Nonmonotone line searches for unconstrained multiobjective optimization problems*

Kanako Mita Ellen H. Fukuda Nobuo Yamashita

June 27, 2019

Abstract

In the last two decades, many descent methods for multiobjective optimization problems were proposed. In particular, the steepest descent and the Newton methods were studied for the unconstrained case. In both methods, the search directions are computed by solving convex subproblems, and the stepsizes are obtained by an Armijo-type line search. As a consequence, the objective function values decrease at each iteration of the algorithms. In this work, we consider nonmonotone line searches, i.e., we allow the increase of objective function values in some iterations. Two well-known types of nonmonotone line searches are considered here: the one that takes the maximum of recent function values, and the one that takes their average. We also propose a new nonmonotone technique specifically for multiobjective problems. Under reasonable assumptions, we prove that every accumulation point of the sequence produced by the nonmonotone version of the steepest descent and Newton methods is Pareto critical. Moreover, we present some numerical experiments, showing that the nonmonotone technique is also efficient in the multiobjective case.

Keywords: multiobjective optimization, steepest descent method, Newton method, nonmonotone line search, Pareto optimality.

1 Introduction

In *multiobjective optimization*, several objective functions have to be minimized or maximized simultaneously. This type of optimization problem has applications in many areas, including engineering, finance and management science. See [6], for instance, for a list of applications and references. Since variables that minimize or maximize all objective functions at once do not necessarily exist, we use the concept of Pareto optimality. More precisely, considering a minimization problem, a point is called *Pareto optimal*, if there does not exist

*Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606–8501, Japan (mita.kanako.45a@kyoto-u.jp, ellen@i.kyoto-u.ac.jp, nobuo@i.kyoto-u.ac.jp).
This is a pre-print of an article published in Journal of Global Optimization. The final authenticated version is available online at https://doi.org/10.1007/s10898-019-00802-0.

a different point with the same or smaller objective function values such that there is a decrease in at least one objective function value.

One well-known method for solving multiobjective optimization problems is the *scalarization* approach [11], which basically converts vector-valued problems into scalar-valued ones, using certain parameters. In particular, the so-called *weighted-sum* method considers a linear combination of the objectives as the objective function of the scalar reformulation. However, when the original problem is nonconvex, it may fail to obtain particular Pareto optimal points. To overcome such drawback, sublinear scalarization methods were also proposed [15]. But as in the classical weighted-sum method, it has the disadvantage of requiring in advance unknown parameters that lead to desired solutions. Another approach, that do not involve scalarizations, is one based on heuristics [10]. In this case, the convergence to Pareto optimal points is not necessarily guaranteed.

Descent methods for multiobjective optimization, that do not require knowledge of unknown parameters and with theoretical guarantee of convergence, have been considered recently. They are basically extensions of methods for single-objective nonlinear programming problems. In particular, for the unconstrained case, the steepest descent method and the Newton method were proposed, respectively by Fliege and Svaiter [5] and Fliege et al. [4]. In both methods, we can find Pareto optimal solutions by choosing appropriate starting points. Moreover, the methods in [4, 5] are iterative, and at each iteration the search directions are computed by solving strongly convex subproblems, and the stepsizes are chosen in order to satisfy an Armijo-type condition. With this, all the objective function values decrease at each iteration. However, since the number of objective functions increases, the Armijo-type condition becomes more strict, possibly leading to smaller stepsizes.

Furthermore, in the single-objective case, it is well-known that enforcing the monotonicity of the function values makes the method to converge slower, in particular when the iterate is in the bottom of a narrow curved valley. During the line search, one can permit some increase in the function values in some iterations to improve the speed of convergence. This technique is called *nonmonotone line search*, and it has been studied considerably for single-objective optimization. For the global convergence, it is required that the current objective function value is smaller than a certain benchmark. In particular, if the maximum of the recent objective function values are taken as a benchmark, then we obtain the nonmonotone line search proposed by Grippo et al. [7]. Instead, Zhang and Hager [19] considered an average-type nonmonotone procedure, which turns out to be even more efficient in practice. Although the nonmonotone techniques can be slower in the case of complex problems that have many optimal points, they are advantageous for improving the convergence speed of many problems, and for being easy to implement. However, the research about nonmonotone line searches for multiobjective descent methods is still recent and insufficient.

In this work, we consider both the max-type and the average-type nonmonotone line searches for unconstrained multiobjective descent methods. We establish the convergence of both the nonmonotone approaches with a general framework, considering that the descent directions satisfy certain assumptions. We then show that these assumptions are satisfied for the steepest descent and the Newton directions. Consequently, we obtain the convergence of the nonmonotone multiobjective steepest descent and Newton methods simultaneously. In addition, we consider a new nonmonotone technique that only makes sense if the problem is multiobjective. We call it hybrid-type nonmonotone line search, because at each iteration, it imposes the Armijo condition for some objective functions, while allowing increase in the remaining objectives. Moreover, we show that the hybrid-type line search also has guarantee of convergence. Finally, we perform some numerical experiments, showing that the nonmonotone approach usually outperforms the monotone version also in the multiobjective case.

We point out that this work started in 2016, during Mita's undergraduate thesis [12], which appears afterwards in the proceedings [13], where the average-type nonmonotone line search for multiobjective steepest descent method was proposed. At the same time, Qu et al. [16] studied the max-type nonmonotone scheme for multiobjective gradient methods, with a different scheme for computing search directions (the subproblem associated to the search directions considers a bound constraint, while here we follow the original works [4, 5], which makes the subproblem unconstrained). Parallel to these works, Fazzio and Schuverdt [3] also obtained convergence results, this time for the average-type nonmonotone projected gradient method. However, no numerical experiments were presented. Although the papers [3, 16] have intersections with ours, the search directions and the scheme used in the convergence proofs are different, as well as the numerical experiments. The hybrid-type nonmonotone line search, which performs better among all the considered line searches, is also brand new as far as we know.

The outline of this paper is as follows. In Section 2, we define some notations, explaining also the basics about Pareto optimality, the multiobjective steepest descent method, the multiobjective Newton method, and nonmonotone procedures for single-objective optimization. In Section 3, we propose two nonmonotone line search algorithms for multiobjective problems. In Section 4, we prove the global convergence of the method with general search directions and nonmonotone line searches, and in Section 5 we consider the more specific nonmonotone steepest descent and Newton methods. In Section 6, we propose a hybrid-type line search specifically for multiobjective problems. In Section 7, we confirm the efficiency of the nonmonotone schemes with numerical experiments. Finally, we present some conclusions in Section 8.

2 Preliminaries

We start with some notations that will be used in the whole paper. We denote by $\mathbb{R}_{++}$ the set of strictly positive real numbers, and $\mathbb{R}^m_{++} := \mathbb{R}_{++} \times \cdots \times \mathbb{R}_{++}$. The Euclidean norm is written as $\|\cdot\|$, and the transpose sign is denoted by $\top$. For a vector $z \in \mathbb{R}^n$, its entries are written as $z_1, z_2, \ldots, z_n$, and so $z := (z_1, z_2, \ldots, z_n)^\top$. The sequence of vectors $z^1, z^2, \ldots$ is denoted by $\{z^k\}_k$, or simply $\{z^k\}$. If $y, z \in \mathbb{R}^n$, then $y \leq z$ ($y < z$) is equivalent to $y_i \leq z_i$ ($y_i < z_i$) for all i. Moreover, for a matrix $A \in \mathbb{R}^{m\times n}$, we denote its image as $\mathrm{Im}(A)$. For a function $\psi\colon \mathbb{R}^n \to \mathbb{R}$, we write its gradient at $z \in \mathbb{R}^n$ as $\nabla\psi(z)$. Let $\Psi\colon \mathbb{R}^n \to \mathbb{R}^m$ be a vector-valued function, where $\Psi(z) := (\Psi_1(z), \ldots, \Psi_m(z))$ for all z, with $\Psi_i\colon \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, m$.

Then, the Jacobian of Ψ at z is given by $J\Psi(z) := (\nabla\Psi_1(z), \dots, \nabla\Psi_m(z))^\top \in \mathbb{R}^{m\times n}$.

2.1 The multiobjective optimization problem

In this paper, we consider the following multiobjective optimization problem:

$$\begin{array}{ll} \min & F(x) \\ \text{s.t.} & x \in \mathbb{R}^n, \end{array} \tag{1}$$

where $F\colon \mathbb{R}^n \to \mathbb{R}^m$ is a vector-valued function. We assume that

$$F(x) := (F_1(x), \dots, F_m(x))^\top$$

for all $x \in \mathbb{R}^n$, with $F_i\colon \mathbb{R}^n \to \mathbb{R}$, $i = 1, \dots, m$. We also suppose that all these component functions F_i are continuously differentiable and bounded from below. Recall that completely optimal solutions of (1), that minimize all objective functions at once, do not necessarily exist. Because of this, the notion of Pareto optimality becomes crucial in this case.

Definition 2.1. *A point $x^* \in \mathbb{R}^n$ is* Pareto optimal *(or* efficient*) for problem* (1) *if there does not exist $x \in \mathbb{R}^n$ satisfying $F(x) \le F(x^*)$ and $F(x) \ne F(x^*)$. Moreover, a point $x \in \mathbb{R}^n$ is* Pareto critical *(or* stationary*) for* (1) *if it satisfies* $\mathrm{Im}(JF(x)) \cap (-\mathbb{R}^m_{++}) = \emptyset$ *(see [5, page 481]).*

The set of Pareto optimal points are commonly called *Pareto frontier*. The next result shows the relation between Pareto optimality and criticality. We observe that, as usual, the stationarity is a necessary condition for optimality.

Lemma 2.1. *[5, Section 2] For problem* (1)*, the following statements hold.*

1. *If $x \in \mathbb{R}^n$ is Pareto optimal, then x is Pareto critical.*
2. *The point $x \in \mathbb{R}^n$ is not Pareto critical if, and only if, there exists a direction $d \in \mathbb{R}^n$ such that $JF(x)d < 0$. Moreover, $x \in \mathbb{R}^n$ is a Pareto critical point if, and only if, for each $d \in \mathbb{R}^n$, there exist at least one index $i \in \{1, \dots, m\}$ such that $\nabla F_i(x)^\top d \ge 0$.*

The direction d such that $JF(x)d < 0$ is also called *descent direction* of F at x. Clearly, the above lemma shows that Pareto criticality implies the non-existence of descent directions from the corresponding point. We also recall that Pareto criticality does not necessarily imply Pareto optimality, even when F_i are all convex. In fact, this implication only holds under the strict convexity [4, Theorem 3.1].

2.2 Multiobjective descent methods

Let us now recall the steepest descent method for unconstrained multiobjective optimization problems, proposed by Fliege and Svaiter in [5]. We can regard this algorithm as an extension of the classical steepest descent method. We begin by explaining how to choose the search direction and the stepsize of each iteration, and afterwards, we show the algorithm and its convergence results.

For a given $x \in \mathbb{R}^n$, we consider the following unconstrained minimization problem:

$$\begin{aligned} \min \quad & \max_{i=1,\ldots,m} \nabla F_i(x)^\top d + \frac{1}{2}\|d\|^2 \\ \text{s.t.} \quad & d \in \mathbb{R}^n. \end{aligned} \tag{2}$$

The first term of the objective function is convex because it is the maximum of linear functions $d \mapsto \nabla F_i(x)^\top d$. So, the whole objective function is strongly convex, which means that (2) always has a unique solution. Let us denote such a solution by $d_{\mathrm{sd}}(x)$ and let $\theta_{\mathrm{sd}}(x)$ be its optimal function value. The result below shows that $d_{\mathrm{sd}}(x)$ and $\theta_{\mathrm{sd}}(x)$ can be used to characterize stationary points of the multiobjective problem (1).

