou, “A smoothing conic trust region filter method for the nonlinear complementarity problem,” *Journal of Computational and Applied Mathematics*, vol. 229, no. 1, pp. 248–263, 2009.

[13] K. Su and D. Pu, “A nonmonotone filter trust region method for nonlinear constrained optimization,” *Journal of Computational and Applied Mathematics*, vol. 223, no. 1, pp. 230–239, 2009.

[14] N. I. M. Gould, S. Leyffer, and P. L. Toint, “A multidimensional filter algorithm for nonlinear equations and nonlinear least-squares,” *SIAM Journal on Optimization*, vol. 15, no. 1, pp. 17–38, 2004.

[15] N. I. M. Gould, C. Sainvitu, and P. L. Toint, “A filter-trust-region method for unconstrained optimization,” *SIAM Journal on Optimization*, vol. 16, no. 2, pp. 341–357, 2005.

[16] 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.

[17] 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.

[18] P.-Y. Nie, “A null space method for solving system of equations,” *Applied Mathematics and Computation*, vol. 149, no. 1, pp. 215–226, 2004.

[19] P.-Y. Nie, “CDT like approaches for the system of nonlinear equations,” *Applied Mathematics and Computation*, vol. 172, no. 2, pp. 892–902, 2006.

[20] P.-Y. Nie, “An SQP approach with line search for a system of nonlinear equations,” *Mathematical and Computer Modelling*, vol. 43, no. 3-4, pp. 368–373, 2006.

[21] P.-Y. Nie, M.-y. Lai, S.-j. Zhu, and P.-a. Zhang, “A line search filter approach for the system of nonlinear equations,” *Computers & Mathematics with Applications*, vol. 55, no. 9, pp. 2134–2141, 2008.

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

[23] P.-Y. Nie, “Sequential penalty quadratic programming filter methods for nonlinear programming,” *Nonlinear Analysis*, vol. 8, no. 1, pp. 118–129, 2007.

[24] R. H. Byrd, M. Marazzi, and J. Nocedal, “On the convergence of Newton iterations to non-stationary points,” *Mathematical Programming*, vol. 99, no. 1, pp. 127–148, 2004.
[25] H. Schichl and A. Neumaier, “Exclusion regions for systems of equations,” *SIAM Journal on Numerical Analysis*, vol. 42, no. 1, pp. 383–408, 2004.