Scaled Proximal Gradient Methods for Multiobjective Optimization: Improved Linear Convergence and Nesterov's Acceleration

Jian Chen · Liping Tang · Xinmin Yang

Abstract Over the past two decades, descent methods have received substantial attention within the multiobjective optimization field. Nonetheless, both theoretical analyses and empirical evidence reveal that existing first-order methods for multiobjective optimization converge slowly, even for well-conditioned problems, due to the objective imbalances. To address this limitation, we incorporate curvature information to scale each objective within the direction-finding subproblem, introducing a scaled proximal gradient method for multiobjective optimization (SPGMO). We demonstrate that the proposed method achieves improved linear convergence, exhibiting rapid convergence in well-conditioned scenarios. Furthermore, by applying small scaling to linear objectives, we prove that the SPGMO attains improved linear convergence for problems with multiple linear objectives. Additionally, integrating Nesterov's acceleration technique further enhances the linear convergence of SPGMO. To the best of our knowledge, this advancement in linear convergence is the first theoretical result that directly addresses objective imbalances in multiobjective first-order methods. Finally, we provide numerical experiments to validate the efficiency of the proposed methods and confirm the theoretical findings.

J. Chen
National Center for Applied Mathematics in Chongqing, Chongqing Normal University, Chongqing 401331, China, and School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China
chenjian_math@163.com

L.P. Tang
National Center for Applied Mathematics in Chongqing, and School of Mathematical Sciences, Chongqing Normal University, Chongqing 401331, China
tanglipings@163.com

✉X.M. Yang
National Center for Applied Mathematics in Chongqing, Chongqing Normal University, Chongqing 401331, China
xmyang@cqnu.edu.cn

1 Introduction

In multiobjective optimization, the primary goal is to optimize multiple objective functions simultaneously. Generally, it is not feasible to identify a single solution that simultaneously achieves optimal values for all objectives. Consequently, the notion of optimality is defined by *Pareto optimality* or *efficiency*. A solution is Pareto optimal or efficient if no objective can be improved without sacrificing the others. As society and the economy progress, the applications of this type of problem have proliferated across a multitude of domains, such as engineering [25], economics [18, 38], management science [13], and machine learning [31].

Solution strategies are pivotal in applications involving multiobjective optimization problems (MOPs). Over the past two decades, multiobjective gradient descent methods have gained escalating attention within the multiobjective optimization community, as they provide common descent directions for all objectives without requiring predefined parameters. Besides, this type of method enjoys an attractive property, as pointed out by Attouch et al. [4], in fields like game theory, economics, social science, and management: *it improves each of the objective functions.* As far as we know, the study of multiobjective gradient descent methods can be traced back to the pioneering works by Mukai [28] and Fliege and Svaiter [15]. The latter elucidated that the multiobjective steepest descent direction reduces to the steepest descent direction when dealing with a single objective. This observation inspired researchers to extend ordinary numerical algorithms for solving MOPs (see, e.g., [2, 7, 9, 14, 16, 19, 22, 26, 27, 29, 30] and references therein).

Although multiobjective gradient descent methods are derived from their single-objective counterparts, a theoretical gap remains between the two types of approaches. Recently, Zeng et al. [40] and Fliege et al. [17] established a linear convergence rate $\mathcal{O}(r^k)$ of the steepest descent method for MOPs (SDMO), where $r = \sqrt{1 - \mu_{\min}/L_{\max}}$, $\mu_{\min} := \min\{\mu_i : i = 1, 2, \cdots, m\}$ and $L_{\max} := \max\{L_i : i = 1, 2, \cdots, m\}$, with $\mu_{\min}$ and L_i representing the strongly convex and smooth parameters, respectively. Tanabe et al. [36] obtained a similar result for the proximal gradient method for MOPs (PGMO) [34]. Notably, when minimizing a μ-strongly convex and L-smooth function using the vanilla gradient method, the rate of convergence in terms of $\{\|x^k - x^*\|\}$ is $\sqrt{1 - \mu/L}$. This highlights that *objective imbalances*, stemming from the substantially distinct curvature information of different objective functions, can result in a small value of $\mu_{\min}/L_{\max}$. Remarkably, even if each of the objective functions is not ill-conditioned (a relatively small L_i/μ_i), the *imbalanced condition number* $L_{\max}/\mu_{\min}$ can be tremendous. This theoretical gap between first-order methods for single-objective optimization problems (SOPs) and MOPs elucidates why each objective is relatively easy to optimize individually but challenging when attempting to optimize them simultaneously. We emphasize that the objective imbalances are intrinsic to MOPs, particularly in large-scale and real-world scenarios, imposing significant challenges for solving MOPs via existing first-order methods [15, 19, 22, 34, 35]. This raises a critical question: How can the theoretical gap between first-order methods for SOPs and MOPs be bridged?

In this paper, we consider the question with the following generic model of unconstrained multiobjective composite optimization problem:

$$\min_{x\in\mathbb{R}^n} F(x), \qquad\qquad\text{(MCOP)}$$

where $F : \mathbb{R}^n \to (\mathbb{R} \cup +\{\infty\})^m$ is a vector-valued function. Each component F_i, $i = 1, 2, \cdots, m$, is defined by

$$F_i := f_i + g_i,$$

with convex and continuously differentiable function $f_i : \mathbb{R}^n \to \mathbb{R}$ and proper convex and lower semicontinuous but not necessarily differentiable function $g_i : \mathbb{R}^n \to \mathbb{R} \cup +\{\infty\}$. This type of problem finds wide applications in machine learning and statistics, and gradient descent methods tailored for it have received increasing attention (see, e.g., [1, 3, 6, 8, 35]). To address the aforementioned challenge, we incorporate curvature information to scale each objective within the direction-finding subproblem and propose a scaled proximal gradient method for multiobjective optimization (SPGMO). We analyze the convergence rates of SPGMO and offer new theoretical insights, helping to explain its observed rapid convergence in practical applications. The primary contributions of this paper are summarized as follows:

(i) When the smooth parameters are known, we apply them to scale each of objectives in the SPGMO and prove that its rate of convergence in terms of $\|x^k - x^*\|$ is $\sqrt{1 - \min_{i=1,2,\cdots,m}\{\mu_i/L_i\}}$. The improved linear convergence bridges the theoretical gap between first-order methods for SOPs and MOPs, providing a theoretical basis for the superior performance of SPGMO over PGMO. Additionally, we establish improved linear convergence for SPGMO when applied to MOPs with both linear and strongly convex objectives. The convergence rate primarily depends on the strongly convex objectives, with the influence of linear objectives mitigated through small scaling. To our knowledge, this is the first result demonstrating linear convergence for descent methods on this class of problems.