Lemma 2.2. *[5, Lemma 1] Let $d_{\mathrm{sd}}(x)$ be the solution of* (2) *and $\theta_{\mathrm{sd}}(x)$ be its optimal value. Then, the statements below hold.*

1. *The following three conditions are equivalent: (a) x is Pareto critical, (b) $d_{\mathrm{sd}}(x) = 0$, and (c) $\theta_{\mathrm{sd}}(x) = 0$.*
2. *The following three conditions are equivalent: (a) x is not Pareto critical, (b) $d_{\mathrm{sd}}(x) \neq 0$, and (c) $\theta_{\mathrm{sd}}(x) < 0$.*
3. *The mappings $x \mapsto d_{\mathrm{sd}}(x)$ and $x \mapsto \theta_{\mathrm{sd}}(x)$ are continuous.*

Moreover, $d_{\mathrm{sd}}(x)$ is a descent direction when x is not Pareto critical.

Now, suppose that we have a descent direction $d \in \mathbb{R}^n$, i.e., $JF(x)d < 0$. From Lemma 2.1, x is not Pareto critical. Letting $\delta \in (0,1)$, from [5, Lemma 4], there exists some $\epsilon > 0$ such that

$$F(x + \alpha d) \leq F(x) + \delta \alpha JF(x)d$$

for any $\alpha \in (0, \epsilon]$. This result shows that the choice of stepsizes can be made by implementing an Armijo-type rule with a backtracking procedure. With this, we describe the multiobjective steepest descent method in Algorithm 1.

Algorithm 1: Steepest descent method for multiobjective optimization [5]

---

1 Choose starting point $x^0 \in \mathbb{R}^n$ and parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\varepsilon > 0$. Set $k = 0$.
2 Compute the solution $d_{\mathrm{sd}}(x^k)$ and the optimal value $\theta_{\mathrm{sd}}(x^k)$ of (2) at $x = x^k$.
3 If $|\theta_{\mathrm{sd}}(x^k)| < \varepsilon$, then stop.
4 Take $\alpha_k = \rho^{h_k}$, where h_k is the smallest nonnegative integer such that

$$F\big(x^k + \alpha_k d_{\mathrm{sd}}(x^k)\big) \leq F(x^k) + \delta \alpha_k JF(x^k) d_{\mathrm{sd}}(x^k).$$

5 Set $x^{k+1} = x^k + \alpha_k d_{\mathrm{sd}}(x^k)$, $k = k + 1$ and return to Step 2.

In general, the parameter ε used in Algorithm 1 is arbitrarily small, so the condition in Step 3 means that x is critical from Lemma 2.2. Moreover, if Step 4 is reached in a certain iteration, then from Step 2, $d_{\mathrm{sd}}(x^k)$ is a descent direction, once again from Lemmas 2.1 and 2.2. Furthermore, Algorithm 1 either stops in Step 3 at a Pareto critical point, or generates an infinite sequence of iterates $\{x^k\}$ of nonstationary points. For the latter case, from Step 4 and the fact that $JF(x^k)d_{\mathrm{sd}}(x^k) < 0$, we have

$$F(x^{k+1}) < F(x^k) \quad \text{for all } k,$$

i.e., the objective function values decrease in each iteration of the algorithm.

We recall that Algorithm 1 was proposed in [5] without the stopping condition. Here, we follow [6], and assume in the following theorem that Algorithm 1 generates an infinite sequence of nonstationary points.

Theorem 2.1. *[5, Theorem 1] Every accumulation point of the sequence $\{x^k\}$ generated by Algorithm 1 is Pareto critical. Moreover, if the level set $\{x \in \mathbb{R}^n \mid F(x) \le F(x^0)\}$ is bounded, then the sequence $\{x^k\}$ stays bounded and has at least one accumulation point.*

We now discuss the multiobjective Newton method proposed by Fliege, Graña Drummond and Svaiter [4]. Here, for all $i = 1, \dots, m$, we assume that F_i is twice continuously differentiable and that $\nabla^2 F_i(x)$ is positive definite for all $x \in \mathbb{R}^n$. In this case, for a given point $x \in \mathbb{R}^n$, the search direction is computed by solving the following unconstrained problem:

$$\begin{array}{ll} \min & \displaystyle\max_{i=1,\dots,m} \left(\nabla F_i(x)^\top d + \frac{1}{2} d^\top \nabla^2 F_i(x) d \right) \\ \text{s.t.} & d \in \mathbb{R}^n. \end{array} \tag{3}$$

Note that by replacing the Hessian matrices $\nabla^2 F_i(x)$ with the identity matrix, we recover (2). The objective function of (3) is also strongly convex, because of the positive definiteness of $\nabla^2 F_i(x)$. Thus, denote its unique optimal solution by $d_{\mathrm{N}}(x)$ and its optimal function value by $\theta_{\mathrm{N}}(x)$. The following result corresponds to Lemma 2.2, but translating to the case of the Newton direction.

Lemma 2.3. *[4, Lemma 3.2] Assume that $\nabla^2 F_i(x)$ is positive definite for all $x \in \mathbb{R}^n$ and $i = 1, \dots, m$. Let $d_{\mathrm{N}}(x)$ be the solution of (3) and $\theta_{\mathrm{N}}(x)$ be its optimal value. Then, the statements below hold:*

1. *The following three conditions are equivalent: (a) x is Pareto critical, (b) $d_{\mathrm{N}}(x) = 0$, and (c) $\theta_{\mathrm{N}}(x) = 0$.*
2. *The following three conditions are equivalent: (a) x is not Pareto critical, (b) $d_{\mathrm{N}}(x) \neq 0$, and (c) $\theta_{\mathrm{N}}(x) < 0$.*
3. *The mappings $x \mapsto d_{\mathrm{N}}(x)$ and $x \mapsto \theta_{\mathrm{N}}(x)$ are continuous.*

Moreover, $d_{\mathrm{N}}(x)$ is a descent direction when x is not Pareto critical.

For completeness, we write out the multiobjective Newton method in the algorithm below.

Algorithm 2: Newton method for multiobjective optimization [4]

---

1 Choose starting point $x^0 \in \mathbb{R}^n$ and parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\varepsilon > 0$. Set $k = 0$.

2 Compute the solution $d_{\mathrm{N}}(x^k)$ and the optimal value $\theta_{\mathrm{N}}(x^k)$ of (3) at $x = x^k$.

3 If $|\theta_{\mathrm{N}}(x^k)| < \varepsilon$, then stop.

4 Take $\alpha_k = \rho^{h_k}$, where h_k is the smallest nonnegative integer such that

$$F\big(x^k + \alpha_k d_{\mathrm{N}}(x^k)\big) \leq F(x^k) + \delta \alpha_k JF(x^k) d_{\mathrm{N}}(x^k).$$

5 Set $x^{k+1} = x^k + \alpha_k d_{\mathrm{N}}(x^k)$, $k = k+1$ and return to Step 2.

In the original Newton algorithm [4], the authors use the inequality

$$F(x^k + \alpha_k d_{\mathrm{N}}(x^k)) \leq F(x^k) + \delta \alpha_k \theta_{\mathrm{N}}(x^k)$$

for line search. Here, we consider using the one in Step 4, in order to align with the steepest descent method. At least for the global convergence for Pareto critical points, that we consider here, the above condition is enough.

2.3 Nonmonotone line search for single-objective optimization

In this section, we briefly explain two nonmonotone line search algorithms for scalar-valued optimization, i.e., problem (1) with $m = 1$. For the sake of completeness, we denote this single-objective optimization problem as follows:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in \mathbb{R}^n, \end{array} \tag{4}$$

where $f \colon \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable. Many iterative methods for (4) produce a sequence of iterates $\{x^k\}$ using the update $x^{k+1} = x^k + \alpha_k d^k$, with a search direction d^k and a positive stepsize α_k. In the classical monotone line search, α_k is chosen such that $f(x^{k+1}) < f(x^k)$. It means that the objective function value decreases in every iteration. In nonmonotone line search, some growth in the function value is permitted, that is we choose $\alpha_k > 0$ satisfying

$$f(x^k + \alpha_k d^k) \leq c_k + \delta \alpha_k \nabla f(x^k)^\top d^k, \tag{5}$$

with $\delta \in (0,1)$ and $c_k \geq f(x^k)$.

Among the ways to choose c_k, we first cite the well-known max-type, proposed by Grippo, Lampariello and Lucidi [7]. The procedure consists in taking the maximum of recent objective function values. If d^k is a descent direction, then the such update of c_k and the condition (5) show that $f(x^k) \leq c_k$ for all k. Hence, the relaxation of the line search is

Procedure Max-type nonmonotone line search

1 Choose starting point $x^0 \in \mathbb{R}^n$ and a nonnegative integer M. Set $k = 0$, $m(k) = 0$.
2 Choose $m(k)$ such that $0 \leq m(k) \leq \min\{m(k-1)+1, M\}$, and set

$$c_k = \max_{0 \leq j \leq m(k)} f(x^{k-j}).$$

well-defined. Another nonmonotone line search that is shown to perform well in practice is the one proposed by Zhang and Hager [19]. Instead of taking the maximum, it considers the average of the function values.

Procedure Average-type nonmonotone line search

1 Choose starting point $x^0 \in \mathbb{R}^n$ and parameters $0 \leq \eta_{\min} \leq \eta_{\max} \leq 1$. Set $k = 0$, $c_0 = f(x^0)$, $q_0 = 1$.
2 Choose $\eta_k \in [\eta_{\min}, \eta_{\max}]$, and update q_k and c_k as follows:

$$q_{k+1} = \eta_k q_k + 1,$$
$$c_{k+1} = \frac{\eta_k q_k c_k + f(x^{k+1})}{q_{k+1}}.$$

In average-type nonmonotone line search, c_{k+1} is a convex combination of c_k and $f(x^{k+1})$. Since $c_0 = f(x^0)$, we observe that c_k is actually a convex combination of all function values until iteration k, i.e., $f(x^0)$, $f(x^1)$, ..., and $f(x^k)$. Moreover, if $\eta_k = 0$ for each k, then $c_k = f(x^k)$, and the approach is the usual monotone line search. On the other hand, if $\eta_k = 1$ for each k, then $c_k = (\sum_{i=0}^k f(x^i))/(k+1)$, which is the average function value. It means that the parameter η_k controls how nonmonotone the line search is. The following lemma shows that the line search is well-defined regardless of $\eta_k \in [0,1]$.

Lemma 2.4. *[19, Lemma 1.1] For each iteration k of the average-type nonmonotone line search, if d^k satisfies $\nabla f(x^k)^\top d^k \leq 0$, then we have $f(x^k) \leq c_k \leq (\sum_{i=0}^k f(x^i))/(k+1)$. Moreover, if d^k is a descent direction, i.e., $\nabla f(x^k)^\top d^k < 0$, and f is bounded from below, then there exists $\alpha_k > 0$ satisfying* (5).

