RAIRO-Oper. Res. 58 (2024) 2783–2795
https://doi.org/10.1051/ro/2024030

RAIRO Operations Research
www.rairo-ro.org

ACCELERATED NONMONOTONE LINE SEARCH TECHNIQUE FOR MULTIOBJECTIVE OPTIMIZATION

Zohre Aminifard[1,*], Saman Babaie-Kafaki[2], Fereidoun Habibian-Dehkordi[3] and Maria Toofan[1]

Abstract. In order to increase the probability of applying more recent information, a forgetting factor is embedded in the nonmonotone line search technique for minimization of the multiobjective problem concerning the partial order induced by a closed, convex, and pointed cone. The method is shown to be globally convergent without convexity assumption on the objective function. Moreover, to improve behavior of the classical steepest descent method, an accelerated scheme is presented. Ultimately, computational advantages of the algorithms are depicted on a class of standard test problems.

Mathematics Subject Classification. 90C29, 90B50, 65K05.

Received January 25, 2023. Accepted January 30, 2024.

1. Introduction

Generally, multiobjective optimization problems deal with minimization of a vector-valued function with respect to the partial order defined by a closed, convex, and pointed cone. This problem appears in many real-world applications such as engineering design problems [8], statistics [7], management science [15], and so on. As one of the significant applications of multiobjective optimization, multi-criteria decision making is concerned with theory and methodology that can treat complex problems encountered in business, engineering and other areas of human activities. Multi-criteria decision making is closely related to multiobjective optimization. A multi-criteria decision making process is a system that helps with making decisions under multiple, but conflicting criteria. Since, the objectives are usually in conflict with each other, the concept of Pareto-optimality has been developed in the literature [14, 17, 18]. Technically, considering a minimization problem, a point is called Pareto optimal, if there does not exist a different point with the same or smaller objective function values.

Multiobjective optimization algorithms that do not scalarize have recently been developed [20]. Some of these techniques borrow heavily from ideas developed in heuristic optimization [37,40]. While the others are extensions of scalar optimization algorithms including several versions of the steepest descent method [13, 14, 17, 19], the conjugate gradient method [29] and the Newton-like methods [5,9,18]. Fliege and Svaiter [17] especially proposed

Keywords. Multiobjective optimization, Pareto-optimality, steepest descent method, accelerated nonmonotone line search, forgetting factor.

[1] UCLouvain, Institute of Information and Communication Technologies, Electronics and Applied Mathematics, Belgium.

[2] Faculty of Engineering, Free University of Bozen-Bolzano, Piazza Università 5, 39100 Bolzano, Italy.

[3] Faculty of Mathematics, Statistics and Computer Science, Semnan University, P.O. Box: 35195–363, Semnan, Iran.

*Corresponding author: zohreh.aminifard@uclouvain.be

a Pareto descent method for multiobjective optimization based on a suitable extension of the classical steepest descent direction for vector-valued functions. They showed that the method converges to a point satisfying certain first-order necessary conditions of Pareto optimality, under appropriate conditions. Then, Drummond and Svaiter [14] extended the method of [17] on a general cone and under some assumptions proved that the extended method is well-defined in the sense that the Armijo line search always performs successfully and that every cluster point of the iterates, if any, is a stationary point of the problem. Also, Drummond and Iusem [13] suggested a projected steepest descent method for solving constrained vector optimization problems. At each iteration of their method, when a certain first-order optimality condition does not hold, a search direction is computed in the image space using an auxiliary nonsmooth problem (*i.e.* the minimization of the maximum ordering scalarization). Then, under convexity assumption as well as some other standard hypotheses, Fukuda and Drummond [19] established convergence of the projected steepest descent method with respect to the ordering cone.

It is worth noting that the merits and demerits of the classical methods of single objective optimization (such as the steepest descent method) can be inherited by the multiobjective optimization algorithms. Especially, the steepest descent method for multiobjective optimization problems may perform poorly and may be badly affected by ill-conditioning; this is the phenomenon that appears for the method in single objective optimization problems as well [39]. Also, decreasing the function value monotonically may reduce the convergence speed, particularly for highly nonlinear objective functions with narrow parabolic shape valleys [41]. In such situations, allowing some controlled increase in the function value while maintaining the convergence in the sense of the nonmonotone schemes [21] may be helpful, enhancing the likelihood of achieving global optima as well [43].

Initially, nonmonotone line search approach was suggested by Grippo *et al.* [21]. In each iteration of their method, the function is decreased in contrast to the maximum of a prespecified number of recent function values. The nonmonotone line search technique of [21] has been successfully applied in the nonlinear optimization algorithms such as the spectral steepest descent method [6, 42], the conjugate gradient method [11, 28], the truncated Newton method [22] as well as the trust region strategy [1, 10, 34, 38, 41]. Recently, a nonmonotone line search approach was developed for multiobjective optimization by Qu *et al.* [33], decreasing the function in contrast to the componentwise maximum of a prespecified number of recent function values. Also, Fazzio and Schuverdt [16] introduced another nonmonotone line search scheme which guarantees a reduction in the function value in contrast to an average of a prespecified number of recent function values. Moreover, Mita *et al.* [31] used maximum, average, and hybrid-type of a prespecified number of recent function values in the nonmonotone line search strategy for multiobjective optimization.

Founded upon the forgetting factor approach, recently a nonmonotone line search strategy for single objective function has been put forward in [2]. Here, considering the procedure of [2], a nonmonotone steepest descent method is developed for multiobjective optimization. The remainder of this work is organized as follows. In Section 2, we summarize some notations and preliminaries of the literature. Then, in Section 3 we present our method and discuss its global convergence without convexity assumption. An acceleration scheme for the method is presented in Section 4. To show advantages of the method, we perform some numerical experiments in Section 5. Ultimately, in Section 6 we give some concluding remarks.

2. Preliminaries

Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a smooth vector-valued function and consider a closed convex and pointed cone $K \subset \mathbb{R}^m$ with $\mathrm{int}(K) \neq \emptyset$, for which the partial order $\preceq_K$ can be defined by [14]

$$u \preceq_K v \Leftrightarrow v - u \in K, \text{ and similarly, } u \prec_K v \Leftrightarrow v - u \in \mathrm{int}(K).$$

With these notations, a general unconstrained multiobjective optimization problem can be given by

$$\min_{x \in \mathbb{R}^n} F(x), \tag{2.1}$$

in which "min" refers to determining the Pareto optimal solutions.