(ii) When the smooth parameters are unknown, we investigate the selection of scaling parameters in SPGMO with line search. We demonstrate that these scaling parameters are crucial for the convergence rate of SPGMO with line search. Theoretical results indicate that optimal linear convergence is achieved by choosing either $\{\mu_i : i = 1, 2, ..., m\}$ or $\{L_i : i = 1, 2, ..., m\}$ as the scaling parameters. Notably, in practical applications, it is advisable to select each scaling parameter from the interval $[\mu_i, L_i]$ to better capture the local geometry of the problem. Consequently, the Barzilai-Borwein method [5, 10] emerges as a judicious choice for scaling parameters.

(iii) We propose an accelerated SPGMO (ASPGMO) which unifies the convex and strongly convex cases by employing different momentum parameters. In the context of convex problems, the ASPGMO recovers the well-known sub-linear convergence rate $\mathcal{O}(1/k^2)$, a result initially rigorously demonstrated by Tanabe et al. [35] for MOPs, with more recent advancements discussed in [32, 33]. For strongly convex problems, we demonstrate that ASPGMO, with an appropriately chosen momentum parameter, converges linearly at a rate of $1 - \sqrt{\min_{i=1,2,\cdots,m}\{\mu_i/L_i\}}$.

The remainder of the paper is organized as follows. In section 2, we present some necessary notations and definitions that will be used later. Section 3 recalls the PGMO [34]. In section

4, we propose a scaled proximal gradient method for MOPs with known smooth parameters and establish an improved linear convergence for strongly convex cases. Section 5 is devoted to a scaled proximal gradient method for MOPs with unknown smooth parameters and discusses the selection of scaling parameters in the proposed approach. In section 6, we incorporate Nesterov's acceleration technique into SPGMO and analyze its convergence rates for both convex and strongly convex scenarios. The numerical experiments are presented in section 7, demonstrating the efficiency of the SPGMO and validating the theoretical results. Finally, we draw some conclusions at the end of the paper.

2 Preliminaries

Throughout this paper, the n-dimensional Euclidean space $\mathbb{R}^n$ is equipped with the inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\| \cdot \|$. We denote by $Jf(x) \in \mathbb{R}^{m \times n}$ the Jacobian matrix of f at x, by $\nabla f_i(x) \in \mathbb{R}^n$ the gradient of f_i at x. We denote $[m] := \{1, 2, \cdots, m\}$, and $\Delta_m := \{\lambda \in \mathbb{R}^m_+ : \sum_{i \in [m]} \lambda_i = 1\}$ the m-dimensional unit simplex. We define order relations $\preceq$ and $\prec$ in $\mathbb{R}^m$ as

$$u \preceq v \ \Leftrightarrow \ v - u \in \mathbb{R}^m_+$$

and

$$u \prec v \ \Leftrightarrow \ v - u \in \mathbb{R}^m_{++},$$

respectively. We denote the level set of F on $c \in \mathbb{R}^m$ as

$$\mathcal{L}_F(c) := \{x : F(x) \preceq c\}.$$

Next, we introduce optimality concepts for (MCOP) in the Pareto sense.

Definition 2.1 A vector $x^* \in \mathbb{R}^n$ is called Pareto solution to (MCOP), if there exists no $x \in \mathbb{R}^n$ such that $F(x) \preceq F(x^*)$ and $F(x) \neq F(x^*)$.

Definition 2.2 A vector $x^* \in \mathbb{R}^n$ is called weakly Pareto solution to (MCOP), if there exists no $x \in \mathbb{R}^n$ such that $F(x) \prec F(x^*)$.

Definition 2.3 A differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is L-smooth if

$$h(y) \leq h(x) + \langle \nabla h(x), y - x \rangle + \frac{L}{2} \|y - x\|^2$$

holds for all $x, y \in \mathbb{R}^n$. And h is μ-strongly convex if

$$h(y) \geq h(x) + \langle \nabla h(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2$$

holds for all $x, y \in \mathbb{R}^n$.

3 Proximal gradient method for MCOPs

For the (MCOP), Tanabe et al. [34] proposed the following proximal gradient subproblem:

$$\min_{x \in \mathbb{R}^n} \max_{i \in [m]} \left\{ \langle \nabla f_i(x^k), d \rangle + g_i(x^k + d) - g_i(x^k) + \frac{\ell}{2} \|d\|^2 \right\}, \tag{1}$$

where $\ell > 0$. Assume that f_i is L_i-smooth for $i \in [m]$, denote $L_{\max} := \max\{L_i : i \in [m]\}$. When $\{L_1, \cdots, L_m\}$ is available, the complete proximal gradient method for MCOPs is described as follows.

Algorithm 1: `Proximal_gradient_method_for_MCOPs` [34]

Require: $x^0 \in \mathbb{R}^n$, $\ell \geq L_{\max}$
1: **for** $k = 0, \dots$ **do**
2:     Compute d_ℓ^k by solving subproblem (1)
3:     **if** $d_\ell^k = 0$ **then**
4:         **return** x^k
5:     **else**
6:         Update $x^{k+1} := x^k + d_\ell^k$
7:     **end if**
8: **end for**

When $m = 1$, Algorithm 1 reduces to the proximal gradient method for SOPs, it is known that $\{x^k\}$ converges linearly to the unique minimizer when f_1 is strongly convex with $\mu_1 > 0$. Tanabe et al. [36] also established the linear convergence for Algorithm 1.

Theorem 3.1 (See Theorem 5.3 of [36]) *Assume that f_i is strongly convex with module $\mu_i > 0$ and L_i-smooth for $i \in [m]$, let $\{x^k\}$ be the sequence generated by Algorithm 1. Then there exists a Pareto solution x^* such that*

$$\left\| x^{k+1} - x^* \right\| \leq \sqrt{1 - \frac{\mu_{\min}}{L_{\max}}} \left\| x^k - x^* \right\|.$$

Although Algorithm 1 enjoys linear convergence in strongly convex cases, the rate of convergence can be very slow even if each of the objectives is well-conditioned. We clarify the statement with the following example.