In order to establish the global convergence of the two nonmonotone line searches, the following assumption concerning the search directions d^k was considered.

Assumption 2.1. *There exists positive constants Γ_1 and Γ_2 such that*

$$\nabla f(x^k)^\top d^k \leq -\Gamma_1 \|\nabla f(x^k)\|^2,$$
$$\|d^k\| \leq \Gamma_2 \|\nabla f(x^k)\|$$

for all sufficiently large k.

The first inequality of the above assumption means that the search direction d^k is a sufficient descent direction, and the second one means that d^k is not too large in magnitude. For example, the steepest descent direction $d^k = -\nabla f(x^k)$ clearly satisfies Assumption 2.1.

Theorem 2.2. *[7, Section 3] Suppose that f is bounded from below, and that Assumption 2.1 holds. Then, the sequence $\{x^k\}$ generated by the max-type nonmonotone line search has the property that*

$$\lim_{k\to\infty} \nabla f(x^k) = 0.$$

Theorem 2.3. *[19, Theorem 2.2] Suppose that f is bounded from below, and that Assumption 2.1 holds. Assume also that ∇f is Lipschitz continuous on the set*

$$\left\{ x \in \mathbb{R}^n \;\middle|\; \exists\, y \in \mathcal{S} \text{ s.t. } \|x - y\| \le \sup_k \|d^k\| \right\}.$$

where $\mathcal{S}$ is the level set $\{x \mid f(x) \le f(x^0)\}$. Then, the sequence $\{x^k\}$ generated by the average-type nonmonotone line search has the property that

$$\liminf_{k\to\infty} \|\nabla f(x^k)\| = 0.$$

Moreover, if $\eta_{\max} < 1$,

$$\lim_{k\to\infty} \nabla f(x^k) = 0.$$

The numerical results given in [19] indicate that the max-type nonmonotone line search converges faster than the monotone one, and the average-type converges faster than the max-type, when using L-BFGS quasi-Newton method [?].

3 Nonmonotone line search for multiobjective optimization

In this section, we return to the multiobjective optimization problem (1), and propose two nonmonotone line search approaches for it. Basically, we extend the two nonmonotone line searches for single-objective optimization of Section 2.3 to the multiobjective setting. Here, instead of considering the steepest descent or the Newton directions, we just state that the search direction d^k at x^k satisfies some conditions, which is given in the assumption below.

Assumption 3.1. *For a sequence of iterates $\{x^k\}$ and search directions $\{d^k\}$, there exist positive constants Γ_1 and Γ_2 such that*

$$\max_{i=1,\dots,m} \left\{ \nabla F_i(x^k)^\top d^k \right\} \le -\Gamma_1 |\theta_{\mathrm{sd}}(x^k)|, \tag{6}$$

$$\|d^k\|^2 \le \Gamma_2 |\theta_{\mathrm{sd}}(x^k)|, \tag{7}$$

recalling that $\theta_{\mathrm{sd}}(x^k)$ is the optimal value of (2).

Observe that (6) and (7) of this assumption is similar to Assumption 2.1, required in both max-type and average-type nonmonotone techniques for scalar-valued problems. Moreover, from (6), we have

$$(JF(x^k)d^k)_i = \nabla F_i(x^k)^\top d^k \le -\Gamma_1 |\theta_{\mathrm{sd}}(x^k)| \le 0 \quad \text{for all } i. \tag{8}$$

Recall from Lemma 2.2 that $\theta_{\mathrm{sd}}(x^k) \neq 0$ when x^k is not Pareto critical. So, the above inequality shows that d^k is a descent direction if x^k is not critical. On the other hand, if x^k is critical, then from (7), we have $d^k = 0$. In Section 5, we will observe that Assumption 3.1 actually holds when d^k is the steepest descent direction $d_{\mathrm{sd}}(x^k)$ or the Newton direction $d_{\mathrm{N}}(x^k)$. For now, we just state our algorithm considering search directions that satisfy Assumption 3.1.

Algorithm 3: Max-type nonmonotone line search for multiobjective optimization

1 Choose starting point $x^0 \in \mathbb{R}^n$, parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\mu > 0$, and a nonnegative integer M. Set $k = 0$.

2 Compute a search direction d^k at x^k that satisfies Assumption 3.1.

3 If x^k is a Pareto critical point, then stop.

4 Let $m(k) = \min(k, M)$ and set

$$C_i^k = \max_{0 \le j \le m(k)} F_i(x^{k-j}), \quad i = 1, \dots, m. \tag{9}$$

5 Take $\alpha_k = \mu \rho^{h_k}$, where h_k is the smallest nonnegative integer such that

$$F(x^k + \alpha_k d^k) \le C^k + \delta \alpha_k JF(x^k) d^k. \tag{10}$$

6 Set $x^{k+1} = x^k + \alpha_k d^k$, $k = k + 1$ and return to Step 2.

The extension of the max-type procedure is given in Algorithm 3. As usual, we denote $C^k := (C_1^k, \dots, C_m^k)^\top \in \mathbb{R}^m$. We also call (10) an *Armijo-type* condition. The next lemma shows that (10) is actually a relaxation of the usual Armijo condition, where $F(x^k)$ is considered instead of C^k.

Lemma 3.1. *For each iteration k of Algorithm 3, we have $F(x^k) \le C^k$.*

Proof. It follows easily from the definition of C^k in (9). □ □

The extension of the average-type procedure to the multiobjective case is shown in Algorithm 4. For each iteration k, let us define the average of the function values as

$$A^k := \frac{1}{k+1} \sum_{i=0}^{k} F(x^i).$$

Algorithm 4: Average-type nonmonotone line search for multiobjective optimization

1 Choose starting point $x^0 \in \mathbb{R}^n$, and parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\mu > 0$, $\eta \in [0,1]$. Set $k = 0$, $C^0 = F(x^0)$ and $q_0 = 1$.
2 Compute a search direction d^k at x^k that satisfies Assumption 3.1.
3 If x^k is a Pareto critical point, then stop.
4 Take $\alpha_k = \mu\rho^{h_k}$, where h_k is the smallest nonnegative integer such that (10) holds.
5 Set $x^{k+1} = x^k + \alpha_k d^k$.
6 Update q_k and C^k as follows:

$$q_{k+1} = \eta q_k + 1, \tag{11}$$

$$C^{k+1} = \frac{\eta q_k}{q_{k+1}} C^k + \frac{1}{q_{k+1}} F(x^{k+1}). \tag{12}$$

7 Set $k = k + 1$ and return to Step 2.

As in the scalar-valued case, C_i^{k+1} is a convex combination of C_i^k and $F_i(x^{k+1})$ for all $i = 1, \ldots, m$. Moreover, as we can see in the next result, for each i, the values $F_i(x^k)$ and A_i^k are, respectively, lower and upper bounds of C_i^k. Although the proof of the lemma below is similar to [7, Lemma 1.1], we show it to be sure that no complication exists when dealing with vector-valued functions.

Lemma 3.2. *For each iteration k of Algorithm 4, we have $F(x^k) \leq C^k \leq A^k$.*

Proof. Let us first define $D^k \colon \mathbb{R} \to \mathbb{R}^m$ as

$$D^k(t) := \frac{1}{t+1}\left(tC^{k-1} + F(x^k)\right).$$

Then, its derivative is given by

$$(D^k)'(t) := \frac{\partial D^k(t)}{\partial t} = \frac{1}{(t+1)^2}\left(C^{k-1} - F(x^k)\right).$$

Since $JF(x^k)d^k \leq 0$ by Assumption 3.1, or more precisely (8), it follows from the Armijo-type condition (10), that $F(x^k) \leq C^{k-1}$. This implies that $(D^k)'(t) \geq 0$ for all $t \neq -1$, that is, D^k is nondecreasing for all $t \geq 0$. Also, from (11) and the fact that $\eta \in [0,1]$ and $q_0 = 1$, we have $q_k \geq 1$ for all k. Hence, $\eta q_{k-1} \geq 0$ holds, and we obtain

$$F(x^k) = D^k(0) \leq D^k(\eta q_{k-1}) = C^k.$$

Now, let us prove $C^k \leq A^k$ by induction. For $k = 0$, this inequality holds because $C^0 = A^0 = F(x^0)$. So, assume that $C^j \leq A^j$ for all $0 \leq j \leq k$. Because $\eta \in [0,1]$ and $q_0 = 1$, from (11) we can write

$$q_k = 1 + \sum_{i=1}^{k} \eta^i \leq k + 1 \quad \text{for all } k > 0.$$

Thus, $0 \leq q_k - 1 \leq k$ holds. Since D^k is nondecreasing for all $t \geq 0$ and $q_k = \eta q_{k-1} + 1$ in Step 6, we obtain

$$C^k = D^k(\eta q_{k-1}) = D^k(q_k - 1) \leq D^k(k).$$

By the induction step, we also have

$$D^k(k) = \frac{1}{k+1}\left(kC^{k-1} + F(x^k)\right) \leq \frac{1}{k+1}\left(kA^{k-1} + F(x^k)\right) = A^k,$$

and the conclusion follows. □ □

In the next proposition, we prove that Algorithms 3 and 4 are well-defined, in the sense that there is always a stepsize satisfying the Armijo-type condition, so that the iterates $\{x^k\}$ can be generated.

Proposition 3.1. *Let x^k be an iterate of Algorithms 3 or 4. If x^k is not Pareto critical, then there exists a stepsize $\alpha_k > 0$ satisfying the Armijo-type condition* (10)*.*

Proof. Since x^k is not a Pareto critical point, $JF(x^k)d^k < 0$ holds from Assumption 3.1. Moreover, $JF(x^k)d^k < \delta JF(x^k)d^k$ because $\delta \in (0, 1)$. Now, since F is differentiable, we also have

$$F(x^k + \alpha d^k) = F(x^k) + \alpha JF(x^k)d^k + o(\alpha). \tag{13}$$

Hence, there exists $\bar{\alpha} \in (0, 1)$ such that

$$F(x^k + \alpha d^k) \leq F(x^k) + \delta\alpha JF(x^k)d^k \quad \text{for all } \alpha \in (0, \bar{\alpha}].$$

Since $F(x^k) \leq C^k$ from Lemmas 3.1 and 3.2, the above inequality yields

$$F(x^k + \alpha d^k) \leq C^k + \delta\alpha JF(x^k)d^k \quad \text{for all } \alpha \in (0, \bar{\alpha}],$$

and the proof is complete. □ □

4 Global convergence

In this section, we give convergence results for the methods with max-type and average-type nonmonotone line searches. Initially, we consider the general case for the search directions, as stated in Algorithms 3 and 4. Note that these algorithms stop at a Pareto critical point, or generate an infinite sequence $\{x^k\}$ of nonstationary points. So, let us assume the second assertion from now on. As a consequence, from Assumption 3.1, the direction d^k satisfy $JF(x^k)d^k < 0$ for all k. Let us first prove some auxiliary results. In the following lemma, we show a property about the sequence $\{C_i^k\}_k$ produced by both Algorithms 3 and 4.

Lemma 4.1. *Let $\{x^k\}$ be the sequence generated by Algorithms 3 or 4. Then, $\{C_i^k\}_k$ is nonincreasing and admits a limit when $k \to \infty$.*

Proof. Since d^k is a descent direction, $JF(x^k)d^k < 0$ for all k. Then, the Armijo-type condition (10) shows that $F_i(x^{k+1}) \leq C_i^k$ for all i.

