

CONVERGENCE OF A NONMONOTONE PROJECTED GRADIENT METHOD FOR NONCONVEX MULTIOBJECTIVE OPTIMIZATION

XIAOPENG ZHAO¹, LATEEF O. JOLAOSO², YEKINI SHEHU³, JEN-CHIH YAO^{4,*}

¹*School of Mathematical Sciences, Tiangong University, Tianjin 300387, China*

²*Department of Mathematics and Applied Mathematics,
Sefako Makgatho Health Sciences University, P.O. Box 94 Medunsa 0204, Pretoria, South Africa*

³*Department of Mathematics, Zhejiang Normal University, Jinhua, China*

⁴*Research Center for Interneural Computing,
China Medical University Hospital, China Medical University, Taichung 40402, Taiwan*

Abstract. We consider a nonmonotone projected gradient method for nonconvex multiobjective optimization problem. Under some mild assumptions, we prove that each accumulation point of the sequence generated by the algorithm, if exists, is Pareto stationary. Furthermore, when the nonmonotone line search in the algorithm is replaced by the monotone one, i.e., the classical Armijo search rule, the full convergence of the generated sequence to a Pareto stationary point (respectively, a weak Pareto optimal point) is established when the multiobjective function is quasiconvex (respectively, pseudoconvex). Numerical experiments are also presented to illustrate the effectiveness of this method.

Keywords. Multiobjective optimization; Projected gradient method; Nonmonotone line search; Pareto optimality.

1. INTRODUCTION

In multiobjective optimization, also known as multicriteria optimization, several objective functions have to be optimized simultaneously. For nontrivial problems, usually no single point will optimize all the given objective functions currently, so the concept of optimality is replaced by the concept of *Pareto optimality or efficiency*. The multiobjective optimization problem has wide applications in many areas, including engineering [1], economy [2], management science [3, 4, 5], environmental analysis [6], design [7], location science [8] and so on.

One popular strategy for solving multiobjective optimization problems is the scalarization method [9, 10], which converts the original vector-valued problems into the parameterized scalar-valued ones. The most widely used scalarization technique is the weighting methods. That is, one minimizes a linear combination of the objectives with the vector of “weights” which is not known in advance. The disadvantage of this procedure, just as shown in [11, 12], is that it may lead to unbounded numerical problems, which therefore may lack minimizers.

*Corresponding author.

E-mail addresses: zhaoxiaopeng.2007@163.com (X.P. Zhao), lateef.jolaoso@smu.ac.za (L.O. Jolaoso), yekini.shehu@unn.edu.ng (Y. Shehu), yaojc@mail.cmu.edu.tw (J.C. Yao).

Received January 11, 2021; Accepted April 6, 2021.

Another main solution strategy for multiobjective optimization is the descent method, which does not require any priori parameter information and has been studied extensively in recent years, see e.g., the projected gradient method [13, 14], the steepest descent method [15], the Newton method [16] and the conjugate gradient method [17]. In these iterative methods, the step length is determined by a monotone line search so that the objective function values decrease at each iteration.

However, it is well-known in the scalar optimization that the enforcing monotonicity will slow the convergence speed of the method, especially in the case when the scheme is forced to creep along the bottom of a narrow curved valley (a common occurrence in difficult nonlinear problems) [18]. In order to avoid this drawback, one can use the nonmonotone line search technique. As pointed out by some researchers [19, 20] for the scalar optimization, nonmonotone schemes can improve the likelihood of finding a global optimum. During this line search, some increase in the function values in some iterations is permitted and so the convergence speed can be improved. Such nonmonotone techniques are verified numerically in the literature, see, e.g., [18, 19, 21, 22, 23]. Particularly, Grippo et al. [18] proposed a nonmonotone line search which requires that a maximum of recent function values decreases. In [22], Zhang and Hager introduced a new nonmonotone line search technique requiring that an average of the successive function values decreases, and this nonmonotone search technique has been shown to be more efficient than either the monotone or the traditional nonmonotone schemes in terms of taking fewer function and gradient evaluations on average.

In contrast to the rich research on the nonmonotone line search techniques for solving scalar optimization problems, the research about nonmonotone line search-based algorithms for solving multiobjective optimization problems is still insufficient. To our best knowledge, Mita et al. [24], respectively, Mahdavi-Amiri et al. [25] considered the nonmonotone multiobjective steepest descent and Newton's methods, respectively, the quasi-Newton method. Qu et al. [26] proposed two nonmonotone gradient methods for convex vector optimization based on the nonmonotone line search given by Grippo et al. [18] in the scalar context. In [27], Fazzio and Schuverdt [27] considered a projected gradient method based on the nonmonotone line search technique of [22].

Recently, in [28], we developed the projected gradient method defined by Graña Drummond and Iusem [14] (with exogenously given square summable step lengths) to include the nonmonotone line search given by Zhang and Hager [22]. By adopting different techniques, we proved that any sequence generated by the proposed method converges to a weak Pareto optimal point when the multiobjective function is convex. Moreover, we established the linear convergence result under some reasonable Lipschitz continuity assumptions.

In this paper, we continue to consider the nonmonotone gradient method proposed in [28] but for the nonconvex multiobjective optimization. Under standard assumptions, we prove that every accumulation point of the sequence generated by this algorithm is Pareto stationary. Further, we consider the special case when the nonmonotone line search rule in the algorithm is replaced by the monotone one, i.e., the Armijo line search. In this situation, we show that the entire generated sequence will converge to the stationary point when the multiobjective function is quasiconvex. Moreover, when the multiobjective function is pseudoconvex, the convergence of the generated sequence to a weak Pareto optimal point is obtained.

The outline of this paper is as follows. In Section 2, we present some definitions and preliminary results. The proposed algorithm and its properties are described in Section 3. Section 4 contains the convergence analysis of the algorithm. In Section 5, we present some numerical results to show the validity of the algorithm. Finally, conclusions and lines for future research are given in Section 6.

2. PRELIMINARIES

The following notations will be used throughout this paper. As usual, we denote by $\langle \cdot, \cdot \rangle$ the inner product of \mathbb{R}^n , and by $\|\cdot\|$ the corresponding norm. The transpose sign is denoted by \top . Let \mathbb{R}_+^m and \mathbb{R}_{++}^m denote the nonnegative orthant and positive orthant of \mathbb{R}^m , respectively. We may consider the partial order \preceq (\prec) induced by \mathbb{R}_+^m (\mathbb{R}_{++}^m): for any $x, y \in \mathbb{R}^m$, $x \preceq y$ ($x \prec y$) means $y - x \in \mathbb{R}_+^m$ ($y - x \in \mathbb{R}_{++}^m$).