Example 3.1 Let us consider the following bi-objective problem:

$$\min_{x \in \mathbb{R}^2} (F_1(x), F_2(x)),$$

where $f_1(x) = \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$, $f_2(x) = \frac{L}{2}x_1^2 + \frac{L}{2}x_2^2$ (L is a very large positive constant), $g_1(x) = g_2(x) = 0$. It is easy to see that $L_{\max} = L$, condition numbers of f_1 and f_2 are both 1, and the unique Pareto solution is $x^* = (0,0)^T$. For any noncritical point x^k, we derive that $x^{k+1} = (1 - 1/L)x^k$, i.e.,

$$\left\| x^{k+1} - x^* \right\| = \left(1 - \frac{1}{L} \right) \left\| x^k - x^* \right\|.$$

Recall that L is a very large positive constant, thus the rate of linear convergence is very slow.

4 Scaled proximal gradient method for MCOPs

In this section, we propose a scaled proximal gradient method for (MCOP) with improved linear convergence. Firstly, we emphasize that the slow convergence of Algorithm 1 is mainly due to the global upper bound used in (1):

$$F_i(x^k + d) - F_i(x^k) \leq \left\langle \nabla f_i(x^k), d \right\rangle + g_i(x^k + d) - g_i(x^k) + \frac{L_{\max}}{2} \|d\|^2$$

for all $i \in [m]$, which may be too conservative for objectives with small global smoothness parameters. Instead, we employ a separate global smoothness parameter for each objective to devise the following scaled proximal gradient subproblem:

$$\min_{x \in \mathbb{R}^n} \max_{i \in [m]} \left\{ \frac{\left\langle \nabla f_i(x^k), d \right\rangle + g_i(x^k + d) - g_i(x^k)}{L_i} + \frac{1}{2} \|d\|^2 \right\}. \tag{2}$$

This strategy can capture the curvature information of each objective and alleviate interference among the objectives. The complete scaled proximal gradient method for MCOPs (SPGMO) is described as follows.

Algorithm 2: `Scaled_proximal_gradient_method_for_MCOPs` [34]

Require: $x^0 \in \mathbb{R}^n$
1: **for** $k = 0, \dots$ **do**
2:     Compute d^k by solving subproblem (2)
3:     **if** $d^k = 0$ **then**
4:         **return** x^k
5:     **else**
6:         Update $x^{k+1} := x^k + d^k$
7:     **end if**
8: **end for**

Let us denote the following scaled multiobjective composite optimization problem:

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

where $F^L : \mathbb{R}^n \to (\mathbb{R} \cup \{+\infty\})^m$ is a vector-valued function. Each component F_i^L, $i \in [m]$, is defined by

$$F_i^L := \frac{f_i + g_i}{L_i}.$$

By scaling the objective functions, we note that Algorithm 2 for (MCOP) can be interpreted as Algorithm 1 for (MCOP$_L$). As a result, the convergence rates analysis for Algorithm 1 in [36] also applies to Algorithm 2. Before we proceed with the convergence analysis, we will first present several properties of (MCOP$_L$).

Proposition 4.1 *For the* (MCOP$_L$), *the following statements hold.*

(i) $x \in \mathbb{R}^n$ *is a Pareto critical point of* (MCOP$_L$) *if and only if* $x \in \mathbb{R}^n$ *is a Pareto critical point of* (MCOP).

(ii) *For* $i \in [m]$, *if* f_i *is* L_i-*smooth, then* f_i/L_i *is* 1-*smooth.*

(iii) *For* $i \in [m]$, *if* f_i *is strongly convex with* $\mu_i \geq 0$, *then* f_i/L_i *is strongly convex with* μ_i/L_i.

Proof The proof can be easily derived by the definition of F^L, we omit it here.

4.1 Improved linear convergence of SPGMO

Firstly, we establish a fundamental inequality.

Lemma 4.1 *Assume that* f_i *is strongly convex with modulus* $\mu_i \geq 0$ *and* L_i-*smooth for* $i \in [m]$. *Let* $\{x^k\}$ *be the sequence generated by Algorithm 2. Then, there exists* $\lambda^k \in \Delta_m$ *such that*

$$
\begin{aligned}
&\sum_{i \in [m]} \lambda_i^k \frac{F_i(x^{k+1}) - F_i(x)}{L_i} \\
&\leq \frac{1}{2}\|x^k - x\|^2 - \frac{1}{2}\|x^{k+1} - x\|^2 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{2L_i}\|x^k - x\|^2, \text{ for all } x \in \mathbb{R}^n.
\end{aligned}
\tag{3}
$$

Proof The assertions can be obtained by using similar arguments as in the proof of [37, Lemma 5.2]

Theorem 4.1 *Assume that* f_i *is strongly convex with modulus* $\mu_i > 0$ *and* L_i-*smooth for* $i \in [m]$. *Let* $\{x^k\}$ *be the sequence generated by Algorithm 2 and* x^* *be the Pareto solution satisfies* $F(x^*) \preceq F(x^k)$ *for all* $k \geq 0$. *Then we have*

$$
\|x^{k+1} - x^*\| \leq \sqrt{1 - \min_{i \in [m]} \left\{ \frac{\mu_i}{L_i} \right\}} \|x^k - x^*\|.
$$

Proof By substituting $x = x^*$ into inequality (3), we obtain

$$
\begin{aligned}
&\sum_{i \in [m]} \lambda_i^k \frac{F_i(x^{k+1}) - F_i(x^*)}{L_i} \\
&\leq \frac{1}{2}\|x^k - x^*\|^2 - \frac{1}{2}\|x^{k+1} - x^*\|^2 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{2L_i}\|x^k - x^*\|^2 \text{ for all } x \in \mathbb{R}^n.
\end{aligned}
$$

Applying $F(x^*) \preceq F(x^k)$, it follows that

$$
\|x^{k+1} - x^*\| \leq \sqrt{1 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_i}} \|x^k - x^*\| \leq \sqrt{1 - \min_{i \in [m]} \left\{ \frac{\mu_i}{L_i} \right\}} \|x^k - x^*\|,
\tag{4}
$$

where the last inequality holds due to the fact $\lambda^k \in \Delta_m$.

Remark 4.1 In the following, we present two remarks on the linear convergence results of the SPGMO and PGMO.

(i) When $m = 1$, both the SPGMO and the PGMO reduce to the proximal gradient method for SOPs. In cases where $m \neq 1$, the condition numbers $\tilde{\kappa} := L_{\max}/\mu_{\min}$ and $\kappa := \max_{i \in [m]}\{L_i/\mu_i\}$ are pivotal for the geometric convergence of PGMO and SPGMO, respectively. Notably, since $\tilde{\kappa} \geq \kappa$, the SPGMO exhibits improved linear convergence compared to the PGMO. Furthermore, the SPGMO exhibits rapid linear convergence provided that all differentiable components are not ill-conditioned. Conversely, in such scenarios, the PGMO may experience slow convergence due to objective imbalances (see Example 3.1).