First, let us consider Algorithm 3. Because $m(k+1) \leq m(k) + 1$, we have

$$\begin{aligned} C_i^{k+1} &= \max_{0 \leq j \leq m(k+1)} \{F_i(x^{k+1-j})\} \\ &\leq \max_{0 \leq j \leq m(k)+1} \{F_i(x^{k+1-j})\} \\ &= \max\{C_i^k, F_i(x^{k+1})\} = C_i^k \end{aligned}$$

for all i and k, which implies that the sequence $\{C_i^k\}_k$ is nonincreasing.

Next, let us consider Algorithm 4. It follows from (12) that

$$C_i^{k+1} = \frac{\eta q_k}{q_{k+1}} C_i^k + \frac{1}{q_{k+1}} F_i(x^{k+1}) \leq C_i^k,$$

which also implies that $\{C_i^k\}_k$ is nonincreasing. Since F_i is bounded from below, and Lemmas 3.1 and 3.2 give $F_i(x^k) \leq C_i^k$ for all i and k, we can conclude that $\{C_i^k\}_k$ admits a limit when $k \to \infty$. □ □

The next two lemmas show that $\lim_{k\to\infty} \alpha_k |\theta_{\mathrm{sd}}(x^k)| = 0$ holds in each algorithm.

Lemma 4.2. *Let $\{x^k\}$ be the sequence generated by Algorithm 3. Suppose that Assumption 3.1 holds. Then, we have*

$$\lim_{k\to\infty} \alpha_k |\theta_{\mathrm{sd}}(x^k)| = 0.$$

Proof. For an index $i = 1, \ldots, m$ and an iteration k, let $\ell_i(k)$ be an integer such that

$$\begin{aligned} k - m(k) \leq \ell_i(k) \leq k, \\ C_i^k = F_i(x^{\ell_i(k)}). \end{aligned} \tag{14}$$

Fix an index i. From (10) for $k > M$, we obtain

$$\begin{aligned} C_i^k &= F_i(x^{\ell_i(k)}) \\ &= F_i(x^{\ell_i(k)-1} + \alpha_{\ell_i(k)-1} d^{\ell_i(k)-1}) \\ &\leq C_i^{\ell_i(k)-1} + \delta \alpha_{\ell_i(k)-1} \nabla F_i(x^{\ell_i(k)-1})^\top d^{\ell_i(k)-1}. \end{aligned}$$

From Lemma 4.1, we have $\lim_{k\to\infty}(C_i^k - C_i^{\ell_i(k)-1}) = 0$. Thus, since $\alpha_{\ell_i(k)-1} > 0$ and $\nabla F_i(x^{\ell_i(k)-1})^\top d^{\ell_i(k)-1} < 0$, the above inequality gives

$$\lim_{k\to\infty} \alpha_{\ell_i(k)-1} \nabla F_i(x^{\ell_i(k)-1})^\top d^{\ell_i(k)-1} = 0. \tag{15}$$

Moreover, from inequality (6) of Assumption 3.1, it follows that

$$\alpha_{\ell_i(k)-1} \nabla F_i(x^{\ell_i(k)-1})^\top d^{\ell_i(k)-1} \leq -\Gamma_1 \alpha_{\ell_i(k)-1} |\theta_{\mathrm{sd}}(x^{\ell_i(k)-1})| \leq 0 \quad \text{for all } k.$$

Since $\alpha_{\ell_i(k)-1} \leq \mu$ for all k and (15) holds, we can take the limit in the above inequalities and obtain

$$\lim_{k\to\infty} \alpha_{\ell_i(k)-1}|\theta_{\rm sd}(x^{\ell_i(k)-1})| = 0. \tag{16}$$

Now, let us prove that $\lim_{k\to\infty} \alpha_k |\theta_{\rm sd}(x^k)| = 0$. First, define

$$\hat{\ell}_i(k) := \ell_i(k + M + 2).$$

Note that $\{\hat{\ell}_i(k)\}_k \subseteq \{\ell_i(k)\}_k$. Let us show by induction that, for any given $j \geq 1$, we have

$$\lim_{k\to\infty} \alpha_{\hat{\ell}_i(k)-j}|\theta_{\rm sd}(x^{\hat{\ell}_i(k)-j})| = 0, \tag{17}$$

$$\lim_{k\to\infty} F_i(x^{\hat{\ell}_i(k)-j}) = \lim_{k\to\infty} C_i^k. \tag{18}$$

In the following we assume that $k \geq j - 1$ without loss of generality. For $j = 1$, it follows from (16) and the fact that $\{\hat{\ell}_i(k)\}_k \subseteq \{\ell_i(k)\}_k$ that (17) holds. Since $\|d^{\hat{\ell}_i(k)-1}\| \leq (\Gamma_2|\theta_{\rm sd}(x^{\hat{\ell}_i(k)-1})|)^{1/2}$ from Assumption 3.1, we obtain

$$\|x^{\hat{\ell}_i(k)} - x^{\hat{\ell}_i(k)-1}\| = \|\alpha_{\hat{\ell}_i(k)-1} d^{\hat{\ell}_i(k)-1}\| \leq \left(\Gamma_2 \alpha^2_{\hat{\ell}_i(k)-1}|\theta_{\rm sd}(x^{\hat{\ell}_i(k)-1})|\right)^{1/2}. \tag{19}$$

Since (17) holds for $j = 1$ and $\alpha_{\hat{\ell}_i(k)-1} \in (0, \mu]$, the above inequality implies that $\|x^{\hat{\ell}_i(k)} - x^{\hat{\ell}_i(k)-1}\| \to 0$. Thus, (18) also holds for $j = 1$ because F_i is continuous. Next, assume that (17) and (18) are satisfied for some $j = j'$. From the Armijo-type condition (10), we get

$$F_i(x^{\hat{\ell}_i(k)-j'}) \leq C_i^{\hat{\ell}_i(k)-j'-1} + \delta \alpha_{\hat{\ell}_i(k)-j'-1} \nabla F_i(x^{\hat{\ell}_i(k)-j'-1})^\top d^{\hat{\ell}_i(k)-j'-1}.$$

Therefore, since (18) holds and $\lim_{k\to\infty} \hat{\ell}_i(k) - j' - 1 = \infty$, we have

$$\lim_{k\to\infty} \alpha_{\hat{\ell}_i(k)-(j'+1)} \nabla F_i(x^{\hat{\ell}_i(k)-(j'+1)})^\top d^{\hat{\ell}_i(k)-(j'+1)} = 0.$$

It follows from (6) of Assumption 3.1 that

$$\lim_{k\to\infty} \alpha_{\hat{\ell}_i(k)-(j'+1)}|\theta_{\rm sd}(x^{\hat{\ell}_i(k)-(j'+1)})| = 0.$$

From (7), and using an inequality similar to (19), we have

$$\|x^{\hat{\ell}_i(k)-j'} - x^{\hat{\ell}_i(k)-(j'+1)}\| \to 0.$$

Moreover, from (18) and the continuity of F_i, we obtain

$$\lim_{k\to\infty} F_i(x^{\hat{\ell}_i(k)-(j'+1)}) = \lim_{k\to\infty} F_i(x^{\hat{\ell}_i(k)-j'}) = \lim_{k\to\infty} C_i^k.$$

Thus, we conclude that (17) and (18) hold for any given $j \geq 1$.

Now, for any k we have

$$x^{k+1} = x^{\hat{\ell}_i(k)} - \sum_{j=1}^{\hat{\ell}_i(k)-k-1} \alpha_{\hat{\ell}_i(k)-j} d^{\hat{\ell}_i(k)-j}. \tag{20}$$

Because $\hat{\ell}_i(k) - k - 1 = \ell_i(k+M+2) - k - 1 \leq M+1$ by (14), we get

$$\lim_{k\to\infty} \|x^{k+1} - x^{\hat{\ell}_i(k)}\| = 0$$

from (7), (17), and (20). Since $\{C_i^k\}_k$, namely $\{F_i(x^{\ell_i(k)})\}_k$, has a limit from Lemma 4.1, it follows from the continuity of F_i that

$$\lim_{k\to\infty} F_i(x^{k+1}) = \lim_{k\to\infty} F_i(x^{\hat{\ell}_i(k)}). \tag{21}$$

Moreover, from (10), we have

$$F_i(x^{k+1}) \leq C_i^k + \delta\alpha_k \nabla F_i(x^k)^\top d^k.$$

This inequality implies $\alpha_k \nabla F_i(x^k)^\top d^k \to 0$ because $\alpha_k \nabla F_i(x^k)^\top d^k < 0$ for all k and (21) holds. Therefore, we conclude from (6) that

$$\lim_{k\to\infty} \alpha_k |\theta_{\rm sd}(x^k)| = 0.$$

□ □

Lemma 4.3. *Let $\{x^k\}$ be the sequence generated by Algorithm 4. Suppose that Assumption 3.1 holds and assume also that $\eta < 1$. Then, we have*

$$\lim_{k\to\infty} \alpha_k |\theta_{\rm sd}(x^k)| = 0.$$

Proof. Recall that from Lemma 4.1, $\{C_i^k\}_k$ admits a limit for $k \to \infty$. From the definition of C^{k+1} in (12), we get

$$\begin{aligned} C_i^{k+1} &= \frac{\eta q_k}{q_{k+1}} C_i^k + \frac{1}{q_{k+1}} F_i(x^{k+1}) \\ &\leq \frac{\eta q_k}{q_{k+1}} C_i^k + \frac{1}{q_{k+1}} (C_i^k + \delta\alpha_k \nabla F_i(x^k)^\top d^k) \\ &= C_i^k + \frac{\delta\alpha_k}{q_{k+1}} \nabla F_i(x^k)^\top d^k, \end{aligned}$$

where the above inequality holds from (10) and the last equality follows from (11). Since $(\delta\alpha_k)/q_{k+1} \geq 0$ and $\nabla F_i(x^k)^\top d^k \leq 0$, we have

$$\lim_{k\to\infty} \frac{\alpha_k}{q_{k+1}} \nabla F_i(x^k)^\top d^k = 0. \tag{22}$$

Moreover, from (6), we obtain

$$\frac{\alpha_k}{q_{k+1}} \nabla F_i(x^k)^\top d^k \leq -\frac{\alpha_k}{q_{k+1}} \Gamma_1 |\theta_{\mathrm{sd}}(x^k)| \leq 0. \tag{23}$$

Now, observe that (11) gives

$$q_{k+1} = 1 + \sum_{j=0}^{k} \eta^{j+1} \leq \sum_{j=0}^{\infty} \eta^j = \frac{1}{1-\eta}.$$

Because $0 < \eta < 1$, we have that $\{1/q_{k+1}\}$ is bounded from below. So, we can take the limit in (23) and use (22) to show that

$$\lim_{k \to \infty} \alpha_k |\theta_{\mathrm{sd}}(x^k)| = 0,$$

and the conclusion follows. □ □

Using the results of Lemmas 4.2 and 4.3, we can establish global convergence of Algorithms 3 and 4. More precisely, we prove that every accumulation point of the sequences generated by these algorithms are Pareto critical.