Definition 2.1 ([31])**.** A point $x^* \in \mathbb{R}^n$ is a K-optimal solution of (2.1) if there does not exist $x \in \mathbb{R}^n$ satisfying $F(x) \preceq_K F(x^*)$ and $F(x) \neq F(x^*)$.

Fliege and Svaiter [17] developed some useful tools for multiobjective optimization such as directional derivative, steepest descent direction and Armijo line search. Since we need these concepts in our analysis, here we briefly review necessary notations, definitions and results of the literature.

Firstly, note that the positive polar cone of K is defined by $K^* := \{w \in \mathbb{R}^m | \langle w, y\rangle \geq 0,\ \forall y \in K\}$, and also, $-\text{int}(K) := \{y \in \mathbb{R}^m | \langle w, y\rangle < 0,\ \forall w \in K^* - \{0\}\}$ in which $\langle , \rangle$ stands for the inner product. Moreover, let JF be the Jacobian matrix of F where its image in $\mathbb{R}^m$ is denoted by Image(JF). In addition, a point $x \in \mathbb{R}^n$ is Pareto critical for (2.1) if it satisfies $\text{Image}(\text{JF}(x)) \cap \mathbb{R}^m_{++} = \varnothing$, in which $\mathbb{R}_{++}$ is the set of strictly positive real numbers and $\text{Image}(\text{JF}(\cdot))$ denotes range of the Jacobian matrix.

Definition 2.2. Focusing on the multiobjective optimization, d is a K-descent direction for F at x if and only if $\text{JF}(x)d \prec 0$, and also, x is a Pareto critical point of F if and only if for all $d \in \mathbb{R}^n$ there exists an index $i = 1, \ldots, m$ for which $(\text{JF}(x)d)_i \geq 0$.

The search direction d can be determined by solving the following convex quadratic programming problem [17]:

$$\min_{d \in \mathbb{R}^n} f_x(d) + \frac{1}{2}\|d\|^2, \tag{2.2}$$

where $f_x(d) = \max_{1 \leq i \leq m} (\text{JF}(x)d)_i$ and $\| \cdot \|$ stands for the Euclidean norm. Due to closeness and uniform convexity of the objective function (2.2), it has a unique optimizer $d(x)$ with the optimal value $\theta(x)$.

Lemma 2.1 ([17])**.** *Let $d(x)$ be the solution of (2.2) and $\theta(x)$ be its optimal value. Then, the statements below hold.*

- *The following three conditions are equivalent: (a) x is Pareto critical, (b) $d(x) = 0$, and (c) $\theta(x) = 0$.*
- *The following three conditions are equivalent: (a) x is not Pareto critical, (b) $d(x) \neq 0$, and (c) $\theta(x) < 0$.*
- *The mappings $x \to d(x)$ and $x \to \theta(x)$ are continuous.*

Based on the above discussions, Fliege and Svaiter [17] developed a steepest descent method for solving the multiobjective optimization problem (2.1) in which successive approximations are generated by

$$x_{k+1} = x_k + \alpha_k d_k, \qquad \text{for all } k \geq 0, \tag{2.3}$$

starting from a given point $x_0 \in \mathbb{R}^n$, where k is an index counter and $\alpha_k > 0$ is a step length commonly determined by some inexact line search strategies along the direction d_k.

As an important step of the iterative methods of unconstrained optimization, an Armijo-type line search condition [39] (as a condition for ensuring sufficient decrease in the objective function value) in multiobjective optimization is suggested in [14] as follows:

$$F(x + \alpha d) \preceq_{\text{KF}} (x) + \gamma\alpha \text{JF}(x)d,$$

where $0 < \gamma < 1$. (See also [17, 29].)

3. Analysis of a nonmonotone line search technique with a forgetting factor for multiobjective optimization

Here, we take advantage of the nonmonotone line search scheme with a special view of the later information. In this context, a forgetting factor is assigned to the recent function values, decreasing the effect of the older iterates. This strategy turned out to be useful in the recursive least squares approach [25, 27, 32].

To be more exact, here we embed a forgetting factor in the backtracking Armijo line search technique of Mita *et al.* [31] in the sense of determining the smallest integer $h \geq 0$ satisfying

$$F\big(x_k + \sigma^h a d_k\big) \preceq_K \mathfrak{F}^{l(k)} + \gamma\sigma^h a \mathrm{JF}^k d_k, \tag{3.1}$$

where $\mathfrak{F}^{l(k)} = (\mathfrak{F}_1^{l_1(k)}, \mathfrak{F}_2^{l_2(k)}, \ldots, \mathfrak{F}_m^{l_m(k)})$ and

$$\mathfrak{F}_i^{l_i(k)} = \max_{0 \leq j \leq m(k)} \eta^{\operatorname{sgn}\left(F_i^{k-j}\right)j} F_i^{k-j}, \qquad \forall i = 1, \ldots, m,$$

in which

- $\operatorname{sgn}(\cdot)$ denotes the sign function;
- $l_i(k) = k - j^*$ where j^* is the index that gives the maximum;
- $F^k = F(x_k)$;
- $a > 0$ is the initial estimation of the step length;
- σ^h is the step length decreasing multiplier to perform the backtracking with $\sigma \in (0, 1)$;
- $\gamma \in (0, 1)$ is the parameter of classical Armijo line search condition;
- $\eta \in (0, 1]$ is the forgetting factor;
- $m(k)$ is the biggest integer in the interval $[0, \min\{m(k-1) + 1, M\}]$, for all $k > 0$, with $m(0) = 0$ and the integer $M > 0$ is a prespecified number of required recent iterates.

Note that if $\eta = 1$, then $\mathfrak{F}_i^{l_i(k)} = F_i^{l_i(k)}$ and so, equation (3.1) reduces to the maximum-type nonmonotone line search condition proposed in [31].