(ii) Objective imbalances pose a significant challenge to existing first-order methods in multi-objective optimization, particularly in large-scale and real-world scenarios where objective imbalances are intrinsic to MOPs. To quantify the objective imbalances in strongly convex cases, we introduce the following parameter:

$$\zeta := \frac{\tilde{\kappa}}{\kappa}. \tag{5}$$

A larger value of this parameter indicates a greater degree of objective imbalances in the problem.

(iii) As outlined in Figure 4.1, for the function F^L we have $L_{\max} = 1$ and $\mu_{\min} = \min_{i \in [m]}\{\mu_i/L_i\}$. Consequently, Theorem 4.1 follows as a corollary of Theorem 3.1, resulting in an objective imbalances parameter ζ of F^L is 1.

4.2 Linear convergence with some linear objective functions

In addition to strongly convex cases, linear objectives often introduce significant objective imbalances in MOPs, decelerating the convergence of first-order methods [10, Examples 2 and 3]. In what follows, we confirm that a small scaling parameter is beneficial to mitigate the influence of the linear objectives from a theoretical perspective. Consider the linear constrained MOP with linear objective functions, which is described as follows:

$$\min_{x \in \mathcal{X}} f(x), \tag{LCMOP}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}^m$ is a vector-valued function; the component f_i is linear for $i \in \mathcal{L}$, and μ_i-strongly convex and L_i-smooth for $i \in [m] \setminus \mathcal{L}$, respectively; $\mathcal{X} = \{x : Ax \leq a,\ Bx = b\}$ with $A \in \mathbb{R}^{|\mathcal{J}| \times n}$, $B \in \mathbb{R}^{|\mathcal{E}| \times n}$. This type of problem has wide applications in portfolio selection [24] and can be reformulated as (MCOP) with $F_i = f_i + \mathbb{I}_{\mathcal{X}}$ for $i \in [m]$, where

$$\mathbb{I}_{\mathcal{X}}(x) := \begin{cases} 0, & x \in \mathcal{X}, \\ +\infty, & x \notin \mathcal{X}. \end{cases}$$

Regarding the first inequality in (4), the linear convergence rate in terms of $\|x^k - x^*\|$ is actually $\sqrt{1 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_i}}$ for Algorithm 2. Consequently, Algorithm 2 exhibits linear convergence as long as $\sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_i} > 0$. A similar statement holds for PGMO. However, such a statement holds under the restrictive condition $\sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_i} > 0$, which is equivalent to $\sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k > 0$. In what follows, we will prove that a small value of L_i for $i \in \mathcal{L}$ leads to larger value of $\sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k$ in SPGMO.

When minimizing (LCMOP) using SPGMO, the subproblem (2) can be expressed as follows:

$$
\begin{aligned}
\min \quad & t + \frac{1}{2}\|d\|^2 \\
\text{s.t.} \quad & \left\langle \frac{\nabla f_i(x^k)}{L_i}, d \right\rangle \le t, \ i \in [m], \\
& A(x^k + d) \le a, \\
& Bd = 0,
\end{aligned}
$$

where $L_i = \alpha_{\min}$ for $i \in \mathcal{L}$, and $\alpha_{\min}$ is a very small positive constant[1]. By KKT conditions, we obtain

$$
d^k = -\left(\sum_{i \in [m]} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} + \sum_{j \in \mathcal{J}} \eta_j^k A_j + \sum_{e \in \mathcal{E}} \xi_e^k B_e \right),
$$

where A_j^T and B_e^T is the j-th and e-th row of A and B, respectively. The vector $(\lambda^k, \eta^k, \xi^k) \in \mathbb{R}^{m+|\mathcal{J}|+|\mathcal{E}|}$ is a solution of the following Lagrangian dual problem:

$$
\begin{aligned}
\min \quad & \frac{1}{2}\left\| \sum_{i \in [m]} \lambda_i \frac{\nabla f_i(x^k)}{L_i} + \sum_{j \in \mathcal{J}} \eta_j A_j + \sum_{e \in \mathcal{E}} \xi_e B_e \right\|^2 - \sum_{j \in \mathcal{J}} \eta_j \left(\langle A_j, x^k \rangle - a_j \right) \\
\text{s.t.} \quad & (\lambda, \eta, \xi) \in \Delta_m \times \mathbb{R}_+^{|\mathcal{J}|} \times \mathbb{R}^{|\mathcal{E}|}.
\end{aligned}
$$

And complementary slackness condition gives that

$$
\eta_j^k \left(\langle A_j, x^k + d^k \rangle - a_j \right) = 0 \text{ for all } j \in \mathcal{J}. \tag{6}
$$

Denote $\mathcal{J}^k := \{ j \in \mathcal{J} : \langle A_j, x^k \rangle = a_j \}$, this together with (6) and the fact that $x^{k+1} = x^k + d^k$ implies

$$
d^k = -\left(\sum_{i \in [m]} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} + \sum_{j \in \mathcal{J}^{k+1}} \eta_j^k A_j + \sum_{e \in \mathcal{E}} \xi_e^k B_e \right).
$$

---

[1] The upper bound $f_i(y) \le f_i(x) + \langle \nabla f_i(x), y - x \rangle \le L_i \|y - x\|^2$ holds for $i \in \mathcal{L}$ with any $L_i \ge 0$,.

Before presenting the linear convergence of SPGMO for (LCMOP), let us first define the following multiobjective linear programming problem:

$$\min_{x \in \mathcal{X}} f_{\mathcal{L}}(x), \tag{MLP}$$

where $f_{\mathcal{L}} : \mathbb{R}^n \to \mathbb{R}^{|\mathcal{L}|}$ is the linear component of f. As a result, every weakly Pareto solution of (MLP) is also a weakly Pareto solution of (LCMOP).

Proposition 4.2 *Let $\{x^k\}$ be the sequence generated by Algorithm 2 with $L_i = \alpha_{\min}$, $i \in \mathcal{L}$ for (LCMOP) and x^* be the weakly Pareto solution satisfies $F(x^*) \preceq F(x^k)$ for all $k \geq 0$. If x^* is not a weakly Pareto solution of (MLP), then*

$$\|x^{k+1} - x^*\| \leq \sqrt{1 - (1 - c_1 \alpha_{\min}) \min_{i \in [m] \setminus \mathcal{L}} \left\{ \frac{\mu_i}{L_i} \right\}} \|x^k - x^*\|,$$

where $c_1 := \max_{i \in [m] \setminus \mathcal{L}} \frac{2(\|\nabla f_i(x^0)\| + L_i R)}{\epsilon L_i}$.