Theorem 4.1. *Suppose that Assumption 3.1 holds. Every accumulation point of the sequence $\{x^k\}$ produced by Algorithm 3 is Pareto critical. Moreover, for Algorithm 4, the same statement follows if $\eta < 1$.*

Proof. We prove convergence of Algorithms 3 and 4 simultaneously. From Lemmas 4.2 and 4.3, we have

$$\lim_{k \to \infty} \alpha_k |\theta_{\mathrm{sd}}(x^k)| = 0. \tag{24}$$

Now let $\bar{x}$ be an accumulation point of $\{x^k\}$ and let $\{x^k\}_{k \in K}$ be a subsequence of $\{x^k\}$ converging to $\bar{x}$. Then, (24) implies that either $\lim_{k \to \infty} |\theta_{\mathrm{sd}}(x^k)| = 0$, or

$$\lim_{k \to \infty, k \in K} \alpha_k = 0. \tag{25}$$

In the former case, $\theta_{\mathrm{sd}}(\bar{x}) = 0$ by the continuity of θ_{sd} in Lemma 2.2, and so $\bar{x}$ is critical.

In the latter case, by (25) and the way to choose the stepsize α_k, there exists an index $\bar{k}$ such that $\alpha_k < \mu$ for all k satisfying $k \geq \bar{k}$ and $k \in K$, which implies that there exists $r_k \in \{1, \dots, m\}$ for each k satisfying $k \geq \bar{k}$ and $k \in K$ such that

$$\begin{aligned} F_{r_k}\left(x^k + \frac{\alpha_k}{\rho} d^k\right) &> C_{r_k}^k + \delta \frac{\alpha_k}{\rho} \nabla F_{r_k}(x^k)^\top d^k \\ &\geq F_{r_k}(x^k) + \delta \frac{\alpha_k}{\rho} \nabla F_{r_k}(x^k)^\top d^k, \end{aligned} \tag{26}$$

where the last inequality follows from Lemmas 3.1 and 3.2. Recall that $d^k \neq 0$ for all k because it is a descent direction. By the mean value theorem we have a point $u^k := x^k + (\omega_k/\|d^k\|)(\alpha_k/\rho)d^k$, with $\omega_k \in (0,1)$, such that

$$\nabla F_{r_k}(u^k)^\top d^k \geq \delta \nabla F_{r_k}(x^k)^\top d^k, \tag{27}$$

from (26). Since $\{d^k/\|d^k\|\}$ is bounded, there exists an index set $\bar{K} \subseteq K$ such that

$$\lim_{k\to\infty, k\in\bar{K}} x^k = \bar{x}, \qquad \lim_{k\to\infty, k\in\bar{K}} \frac{d^k}{\|d^k\|} = \bar{d}.$$

Let $K_r := \{k \in \bar{K} \mid r_k = r\}$. Then, there exists $\bar{r} \in \{1,\dots,m\}$ such that $K_{\bar{r}} \subseteq \bar{K}$ is an infinite set. It follows from (25) that

$$\lim_{k\to\infty, k\in K_{\bar{r}}} u^k = \lim_{k\to\infty, k\in K_{\bar{r}}} x^k + \lim_{k\to\infty, k\in K_{\bar{r}}} \left(\frac{\alpha_k\,\omega_k}{\rho}\right) \frac{d^k}{\|d^k\|} = \bar{x}.$$

Hence, dividing both sides of (27) by $\|d^k\|$ and taking the limit $k \to \infty$ with $k \in K_{\bar{r}}$, we obtain

$$(1-\delta)\nabla F_{\bar{r}}(\bar{x})^\top \bar{d} \geq 0.$$

Since $1-\delta > 0$ and $\nabla F_{\bar{r}}(x^k)^\top d^k < 0$ for all k, it yields $\nabla F_{\bar{r}}(\bar{x})^\top \bar{d} = 0$. Moreover, it follows from (6) of Assumption 3.1 that $\nabla F_{\bar{r}}(x^k)^\top d^k \leq -\Gamma_1|\theta_{\mathrm{sd}}(x^k)| \leq 0$ for all k. Dividing these inequalities by $\|d^k\|$ and once again taking the limit $k \to \infty$ with $k \in K_{\bar{r}}$, it yields

$$\lim_{k\to\infty, k\in K_{\bar{r}}} \frac{|\theta_{\mathrm{sd}}(x^k)|}{\|d^k\|} = 0.$$

Now, by using (7), we conclude that $\lim_{k\to\infty, k\in K_{\bar{r}}} \|d^k\| = 0$. This equality, together with the above equality and the continuity of θ_{sd}, yields $|\theta_{\mathrm{sd}}(\bar{x})| = 0$. Therefore, $\bar{x}$ is Pareto critical from Lemma 2.2. □ □

5 Nonmonotone steepest descent and Newton methods

Now, let us consider Algorithms 3 and 4, with either the steepest descent or the Newton direction as the search direction. As we already mentioned in Lemmas 2.2 and 2.3, both are descent directions if the corresponding iterate is nonstationary. So, here we prove that Assumption 3.1 required in the global convergence results of Section 4 holds automatically for multiobjective steepest descent and Newton directions.

Proposition 5.1. *For a given $x \in \mathbb{R}^n$, the steepest descent direction $d_{\mathrm{sd}}(x)$ satisfies the following two statements:*

$$\max_{i=1,\dots,m} \left\{\nabla F_i(x)^\top d_{\mathrm{sd}}(x)\right\} \leq -|\theta_{\mathrm{sd}}(x)|, \tag{28}$$

$$\|d_{\mathrm{sd}}(x)\|^2 = 2|\theta_{\mathrm{sd}}(x)|. \tag{29}$$

Proof. Define $f_x\colon \mathbb{R}^n \to \mathbb{R}$ as

$$f_x(d) := \max_{i=1,\dots,m} \left\{ \nabla F_i(x)^\top d \right\}.$$

From (2), and writing out the definitions of d_{sd} and θ_{sd}, we have

$$\begin{aligned} d_{\mathrm{sd}}(x) &= \operatorname*{argmin}_d \left(f_x(d) + \frac{1}{2}\|d\|^2 \right), \\ \theta_{\mathrm{sd}}(x) &= f_x(d_{\mathrm{sd}}(x)) + \frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2. \end{aligned} \tag{30}$$

Observe first that problem (2) is equivalent to

$$\begin{aligned} \min_{(d,\beta)} \quad & \beta + \frac{1}{2}\|d\|^2 \\ \text{s.t.} \quad & \nabla F_i(x)^\top d \le \beta \quad i = 1,\dots,m. \end{aligned}$$

So, the optimal solution of the above problem is given by $(d_{\mathrm{sd}}(x),\ f_x(d_{\mathrm{sd}}(x)))$. Moreover, there exists Lagrange multipliers $w_i(x) \in \mathbb{R}$, $i = 1,\dots,m$ such that

$$d_{\mathrm{sd}}(x) + \sum_{i=1}^m w_i(x) \nabla F_i(x) = 0, \tag{31}$$

$$\sum_{i=1}^m w_i(x) = 1, \tag{32}$$

$$w_i(x) \ge 0, \quad i = 1,\dots,m, \tag{33}$$

$$\nabla F_i(x)^\top d_{\mathrm{sd}}(x) - f_x(d_{\mathrm{sd}}(x)) \le 0, \quad i = 1,\dots,m,$$

$$w_i(x)\left(\nabla F_i(x)^\top d_{\mathrm{sd}}(x) - f_x(d_{\mathrm{sd}}(x)) \right) = 0, \quad i = 1,\dots,m.$$

From the complementarity condition, we have $\nabla F_i(x)^\top d_{\mathrm{sd}}(x) = f_x(d_{\mathrm{sd}}(x))$ for every index i such that $w_i(x) > 0$. Therefore, it follows from (30), (31), (32), and (33) that

$$\begin{aligned} \theta_{\mathrm{sd}}(x) &= \sum_{i=1}^m w_i(x) \nabla F_i(x)^\top d_{\mathrm{sd}}(x) + \frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2 \\ &= -d_{\mathrm{sd}}(x)^\top d_{\mathrm{sd}}(x) + \frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2 \\ &= -\frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2. \end{aligned}$$

Since $\theta_{\mathrm{sd}}(x) \le 0$ for all x, we conclude that (29) holds.

Moreover, $\theta_{\mathrm{sd}}(x) \le 0$ also gives

$$\frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2 \le -f_x(d_{\mathrm{sd}}(x)).$$

From this inequality and (29), it yields

$$|\theta_{\mathrm{sd}}(x)| \leq -f_x\left(d_{\mathrm{sd}}(x)\right),$$

which is precisely the inequality (28). □ □

Proposition 5.2. *Suppose that F_i is twice continuously differentiable and $\nabla^2 F_i(x)$ is positive definite for all $i = 1, \ldots, m$. Let $\lambda_{\max}$ and $\lambda_{\min}$ be, respectively, the maximum and the minimum eigenvalues of $\nabla^2 F_i(x)$ for all i. Then, the Newton direction $d_{\mathrm{N}}(x)$ satisfies the following two inequalities:*

$$\max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top d_{\mathrm{N}}(x)\right\} \leq -\frac{1}{\zeta}|\theta_{\mathrm{sd}}(x)|, \tag{34}$$

$$\|d_{\mathrm{N}}(x)\|^2 \leq \frac{2}{\xi^2}|\theta_{\mathrm{sd}}(x)|, \tag{35}$$

for any ζ and ξ such that $\zeta \geq \lambda_{\max}$ and $0 < \xi \leq \lambda_{\min}$.

Proof. From Lemmas 2.2 and 2.3, when x is Pareto critical, $d_{\mathrm{N}}(x) = 0$ and $\theta_{\mathrm{sd}}(x) = 0$, so (34) and (35) holds. Thus, we can assume that $d_{\mathrm{N}}(x) \neq 0$ and $d_{\mathrm{sd}}(x) \neq 0$. Since $\nabla^2 F_i(x)$ is positive definite, we have $d_{\mathrm{N}}(x)^\top \nabla^2 F_i(x) d_{\mathrm{N}}(x) > 0$ for all $i = 1, \ldots, m$. It follows that

$$\begin{aligned}
&\max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top d_{\mathrm{N}}(x)\right\} \\
\leq &\max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top d_{\mathrm{N}}(x) + \frac{1}{2} d_{\mathrm{N}}(x)^\top \nabla^2 F_i(x) d_{\mathrm{N}}(x)\right\} \\
= &\ \theta_{\mathrm{N}}(x),
\end{aligned} \tag{36}$$

where the equality follows from the definition of $\theta_{\mathrm{N}}(x)$ and (3). Also, we can see that

$$\begin{aligned}
\frac{1}{\zeta}\theta_{\mathrm{sd}}(x) &= \frac{1}{\zeta}\max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top d_{\mathrm{sd}}(x) + \frac{1}{2}\|d_{\mathrm{sd}}(x)\|^2\right\} \\
&= \max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top\left(\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right) + \frac{\zeta}{2}\left\|\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right\|^2\right\} \\
&\geq \max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top\left(\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right) + \frac{\lambda_{\max}}{2}\left\|\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right\|^2\right\},
\end{aligned}$$

where the inequality follows from the assumption that $\zeta \geq \lambda_{\max}$. Also, recalling that $0 < d^\top \nabla^2 F_i(x) d \leq \lambda_{\max}\|d\|^2$ for all $d \neq 0$, we obtain

$$\begin{aligned}
&\frac{1}{\zeta}\theta_{\mathrm{sd}}(x) \\
\geq &\max_{i=1,\ldots,m}\left\{\nabla F_i(x)^\top\left(\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right) + \frac{1}{2}\left(\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right)^\top \nabla^2 F_i(x)\left(\frac{1}{\zeta}d_{\mathrm{sd}}(x)\right)\right\} \\
\geq &\ \theta_{\mathrm{N}}(x).
\end{aligned}$$