As seen, the factor $\eta^{\operatorname{sgn}(F_i^{k-j})j}$ controls the effect of the corresponding function value on the line search condition. More precisely, in the right hand side of (3.1), F_i^k appears with the weight one while the other function values imposed to decrease by appropriate powers of η in the sense that $\eta^{\operatorname{sgn}(F_i^{k-j})j} F_i^{k-j} \leq F_i^{k-j}$. In light of convergence analysis conducted by Grippo *et al.* [21], now we should deal with the following theorem.

Theorem 3.1. *Assume that for the sequence $\{x_k\}_{k\geq 0}$ defined by (2.3) with the backtracking line search (3.1),*

(i) $\Omega_0 := \{x : F(x) \preceq_{\mathrm{K}} F(x_0)\}$ *is compact;*
(ii) *there exist scalars $c_1 > 0$ and $c_2 > 0$ such that*

$$\max_{i=1,\cdots,m} \Big(\mathrm{JF}^k d_k\Big)_i \leq -c_1 |\theta(x_k)|,$$
$$\|d_k\|^2 \leq c_2 |\theta(x_k)|;$$

(iii) $\alpha_k = \sigma^{h_k} a$ *in which h_k is the smallest integer $h > 0$ satisfying (3.1).*

Then, $\{x_k\}_{k\geq 0} \subseteq \Omega_0$ and for every accumulation point $\bar{x}$ of $\{x_k\}_{k\geq 0}$ we have $\theta(\bar{x}) = 0$.

Proof. We proceed by the following four assertions.

Phase 1. The sequences $\{\mathfrak{F}_i^{l_i(k)}\}_{k\geq 0}$, $i = 1, \ldots, m$, are convergent.

Since $(\mathrm{JF}^k d_k)_i < 0$ and $m(k+1) \leq m(k) + 1$, from (3.1) for each $i = 1, \ldots, m$, we have

$$\begin{aligned}
\mathfrak{F}_i^{l_i(k+1)} &= \max_{0 \leq j \leq m(k+1)} \Big\{\eta^{\operatorname{sgn}\left(F_i^{k+1-j}\right)j} F_i^{k+1-j}\Big\} \\
&\leq \max_{0 \leq j \leq m(k)+1} \Big\{\eta^{\operatorname{sgn}\left(F_i^{k+1-j}\right)j} F_i^{k+1-j}\Big\} \\
&\leq \max\Big\{\mathfrak{F}_i^{l_i(k)}, F_i^{k+1}\Big\} = \mathfrak{F}_i^{l_i(k)},
\end{aligned}$$

which ensures that the sequence $\{\mathfrak{F}_i^{l_i(k)}\}_{k\geq 0}$ is nonincreasing. Moreover, since $F^k \preceq F(x_0)$, for every $k > 0$, by taking (3.1) into consideration, for all i we have $\mathfrak{F}_i^{l_i(k)} \leq F_i^{l_i(k)} \leq F_i(x_0)$, and so $\mathfrak{F}^{l(k)} \preceq_K F(x_0)$ which leads to $\{x_k\}_{k\geq 0} \subset \Omega_0$. Now, because F_i is continuous, the sequence $\{F_i^k\}_{k\geq 0}$ is bounded on Ω_0 and so, since $\mathfrak{F}_i^{l_i(k)} \geq F_i^k$, the sequence $\{\mathfrak{F}_i^{l_i(k)}\}_{k\geq 0}$ is bounded below and consequently, it is convergent.

Phase 2. $\lim_{k\to\infty} \mathfrak{F}_i^{l_i(k)} = \lim_{k\to\infty} F_i^k$, $i = 1, \ldots, m$.

From (3.1), for all $k > M$ we have

$$\begin{aligned}
\mathfrak{F}_i^{l_i(k)} &\leq F_i^{l_i(k)} = F_i\big(x_{l_i(k)-1} + \alpha_{l_i(k)-1} d_{l_i(k)-1}\big) \\
&\leq \max_{0\leq j\leq m(l_i(k)-1)} \left\{ \eta^{\operatorname{sgn}\left(F_i^{l_i(k)-1-j}\right) j} F_i^{l_i(k)-1-j} \right\} + \gamma \alpha_{l_i(k)-1} \Big(\mathrm{JF}^{l_i(k)-1} d_{l_i(k)-1}\Big)_i \\
&= \mathfrak{F}_i^{l_i(l_i(k)-1)} + \gamma \alpha_{l_i(k)-1} \Big(\mathrm{JF}^{l_i(k)-1} d_{l_i(k)-1}\Big)_i \\
&\leq F_i^{l_i(l_i(k)-1)} + \gamma \alpha_{l_i(k)-1} \Big(\mathrm{JF}^{l_i(k)-1} d_{l_i(k)-1}\Big)_i.
\end{aligned}$$

Hence, similar to the proof of the Lemma 8 of [31], it can be seen that $\lim_{k\to\infty} F_i^{l_i(k)} = \lim_{k\to\infty} F_i^k$. Also, we have $F_i^k \leq \mathfrak{F}_i^{l_i(k)} \leq F_i^{l_i(k)}$ which yields the desired result.

Phase 3. The backtracking line search with the condition (3.1) terminates in a finite number of iterations.

Let $\bar{x}$ be an arbitrary accumulation point of $\{x_k\}_{k\geq 0}$ and set $\{x_k\}_{k\in I} \subseteq \{x_k\}_{k\geq 0}$ as a subsequence which converges to $\bar{x}$. In addition, contrary to our claim, suppose that there exist $r \in \{1, \ldots, m\}$ and an index $\bar{k}$ so that for each index $k \in I \cap [\bar{k}, +\infty]$ we have

$$\begin{aligned}
F_r\Big(x_k + \frac{\alpha_k}{\sigma} d_k\Big) &> \mathfrak{F}_r^{l_r(k)} + \gamma \frac{\alpha_k}{\sigma} \Big(\mathrm{JF}^k d_k\Big)_r \\
&\geq F_r^k + \gamma \frac{\alpha_k}{\sigma} \Big(\mathrm{JF}^k d_k\Big)_r.
\end{aligned}$$