Proof We refer to Theorem 4.1(ii), which states

$$\|x^{k+1} - x^*\| \leq \sqrt{1 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_i}} \|x^k - x^*\|. \tag{7}$$

Given that x^* is not a weakly Pareto solution of (MLP), we deduce $0 \notin C^k := \{ \sum_{i \in \mathcal{L}} \lambda_i g_i + \sum_{j \in \mathcal{J}^k} \eta_j A_j + \sum_{e \in \mathcal{E}} \xi_e B_e : \lambda \in \Delta_{|\mathcal{L}|}, \eta \in \mathbb{R}_+^{|\mathcal{J}^k|}, \xi \in \mathbb{R}^{|\mathcal{E}|} \}$, where g_i represents the gradient of the linear function f_i, $i \in \mathcal{L}$. Furthermore, since $\mathcal{J}^k \subset \mathcal{J}$, we can infer that $|\{C^k\}|$ is finite. Consequently, we can define $\epsilon := \min_{x \in \{C^k\}} \|x\| > 0$. By direct calculation, we have

$$
\begin{aligned}
\|d^k\| &= \left\| \sum_{i \in [m]} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} + \sum_{j \in \mathcal{J}^{k+1}} \eta_j^k A_j + \sum_{e \in \mathcal{E}} \xi_e^k B_e \right\| \\
&\geq \left\| \sum_{i \in \mathcal{L}} \lambda_i^k \frac{g_i}{L_i} + \sum_{j \in \mathcal{J}^{k+1}} \eta_j^k A_j + \sum_{e \in \mathcal{E}} \xi_e^k B_e \right\| - \left\| \sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} \right\| \\
&= \frac{\sum_{i \in \mathcal{L}} \lambda_i^k}{\alpha_{\min}} \left\| \sum_{i \in \mathcal{L}} \frac{\lambda_i^k}{\sum_{i \in \mathcal{L}} \lambda_i^k} g_i + \sum_{j \in \mathcal{J}^{k+1}} \frac{\alpha_{\min} \eta_j^k}{\sum_{i \in \mathcal{L}} \lambda_i^k} A_j + \sum_{e \in \mathcal{E}} \frac{\alpha_{\min} \xi_e^k}{\sum_{i \in \mathcal{L}} \lambda_i^k} B_e \right\| - \left\| \sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} \right\| \\
&\geq \frac{\epsilon \sum_{i \in \mathcal{L}} \lambda_i^k}{\alpha_{\min}} - \left\| \sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} \right\|,
\end{aligned}
\tag{8}
$$

where the second equality follows from the fact that $L_i = \alpha_{\min}$ for all $i \in \mathcal{L}$, and the last inequality is given by the definition of ϵ. By simple calculation, we have

$$
\begin{aligned}
\frac{1}{2}\|d^k\|^2 &= \frac{1}{2}\left\| \sum_{i\in[m]} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} + \sum_{j\in\mathcal{J}} \eta_j^k A_j + \sum_{e\in\mathcal{E}} \xi_e^k B_e \right\|^2 \\
&\leq \frac{1}{2}\left\| \sum_{i\in[m]} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} + \sum_{j\in\mathcal{J}} \eta_j^k A_j + \sum_{e\in\mathcal{E}} \xi_e^k B_e \right\|^2 - \sum_{j\in\mathcal{J}} \eta_j^k \left(\langle A_j, x^k \rangle - a_j \right) \\
&\leq \frac{1}{2} \max_{i\in[m]\setminus\mathcal{L}} \left\| \frac{\nabla f_i(x^k)}{L_i} \right\|^2,
\end{aligned}
$$

where the second inequality follows by the non-negativeness of η_j^k and $a_j - \langle A_j, x^k \rangle$, and the last inequality is due to the fact that $(\lambda^k, \eta^k, \xi^k)$ is a solution of the dual problem. This together with (8) implies

$$
\frac{\epsilon \sum_{i\in\mathcal{L}} \lambda_i^k}{\alpha_{\min}} \leq \max_{i\in[m]\setminus\mathcal{L}} \left\| \frac{\nabla f_i(x^k)}{L_i} \right\| + \left\| \sum_{i\in[m]\setminus\mathcal{L}} \lambda_i^k \frac{\nabla f_i(x^k)}{L_i} \right\| \leq 2 \max_{i\in[m]\setminus\mathcal{L}} \left\| \frac{\nabla f_i(x^k)}{L_i} \right\|.
$$

Notice that f_i is strongly convex for $i \in [m] \setminus \mathcal{L}$, then we deduce that $\mathcal{L}_F(F(x^0))$ is bounded. Denoting $R := \max\{\|x - y\| : x, y \in \mathcal{L}_F(F(x^0))\}$, and utilizing the L_i-smoothness of $f_i, i \in [m] \setminus \mathcal{L}$, we derive an upper bound of $\|\nabla f_i(x^k)\|$:

$$
\|\nabla f_i(x^k)\| \leq \|\nabla f_i(x^0)\| + \|\nabla f_i(x^k) - \nabla f_i(x^0)\| \leq \|\nabla f_i(x^0)\| + L_i R.
$$

Therefore, we obtain

$$
\sum_{i\in\mathcal{L}} \lambda_i^k \leq \max_{i\in[m]\setminus\mathcal{L}} \frac{2(\|\nabla f_i(x^0)\| + L_i R)}{\epsilon L_i} \alpha_{\min}. \tag{9}
$$

Substituting the preceding bound into (7), the desired result follows.

Remark 4.2 From Proposition 4.2, the sum of dual variables of linear objectives $\sum_{i\in\mathcal{L}} \lambda_i^k$ is dominated by $\alpha_{\min}$. Consequently, the impact of the linear objectives can be effectively mitigated by employing a small scaling parameter $\alpha_{\min}$ in the direction-finding subproblems.

Remark 4.3 The assumption x^* is not a weakly Pareto solution of (MLP) seems restrictive. In practice, the Pareto set of (LCMOP) can be a $(m-1)$-dimensional manifold, and the Pareto set of (MLP) can be a $(|\mathcal{L}|-1)$-dimensional sub-manifold within the $(m-1)$-dimensional manifold. As a result, for a random initial point x^0, the probability that x^* is not a weakly Pareto solution of (MLP) can be 1. Additionally, when $|\mathcal{J}| = 0$, meaning (MLP) has no inequality constraints, then the problem either has no weakly Pareto solution or every feasible point is a weakly Pareto solution. In other words, SPGMO converges linearly for equality constrained and unconstrained (LCMOP).

In the following, we also analyze the convergence rate of PGMO for (LCMOP).