The above inequality, together with (36) gives

$$\max_{i=1,\dots,m}\{\nabla F_i(x)^\top d_{\rm N}(x)\} \le \frac{1}{\zeta}\theta_{\rm sd}(x).$$

Recalling that $\theta_{\rm sd}(x) = -|\theta_{\rm sd}(x)|$ because $\theta_{\rm sd}(x) \le 0$ for all x, we conclude that (34) follows. Now, once again from the definition of $\theta_{\rm N}(x)$, we have

$$\begin{aligned}
\xi\theta_{\rm N}(x) &= \xi \max_{i=1,\dots,m}\left\{\nabla F_i(x)^\top d_{\rm N}(x) + \frac{1}{2} d_{\rm N}(x)^\top \nabla^2 F_i(x) d_{\rm N}(x)\right\} \\
&\ge \xi \max_{i=1,\dots,m}\left\{\nabla F_i(x)^\top d_{\rm N}(x) + \frac{\lambda_{\min}}{2}\|d_{\rm N}(x)\|^2\right\} \\
&= \max_{i=1,\dots,m}\left\{\nabla F_i(x)^\top (\xi d_{\rm N}(x)) + \frac{\lambda_{\min}}{2\xi}\|\xi d_{\rm N}(x)\|^2\right\},
\end{aligned}$$

where the inequality follows from the fact that $0 < \lambda_{\min}\|d\|^2 \le d^\top \nabla^2 F_i(x) d$ for all $d \ne 0$. Since $0 < \xi \le \lambda_{\min}$ by assumption,

$$\begin{aligned}
\xi\theta_{\rm N}(x) &\ge \max_{i=1,\dots,m}\left\{\nabla F_i(x)^\top (\xi d_{\rm N}(x)) + \frac{1}{2}\|\xi d_{\rm N}(x)\|^2\right\} \\
&\ge \theta_{\rm sd}(x).
\end{aligned}$$

Since $|\theta_{\rm N}(x)| = -\theta_{\rm N}(x)$ because $\theta_{\rm N}(x) \le 0$ for all x, and similarly $|\theta_{\rm sd}(x)| = -\theta_{\rm sd}(x)$, the above inequality is equivalent to

$$|\theta_{\rm sd}(x)| \ge \xi|\theta_{\rm N}(x)|. \tag{37}$$

From [4, Lemma 4.2], we also obtain

$$\frac{\xi}{2}\|d_{\rm N}(x)\|^2 \le |\theta_{\rm N}(x)|.$$

The above inequality, together with (37) gives (35), as we claimed. □ □

From Propositions 5.1 and 5.2, Assumption 3.1 is clearly satisfied. So, from Theorem 4.1, we can state our final theorem, which states the global convergence of the nonmonotone multiobjective steepest descent and Newton methods.

Theorem 5.1. *(a) Every accumulation point of the sequence* $\{x^k\}$ *produced by the nonmonotone multiobjective steepest descent method (i.e., Algorithms 3 or 4 with* $\eta < 1$ *and* $d^k = d_{\rm sd}(x^k)$ *for all* k*) is Pareto critical.*
(b) Suppose that F_i *is twice continuously differentiable and* $\nabla^2 F_i(x)$ *is positive definite for all* $i = 1, \dots, m$*. Then, every accumulation point of the sequence* $\{x^k\}$ *produced by the nonmonotone multiobjective Newton method (i.e., Algorithms 3 or 4 with* $\eta < 1$ *and* $d^k = d_{\rm N}(x^k)$ *for all* k*) is Pareto critical.*

6 New nonmonotone line searches for multiobjective optimization

In general line searches for multiobjective optimization (Algorithms 1 to 4), we choose stepsizes in order to satisfy the same Armijo-type conditions for all objective functions. In this section, we propose a new line search approach, which do not impose the same condition on all objective functions. One easy algorithm that chooses a stepsize satisfying Armijo condition for not all but some objective functions is given below. In Step 5, we use the notation $\#\mathcal{S}$ to denote the number of elements in the set $\mathcal{S}$.

Algorithm 5: New (ad hoc) nonmonotone line search for multiobjective optimization

1 Choose starting point $x^0 \in \mathbb{R}^n$ and parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\mu > 0$. Set $k = 0$.
2 Compute a search direction d^k at x^k that satisfies Assumption 3.1.
3 If x^k is a Pareto critical point, then stop.
4 Choose an integer m_k with $1 \leq m_k \leq m$.
5 Take $\alpha_k = \mu\rho^{h_k}$. Here, h_k is the largest integer satisfying

$$\#\{i \mid F_i\big(x^k + \alpha_k d^k\big) \leq F_i(x^k) + \delta\alpha_k \nabla F_i(x^k)^\top d^k\} \geq m_k.$$

6 Set $x^{k+1} = x^k + \alpha_k d^k$, $k = k+1$ and return to Step 2.

In Algorithm 5, the set of functions for which Armijo condition is applied may oscillate. For example, when $m = 2$ and $m_k = 1$, it is possible that the set is chosen as $\{F_1\}, \{F_2\}, \{F_1\}, \{F_2\}, \{F_1\}, \dots$ for each iteration. Thus, the sequence of points does not converge to a Pareto critical point. Therefore, in order to guarantee the convergence, we consider combining Algorithm 5 with the approaches in Section 3 as follows.

Algorithm 6: Hybrid-type nonmonotone line search for multiobjective optimization

1 Choose starting point $x^0 \in \mathbb{R}^n$ and parameters $\delta \in (0,1)$, $\rho \in (0,1)$, $\mu > 0$. Set $k = 0$.
2 Compute a search direction d^k at x^k that satisfies Assumption 3.1.
3 If x^k is a Pareto critical point, then stop.
4 Choose an integer m_k with $1 \leq m_k \leq m$.
5 Take $\alpha_k = \mu\rho^{h_k}$. Here, h_k is the largest integer satisfying

$$\#\{i \mid F_i\big(x^k + \alpha_k d^k\big) \leq F_i(x^k) + \delta\alpha_k \nabla F_i(x^k)^\top d^k\} \geq m_k$$

and

$$F_i\big(x^k + \alpha_k d^k\big) \leq C_i^k + \delta\alpha_k \nabla F_i(x^k)^\top d^k, \quad i = 1, \dots, m.$$

6 Set $x^{k+1} = x^k + \alpha_k d^k$, $k = k+1$ and return to Step 2.

In Step 5, we set C_i^k using either (9) or (12) with $\eta < 1$. We call this method hybrid-type because it can be seen as a combination of monotone and nonmonotone techniques. Because of the addition of the second condition of Step 5, the result below clearly follows from Theorem 4.1.

Theorem 6.1. *Suppose that Assumption 3.1 holds. Every accumulation point of the sequence $\{x^k\}$ produced by Algorithm 6 (with C_i^k updated with either* (9) *or* (12) *with $\eta < 1$) is Pareto critical.*

7 Numerical experiments

In this section, we present some numerical results to show the validity of the proposed nonmonotone techniques for multiobjective optimization, and to verify if the nonmonotone approach is also valuable in the multiobjective case. The experiments were done using MATLAB R2016b on a computer with CPU Intel Core i7 3.6GHz and 8GB of memory. For the tests, we considered the multiobjective problems used in [4]. In order to have a more reasonable quantity of tests, we also added extensions of single-objective optimization problems from [14, 18] to the multiobjective setting. This gives 16 test problems in total, and for the sake of completeness, we list them in Appendix A.

As we can see in Appendix A, we solved problems under box constraints $L \leq x \leq U$. As it was done in [4, Section 8], we can handle these boxes considering them in the constraints of the subproblems associated to the directions. More precisely, the subproblem (2) is replaced with the following quadratic problem:

$$\begin{aligned} \min_{(d,\beta)} \quad & \beta + \frac{1}{2}\|d\|^2 \\ \text{s.t.} \quad & \nabla F_i(x)^\top d \leq \beta, \quad i = 1, \dots, m, \\ & \frac{L - x}{\mu} \leq d \leq \frac{U - x}{\mu}. \end{aligned}$$

Newton directions can be computed in the same way. We used `quadprog`, MATLAB Optimization Toolbox, as a solver for steepest descent direction search problems, and `fmincon` as a solver for Newton direction search problems. In the case of the Newton direction, if

$$\max_{i=1,\dots,m} \left\{ \nabla F_i(x^k)^\top d_{\mathrm{N}}(x^k) \right\} \geq \varsigma \|d_{\mathrm{N}}(x^k)\|^2$$

with $\varsigma = \frac{1}{100}$ holds in a iteration k, then we take $d_{\mathrm{sd}}(x^k)$ instead of $d_{\mathrm{N}}(x^k)$ in order to get convergence of nonconvex problems.

We solve all test problems of Appendix A using the steepest descent and Newton methods, each one with the following four line searches:

(i) The usual monotone line search;

(ii) The max-type nonmonotone line search (Algorithm 3) with $M = 4$;

(iii) The average-type nonmonotone line search (Algorithm 4) with $\eta = 0.85$;

(iv) The hybrid-type nonmonotone line search
($k < 30$: Algorithm 5 with $m_k = \lceil m/2 \rceil$,
$k \geq 30$: Algorithm 6 using (9) with $M = 29$ and $m_k = \lceil m/2 \rceil$).

The values $M = 4$ and $\eta = 0.85$, that controls the level of nonmonotonicity, were chosen based on the single-objective case [7, 19]. It can be seen in [12, 13] that these values actually minimize the necessary time to solve the set of problems of Appendix A. Concerning other parameters, we use $\delta = 10^{-4}$, $\mu = 1$, and $\rho = 0.5$. For the termination criterion (Step 3 of Algorithms 3 to 6), we use $|\theta_{\mathrm{sd}}(x)| < \varepsilon$ or $|\theta_{\mathrm{N}}(x)| < \varepsilon$ with $\varepsilon = 10^{-6}$ as usual, but we finished solving some problems when $\varepsilon = 10^{-4}$ when the problem is difficult enough (problems ZDT1 and Rosenbrock from Appendix A).

We compare the methods with the number of objective function evaluations. The results are similar if instead we consider the number of iterations. The results are shown using performance profiles [2], which we briefly explain here for completeness. Let S be the set of solvers, P be the set of problems, and $t_{p,s}$ be the performance, in this case the number of function evaluations, of the solver $s \in S$ on the problem $p \in P$. Then, performance profiles are depicted by plotting the following cumulative distribution functions $t \mapsto R_s$ of the performance ratio $r_{p,s}$ for different methods:

$$
\begin{aligned}
r_{p,s} &:= \frac{t_{p,s}}{\min\{t_{p,s} \mid s \in S\}}, \\
R_s(t) &:= \frac{1}{|P|} \left| p \in P \mid r_{p,s} \leq t \right|.
\end{aligned}
$$

Figure 1 compares the four line searches (i) to (iv) stated above, with the steepest descent direction. Since the hybrid-type outperforms the other line searches, we also plot Figure 2, comparing the three line searches (i) to (iii), so the difference between them becomes more clear. In this case, we can observe that nonmonotone line searches are faster than the monotone version, and the hybrid-type nonmonotone line search is the most efficient. The number of problems that the hybrid-type does not perform well is just one, which is the problem Shifted TRIDIA.