Accordingly, we can write

$$\frac{F_r\big(x_k + \frac{\alpha_k}{\sigma} d_k\big) - F_r^k}{\frac{\alpha_k}{\sigma}} > \gamma \Big(\mathrm{JF}^k d_k\Big)_r.$$

Thus, there exists some $\xi_k \in (0,1)$ such that

$$\nabla F_r\Big(x_k + \xi_k \frac{\alpha_k}{\sigma} d_k\Big)^T d_k > \gamma \Big(\mathrm{JF}^k d_k\Big)_r = \gamma \big(\nabla F_r^k\big)^T d_k. \tag{3.2}$$

Now, consider $I' \subseteq I$ such that

$$\lim_{k\to\infty,\ k\in I'} \frac{d_k}{\|d_k\|} = \bar{d}.$$

Dividing both sides of (3.2) by $\|d_k\|$ and taking the limits when k tends to infinity, we get $(1-\gamma)\nabla F_r(\bar{x})^T \bar{d} > 0$, which since $1-\gamma > 0$, contradicts (ii). Therefore, line search (3.1) is well-defined.

Phase 4. $\lim_{k\to\infty} |\theta(x_k)| = 0$.

From (3.1) and Phase 2, we get

$$\lim_{k\to\infty} \alpha_k \max_{1\le i\le m} \left(\mathrm{JF}^k d_k\right)_i = 0,$$

and so, from (ii), we have

$$\lim_{k\to\infty} \alpha_k |\theta(x_k)| = 0,$$

which by Phase 3 yields $\lim_{k\to\infty} |\theta(x_k)| = 0$. Therefore, considering Lemma 2 of [31], the sequence $\{x_k\}_{k\ge 0}$ converges to a Pareto critical point. □

It is worth noting that from Proposition 2 of [31], for the steepest descent method condition (ii) of Theorem 3.1 holds with $c_1 = 1$ and $c_2 = 2$; ensuring global convergence of the nonmonotone steepest descent method with forgetting factor in multiobjective optimization. Now, considering (3.1), the following algorithm can be rendered.

Algorithm 1. Nonmonotone steepest descent algorithm with forgetting factor for multiobjective optimization

Step 0. {Initialization} Choose $x_0 \in \mathbb{R}^n$, the termination tolerance $\epsilon > 0$, and the constants $M \ge 0$, $\gamma \in (0,1)$, $\sigma \in (0,1)$ and $\eta \in (0,1]$. Set $k = 0$.
Step 1. {Stopping criterion} **If** $|\theta(x_k)| < \epsilon$, **then stop**.
Step 2. Compute d_k by solving (2.2) and set $\alpha_k = \alpha_{k_0}$.
Step 3. **If** $k = 0$ **then** set $m(k) = 0$; **else** select the largest integer $m(k) \in [0, \min\{m(k-1)+1, M\}]$.
Step 4. {Line search} **If** α_k satisfies (3.1), **then** set $x_{k+1} = x_k + \alpha_k d_k$, $k = k+1$ and **goto** Step 1.
Step 5. Set $\alpha_k = \sigma\alpha_k$ and **goto** Step 4.

In Step 2 of Algorithm 1, α_{k_0} is the initial guess of the step length at the kth iteration, can be computed using the available data. Here, since when convergence occurs the sequence $\{\alpha_k\|d_k\|\}_{k\ge 0}$ decreasingly tends to zero, for sufficiently large values of k we may have $\alpha_k\|d_k\| \le \alpha_{k-1}\|d_{k-1}\|$. Hence, we follow the approach of [4] and determine α_{k_0} by

$$\alpha_{k_0} = \begin{cases} \frac{1}{\|d_k\|_\infty}, & k = 0, \\ \alpha_{k-1}\frac{\|d_{k-1}\|}{\|d_k\|}, & \text{otherwise.} \end{cases} \tag{3.3}$$

4. Acceleration strategy

In an effort to accelerate convergence of the classical steepest descent method, Andrei [3] suggested an acceleration scheme to be used in the backtracking Armijo line search. Here, we develop Andrei's approach for the vector-valued functions.

Following the analysis conducted in [3], after determining the step length α_k, a positive parameter λ_k is embedded to (2.3) as follows:

$$x_{k+1} = x_k + \lambda_k \alpha_k d_k.$$

Then, λ_k is computed employing the second-order information effectively. To be more exact, having used the Taylor expansion, for each $i = 1, \cdots, m$, we have

$$F_i(x_k + \alpha_k d_k) = F_i(x_k) + \alpha_k \nabla F_i(x_k)^T d_k \frac{1}{2}\alpha_k^2 d_k^T \nabla^2 F_i(x_k) d_k + o\Big(\|\alpha_k d_k\|^2\Big),$$

and similarly,

$$F_i(x_k + \lambda\alpha_k d_k) = F_i(x_k) + \lambda\alpha_k \nabla F_i(x_k)^T d_k + \frac{1}{2}\lambda^2\alpha_k^2 d_k^T \nabla^2 F_i(x_k) d_k + o\big(\|\lambda\alpha_k d_k\|^2\big).$$

So, we can write

$$F_i(x_k + \lambda\alpha_k d_k) = F_i(x_k + \alpha_k d_k) + \Psi_{k,i}(\lambda),$$

where

$$\Psi_{k,i}(\lambda) = -(1-\lambda)\alpha_k \nabla F_i(x_k)^T d_k - \frac{1}{2}\big(1-\lambda^2\big)\alpha_k^2 d_k^T \nabla^2 F_i(x_k) d_k + \lambda^2 \alpha_k o\big(\alpha_k \|d_k\|^2\big) - \alpha_k o\big(\alpha_k \|d_k\|^2\big).$$

Now, letting

$$\begin{aligned} a_{k,i} &= -\alpha_k \nabla F_i(x_k)^T d_k \geq 0, \\ b_{k,i} &= \alpha_k^2 d_k^T \nabla^2 F_i(x_k) d_k, \\ \varepsilon_{k,i} &= o\big(\alpha_k \|d_k\|^2\big), \end{aligned}$$

we have

$$\Psi_{k,i}(\lambda) = -\frac{1}{2}\big(1-\lambda^2\big) b_{k,i} + (1-\lambda) a_{k,i} + \lambda^2 \alpha_k \varepsilon_{k,i} - \alpha_k \varepsilon_{k,i}.$$