Proposition 4.3 *Let $\{x^k\}$ be the sequence generated by Algorithm 1 for (LCMOP) and x^* be the weakly Pareto solution satisfies $F(x^*) \preceq F(x^k)$ for all $k \geq 0$. If x^* is not a weakly Pareto solution of (MLP), then there exists $\kappa \in (0,1)$ such that $\|x^k - x^*\| \leq \kappa^k \|x^0 - x^*\|$.*

Proof Setting $L_i = L_{\max}$ in SPGMO, it coincides with PGMO. Consequently, (7) collapses to

$$\|x^{k+1} - x^*\| \leq \sqrt{1 - \sum_{i \in [m]} \lambda_i^k \frac{\mu_i}{L_{\max}}} \|x^k - x^*\|. \tag{10}$$

Substituting $L_i = L_{\max}$ into (8), it follows that

$$\epsilon \sum_{i \in \mathcal{L}} \lambda_i^k \leq \left\| \sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k \nabla f_i(x^k) \right\| + L_{\max} \|d^k\|$$

$$\leq \left(\sum_{i \in [m] \setminus \mathcal{L}} \lambda_i^k \right) \max_{i \in [m] \setminus \mathcal{L}} \{\|\nabla f_i(x^0)\| + L_i R\} + L_{\max} \|d^k\|.$$

Rearranging the above inequality, we have

$$\sum_{i \in \mathcal{L}} \lambda_i^k \leq \frac{\max\limits_{i \in [m] \setminus \mathcal{L}} \{\|\nabla f_i(x^0)\| + L_i R\} + L_{\max} \|d^k\|}{\max\limits_{i \in [m] \setminus \mathcal{L}} \{\|\nabla f_i(x^0)\| + L_i R\} + \epsilon}.$$

Since $\{x^k\}$ converges to a weakly Pareto point, it follows that $d^k \to 0$. Then there exists $K > 0$ such that $\|d^k\| \leq \frac{\epsilon}{2L_{\max}}$ for all $k \geq K$. This implies that

$$\sum_{i \in \mathcal{L}} \lambda_i^k \leq c_2 \text{ for all } k \geq K,$$

where $c_2 := \frac{\max\limits_{i \in [m] \setminus \mathcal{L}} \{\|\nabla f_i(x^0)\| + L_i R\} + \frac{\epsilon}{2}}{\max\limits_{i \in [m] \setminus \mathcal{L}} \{\|\nabla f_i(x^0)\| + L_i R\} + \epsilon} < 1$. By substituting the above bound into (10), we obtain

$$\|x^{k+1} - x^*\| \leq \sqrt{1 - (1 - c_2) \frac{\min\limits_{i \in [m] \setminus \mathcal{L}} \mu_i}{L_{\max}}} \|x^k - x^*\| \text{ for all } k \geq K.$$

Without loss of generality, there exists $\kappa \in (0,1)$ such that $\|x^k - x^*\| \leq \kappa^k \|x^0 - x^*\|$.

From Proposition 4.3, PGMO only achieves R-linear convergence. The following example illustrates that $\sum_{i \in \mathcal{L}} \lambda_i^k = 1$ at the early stage in the iterations. In other words, we can not obtain Q-linear convergence of the method for (LCMOP). Beside, the rate of R-linear convergence can be very slow.

Example 4.1 Consider the following multiobjective optimization problem:

$$\min_{x \in \mathcal{X}} (f_1(x), f_2(x)),$$

where $f_1(x) = \frac{1}{2}x_1^2 + \frac{1}{2}x_2^2$, $f_2(x) = cx_1$ (c is a relative small positive constant), and $\mathcal{X} = \{(x_1, x_2) : x_1 \geq 0, x_2 = 0\}$. By simple calculations, we have

$$\nabla f_1(x) = (x_1, x_2)^T, \ \nabla f_2(x) = (c, 0)^T, \ L_{\max} = 1$$

and the unique Pareto solution is $(0,0)^T$. Given a feasible x^0, at the early stage ($x_1^k > c$) of PGMO without line search, we have $\lambda_2^k = 1$, and $d^k = (-c, 0)$. At this stage, we have $\|x^{k+1} - x^*\|/\|x^k - x^*\| = (x_1^k - c)/x_1^k$, which tends to 1 for sufficient small c.

5 Scaled proximal gradient method with unknown smoothness parameters for MCOPs

In the previous section, global smoothness parameters were employed to scale each objective; however, these parameters are often unknown and can be overly conservative. This section is devoted to a scaled proximal gradient method with unknown smoothness parameters.

5.1 Estimating the local smoothness parameters

Firstly, we propose a backtracking method to estimate the local smoothness parameters.

Algorithm 3: Backtracking

Require: $0 < \alpha_i^k \leq L_i, s_i = 0, i \in [m], \tau > 1$
1: **repeat**
2:     Update $\alpha_i^k = \tau^{s_i}\alpha_i^k, i \in [m]$
3:     Update $x^{k+1} := \arg\min\limits_{x \in \mathbb{R}^n} \max\limits_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x^k), x - x^k \rangle + g_i(x) - g_i(x^k)}{\alpha_i^k} + \frac{1}{2}\|x - x^k\|^2 \right\}$
4:     **for** $i = 1, \cdots, m$ **do**
5:         **if** $f_i(x^{k+1}) - f_i(x^k) > \langle \nabla f_i(x^k), x^{k+1} - x^k \rangle + \frac{\alpha_i^k}{2}\|x^{k+1} - x^k\|^2$ **then**
6:             Update $s_i = s_i + 1$
7:         **end if**
8:     **end for**
9: **until** $f_i(x^{k+1}) - f_i(x^k) \leq \langle \nabla f_i(x^k), x^{k+1} - x^k \rangle + \frac{\alpha_i^k}{2}\|x^{k+1} - x^k\|^2, \ i \in [m]$

Since f_i is L_i-smooth for $i \in [m]$, the *repeat loop* terminates in a finite number of iterations, and $\alpha_i^k < \tau L_i, \ i \in [m]$. A notable advantage of this approach is its ability to adapt to the local smoothness based on the trajectory of the iterations. However, the use of backtracking increases the computational cost per iteration due to the repeated solving of subproblems.

5.2 Line search

To avoid solving a subproblem in each backtracking procedure, we apply the Armijo line search proposed by Tanabe et al. [34].