For the case of the Newton direction, since the test problems were in general small-sized, all algorithms finished in few iterations. Even so, we can conclude from Figures 3 and 4 that the hybrid-type is more efficient than the monotone line search, and that the max-type and the average-type are as efficient as the monotone one.

Until now, we are just concerned with the speed of convergence of the methods. However, since we are dealing with multiobjective optimization, it is interesting to see if the methods are able to generate Pareto frontiers properly. For this, we considered two examples from Appendix A, and for each problem, we generated 100 starting points randomly. We chose two nonconvex problems, called DD1 and KW2, plotting the obtained Pareto critical points given by the steepest descent direction with line searches (i), (iii), and (iv). The case (ii) is omitted because it was similar to (iii). In Figure 5, for problem DD1, the Pareto frontiers of the average-type and the hybrid-type are more proper than the monotone one. However, in Figure 6, for problem KW2, the Pareto frontier of the hybrid-type does not seem good enough.

Finally, we want to check if the stepsizes of the nonmonotone line searches tend in fact to be larger, and if this is the reason for the faster convergence. For this experiment, we

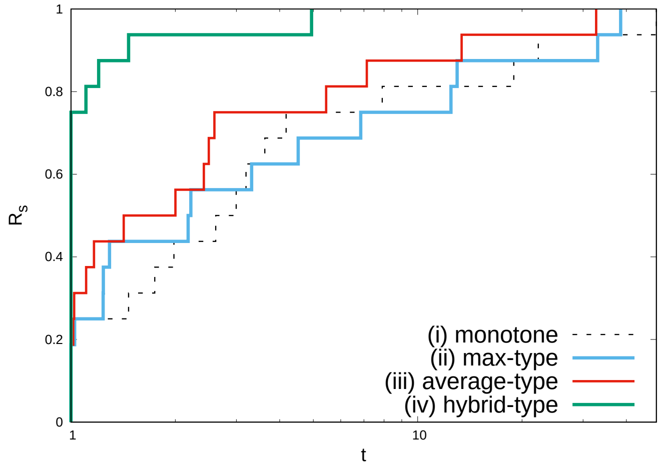


Figure 1: Performance profile with the steepest descent direction using line searches (i)–(iv)

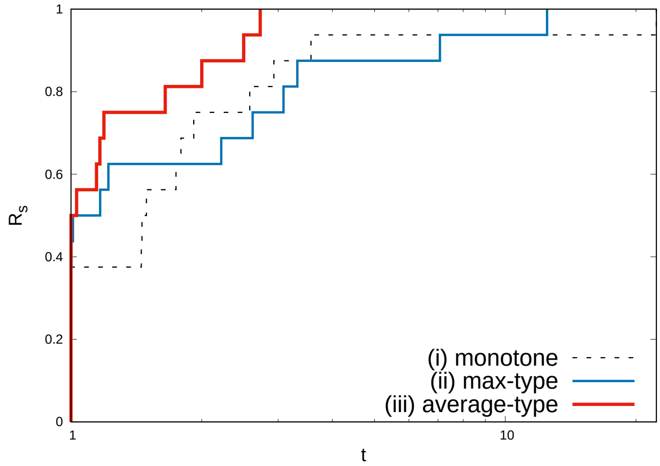


Figure 2: Performance profile with the steepest descent direction using line searches (i)–(iii)

took problems from Appendix A, where the number of objective functions m is variable. We also considered the steepest descent direction and line searches (i), (iii) and (iv). We omitted (ii), because we already know from Figure 1 that the average-type performs better than the max-type, and both are pure nonmonotone techniques.

Table 1 shows the average of all the stepsizes, defined as $\overline{\alpha}$. Moreover, "iter" means average number of iterations, and "eval" means average number of objective functions evaluations. From this table, we can see for all the three problems that, as m increases, $\overline{\alpha}$ of the monotone version becomes smaller. For problem 15, $\overline{\alpha}$ of the average-type also be-

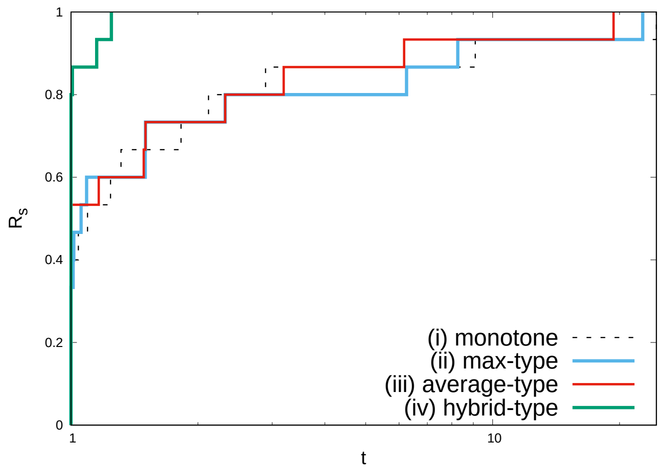


Figure 3: Performance profile with the Newton direction using line searches (i)–(iv)

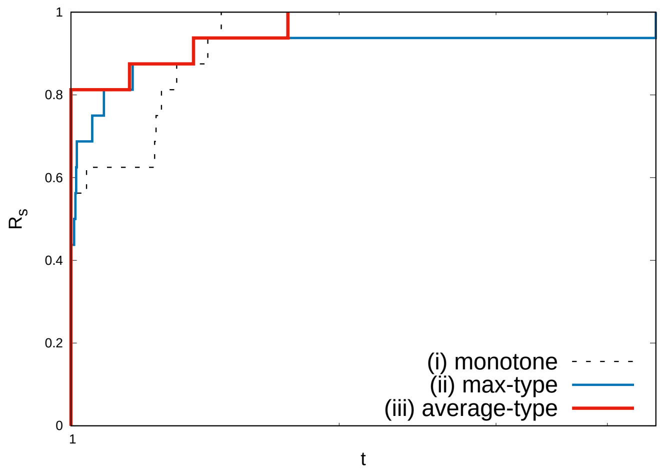
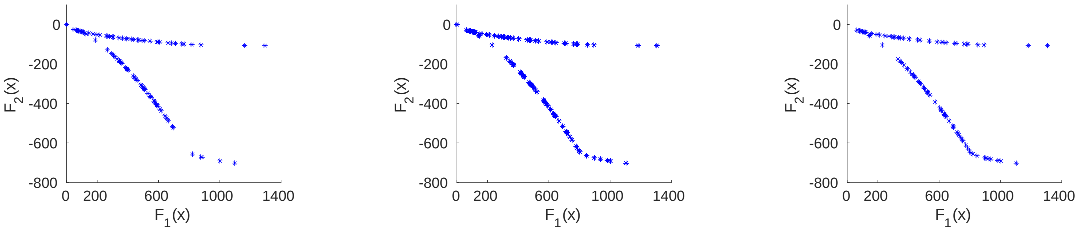


Figure 4: Performance profile with the Newton direction using line searches (i)–(iii)

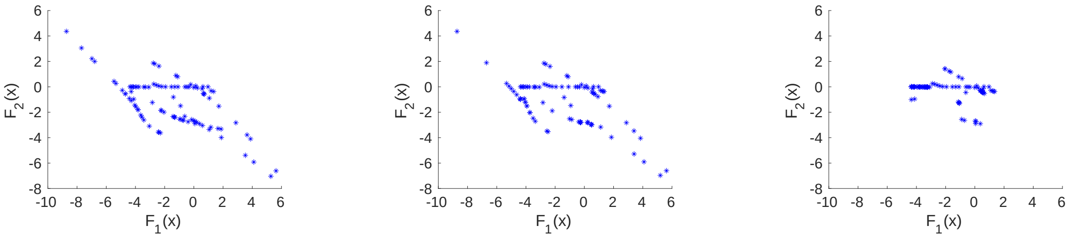
comes smaller as m increases. On the other hand, when we used the hybrid-type one, $\overline{\alpha}$ hardly changed. Furthermore, for nonconvex problems 14 and 15, we can observe that the nonmonotone versions are more efficient, taking larger stepsizes in average.

From all the above experiments, we conclude that the nonmonotone line searches tend to be more efficient, in the sense that they converge faster. In particular, the hybrid-type has not only theoretical guarantee of convergence, but seems to perform better, at least in our test problems. However, depending on the nature of the multiobjective optimization problem, it does not necessarily guarantee good enough Pareto frontiers. As in the classical



(i) monotone (iii) average-type (iv) hybrid-type

Figure 5: Pareto critical points obtained for DD1 with the steepest descent method



(i) monotone (iii) average-type (iv) hybrid-type

Figure 6: Pareto critical points obtained for KW2 with the steepest descent method

Table 1: Relation between the stepsize and the number of functions evaluations/iterations

	m	algorithm	$\overline{\alpha}$	iter	eval
14. Brown and Dennis	5	(i) monotone	0.6999	64.91	141.19
		(iii) average-type	0.7904	56.73	87.71
		(iv) hybrid-type	0.9253	28.31	33.83
	7	(i) monotone	0.5638	129.99	408.29
		(iii) average-type	0.8788	67.49	88.62
		(iv) hybrid-type	0.9552	46.64	51.99
15. Trigonometric	4	(i) monotone	0.7555	17.3	26.31
		(iii) average-type	0.9637	13.73	14.69
		(iv) hybrid-type	0.9881	12.99	13.28
	6	(i) monotone	0.3978	25.64	64.5
		(iii) average-type	0.7515	15.02	22.8
		(iv) hybrid-type	0.9835	10.4	10.79
16. Linear function	5	(i) monotone	0.2147	2.58	6.93
		(iii) average-type	0.2017	5.6	17.31
		(iv) hybrid-type	0.2230	3.05	8.33
	7	(i) monotone	0.2140	2.52	6.65
		(iii) average-type	0.2116	4.82	14.27
		(iv) hybrid-type	0.2248	3.08	8.48

single-objective optimization, this means that the nonmonotone approach is not a substitute for the standard monotone line search, but only an alternative.

8 Conclusion

In this work, we confirmed, with numerical experiments, the efficiency of the nonmonotone line searches for multiobjective steepest descent and Newton methods. From the theoretical point of view, we obtained the global convergence for Pareto critical points, under the same assumptions required for the monotone version of the multiobjective descent methods. We considered a general framework for the proof, by requiring only that we have descent directions that satisfy Assumption 3.1. We then proved that both steepest descent and Newton directions satisfy such an assumption. A future will be to propose novel nonmonotone techniques, to be able to find good Pareto frontiers for problems with many complex objective functions.

Acknowledgement. We would like to thank the anonymous referees for their suggestions, which improved the original version of the paper.

Appendix A

Here, we list the test problems used in Section 7. For each problem, we state the original reference, the number of variables n, the number of objective functions m, the convexity property, the objective functions, and the bounds L and U of the box constraints.

1. **Das and Dennis (DD1)** [1]: $n = 5$, $m = 2$, nonconvex,[1][2]

$$F_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2,$$
$$F_2(x) = 3x_1 + 2x_2 - \frac{x_3}{3} + 0.01(x_4 - x_5)^3,$$

   $L = (-20, \ldots, -20)^\top$, and $U = (20, \ldots, 20)^\top$.