After some algebraic manipulations, it can be seen that

$$\widetilde{\lambda}_{k,i} = \frac{a_{k,i}}{b_{k,i} + 2\alpha_k \varepsilon_{k,i}}, \tag{4.1}$$

is a minimizer of $\Psi_{k,i}(\lambda)$ provided that $b_{k,i} + 2\alpha_k \varepsilon_{k,i} > 0$. As discussed in [3], contribution of $\varepsilon_{k,i}$ can be ignored in (4.1). Also, $b_{k,i}$ can be approximated by $b_{k,i} = -\alpha_k y_{k,i}^T d_k$ in which $y_{k,i} = \nabla F_i(x_k) - \nabla F_i(z_k)$ with $z_k = x_k + \alpha_k d_k$. Hence, for $i = 1, \cdots, m$, $\lambda_{k,i}$ is computed as follows:

$$\lambda_{k,i} = \begin{cases} \frac{a_{k,i}}{b_{k,i}}, & b_{k,i} > 0, \\ 1, & \text{otherwise}, \end{cases}$$

which is a modified version of $\widetilde{\lambda}_{k,i}$. Now, we define

$$\lambda_k = \min_{1 \leq i \leq m} \{\lambda_{k,i}\}, \tag{4.2}$$

as an acceleration parameter to be embedded in (2.3). Based on the above discussion, the following accelerated version of Algorithm 1 can be proposed.

Algorithm 2. Accelerated nonmonotone steepest descent algorithm with forgetting factor for multiobjective optimization

Step 0. {Initialization} Choose $x_0 \in \mathbb{R}^n$, the termination tolerance $\epsilon > 0$, and the constants $M \geq 0$, $\gamma \in (0,1)$, $\sigma \in (0,1)$ and $\eta \in (0,1]$. Set $k = 0$.
Step 1. {Stopping criterion} **If** $|\theta(x_k)| < \epsilon$, **then stop**.
Step 2. Compute d_k by solving (2.2) and set $\alpha_k = \alpha_{k_0}$.
Step 3. **If** $k = 0$ **then** set $m(k) = 0$; **else** select the largest integer $m(k) \in [0, \min\{m(k-1)+1, M\}]$.
Step 4. {Line search} **If** α_k satisfies (3.1), **then** set $z_k = x_k + \alpha_k d_k$ and **goto** Step 6.
Step 5. Set $\alpha_k = \sigma\alpha_k$ and **goto** Step 4.
Step 6. {Acceleration} Compute λ_k by (4.2), set $x_{k+1} = x_k + \lambda_k \alpha_k d_k$ and **goto** Step 1.

5. Numerical experiments

Here, we deal with computational results obtained by applying MATLAB 8.5 (R2015a) implementations of nonmonotone line search technique with and without forgetting factor as well as the accelerated nonmonotone line search strategy with forgetting factor, on the steepest descent method for multiobjective optimization; the corresponding methods are respectively called:

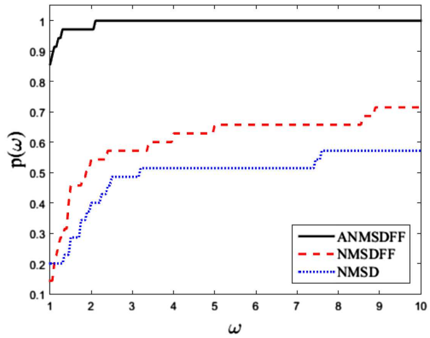
Table 1. Outputs.

Function	n	m	Ref.	NMSD NI	NMSD TNF	NMSDFF NI	NMSDFF TNF	ANMSDFF NI	ANMSDFF TNF
DGO1	1	2	[24]	200.6	1608.8	1.8	18.6	1	15
DGO2	1	2	[24]	1000	8004	12.4	105	1.4	19.4
MOP1	1	2	[24]	1000	8004	1000	8004	1	15
MOP2	2	2	[24]	1	4	1	4	1	4
MOP3	2	2	[24]	11	96.2	8.4	75	10.8	124.2
MLF1	1	2	[24]	1.6	9	1.6	9	1.4	8.6
MLF2	2	2	[24]	53.6	437.6	56.4	459.2	16.8	188.8
IM1	2	2	[24]	76.8	618.6	52.2	421.6	39.2	435.2
LE1	2	2	[24]	408.8	3276.6	13	111.4	9	105.6
SSFYY1	2	2	[24]	1000	8004	95.6	772.6	1	15
SSFYY2	1	2	[24]	34.6	280.8	34.6	280.8	2.4	30.4
FF1	2	2	[24]	6.8	61.2	7.4	66.2	6.8	81.2
LTDZ1	2	2	[26]	1000	8004	41	336.2	1	15
SLC1	2	2	[35]	1000	8004.2	8.4	75.2	5.8	68.8
L1	2	2	[24]	91.2	738.2	100.6	813	3.4	42
L3	2	2	[24]	56.8	461.2	54.8	445.8	2	26.8
L4	2	2	[24]	442.4	3545.6	63.4	514.8	1	15
MMR1	2	2	[30]	14.4	124.4	5.8	56	6.2	77.2
SP1	2	2	[36]	83.4	675.8	95.8	775.4	11.2	127.4
SK2	4	2	[24]	355.6	2851.4	307.8	2469.8	208.8	2303.4
TKLY1	4	2	[24]	225.6	1811.4	266.8	2143.8	463	5098.6
Kur1	10	2	[24]	200	1604.4	200	1604.8	200	2211.6
JOS1	50	2	[24]	998.8	7994.4	1000	8004	1	15
MHHM1	1	3	[24]	402	3213	5	37.8	2	16
MHHM2	2	3	[24]	8.6	66.6	6	45.4	2	15.4
VFM1	2	3	[24]	2.4	15.6	2.6	17.2	1.8	13.2
Far1	2	3	[24]	13.6	116	16.4	138.6	16.2	184.2
MOP7	2	3	[24]	293.4	2352.8	335.6	2690.2	21.8	247.8
MOP5	2	3	[24]	1	4	1	4	1	4
IKK1	2	3	[24]	204.4	1631.6	9.8	75.4	1.8	13
AP1	2	3	[5]	610.2	4896.2	605.4	4859.6	255	2827.4
AP4	3	3	[5]	912	7320.4	911.8	7319.8	482	5419
DTLZ5	3	3	[24]	4.8	36.8	3.8	29.4	3.6	36
FDSa	10	3	[18]	224.2	1802.8	247.4	1988.4	231.4	2551
FDSb	50	3	[18]	426.2	3419	269.4	2164.4	190.4	2098.6
DTLZ1	6	5	[24]	1000	8010.6	1000	8012	1000	11309.6
DTLZ5	5	5	[24]	1000	8004	980.8	7852.2	1	15