Algorithm 4: `Armijo_line_search`

Require: $x^k \in \mathbb{R}^n, d^k \in \mathbb{R}^n, Jf(x^k) \in \mathbb{R}^{m \times n}, \sigma \in (0,1), t_k = 1$
1: **while** $F(x^k + t_k d^k) - F(x^k) \not\preceq t_k \sigma(Jf(x^k)d^k + g(x^k + d^k) - g(x^k))$ **do**
2:      Update $t_k := t_k/2$
3: **end while**
4: **return** t_k

The scaled proximal gradient method with line search for MCOPs is described as follows.

Algorithm 5: `Scaled_proximal_gradient_method_with_line_search_for_MCOPs`

Require: $x^0 \in \mathbb{R}^n$
1: **for** $k = 0, \cdots$ **do**
2:      Update $\alpha^k \in \mathbb{R}_{++}^m$
3:      Update $d^k := \arg\min\limits_{x \in \mathbb{R}^n} \max\limits_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x^k), d \rangle + g_i(x^k + d) - g_i(x^k)}{\alpha_i^k} + \frac{1}{2}\|d\|^2 \right\}$
4:      **if** $d^k = 0$ **then**
5:          **return** x^k
6:      **else**
7:          Update $t_k := $ `Armijo_line_search`$(x^k, d^k, Jf(x^k))$
8:          Update $x^{k+1} := x^k + t_k d^k$
9:      **end if**
10: **end for**

It is worth noting that we don't specify how to select α^k in Algorithm 5. The direct question arises: Does α^k affect the performance of Algorithm 5? In the following, we will examine the role that α^k plays in the Algorithm 5.

Firstly, we prove that d^k is a descent direction for F.

Lemma 5.1 *For the d^k in Algorithm 5, we have*

$$\langle \nabla f_i(x^k), d^k \rangle + g_i(x^k + d^k) - g_i(x^k) \leq -\alpha_i^k \|d^k\|^2 \text{ for all } i \in [m]. \tag{11}$$

Proof The assertion can be obtained by using the same arguments as in the proof of [34, Lemma 4.1].

Next, we give a lower bound of stepsize in each iteration.

Lemma 5.2 *Assume f_i is L_i-smooth for $i \in [m]$, then the k-th stepsize generated by Algorithm 4 satisfies $t_k \geq t_k^{\min} := \min\left\{\min_{i \in [m]}\{(1 - \sigma)\alpha_i^k/L_i\}, 1\right\}$.*

Proof It is sufficient to prove $t_k < 1$, then backtracking is conducted, leading to the inequality:

$$F_i\left(x^k + 2t_k d^k\right) - F_i(x^k) > 2\sigma t_k(\langle \nabla f_i(x^k), d^k \rangle + g_i(x^k + d^k) - g_i(x^k)) \tag{12}$$

for some $i \in [m]$. Since f_i is L_i-smooth for $i \in [m]$, we can derive the following inequalities:

$$
\begin{aligned}
F_i &\left(x^k + 2t_k d^k \right) - F_i(x^k) \\
&\leq 2t_k \left\langle \nabla f_i(x^k), d^k \right\rangle + g_i(x^k + 2t_k d^k) - g_i(x^k) + \frac{L_i}{2} \left\| 2t_k d^k \right\|^2 \\
&\leq 2t_k (\left\langle \nabla f_i(x^k), d^k \right\rangle + g_i(x^k + d^k) - g_i(x^k)) + \frac{L_i}{2} \left\| 2t_k d^k \right\|^2,
\end{aligned}
\tag{13}
$$

where the second inequality follows from the convexity of g_i and the fact that $2t_k \in (0, 1]$. Combining this inequality with (12), we obtain

$$
(\sigma - 1)(\left\langle \nabla f_i(x^k), d^k \right\rangle + g_i(x^k + d^k) - g_i(x^k)) \leq L_i t_k \left\| d^k \right\|^2
$$

for some $i \in [m]$. Utilizing (11), we arrive at

$$
t_k \geq \frac{(1 - \sigma)\alpha_i^k}{L_i}
\tag{14}
$$

for some $i \in [m]$, it holds that $t_k \geq t_{\min}$. This completes the proof.

Before presenting the convergence analysis of Algorithm 5, we introduce two types of merit functions for (MCOP) that quantify the gap between the current point and the optimal solution.

$$
u_0^\alpha(x) := \sup_{y \in \mathbb{R}^n} \min_{i \in [m]} \left\{ \frac{F_i(x) - F_i(y)}{\alpha_i} \right\},
\tag{15}
$$

$$
w_\ell^\alpha(x) := \max_{y \in \mathbb{R}^n} \min_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x), x - y \rangle + g_i(x) - g_i(y)}{\alpha_i} - \frac{\ell}{2} \|x - y\|^2 \right\},
\tag{16}
$$

where $\alpha \in \mathbb{R}_{++}^m$, $\ell > 0$.

We can demonstrate that u_0^α and w_ℓ^α serve as merit functions, satisfying the criteria of weak Pareto and critical points, respectively.

Proposition 5.1 *Let u_0^α and w_ℓ^α be defined as (15) and (16), respectively. Then, for all $\alpha \in \mathbb{R}_{++}^m$ and $\ell > 0$ the following statements hold.*

(i) *$x \in \mathbb{R}^n$ is a weak Pareto solution of (MCOP) if and only if $u_0^\alpha(x) = 0$.*
(ii) *$x \in \mathbb{R}^n$ is a Pareto critical point of (MCOP) if and only if $w_\ell^\alpha(x) = 0$.*

Proof The assertions (i) and (ii) can be obtained by using similar arguments as in the proofs of [37, Theorem 3.1] and [37, Theorem 3.9], respectively.

Proposition 5.2 *Let w_ℓ^α be defined as (16). If $0 < \ell \leq r$, then*

$$
w_r^\alpha(x) \leq w_\ell^\alpha(x) \leq \frac{r}{\ell} w_r^\alpha(x) \text{ for all } x \in \mathbb{R}^n.
$$

Proof The assertion can be obtained by using the similar arguments as in the proof of [37, Theorem 4.2].

We are now in a position to establish the linear convergence of Algorithm 5.

Theorem 5.1 *Assume that f_i is L_i-smooth and strongly convex with modulus $\mu_i > 0$, for $i \in [m]$. Let $\{x^k\}$ be the sequence generated by Algorithm 5. Then, we have*

$$u_0^{\alpha^k}(x^{k+1}) \leq \left(1 - \sigma t_k^{\min} r_k\right) u_0^{\alpha^k}(x^k)$$

for all $k \geq 0$, where $r_k := \min\{\min_{i \in [m]}\{\mu_i/\alpha_i^k\}, 1\}$.