2. **Fliege, Graña Drummond and Svaiter (FDS)** [4]: $n = 10$, $m = 3$, convex,[2]

$$F_1(x) = \frac{1}{n^2} \sum_{i=1}^{n} i(x_i - i)^4,$$
$$F_2(x) = \exp\left(\sum_{i=1}^{n} \frac{x_i}{n}\right) + \|x\|_2^2,$$
$$F_3(x) = \frac{1}{n(n+1)} \sum_{i=1}^{n} i(n - i + 1)e^{-x_i},$$

[1] This is actually a modified version of [1] that can be found in [4].

[2] In the original version, either the bounds L and U, or the variable n can be modified.

$L = (-2, \ldots, -2)^\top$, and $U = (2, \ldots, 2)^\top$.

3. **Jin, Olhofer and Sendhoff (JOS1)** [8]: $n = 5$, $m = 2$, quadratic convex,[2]

$$F_1(x) = \frac{1}{n} \sum_{i=1}^{n} x_i^2,$$
$$F_2(x) = \frac{1}{n} \sum_{i=1}^{n} (x_i - 2)^2,$$

$L = (-2, \ldots, -2)^\top$, and $U = (2, \ldots, 2)^\top$.

4. **Kim and Weck (KW2)** [9]: $n = 2$, $m = 2$, nonconvex,

$$\begin{aligned} F_1(x) = & -3(1-x_1)^2 \exp(-x_1^2 - (x_2+1)^2) \\ & + 10\left(\frac{x_1}{5} - x_1^3 - x_2^5\right) \exp(-x_1^2 - x_2^2) \\ & + 3\exp(-(x_1+2)^2 - x_2^2) - 0.5(2x_1 + x_2), \\ F_2(x) = & -3(1+x_2)^2 \exp(-x_2^2 - (1-x_1)^2) \\ & + 10\left(-\frac{x_2}{5} + x_2^3 + x_1^5\right) \exp(-x_1^2 - x_2^2) \\ & + 3\exp(-(2-x_2)^2 - x_1^2), \end{aligned}$$

$L = (-3, -3)^\top$, and $U = (3, 3)^\top$.

5. **Stadler and J. Dauer (SD)** [17]: $n = 4$, $m = 2$, convex,

$$F_1(x) = 2x_1 + \sqrt{2}x_2 + \sqrt{2}x_3 + x_4,$$
$$F_2(x) = \frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} + \frac{2\sqrt{2}}{x_3} + \frac{2}{x_4},$$

$L = \left(1, \sqrt{2}, \sqrt{2}, 1\right)^\top$, and $U = (3, 3, 3, 3)^\top$.

6. **Zitzler, Deb and Thiele (ZDT1)** [20]: $n = 30$, $m = 2$, convex,[2]

$$F_1(x) = x_1,$$
$$F_2(x) = g(x)\left(1 - \sqrt{\frac{x_1}{g(x)}}\right),$$

with $g(x) = 1 + 9\sum_{i=2}^{n} x_i/(n-1)$, $L = (0, \ldots, 0)^\top$, and $U = \left(\frac{1}{100}, \ldots, \frac{1}{100}\right)^\top$.

7. **Zitzler, Deb and Thiele (ZDT4)** [20]: $n = 10$, $m = 2$, nonconvex,[2]

$$F_1(x) = x_1,$$
$$F_2(x) = g(x)\left(1 - \sqrt{\frac{x_1}{g(x)}}\right),$$

with $g(x) = 1 + 10(n-1) + \sum_{i=2}^{n} \left(x_i^2 - 10\cos(4\pi x_i)\right)$,
$L = \left(\frac{1}{100}, -5, \ldots, -5\right)^\top$, and $U = (1, 5, \ldots, 5)^\top$.

8. **Toint (TOI4)** [18, Problem 4]: $n = 4$, $m = 2$, convex,[3]

$$\begin{aligned} F_1(x) &= x_1^2 + x_2^2 + 1, \\ F_2(x) &= 0.5\left((x_1 - x_2)^2 + (x_3 - x_4)^2\right) + 1, \end{aligned}$$

$L = (-2, \ldots, -2)^\top$, and $U = (5, \ldots, 5)^\top$.

9. **TRIDIA** [18, Problem 8]: $n = 3$, $m = 3$, convex,[23]

$$\begin{aligned} F_1(x) &= (2x_1 - 1)^2, \\ F_2(x) &= 2(2x_1 - x_2)^2, \\ F_3(x) &= 3(2x_2 - x_3)^2, \end{aligned}$$

$L = (-1, -1, -1)^\top$, and $U = (1, 1, 1)^\top$.

10. **Shifted TRIDIA** [18, Problem 9]: $n = 4$, $m = 4$, nonconvex,[23]

$$\begin{aligned} F_1(x) &= (2x_1 - 1)^2 + x_2^2, \\ F_i(x) &= i(2x_{i-1} - x_i)^2 - (i-1)x_{i-1}^2 + ix_i^2 \quad i = 2, 3, \\ F_4(x) &= 4(2x_3 - x_4)^2 - 3x_3^2, \end{aligned}$$

$L = (-1, \ldots, -1)^\top$, and $U = (1, \ldots, 1)^\top$.

11. **Rosenbrock** [18, Problem 10]: $n = 4$, $m = 3$, nonconvex,[23]

$$F_i(x) = 100(x_{i+1} - x_i^2)^2 + (x_{i+1} - 1)^2, \quad i = 1, 2, 3,$$

$L = (-2, \ldots, -2)^\top$, and $U = (2, \ldots, 2)^\top$.

12. **Helical valley** [14, Problem (7)]: $n = 3$, $m = 3$, nonconvex,[3]

$$\begin{aligned} F_1(x) &= \begin{cases} \left[10\left(x_3 - \dfrac{5}{\pi}\arctan\left(\dfrac{x_2}{x_1}\right)\right)\right]^2, & \text{if } x_1 > 0, \\ \left[10\left(x_3 - \dfrac{5}{\pi}\arctan\left(\dfrac{x_2}{x_1}\right) - 5\right)\right]^2, & \text{if } x_1 < 0 \end{cases} \\ F_2(x) &= \left(10\left((x_1^2 + x_2^2)^{1/2} - 1\right)\right)^2, \\ F_3(x) &= x_3^2, \end{aligned}$$

$L = (-2, -2, -2)^\top$, and $U = (2, 2, 2)^\top$.

[3]It is an adaptation of a single-objective optimization problem to the multiobjective setting. Since the problem is originally unconstrained, we also added some bound constraints.

13. **Gaussian** [14, Problem (9)]: $n = 3$, $m = 15$, nonconvex,[3]

$$F_i(x) = x_1 \exp\left(\frac{-x_2(t_i - x_3)^2}{2}\right) - y_i,$$

where $t_i = (8 - i)/2$, $i = 1, \ldots, m$ and y_i is given as

i	1,15	2,14	3,13	4,12	5,11	6,10	7,9	8
y_i	0.0009	0.0044	0.0175	0.0540	0.1295	0.2420	0.3521	0.3989

$L = (-2, -2, -2)^\top$, and $U = (2, -2, 2)^\top$.

14. **Brown and Dennis** [14, Problem (16)]: $n = 4$, $m = 5$, nonconvex,[3]

$$F_i(x) = \left(x_1 + t_i x_2 - e^{t_i}\right)^2 + \left(x_3 + x_4 \sin(t_i) - \cos(t_i)\right)^2,$$

where $t_i = i/5$, $L = (-25, -5, -5, -1)^\top$, and $U = (25, 5, 5, 1)^\top$.

15. **Trigonometric** [14, Problem (26)]: $n = 4$, $m = 4$, nonconvex,[2,3]

$$F_i(x) = \left(n - \sum_{j=1}^{n} \cos x_j + i\,(1 - \cos x_i) - \sin x_i\right)^2, \quad i = 1, \ldots, 4,$$

$L = (-1, \ldots, -1)^\top$, and $U = (1, \ldots, 1)^\top$.

16. **Linear function – rank 1** [14, Problem (33)]: $n = 10$, $m = 4$, convex,[2,3]

$$F_i(x) = \left(i\left(\sum_{j=1}^{n} j x_j\right) - 1\right)^2, \quad i = 1, \ldots, 4,$$

$L = (-1, \ldots, -1)^\top$, and $U = (1, \ldots, 1)^\top$.

References

[1] I. Das and J. E. Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.*, 8(3):631–657, 1998.

[2] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Math. Program.*, 91(2):201–213, 2002.

[3] N. Fazzio and M. L. Schuverdt. Convergence analysis of a nonmonotone projected gradient method for multiobjective optimization problems. *Optim. Lett.*, 2018. To appear.

[4] J. Fliege, L. M. Graña Drummond, and B. F. Svaiter. Newton's method for multiobjective optimization. *SIAM J. Optim.*, 20(2):602–626, 2009.

[5] J. Fliege and B. F. Svaiter. Steepest descent methods for multicriteria optimization. *Math. Methods Oper. Res.*, 51(3):479–494, 2000.

[6] E. H. Fukuda and L. M. Graña Drummond. A survey on multiobjective descent methods. *Pesq. Oper.*, 34(3):585–620, 2014.

[7] L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for Newton's method. *SIAM J. Numer. Anal.*, 23(4):707–716, 1986.

[8] Y. Jin, M. Olhofer, and B. Sendhoff. Dynamic weighted aggregation for evolutionary multi-objective optimization: Why does it work and how? In *GECCO'01 Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, pages 1042–1049, 2001.

[9] I. Y. Kim and O. L. de Weck. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Struct. Multidiscipl. Optim.*, 29(2):149–158, 2005.

[10] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolut. Comput.*, 10(3):263–282, 2002.

[11] D. T. Luc. Scalarization of vector optimization problems. *J. Optim. Theory Appl.*, 55(1):85–102, 1987.

[12] K. Mita. Nonmonotone line search in multiobjective settings (in Japanese), March 2017. Undergraduate research, Kyoto University.

[13] K. Mita, E. H. Fukuda, and N. Yamashita. On using nonmonotone line search techniques in steepest descent methods for multiobjective optimization (in Japanese). In *Proceedings of the 61st Annual Conference of the Institute of Systems, Control and Information Engineers*, May 2017.

[14] J. J. Moré, B. S. Garbow, and K. E. Hillstrom. Testing unconstrained optimization software. 7(1):17–41, 1981.

[15] Y. Ogata, Y. Saito, T. Tanaka, and S. Yamada. Sublinear scalarization methods for sets with respect to set-relations. *Linear Nonlinear Anal.*, 3(1):121–132, 2017.

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

[17] W. Stadler and J. Dauer. Multicriteria optimization in engineering: a tutorial and survey. In M. P. Kamat, editor, *Progress in Aeronautics and Astronautics: Structural Optimization: Status and Promise*, volume 150, pages 209–249. American Institute of Aeronautics and Astronautics, 1992.

[18] Ph. L. Toint. Test problems for partially separable optimization and results for the routine PSPMIN, 1983.

[19] H. Zhang and W. W. Hager. A nonmonotone line search technique and its application to unconstrained optimization. *SIAM J. Optim.*, 14(4):1043–1056, 2004.

[20] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: empirical results. *Evolut. Comput.*, 8(2):173–195, 2000.