For a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, we say that f is differentiable at $\bar{x} \in \mathbb{R}^n$ if there exists a vector $v \in \mathbb{R}^n$ such that

$$f(x) = f(\bar{x}) + \langle v, x - \bar{x} \rangle + o(\|x - \bar{x}\|),$$

where $\lim_{x \rightarrow \bar{x}} \frac{o(\|x - \bar{x}\|)}{\|x - \bar{x}\|} = 0$. Such a vector v is called the gradient of f at \bar{x} , and is denoted by $\nabla f(\bar{x})$.

In what follows, let $C \subseteq \mathbb{R}^n$ be a closed and convex set, and let $F : C \rightarrow \mathbb{R}^m$ be a vector-valued function denoted by

$$F := (f_1, \dots, f_m)^\top,$$

with each $f_i : C \rightarrow \mathbb{R}$ being a continuously differentiable function. In the present paper, we are interested in the following multiobjective optimization problem:

$$\min_{x \in C} F(x). \quad (2.1)$$

Recall that a point $x^* \in C$ is said to be a Pareto optimal point (or Pareto efficient solution) of F on C if there does not exist $x \in C$ such that $F(x) \preceq F(x^*)$ and $F(x) \neq F(x^*)$. Likewise, a point $x^* \in C$ is said to be a weak Pareto optimal point (or weak Pareto efficient solution) of F on C if there does not exist $x \in C$ such that $F(x) \prec F(x^*)$. It is well-known that every Pareto optimal point is also a weak Pareto optimal point, but the converse is not always true.

We say that a point $x^* \in C$ is a Pareto stationary point (or a Pareto critical point) of F if

$$J_F(x^*)(C - x^*) \cap (-\mathbb{R}_{++}^m) = \emptyset, \quad (2.2)$$

where $J_F(x^*)$ is the Jacobian matrix of F at x^* given by $J_F(x) = (\nabla f_1(x), \dots, \nabla f_m(x))^\top$. This notion is necessary (but in general not sufficient) for a point to be weak Pareto efficient and was first used in [15] to investigate a steepest descent algorithm. Note that in the case when $m = 1$, (2.2) is reduced to the classical optimality condition in scalar optimization.

We will consider a special case of nonconvex multiobjective functions, namely, those in which each component is quasiconvex or pseudoconvex. Recall that a scalar function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is said to be quasiconvex if, for any $x, y \in \mathbb{R}^n$ and for all $t \in [0, 1]$,

$$f(tx + (1-t)y) \leq \max\{f(x), f(y)\}.$$

It is known from [29] that a differentiable function f is quasiconvex if and only if, for any $x, y \in \mathbb{R}^n$,

$$f(x) \leq f(y) \quad \text{implies that} \quad \langle \nabla f(y), x - y \rangle \leq 0. \quad (2.3)$$

A differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is called pseudoconvex if $\langle \nabla f(y), x - y \rangle \geq 0$ implies $f(x) \geq f(y)$. It is well-known that pseudoconvex functions are quasiconvex.

We say that the multiobjective function $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is quasiconvex (or pseudoconvex) if each component function $f_i, i = 1, \dots, m$, of F is quasiconvex (or pseudoconvex). It can be verified by definition that when F is pseudoconvex, the condition (2.2) is also sufficient for a point to be weak Pareto efficient (or see [13]); but in general, it does not hold for quasiconvex functions (see [30]).

The following quasi-Fejér convergence theorem will be needed in the convergence analysis of the proposed algorithm. Recall that a sequence $\{u_k\}$ is said to be quasi-Fejér convergent to a nonempty set $U \subseteq \mathbb{R}^n$ if, for each $u \in U$, there exists a sequence $\{\varepsilon_k\} \subseteq \mathbb{R}_+$ such that $\sum_{k=0}^{\infty} \varepsilon_k < \infty$ and

$$\|u_{k+1} - u\|^2 \leq \|u_k - u\|^2 + \varepsilon_k.$$

Proposition 2.1. ([31, Theorem 1]) *If $\{u_k\} \subseteq \mathbb{R}^n$ is quasi-Fejér convergent to a nonempty set $U \subseteq \mathbb{R}^n$, then*

- (i) *the sequence $\{u_k\}$ is bounded.*
- (ii) *if an accumulation point \bar{u} of $\{u_k\}$ belongs to U , then $\lim_{k \rightarrow \infty} u_k = \bar{u}$.*

3. THE ALGORITHM

In this section, we give the nonmonotone projected gradient algorithm considered here and present some basic properties that will be needed later.

Let the sequence $\{\beta_k\} \subseteq \mathbb{R}_+$ of parameters be such that

$$\sum_{k=0}^{+\infty} \beta_k = +\infty, \quad \sum_{k=0}^{+\infty} \beta_k^2 < +\infty. \quad (3.1)$$

Algorithm 3.1. Step 1 Choose parameters $\sigma \in (0, 1)$ and $0 \leq \theta_{\min} \leq \theta_{\max} \leq 1$. Let $x_0 \in C$ be an arbitrary initial point. Set $C_0 = F(x_0)$, $Q_0 = 1$ and $k = 0$.

Step 2 If $\min_{1 \leq i \leq m} \|\nabla f_i(x_k)\| = 0$, then **stop**. Otherwise, take $\eta_k = \max_{1 \leq i \leq m} \|\nabla f_i(x_k)\|$. Compute the search direction v_k :

$$v_k := \operatorname{argmin}_{v \in C - x_k} \varphi_k(v), \quad (3.2)$$

where $\varphi_k(v) = \frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v \rangle + \frac{\|v\|^2}{2}$.

If $v_k = 0$, then **stop**. Otherwise,

Step 3 Compute

$$j(k) := \min\{j \in \mathbb{Z}_+ : F(x_k + 2^{-j}v_k) \preceq C_k + \sigma 2^{-j} J_F(x_k)v_k\}. \quad (3.3)$$

Set

$$x_{k+1} = x_k + \gamma_k v_k, \quad (3.4)$$

where $\gamma_k = 2^{-j(k)}$.

Step 4 Choose $\theta_k \in [\theta_{\min}, \theta_{\max}]$ and define

$$Q_{k+1} = \theta_k Q_k + 1, \quad C_{k+1} = \frac{\theta_k Q_k C_k + F(x_{k+1})}{Q_{k+1}}.$$

Set $k := k + 1$ and go back to Step 2.

As mentioned in [22], the choice of θ_k controls the degree of the non-monotonicity of the line search. If $\theta_k = 0$ for each k , then $C_k = F(x_k)$ and so the line search (3.3) is the usual Armijo line search; while if $\theta_k = 1$ for each k , then $C_k = \frac{1}{k+1} \sum_{i=0}^k f_i(x_i)$ is the average of all the previous function values.