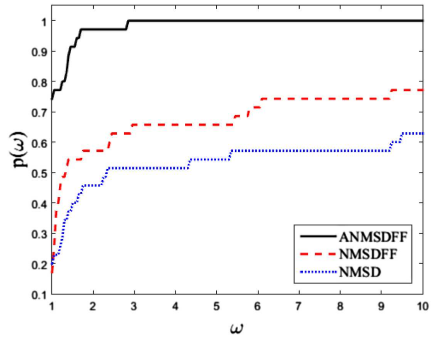
– NMSD: nonmonotone steepest descent method (Algorithm 1 with $\eta = 1$);
– NMSDFF: nonmonotone steepest descent method with the forgetting factor (Algorithm 1 with $\eta = 0.95$);
– ANMSDFF: accelerated nonmonotone steepest descent method with the forgetting factor (Algorithm 2).

The runs were performed on a set of 37 unconstrained multiobjective optimization test problems of [18,24,29], as given in Table 1, using a computer with Intel(R) Core(TM)2 Duo CPU 2.3 GHz with 8 GB of RAM.

In the line search procedure, we set $a = \alpha_{k_0}$ as given in (3.3), $\gamma = 0.0001$, $\sigma = 0.25$ and $M = 5$. For solving the subproblem (2.2), we applied the "quadprog" function of MATLAB library. The algorithms were terminated when $k > 1000$ or $|\theta(x_k)| < \epsilon = 10^{-5}$ (declaring convergence). All the problems were solved 10 times with different starting points obtained from a uniform random distribution inside the parameter domain specified in



(A)



(B)

FIGURE 1. Results of comparisons with accordance to NI (A) and TNF (B) using the backtracking Armijo line search.

[24]. The results are reported in Table 1 in which NI stands for the average number of iterations and TNF is the average of total number of function and Jacobian matrix evaluations being equal to NF + 3NJ, where NF and NJ respectively denote the number of function and Jacobian matrix evaluations [23].

The efficiency of the algorithms was compared applying the performance profile proposed by Dolan and Moré [12] on the NI and TNF. Figure 1 illustrates the results of comparisons. As the results show, NMSDFF and ANMSDFF outperform NMSD. Especially, ANMSDFF is preferable to NMSDFF. Thus, using the nonmonotone line search strategy and particularly, employing Andrei's acceleration scheme in the steepest descent method for multiobjective optimization can be practically effective.

We also investigated effect of the proposed nonmonotone strategy on the strong Wolfe-like conditions suggested in [29] in the sense of

$$F(x_k + \alpha_k d_k) \preceq_K \mathfrak{F}^{l(k)} + \delta \alpha_k f(x_k, d_k) e, \tag{5.1}$$

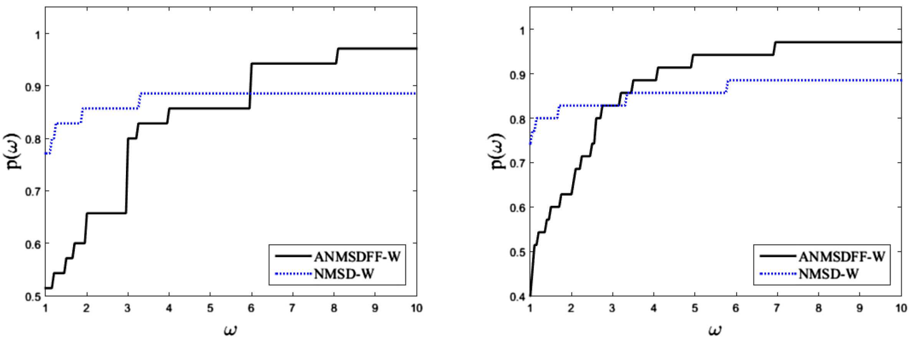
$$|f(x_k + \alpha_k d_k)| \leq \varrho |f(x_k, d_k)|, \tag{5.2}$$

in which $f(x,d) = \sup\{\langle \mathrm{JF}(x)d, w\rangle |\ w \in K^*,\ \|w\| = 1\}$ and $e \in K$ is a vector such that $0 < \langle w, e\rangle \leq 1$ for every unitary vector $w \in K^*$, with $\delta = 10^{-4}$ and $\varrho = 0.9$. Here, substituting (3.1) by (5.1) and (5.2) in Step 4 of Algorithms 1 and 2, we get two other algorithms respectively named by NMSD-W and ANMSDFF-W. As seen in Figure 2, although NMSD-W solves most of the problems faster, there is a category of the test problems for which ANMSDFF-W is considerably preferable.

As known, the ability to generate a well-distributed approximate Pareto front is considered as a positive feature of the multiobjective optimization algorithms [18, 24]. So, we investigated the mentioned ability for the three algorithms NMSD, NMSDFF, and ANMSDFF on the two test problems MOP2 and MLF1. The results have been illustrated in Figures 3 and 4. As seen, all the three algorithms can competitively generate well-distributed approximate Pareto fronts.

6. CONCLUSIONS

For solving unconstrained multiobjective optimization problems, a modified nonmonotone line search technique has been suggested by embedding forgetting factor in the nonmonotone line search condition proposed by Mita *et al.* [31]. Forgetting factor has been added to enhance probability of applying more recent function values. The method has been established to have global convergent property. Moreover, an acceleration scheme has been developed which can be used after the line search. Numerical performance of the nonmonotone and the accelerated nonmonotone methods with forgetting factor have been compared with a recent nonmonotone



(A) (B)

FIGURE 2. Results of comparisons with accordance to NI (A) and TNF (B) using the Wolfe-like line search.