Remark 3.1. Algorithm 3.1 was proposed recently by the authors in [28] for the multiobjective optimization (2.1) with F being assumed to be convex. In [28], under some Lipschitz continuity assumption of the gradients of objective functions, they established the linear convergence of the generated sequence to weak Pareto efficient solutions. Here, in the present work, we consider the multiobjective optimization problem (2.1) but with the general nonconvex function F . In this situation, we will show that every accumulation point of the generated sequence is a Pareto stationary point of (2.1).

The following lemma lists some basic properties about Algorithm 3.1. For (i) one can refer to [22, Lemma 1.1], for (ii) one can refer to [28, Lemma 2]. We omit the proof here.

Lemma 3.1. (i) For each $k \in \mathbb{N}$, if $J_F(x_k)v_k \prec 0$, then $F(x_k) \preceq C_k$.
(ii) Algorithm 3.1 is well-defined and the generated sequence belongs to C .

Now, we show the stopping criteria is valid. For this, we need a useful inequality. Fix $k \geq 0$. Note that the function φ_k is strongly convex. Therefore the first order optimality condition for $\min_{v \in C - x_k} \varphi_k(v)$ is necessary and sufficient. So there exists $u_k \in \partial \varphi_k(v_k)$ such that

$$\langle u_k, v - v_k \rangle \geq 0, \quad \forall v \in C - x_k. \quad (3.5)$$

Then, it follows from the expression of φ_k and the formula for the subdifferential of the maximum of convex functions (see, e.g., [32, Theorem 4.4.2]) that there exist $\emptyset \neq J_k \subseteq \{1, \dots, m\}$ and $\lambda_j^k > 0$ with $j \in J_k$ such that

$$\sum_{j \in J_k} \lambda_j^k = 1, \quad \langle \nabla f_j(x_k), v_k \rangle = \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v_k \rangle, \quad \forall j \in J_k \quad (3.6)$$

and

$$u_k = v_k + \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \nabla f_j(x_k). \quad (3.7)$$

Combining (3.5) and (3.7) yields

$$\left\langle v_k + \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \nabla f_j(x_k), v - v_k \right\rangle \geq 0, \quad \forall v \in C - x_k. \quad (3.8)$$

Proposition 3.1. If the Algorithm 3.1 stops at iteration k , then x_k is a Pareto stationary point.

Proof. Take $k \in \mathbb{N}$. If there exists $i_0 \in \{1, \dots, m\}$ such that $\nabla f_{i_0}(x_k) = 0$, then, for any $x \in C$, $\langle \nabla f_{i_0}(x_k), x - x_k \rangle = 0$, and $J_F(x_k)(C - x_k) \cap (-\mathbb{R}_{++}^m) = \emptyset$. That is, x_k is a Pareto stationary point.

Now, assume that $v_k = 0$. It follows from (3.8) that

$$\left\langle \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \nabla f_j(x_k), x - x_k \right\rangle \geq 0, \quad \forall x \in C.$$

Thus, for any $x \in C$, there exists $j \in J_k \subseteq \{1, \dots, m\}$ such that $\langle \nabla f_j(x_k), x - x_k \rangle \geq 0$. Consequently, $J_F(x_k)(C - x_k) \cap (-\mathbb{R}_{++}^m) = \emptyset$, and x_k is a Pareto stationary point. \square

From now on, we assume that the sequence $\{x_k\}$ generated by Algorithm 3.1 is infinite, and this means that $v_k \neq 0$ for each k . Note by Lemma 3.1(ii) that $0 \in C - x_k$ for each k , and the definition of v_k in (3.2). Then, it follows from $\varphi_k(v_k) \leq \varphi_k(0) = 0$ that

$$\frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v_k \rangle \leq -\frac{\|v_k\|^2}{2} < 0. \quad (3.9)$$

This implies that, for each k , $J_F(x_k)v_k \prec 0$. Hence, $F(x_k) \preceq C_k$ due to Lemma 3.1(i).

Lemma 3.2. *Let $\{v_k\}$ be the sequence generated by Algorithm 3.1. Then, for all $k \in \mathbb{N}$, $\|v_k\| \leq \beta_k$.*

Proof. Take $k \in \mathbb{N}$. Then, by applying (3.8) with $v = 0$, one has

$$\|v_k\|^2 \leq -\frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), v_k \rangle. \quad (3.10)$$

In view of (3.6) and noting (3.9), one can obtain that

$$\begin{aligned} -\frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), v_k \rangle &= -\frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v_k \rangle \\ &\leq \frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} |\langle \nabla f_i(x_k), v_k \rangle| \leq \frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \|\nabla f_i(x_k)\| \|v_k\| = \beta_k \|v_k\|, \end{aligned} \quad (3.11)$$

where the last equality holds by the definition of η_k . This, together with (3.10), implies the desired result. \square

We end this section by the following fundamental inequality.

Proposition 3.2. *Let $x \in C$ and $k \in \mathbb{N}$. It holds that*

$$\|x_{k+1} - x\|^2 \leq \|x_k - x\|^2 + 3\beta_k^2 + \frac{2\gamma_k\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), x - x_k \rangle.$$

Proof. From (3.4), we have

$$\|x_{k+1} - x\|^2 = \|x_k - x\|^2 + \gamma_k^2 \|v_k\|^2 + 2\gamma_k \langle v_k, x_k - x \rangle. \quad (3.12)$$

Taking $v = x - x_k$ in (3.8) and noting (3.11), it follows that

$$\begin{aligned} \langle v_k, x_k - x \rangle &\leq -\|v_k\|^2 + \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), x - x_k \rangle - \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), v_k \rangle \\ &\leq \frac{\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), x - x_k \rangle + \beta_k \|v_k\|. \end{aligned}$$

Combining this with (3.12), also noting that $\gamma_k \in (0, 1]$, one then gets

$$\|x_{k+1} - x\|^2 \leq \|x_k - x\|^2 + (\|v_k\|^2 + 2\beta_k \|v_k\|) + \frac{2\gamma_k\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), x - x_k \rangle. \quad (3.13)$$

Thus, the conclusion follows immediately from (3.13) and Lemma 3.2. \square

4. CONVERGENCE ANALYSIS

In this section, we will investigate the convergence property of the sequence generated by Algorithm 3.1. The following theorem is the first main result of this section, which establishes the stationarity of the accumulation points of the generated sequence.