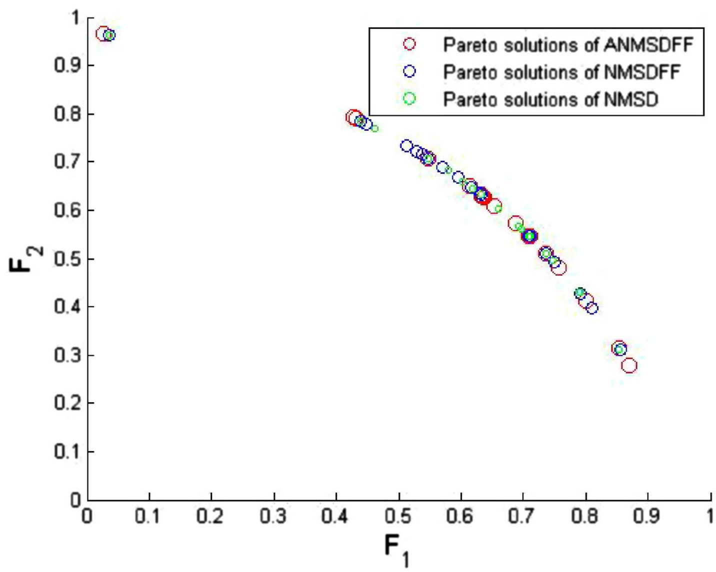


FIGURE 3. Approximate Pareto front for the problem MOP2 with 100 uniformly distributed random starting points.

steepest descent method using 37 standard test problems. The results showed that the proposed methods are practically effective in reducing the number of iterations as well as the total number of function and Jacobian matrix evaluations. Thus, the nonmonotone/acceleration approach with forgetting factor may enhance efficiency of the steepest descent method for solving multiobjective optimization problems.

As possible future works, one can investigate possibility of applying nonmonotone line search techniques to the recent Newton-like methods for minimizing vector-valued functions. Also, developing other acceleration schemes can be studied. Besides, the proposed line search strategy can be embedded in evolutionary algorithms (for example as a local search technique) to find Pareto optimal set. Especially, the proposed method can be combined with various algorithms such as reference set method [37] to solve the well-known multi-criteria decision making problem.

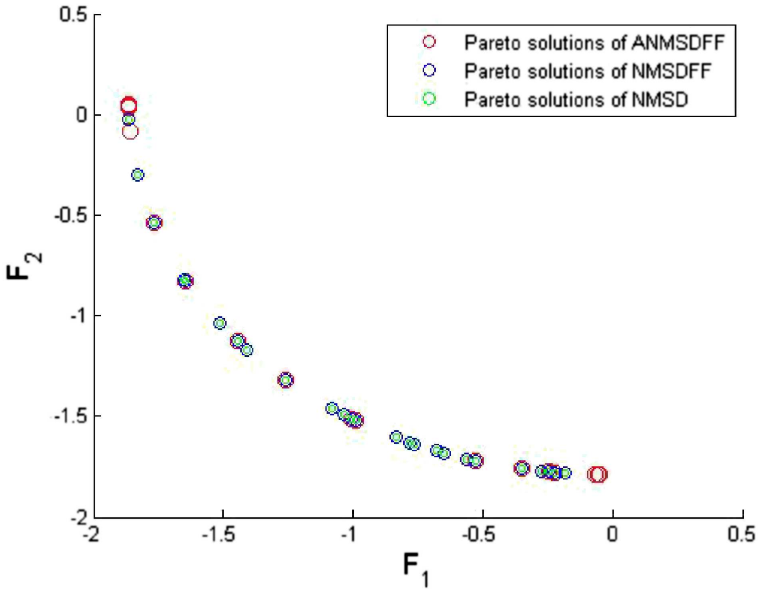


FIGURE 4. Approximate Pareto front for the problem MLF1 with 100 uniformly distributed random starting points.

Acknowledgements

The authors owe a major dept of gratitude to Professor Leandro F. Prudente from Federal University of Goiás (UFG) for providing the strong Wolfe-like line search code. Also, they thank the anonymous reviewers for the valuable comments helped to improve the quality of this work.

References

[1] M. Ahookhoosh, K. Amini and M. Peyghami, A nonmonotone trust region line search method for large-scale unconstrained optimization. *Appl. Math. Model.* **36** (2012) 478–487.

[2] Z. Aminifard, S. Babaie-Kafaki and F. Dargahi, Nonmonotone quasi-Newton-based conjugate gradient methods with application to signal processing. *Numer. Algorithms* **93** (2023) 1527–1541.

[3] N. Andrei, An acceleration of gradient descent algorithm with backtracking for unconstrained optimization. *Numer. Algorithms* **42** (2006) 63–73.

[4] N. Andrei, Another hybrid conjugate gradient algorithm for unconstrained optimization. *Numer. Algorithms* **47** (2008) 143–156.

[5] M.A.T. Ansary and G. Panda, A modified quasi-Newton method for vector optimization problem. *Optimization* **64** (2015) 2289–2306.

[6] E. Birgin, J.M. Martínez and M. Raydan, Nonmonotone spectral projected gradient methods on convex sets. *SIAM J. Optim.* **10** (2000) 1196–1211.

[7] E. Carrizosa and J.B.G. Frenk, Dominating sets for convex functions with some applications. *J. Optim. Theory Appl.* **96** (1998) 281–295.

[8] G. Chiandussi, M. Codegone, S. Ferrero and F.E. Varesio, Comparison of multi-objective optimization methodologies for engineering applications. *Comput. Math. Appl.* **63** (2012) 912–942.

[9] T.D. Chuong, Newton-like methods for efficient solutions in vector optimization. *Comput. Optim. Appl.* **54** (2013) 495–516.

[10] Z. Cui and B. Wu, A new modified nonmonotone adaptive trust region method for unconstrained optimization. *Comput. Optim. Appl.* **53** (2012) 795–806.

[11] Y.H. Dai, A nonmonotone conjugate gradient algorithm for unconstrained optimization. *J. Syst. Sci. Complex.* **15** (2002) 139–145.

[12] E.D. Dolan and J.J. Moré, Benchmarking optimization software with performance profiles. *Math. Program.* **91** (2002) 201–213.

[13] L.M.G. Drummond and A.N. Iusem, A projected gradient method for vector optimization problems. *Comput. Optim. Appl.* **28** (2004) 5–29.

[14] L.M.G. Drummond and B.F. Svaiter, A steepest descent method for vector optimization. *J. Comput. Appl. Math.* **175** (2005) 395–414.

[15] G.W. Evans, An overview of techniques for solving multiobjective mathematical programs. *Manage. Sci.* **30** (1984) 1268–1282.

[16] N.S. Fazzio and M.L. Schuverdt, Convergence analysis of a nonmonotone projected gradient method for multiobjective optimization problems. *Optim. Lett.* **13** (2019) 1365–1379.

[17] J. Fliege and B.F. Svaiter, Steepest descent methods for multicriteria optimization. *Math. Methods Oper. Res.* **51** (2000) 479–494.

[18] J. Fliege, L.M.G. Drummond and B.F. Svaiter, Newton's method for multiobjective optimization. *SIAM J. Optim.* **20** (2009) 602–626.

[19] E.H. Fukuda and L.M.G. Drummond, On the convergence of the projected gradient method for vector optimization. *Optimization* **60** (2011) 1009–1021.

[20] E.H. Fukuda and L.M.G. Drummond, A survey on multiobjective descent methods. *Pesqui. Oper.* **34** (2014) 585–620.

[21] L. Grippo, F. Lampariello and S. Lucidi, A nonmonotone line search technique for Newton's method. *SIAM J. Numer. Anal.* **23** (1986) 707–716.

[22] L. Grippo, F. Lampariello and S. Lucidi, A truncated Newton method with nonmonotone line search for unconstrained optimization. *J. Optim. Theory Appl.* **60** (1989) 401–419.

[23] W.W. Hager and H. Zhang, Algorithm 851: CG_Descent, a conjugate gradient method with guaranteed descent. *ACM Trans. Math. Softw.* **32** (2006) 113–137.

[24] S. Huband, P. Hingston, L. Barone and L. While, A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans. Evol. Comput.* **10** (2006) 477–506.

[25] R.M. Johnstone, C.R. Johnson Jr., R.R. Bitmead and B.D. Anderson, Exponential convergence of recursive least squares with exponential forgetting factor. *Syst. Control Lett.* **2** (1982) 77–82.

[26] M. Laumanns, L. Thiele, K. Deb and E. Zitzler, Combining convergence and diversity in evolutionary multiobjective optimization. *Evol. Comput.* **10** (2002) 263–282.

[27] S.H. Leung and C.F. So, Gradient-based variable forgetting factor RLS algorithm in time-varying environments. *IEEE T. Signal Process.* **53** (2005) 3141–3150.

[28] G.H. Liu, L.L. Jing, L.X. Han and D. Han, A class of nonmonotone conjugate gradient methods for unconstrained optimization. *J. Optim. Theory Appl.* **101** (1999) 127–140.

[29] L.R. Lucambio Pérez and L.F. Prudente, Nonlinear conjugate gradient methods for vector optimization. *SIAM J. Optim.* **28** (2018) 2690–2720.

[30] E. Miglierina, E. Molho and M.C. Recchioni, Box-constrained multi-objective optimization: a gradient-like method without "a priori" scalarization. *Eur. J. Oper. Res.* **188** (2008) 662–682.

[31] K. Mita, E.H. Fukuda and N. Yamashita, Nonmonotone line searches for unconstrained multiobjective optimization problems. *J. Global Optim.* **75** (2019) 63–90.

[32] C. Paleologu, J. Benesty and S. Ciochina, A robust variable forgetting factor recursive least squares algorithm for system identification. *IEEE Signal Process. Lett.* **15** (2008) 597–600.

[33] S. Qu, Y. Ji, J. Jiang and Q. Zhang, Nonmonotone gradient methods for vector optimization with a portfolio optimization application. *Eur. J. Oper. Res.* **263** (2017) 356–366.

[34] S. Rezaee and S. Babaie-Kafaki, A modified nonmonotone trust region line search method. *J. Appl. Math. Comput.* **57** (2018) 421–436.

[35] O. Schütze, A. Lara and C.C. Coello, The directed search method for unconstrained multi-objective optimization problems, in Proceedings of the EVOLVE – A Bridge Between Probability, Set Oriented Numerics, and Evolutionary Computation. (2011) 1–4.

[36] M. Sefrioui and J. Perlaux, Nash genetic algorithms: examples and applications, in Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No. 00TH8512). Vol. 1. IEEE (2000) 509–516.

[37] A.M.J. Skulimowski, Methods of multicriteria decision support based on reference sets, in Advances in Multiple Objective and Goal Programming: Proceedings of the Second International Conference on Multi-Objective Programming and Goal Programming, Torremolinos, Spain, May 16–18, 1996. Springer (1997) 282–290.

[38] W. Sun, Nonmonotone trust region method for solving optimization problems. *Appl. Math. Comput.* **156** (2004) 159–174.

[39] W. Sun and Y.X. Yuan, Optimization Theory and Methods: Nonlinear Programming. Springer, New York (2006).

[40] R. Tanabe and H. Ishibuchi, A review of evolutionary multimodal multiobjective optimization. *IEEE Trans. Evol. Comput.* **24** (2019) 193–200.

[41] P.L. Toint, An assessment of nonmonotone line search techniques for unconstrained optimization. *SIAM J. Sci. Comput.* **17** (1996) 725–739.

[42] Z. Yu, J. Zang and J. Liu, A class of nonmonotone spectral memory gradient method. *J. Korean Math. Soc.* **47** (2010) 63–70.

[43] H. Zhang and W.W. Hager, A nonmonotone line search technique and its application to unconstrained optimization. *SIAM J. Optim.* **14** (2004) 1043–1056.