Theorem 4.1. *Every accumulation point, if any, of the sequence $\{x_k\}$ generated by Algorithm 3.1 is a Pareto stationary point.*

Proof. Let x^* be an accumulation point of $\{x_k\}$, and let $\{x_{j_k}\} \subseteq \{x_k\}$ be the subsequence such that $\lim_{k \rightarrow \infty} x_{j_k} = x^*$. Since $\{x_k\} \subseteq C$ (see Lemma 3.1(ii)) and C is closed, it follows that $x^* \in C$. In the next, we prove the conclusion by contradiction. Suppose that x^* is not Pareto stationary. Then, there exists $\hat{x} \in C$ such that $J_F(x^*)(\hat{x} - x^*) \in -\mathbb{R}_{++}^m$, i.e.,

$$\max_{1 \leq i \leq m} \langle \nabla f_i(x^*), \hat{x} - x^* \rangle < 0. \quad (4.1)$$

We will show the contradiction by considering the following two cases: (a) $\limsup_{k \rightarrow \infty} \gamma_k > 0$, (b) $\limsup_{k \rightarrow \infty} \gamma_k = 0$.

First, assume that (a) holds. By applying Proposition 3.2 with $x = \hat{x}$, we have that, for all k ,

$$\|x_{k+1} - \hat{x}\|^2 \leq \|x_k - \hat{x}\|^2 + 3\beta_k^2 + \frac{2\gamma_k\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), \hat{x} - x_k \rangle. \quad (4.2)$$

Take $N \in \mathbb{N}$. Summing both sides of (4.2) with k between 0 and N , we obtain

$$\begin{aligned} & -2 \sum_{k=0}^N \frac{\gamma_k\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), \hat{x} - x_k \rangle \\ & \leq \|x_0 - \hat{x}\|^2 - \|x_{N+1} - \hat{x}\|^2 + 3 \sum_{k=0}^N \beta_k^2 \\ & \leq \|x_0 - \hat{x}\|^2 + 3 \sum_{k=0}^N \beta_k^2. \end{aligned} \quad (4.3)$$

Letting N go to ∞ in (4.3) and using (3.1), it follows that

$$\liminf_{k \rightarrow \infty} \left(-\frac{\gamma_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), \hat{x} - x_k \rangle \right) \leq 0.$$

By this and the assumption (a), without loss of generality, we may assume that the subsequence $\{j_k\} \subseteq \{k\}$ satisfies

$$\lim_{k \rightarrow \infty} \left(-\frac{\gamma_{j_k}}{\eta_{j_k}} \max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle \right) \leq 0 \quad (4.4)$$

and

$$\lim_{k \rightarrow \infty} \gamma_{j_k} > 0. \quad (4.5)$$

By (4.5), one has that for large enough k , $\gamma_{j_k} \geq \bar{\gamma}$ for some $\bar{\gamma} > 0$. Note that $\{x_{j_k}\}$ is bounded. So $\{\eta_{j_k}\}$ is bounded thanks to the continuous differentiability of f_{j_k} , i.e., there exists $M > 0$ such that $\eta_{j_k} \leq M$ for all k . Moreover, in view of (4.1), one can get from the continuity of the max function and the gradient function ∇f_i that

$$\lim_{k \rightarrow \infty} \max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle = \max_{1 \leq i \leq m} \langle \nabla f_i(x^*), \hat{x} - x^* \rangle < 0,$$

which implies that there is some $\bar{\alpha} > 0$ such that, for large enough k ,

$$\max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle \leq -\bar{\alpha}.$$

Consequently, from the above discussion, one can obtain that

$$-\frac{\gamma_{j_k}}{\eta_{j_k}} \max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle \geq \frac{\bar{\gamma}\bar{\alpha}}{M} > 0,$$

which contradicts (4.4).

Now, assume that (b) holds. For each k , let $\varepsilon_k = \|\hat{x} - x_k\|$ and $u_k = \frac{\|v_k\|}{\varepsilon_k}(\hat{x} - x_k)$. Then, $\lim_{k \rightarrow \infty} \varepsilon_{j_k} = \|\hat{x} - x^*\|$. By Lemma 3.2 and (3.1), one has $\lim_{k \rightarrow \infty} \|v_k\| = 0$. Hence, $\lim_{k \rightarrow \infty} \frac{\|v_{j_k}\|}{\varepsilon_{j_k}} = 0$, which implies that for large enough k , $\frac{\|v_{j_k}\|}{\varepsilon_{j_k}} < 1$. Then, for large enough k ,

$$u_{j_k} = \left(1 - \frac{\|v_{j_k}\|}{\varepsilon_{j_k}}\right) x_{j_k} + \frac{\|v_{j_k}\|}{\varepsilon_{j_k}} \hat{x} - x_{j_k} \in C - x_{j_k},$$

and so

$$\varphi_{j_k}(v_{j_k}) \leq \varphi_{j_k}(u_{j_k}) = \frac{\|u_{j_k}\|^2}{2} + \frac{\beta_{j_k}\|v_{j_k}\|}{\eta_{j_k}\varepsilon_{j_k}} \max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle. \quad (4.6)$$

Note that $\|u_{j_k}\| = \|v_{j_k}\|$. Then, it follows from (4.6) that, for large enough k ,

$$\max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), v_{j_k} \rangle \leq \frac{\|v_{j_k}\|}{\varepsilon_{j_k}} \max_{1 \leq i \leq m} \langle \nabla f_i(x_{j_k}), \hat{x} - x_{j_k} \rangle. \quad (4.7)$$

Without loss of generality, we assume that $\lim_{k \rightarrow \infty} \frac{v_{j_k}}{\|v_{j_k}\|} = \bar{v}$. Dividing both sides of (4.7) by $\|v_{j_k}\|$ and taking limits with $k \rightarrow \infty$, one gets that

$$\max_{1 \leq i \leq m} \langle \nabla f_i(x^*), \bar{v} \rangle \leq \frac{1}{\|\hat{x} - x^*\|} \max_{1 \leq i \leq m} \langle \nabla f_i(x^*), \hat{x} - x^* \rangle < 0, \quad (4.8)$$

where the second inequality holds due to (4.1).

Fix $i \in \{1, \dots, m\}$. Since f_i is continuously differentiable, it follows that

$$\begin{aligned} & \lim_{k \rightarrow \infty} \frac{f_i(x_{j_k}) - f_i(x_{j_k} + v_{j_k}) + \sigma \langle \nabla f_i(x_{j_k}), v_{j_k} \rangle}{\|v_{j_k}\|} \\ &= \lim_{k \rightarrow \infty} \left((\sigma - 1) \left\langle \nabla f_i(x_{j_k}), \frac{v_{j_k}}{\|v_{j_k}\|} \right\rangle + \frac{o(\|v_{j_k}\|)}{\|v_{j_k}\|} \right) \\ &= (\sigma - 1) \langle \nabla f_i(x^*), \bar{v} \rangle > 0, \end{aligned}$$

where the inequality holds thanks to (4.8) and the fact that $\sigma \in (0, 1)$. This implies that for large enough k ,

$$\frac{f_i(x_{j_k}) - f_i(x_{j_k} + v_{j_k}) + \sigma \langle \nabla f_i(x_{j_k}), v_{j_k} \rangle}{\|v_{j_k}\|} > 0,$$

and so

$$f_i(x_{j_k} + v_{j_k}) < f_i(x_{j_k}) + \sigma \langle \nabla f_i(x_{j_k}), v_{j_k} \rangle.$$

Consequently, by the arbitrariness of i and Lemma 3.1(i), we have that for large enough k ,

$$F(x_{j_k} + v_{j_k}) \prec F(x_{j_k}) + \sigma J_F(x_{j_k}) v_{j_k} \preceq C_{j_k} + \sigma J_F(x_{j_k}) v_{j_k}.$$

This, together with (3.3), implies that $j(j_k) = 0$ and henceforth $\gamma_{j_k} = 1$ for large enough k , which is in contradiction to the assumption (b).

Therefore, we conclude that x^* is a Pareto stationary point. The proof is complete. \square

When the line search rule in Algorithm 3.1 is removed, we obtain the following gradient method without line search.

Algorithm 4.1. Step 1 Let $x_0 \in C$ be an arbitrary initial point. Set $k = 0$.

Step 2 If $\min_{1 \leq i \leq m} \|\nabla f_i(x_k)\| = 0$, then **stop**. Otherwise, take $\eta_k = \max_{1 \leq i \leq m} \|\nabla f_i(x_k)\|$. Compute the search direction v_k :

$$v_k := \operatorname{argmin}_{v \in C - x_k} \phi_k(v),$$

$$\text{where } \phi_k(v) = \frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v \rangle + \frac{\|v\|^2}{2}.$$

If $v_k = 0$, then **stop**. Otherwise,

Step 3 Set $x_{k+1} = x_k + v_k$. Let $k := k + 1$ and go back to Step 2.

It can be verified that the conclusions in Lemma 3.1(ii), Lemma 3.2, Propositions 3.1 and 3.2 (with $\gamma_k = 1$) that originally for Algorithm 3.1 are still hold for Algorithm 4.1. Then, by using the similar proof to that for the case (a) of Theorem 4.1, one can obtain the following result.

Theorem 4.2. *Every accumulation point, if any, of the sequence $\{x_k\}$ generated by Algorithm 4.1 is a Pareto stationary point.*

In the rest of this section, we consider a special case of Algorithm 3.1, that is, the nonmonotone line search in Algorithm 3.1 is replaced by the monotone Armijo line search (i.e., for each k , $\theta_k = 0$ and so $C_k = F(x_k)$, $Q_k = 1$). In this case, Algorithm 3.1 reduces to the following form.

Algorithm 4.2. Step 1 Choose parameter $\sigma \in (0, 1)$. Let $x_0 \in C$ be an arbitrary initial point. Set $k = 0$.

Step 2 If $\min_{1 \leq i \leq m} \|\nabla f_i(x_k)\| = 0$, then **stop**. Otherwise, take $\eta_k = \max_{1 \leq i \leq m} \|\nabla f_i(x_k)\|$. Compute the search direction v_k :

$$v_k := \operatorname{argmin}_{v \in C - x_k} \phi_k(v),$$

$$\text{where } \phi_k(v) = \frac{\beta_k}{\eta_k} \max_{1 \leq i \leq m} \langle \nabla f_i(x_k), v \rangle + \frac{\|v\|^2}{2}.$$

If $v_k = 0$, then **stop**. Otherwise,

Step 3 Compute

$$j(k) := \min\{j \in \mathbb{Z}_+ : F(x_k + 2^{-j}v_k) \preceq F(x_k) + \sigma 2^{-j} J_F(x_k)v_k\}. \quad (4.9)$$

Set

$$x_{k+1} = x_k + \gamma_k v_k,$$

where $\gamma_k = 2^{-j(k)}$.

Set $k := k + 1$ and go back to Step 2.

Note from (3.9) that for each k , $J_F(x_k)v_k \prec 0$, and thus it follows from the line search (4.9) that $\{F(x_k)\}_{k=0}^\infty$ is monotone decreasing with respect to the partial order “ \preceq ”.

Remark 4.1. The Algorithm 4.2 is the multiobjective case of the algorithm that considered in [14], where the authors used the algorithm therein to solve the convex vector optimization problem. In [13], the authors considered a different algorithm from Algorithm 4.2, that is the exogenous parameter $\frac{\beta_k}{\eta_k}$ in Algorithm 4.2 was replaced by a fixed positive constant. By applying the algorithm therein, the authors in [13] studied the problem (2.1) in the case when F is quasiconvex/pseudoconvex. Inspired by this, we will establish here the convergence results of Algorithm 4.2 for problem (2.1) when F is quasiconvex/pseudoconvex.

Define the auxiliary set T as

$$T := \{x \in C : F(x) \preceq F(x_k), \forall k \in \mathbb{N}\}.$$

The condition $T \neq \emptyset$ has been extensively applied in the convergence analysis of classical method extensions to vector optimization, such as the projected gradient method [13, 14, 33], steepest descent method [34], proximal point method [35, 36] and so on. This assumption is related to the completeness of the image of F , which is standard for ensuring the existence of Pareto optimal points for vector optimization problems [37].

The following lemma can be obtained directly from the characterization of the differentiable quasiconvex functions (see (2.3)), or one can refer to [13, Proposition 3].

Lemma 4.1. *Assume that F is quasiconvex and $x \in T$. Then, $\langle \nabla f_i(x_k), x - x_k \rangle \leq 0$ for all $k \in \mathbb{N}$ and $i = 1, \dots, m$.*

The next theorem provides a sufficient condition for the convergence of the sequence $\{x_k\}$ generated by Algorithm 4.2.

Theorem 4.3. *Assume that F is quasiconvex and $T \neq \emptyset$. Then the sequence $\{x_k\}$ generated by Algorithm 4.2 converges to a Pareto stationary point.*

Proof. Take $x \in T$. Then, by Proposition 3.2 and Lemma 4.1, one can get that, for all $k \in \mathbb{N}$,

$$\begin{aligned} \|x_{k+1} - x\|^2 &\leq \|x_k - x\|^2 + 3\beta_k^2 + \frac{2\gamma_k\beta_k}{\eta_k} \sum_{j \in J_k} \lambda_j^k \langle \nabla f_j(x_k), x - x_k \rangle \\ &\leq \|x_k - x\|^2 + 3\beta_k^2. \end{aligned}$$

Noting (3.1), the series $\sum_{k=0}^{+\infty} 3\beta_k^2$ is convergent. Thus, $\{x_k\}$ is quasi-Fejér convergent to T and so is bounded by Proposition 2.1(i). Let \bar{x} be an accumulation point of $\{x_k\}$. Since the sequence $\{F(x_k)\}$ is monotone decreasing, it follows that $\bar{x} \in T$. Then, Proposition 2.1(ii) ensures that $\{x_k\}$ converges to \bar{x} , which is Pareto stationary by Theorem 4.1. \square

Note that the pseudoconvexity implies the quasiconvexity, and the Pareto stationary point of (2.1) is also the weak Pareto optimal point when F is pseudoconvex. Thus, one can obtain directly from Theorem 4.3 the following convergence result.

Corollary 4.1. *Assume that F is pseudoconvex and $T \neq \emptyset$. Then the sequence $\{x_k\}$ generated by Algorithm 4.2 converges to a weak Pareto optimal point.*

5. NUMERICAL ILLUSTRATIONS

In this section, we present some numerical results to show the validity of our proposed Algorithm 3.1. The experiments were performed by using MATLAB 2019b on a PC with the specification: Processor AMD Ryzen 53500 U & 2.10 GHz, 8.00 GB RAM. We compare the performance of the proposed method with the projected gradient method in [27, Algorithm 2.1]. The multiobjective optimization problems are solved under the box constraints $L \leq x \leq U$. We used the quadprog function in the MATLAB Optimization Toolbox to solve the optimization subproblems in the line searches. For the termination criterion (Step 2 of Algorithm 3.1), we use $|v_k| < \varepsilon$ with $\varepsilon = 10^{-4}$. We compare our proposed Algorithm 3.1 and [27, Algorithm 2.1] by using the number of iterations, time of computation and the objective function evaluations. We consider the nonconvex multiobjective optimization problems listed in Example 5.1 - Example 5.13 and the convex multiobjective optimization problems listed in Example 5.14 - Example 5.18. For Algorithm 3.1, we choose $\sigma = 0.03$, $\theta_{\min} = 0$, $\theta_{\max} = 1$, $\beta_k = \frac{1}{k+1}$ and for [27, Algorithm 2.1], we choose $\sigma = 0.03$, $\eta_{\min} = 0$, $\eta_{\max} = 1$, $\beta_{\min} = 0$, $\beta_{\max} = 1$. The initial point is generated randomly for both algorithm in the experiments. In Examples 5.1-5.18, n denotes the number of variables, m is the number of objective functions, L is the lower bound and U is the upper bound of the box constraint used in each example. Table 1 contains the computational results for the numerical experiments. From it, we see that Algorithm 3.1 solves the multiobjective optimization problems more efficiently than [27, Algorithm 2.1].

Example 5.1. [24] $n = 2$, $m = 2$, $L = -10$, $U = 10$,

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

Example 5.2. $n = 2$, $m = 2$, $L = -100$, $U = 100$,

$$\begin{aligned} F_1(x) &= \frac{1}{4}(x_1-1)^4 + 2(x_2-2)^4, \\ F_2(x) &= (x_2-x_1)^2 + (1-x_1)^2. \end{aligned}$$

Example 5.3. $n = 2$, $m = 2$, $L = -1$, $U = 1$,

$$\begin{aligned} F_1(x) &= 2 \exp(15(-x_1-0.1)^2 - x_2^2) - \exp(20(-x_1-0.6)^2 - (x_2-0.6)^2) \\ &\quad + \exp(20(-x_1+0.6)^2 - (x_2+0.6)^2) + \exp(20(-x_1-0.6)^2 - (x_2+0.6)^2) \\ &\quad + \exp(20(-x_1+0.6)^2 - (x_2-0.6)^2), \\ F_2(x) &= 2 \exp(20(-x_1^2 - x_2^2)) + \exp(20(-x_1-0.4)^2 - (x_2-0.6)^2) \\ &\quad - \exp(20(-x_1+0.5)^2 - (x_2+0.7)^2) - \exp(-20(-x_1+0.4)^2 - (x_2+0.8)^2). \end{aligned}$$

TABLE 1. Comparison between Algorithm 3.1 and [27, Algorithm 2.1]

	Algorithm 3.1			[27, Algorithm 2.1]		
	Iter.	Time (s)	Objective Functions Eval.	Iter	Time (s)	Objective Functions Eval.
Example 5.1	6	0.0117	-2.4360, 7.2566	29	0.0422	-3.6334, 5.2295
Example 5.2	1	0.0048	0.5000, 0.0000	1	0.0065	0.500, 0.0000
Example 5.3	2	0.0099	-0.0017, 3.0432E-5	2	0.0118	0.0017, 3.0432E-5
Example 5.4	2	0.0082	0.9817, 0.9817	2	0.0141	0.9817, 0.9817
Example 5.5	54	0.0550	0.3333, 3.000	54	0.0557	0.3333, 3.0000
Example 5.6	47	0.0444	-2.5425, -27.2500	1001	10.6146	-18.2259, -18.2229
Example 5.7	9	0.0132	2.4394, 2.4394	9	0.0914	1.0481, 1.0481
Example 5.8	9	0.128	1.0000, 1.7057	9	0.0175	1.0000, 1.7057
Example 5.9	9	0.0279	1.0000, 2.1212	9	0.2018	1.0000, 2.1212
Example 5.10	5	0.0092	0.1577, 0.9971	5	0.0464	0.1577, 0.9971
Example 5.11	26	0.0269	-44.3754, 135.6293	42	0.0401	-44.4548, -85.4817
Example 5.12	6	0.0134	2.0231, 2.0231	8	0.0137	1.0333, 1.03333
Example 5.13	9	0.0358	4.4700, 4.8700	1001	0.4869	-8.8256, -1.4120
Example 5.14	38	0.0547	2.4167, -5.6133	51	0.0623	5.0000, 4.6667
Example 5.15	7	0.0122	5.5556, 5.7183, 0.3066	7	0.0279	5.5556, 5.7183, 0.3066
Example 5.16	8	0.0252	1.0036E4, 52.7183, 3.1883	12	0.0355	1.0036E4, 52.7183, 3.1883
Example 5.17	1	0.0206	0.0000, 4.1200	1	0.3938	0.0000, 4.1200
Example 5.18	1	0.0132	0.0000, 4.0000	1	0.0777	0.0000, 4.0000

Example 5.4. $n = 2, m = 2, L = 0, U = 1$.

$$\begin{aligned}
 F_1(x) &= 1 - \exp(-(x_1 - 1)^2 - (x_2 + 1)^2), \\
 F_2(x) &= 1 - \exp(-(x_1 + 1)^2 - (x_2 - 1)^2).
 \end{aligned}$$

Example 5.5. $n = 2, m = 2, L = -3, U = 3,$

$$\begin{aligned} F_1(x) &= \frac{1}{x_1^2 + x_2^2 + 1}, \\ F_2(x) &= x_1^2 + 3x_2^2 + 1. \end{aligned}$$

Example 5.6. $n = 2, m = 2, L = -100, U = 100,$

$$\begin{aligned} F_1(x) &= x_1^2 - x_2^2 - 4 \left(\exp(-(x_1 + 2)^2 - x_2^2) + \exp(-(x_1 - 2)^2 - x_2^2) \right), \\ F_2(x) &= (x_1 - 6)^2 + (x_2 + 0.5)^2. \end{aligned}$$

Example 5.7. $n = 2, m = 2, L = -5, U = 5,$

$$\begin{aligned} F_1(x) &= \frac{1}{2} \left(\sqrt{1 + (x_1 + x_2)^2} + \sqrt{1 + (x_1 - x_2)^2 + x_1 - x_2} \right), \\ F_2(x) &= \frac{1}{2} \left(\sqrt{1 + (x_1 + x_2)^2} + \sqrt{1 + (x_1 - x_2)^2 - x_1 + x_2} \right), \\ &\quad + k \exp(-x_1 - x_2)^2, \end{aligned}$$

where $k = 3\text{rand}(1, 1)$.

Example 5.8. $n = 2, m = 2, L = 0, U = 2,$

$$\begin{aligned} F_1(x) &= x_1, \\ F_2(x) &= x_1 \left(2 - 0.8 \exp \left(-\frac{2}{5} (x_2 - 0.6)^2 \right) - \exp \left(\frac{-1}{5} (-x_2 - 0.2) \right) \right). \end{aligned}$$

Example 5.9. $n = 100, m = 2, L = -5, U = 5,$

$$\begin{aligned} F_1(x) &= \left[\frac{1}{n} \sum_{i=1}^n x_i^2 \cos(2\pi x) + 10 \right]^{\frac{1}{4}}, \\ F_2(x) &= \left[\frac{1}{n} \sum_{i=1}^n \left(x_i - \frac{3}{2} \right)^2 - 10 \cos \left(2\pi \left(x_i - \frac{3}{2} \right) \right) + 10 \right]^{\frac{1}{4}}. \end{aligned}$$

Example 5.10. $n = 2, m = 2, L = -1, U = 1,$

$$\begin{aligned} F_1(x) &= 1 - \exp \left(-\sum_{i=1}^n \left(x_i - \frac{1}{\sqrt{i}} \right) \right), \\ F_2(x) &= 1 - \exp \left(-\sum_{i=1}^n \left(x_i + \frac{1}{\sqrt{i}} \right) \right). \end{aligned}$$

Example 5.11. $n = 2, m = 2, L = -\pi, U = \pi,$

$$F_1(x) = (1 - A - B)^2 - (C - D)^2,$$

where

$$\begin{aligned} A &= \frac{1}{2} \sin(1) - 2 \cos(1) + \sin(2) - \frac{3}{2} \cos(2), \\ B &= \frac{3}{2} \sin(1) - \cos(1) + 2 \sin(2) - \frac{1}{2} \cos(2), \\ C &= \frac{1}{2} \sin(x_1) - 2 \cos(x_1) + \sin(x_2) - \frac{3}{2} \cos(x_2), \\ D &= \frac{3}{2} \sin(x_1) - \cos(x_1) + 2 \sin(x_2) - \frac{1}{2} \cos(x_2), \end{aligned}$$

and

$$F_2(x) = (x_1 + 3)^2 - (x_2 + 1)^2.$$

Example 5.12. $n = 4, m = 2, L = -10, U = 10$,

$$\begin{aligned} F_1(x) &= (x_1 - 2)^2 - (x_2 + 3)^2 - (x_3 - 5)^2 - (x_4 - 4)^2 + 5, \\ F_2(x) &= \frac{\sin(x_1) + \sin(x_2) + \sin(x_3) + \sin(x_4)}{100(1 + x_1^2 + x_2^2 + x_3^2 + x_4^2)}. \end{aligned}$$

Example 5.13. $n = 2, m = 2, L = -100, U = 100$,

$$\begin{aligned} F_1(x) &= 5 - \frac{1}{200} \left((x_1^2 + x_2^2 - 1)^2 + (x_1 + x_2 - 7)^2 \right), \\ F_2(x) &= 5 - \frac{1}{200} \left((4x_1^2 + 2x_2 - 11)^2 + (2x_1 + 4x_2^2 - 7)^2 \right). \end{aligned}$$

Example 5.14. [24] $n = 5, m = 2, L = -20, U = 20$,

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

Example 5.15. $n = 2, m = 3, L = -100, U = 100$,

$$\begin{aligned} F_1(x) &= \frac{1}{9}(x_1 - 1)^2 + 2(x_2 - 2)^4 + 3(x_3 - 3)^4, \\ F_2(x) &= \frac{1}{3} \exp(x_1 + x_2 + x_3) + 3(x_3 - 3)^4, \\ F_3(x) &= \frac{1}{4} \exp(-x_1) + \frac{1}{3} \exp(-x_2) + \frac{1}{2} \exp(-x_3). \end{aligned}$$

Example 5.16. [24] $n = 50, m = 3, L = -1E6, U = 1E6$,

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

Example 5.17. $n = 100, m = 2, L = -100, U = 100$,

$$F_1(x) = (x_1 - 1)^4 + \sum_{i=2}^n (x_i - 1)^2,$$

$$F_2(x) = (x_1 - 1)^2 + \sum_{i=2}^n (x_i - 1)^4.$$

Example 5.18. $n = 2, m = 2, L = -100, U = 100$,

$$F_1(x) = (x_1 - 1)^2 + (x_1 - x_2)^2,$$

$$F_2(x) = (x_1 - 3)^2 + (x_1 - x_2)^2.$$

6. CONCLUSION

In this paper, we considered a projected gradient method equipped with the nonmonotone line search rule for nonconvex constrained multiobjective optimization problem. Under some mild assumptions, we showed that each accumulation point of the generated sequence is a Pareto stationary point. This result still holds for the algorithm without line search (i.e., when the nonmonotone line search in the algorithm is dropped). Furthermore, we considered the special case of the nonmonotone algorithm, that is, the nonmonotone line search is replaced by the monotone Armijo line search. In this case, we established the convergence of the generated sequence to a Pareto stationary point (respectively, a weak Pareto optimal point) when the multiobjective function is quasiconvex (respectively, pseudoconvex). Moreover, we confirmed the efficiency of the method with some numerical examples.

We leave for a future work the applications of this nonmonotone method in some practical problems. Besides, extending this method to solve nonsmooth multiobjective optimization problems is also an interesting future direction.

Acknowledgments

X.P. Zhao was supported by the National Natural Science Foundation of China (Grant 11801411) and the Natural Science Foundation of Tianjin (Grant 18JCQNJC01100). L.O. Jolaoso was supported by the postdoctoral research fund at the Sefako Makgatho Health Sciences University, Pretoria, South Africa. He also acknowledges the support and assistance from the Department of Mathematics and Applied Mathematics at the institution. The research of J.C. Yao was supported by the Grant MOST (108- 2115-M-039-005-MY3). The authors also acknowledge the valuable contributions of A.H. Ibrahim and A. Kamandi in the numerical computations.

REFERENCES

- [1] H. Eschenauer, J. Koski, A. Osyczka, *Multicriteria Design Optimization*, Springer, Berlin, 1990.
- [2] M. M. Wiecek, Advances in cone-based preference modeling for decision making with multiple criteria, *Decis. Mak. Manuf. Serv.* 1 (2007), 153-173.
- [3] U. Bagchi, Simultaneous minimization of mean and variation of flow time and waiting time in single machine systems, *Oper. Res.* 37 (1989), 118-125.
- [4] D. Prabuddha, J. B. Ghosh, C. E. Wells, On the minimization of completion time variance with a bicriteria extension, *Oper. Res.* 40 (1992), 1148-1155.
- [5] D. J. White, Epsilon-dominating solutions in mean-variance portfolio analysis, *Eur. J. Oper. Res.* 105 (1998), 457-466.

- [6] T. M. Leschine, H. Wallenius, W. A. Verdini, Interactive multiobjective analysis and assimilative capacity-based ocean disposal decisions, *Eur. J. Oper. Res.* 56 (1992), 278-289.
- [7] Y. Fu, U. M. Diwekar, An efficient sampling approach to multiobjective optimization, *Ann. Oper. Res.* 132 (2004), 109-134.
- [8] E. Carrizosa, E. Conde, M. Munoz, J. Puerto, Planar point objective location problems with nonconvex constraints: A geometrical construction, *J. Global Optim.* 6 (1995), 77-86.
- [9] R. S. Burachik, C. Y. Kaya, M. M. Rizvi, A new scalarization technique and new algorithms to generate Pareto fronts, *SIAM J. Optim.* 27 (2017), 1010-1034.
- [10] G. Eichfelder, *Adaptive Scalarization Methods in Multiobjective Optimization*, Springer-Verlag, Berlin, Heidelberg, 2008.
- [11] L. M. Graña Drummond, N. Maculan, B. F. Svaiter, On the choice of parameters for the weighting method in vector optimization, *Math. Program.* 111 (2008), 201-216.
- [12] J. Fliege, L. M. Graña Drummond, B. F. Svaiter, Newton's method for multiobjective optimization, *SIAM J. Optim.* 20 (2009), 602-626.
- [13] J. Y. Bello Cruz, L. R. Lucambio Pérez, J. G. Melo, Convergence of the projected gradient method for quasiconvex multiobjective optimization, *Nonlinear Anal.* 74 (2011), 5268-5273.
- [14] L. M. Graña Drummond, A. N. Iusem, A projected gradient method for vector optimization problems, *Comput. Optim. Appl.* 28 (2004), 5-29.
- [15] J. Fliege, B. F. Svaiter, Steepest descent methods for multicriteria optimization, *Math. Methods Oper. Res.* 51 (2000), 479-494.
- [16] J. H. Wang, Y. H. Hu, C. K. W. Yu, C. Li, X. Q. Yang, Extended Newton methods for multiobjective optimization: Majorizing function technique and convergence analysis, *SIAM J. Optim.* 29 (2019), 2388-2421.
- [17] L. R. Lucambio Pérez, L. F. Prudente, Nonlinear conjugate gradient methods for vector optimization, *SIAM J. Optim.* 28 (2018), 2690-2720.
- [18] L. Grippo, F. Lampariello, S. Lucidi, A nonmonotone line search technique for Newton's method, *SIAM J. Numer. Anal.* 23 (1986), 707-716.
- [19] Y. H. Dai, On the nonmonotone line search, *J. Optim. Theory Appl.* 112 (2002), 315-330.
- [20] P. L. Toint, An assessment of non-monotone line search techniques for unconstrained optimization, *SIAM J. Sci. Comput.* 17 (1996), 725-739.
- [21] Y. H. Dai, A nonmonotone conjugate gradient algorithm for unconstrained optimization, *J. Syst. Sci. Complex.* 15 (2002), 139-145.
- [22] H. Zhang, W. W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, *SIAM J. Optim.* 14 (2004), 1043-1056.
- [23] S. Y. Cho, A monotone Bregan projection algorithm for fixed point and equilibrium problems in a reflexive Banach space, *Filomat* 34 (2020), 1487-1497.
- [24] K. Mita, E. H. Fukuda, N. Yamashita, Nonmonotone line searches for unconstrained multiobjective optimization problems, *J. Global Optim.* 75 (2019), 63-90.
- [25] N. Mahdavi-Amiri, F. Salehi Sadaghiani, A superlinearly convergent nonmonotone quasi-Newton method for unconstrained multiobjective optimization, *Optim. Method Softw.* 35 (2020), 1223-1247.
- [26] S. J. Qu, Y. Ji, J. L. Jiang, Q. P. Zhang, Nonmonotone gradient methods for vector optimization with a portfolio optimization application, *Eur. J. Oper. Res.* 263 (2017), 356-366.
- [27] N. S. Fazzio, M. L. Schuverdt, Convergence analysis of a nonmonotone projected gradient method for multiobjective optimization problems, *Optim. Lett.* 13 (2019), 1365-1379.
- [28] X. P. Zhao, J. C. Yao, Linear convergence of a nonmonotone projected gradient method for multiobjective optimization, *J. Global Optim.* submitted.
- [29] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.
- [30] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, 2007.
- [31] R. Burachik, L. M. Graña Drummond, A. N. Iusem, B. F. Svaiter, Full convergence of the steepest descent method with inexact line searches, *Optimization* 32 (1995), 137-146.
- [32] J.-B. Hiriart Urruty, C. Lemaréchal, *Convex Analysis and Minimization Algorithms I*, Springer, Berlin, 1993.
- [33] J. Y. Bello Cruz, A subgradient method for vector optimization problems, *SIAM J. Optim.* 23 (2013), 2169-2182.

- [34] G. C. Bento, J. X. Cruz Neto, P. R. Oliveira, A. Soubeyran, The self regulation problem as an inexact steepest descent method for multicriteria optimization, *Eur. J. Oper. Res.* 235 (2014), 494-502.
- [35] L. Ceng, J. C. Yao, Approximate proximal methods in vector optimization, *Eur. J. Oper. Res.* 183 (2007), 1-19.
- [36] E. A. Papa Quiroz, H. C. F. Apolinário, K. D. Villacorta, P. R. Oliveria, A linear scalarization proximal point method for quasiconvex multiobjective minimization, *J. Optim. Theory Appl.* 183 (2019), 1028-1052.
- [37] D. T. Luc, *Theory of Vector Optimization*, Lecture Notes in Economics and Mathematical Systems 319, Springer, Berlin, 1989.