Proof The Armijo line search satisfies

$$F_i(x^{k+1}) - F_i(x^k) \leq t_k \sigma(\langle \nabla f_i(x^k), d^k \rangle + g_i(x^k + d^k) - g_i(x^k)).$$

A direct calculation gives

$$
\begin{aligned}
&\frac{F_i(x^{k+1}) - F_i(x^k)}{\alpha_i^k} \\
&\leq t_k \sigma \left(\frac{\langle \nabla f_i(x^k), d^k \rangle + g_i(x^k + d^k) - g_i(x^k)}{\alpha_i^k} + \frac{1}{2}\|d^k\|^2 \right) \\
&\leq t_k \sigma \max_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x^k), d^k \rangle + g_i(x^k + d^k) - g_i(x^k)}{\alpha_i^k} + \frac{1}{2}\|d^k\|^2 \right\} \\
&= -t_k \sigma w_1^{\alpha^k}(x^k) \\
&\leq -t_k^{\min} \sigma w_1^{\alpha^k}(x^k).
\end{aligned}
\tag{17}
$$

Furthermore, due to the strong convexity of f_i for $i \in [m]$, we have

$$
\begin{aligned}
&\frac{F_i(x^k) - F_i(x)}{\alpha_i^k} \\
&\leq \frac{\langle \nabla f_i(x^k), x^k - x \rangle + g_i(x^k) - g_i(x)}{\alpha_i^k} - \frac{\min_{i \in [m]}\{\mu_i/\alpha_i^k\}}{2}\|x - x^k\|^2
\end{aligned}
\tag{18}
$$

for all $i \in [m]$ and $x \in \mathbb{R}^n$. Taking the supremum and minimum with respect to $x \in \mathbb{R}^n$ and $i \in [m]$ on both sides, respectively, we obtain

$$
\begin{aligned}
&\sup_{x \in \mathbb{R}^n} \min_{i \in [m]} \left\{ \frac{F_i(x^k) - F_i(x)}{\alpha_i^k} \right\} \\
&\leq \sup_{x \in \mathbb{R}^n} \min_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x^k), x^k - x \rangle + g_i(x^k) - g_i(x)}{\alpha_i^k} - \frac{\min_{i \in [m]}\{\mu_i/\alpha_i^k\}}{2}\|x - x^k\|^2 \right\} \\
&\leq \frac{1}{r_k} \sup_{x \in \mathbb{R}^n} \min_{i \in [m]} \left\{ \frac{\langle \nabla f_i(x^k), x^k - x \rangle + g_i(x^k) - g_i(x)}{\alpha_i^k} - \frac{1}{2}\|x - x^k\|^2 \right\},
\end{aligned}
$$

where the second inequality is due to Proposition 5.2. The above inequalities can be rewritten as

$$u_0^{\alpha^k}(x^k) \leq \frac{1}{r_k} w_1^{\alpha^k}(x^k).$$

Together with (17), the preceding inequality yields

$$\frac{F_i(x^{k+1}) - F_i(x^k)}{\alpha_i^k} \leq -\sigma t_k^{\min} r_k u_0^{\alpha^k}(x^k).$$

Then, for all $x \in \mathbb{R}^n$, we have

$$\frac{F_i(x^{k+1}) - F_i(x)}{\alpha_i^k} \leq \frac{F_i(x^k) - F_i(x)}{\alpha_i^k} - \sigma t_k^{\min} r_k u_0^{\alpha^k}(x^k).$$

Taking the supremum and minimum with respect to $x \in \mathbb{R}^n$ and $i \in [m]$ on both sides, respectively, we obtain

$$u_0^{\alpha^k}(x^{k+1}) \leq \left(1 - \sigma t_k^{\min} r_k\right) u_0^{\alpha^k}(x^k).$$

This completes the proof.

The following theorem shows that $\alpha^k \in \mathbb{R}_{++}^m$ plays a significant role in the convergence rate of Algorithm 5.

Theorem 5.2 *Assume that f_i is L_i-smooth and strongly convex with modulus $\mu_i > 0$, for $i \in [m]$. Let $\{x^k\}$ be the sequence generated by Algorithm 5. Then, the following statements hold.*

(i) *For any $\alpha^k \in \mathbb{R}_{++}$, we have*

$$(1-\sigma) \min\left\{\frac{\alpha_{\min}^k \mu_{\min}}{\alpha_{\max}^k L_{\max}}, \frac{\alpha_{\min}^k}{L_{\max}}, \frac{\mu_{\min}}{\alpha_{\max}^k}\right\} \leq t_k^{\min} r_k \leq (1-\sigma) \min_{i \in [m]}\left\{\frac{\mu_i}{L_i}\right\},$$

where $\alpha_{\min}^k := \min\{\alpha_i^k, i \in [m]\}$ and $\alpha_{\max}^k := \max\{\alpha_i^k, i \in [m]\}$.

(ii) *If $\alpha_i^k = \ell > 0$ for $i \in [m]$, we have $t_k^{\min} r_k = (1-\sigma) \min\left\{\frac{\mu_{\min}}{L_{\max}}, \frac{\ell}{L_{\max}}, \frac{\mu_{\min}}{\ell}\right\}$, then*

$$u_0(x^{k+1}) \leq \left(1 - \sigma(1-\sigma) \min\left\{\frac{\mu_{\min}}{L_{\max}}, \frac{\ell}{L_{\max}}, \frac{\mu_{\min}}{\ell}\right\}\right) u_0(x^k).$$

(iii) *If $\alpha_i^k = \mu_i$ for $i \in [m]$ or $\alpha_i^k = L_i$ for $i \in [m]$, we have $t_k^{\min} r_k = (1-\sigma) \min_{i \in [m]} \mu_i/L_i$, then*

$$u_0^{\alpha^k}(x^{k+1}) \leq \left(1 - \sigma(1-\sigma) \min_{i \in [m]}\left\{\frac{\mu_i}{L_i}\right\}\right) u_0^{\alpha^k}(x^k).$$

Proof (i) Firstly, we prove the right hand side of the inequality. With loss of generality, let us assume that

$$\frac{\mu_1}{L_1} = \min_{i \in [m]}\left\{\frac{\mu_i}{L_i}\right\}.$$

Recall the definition of $t_k^{\min}$ and r_k, we have $t_k^{\min} \leq (1-\sigma)\alpha_1^k/L_1$ and $r_k \leq \mu_1/\alpha_1^k$, it follows that

$$t_k^{\min} r_k \leq (1-\sigma)\frac{\mu_1}{L_1} = (1-\sigma) \min_{i \in [m]}\left\{\frac{\mu_i}{L_i}\right\}.$$

For the left hand side of the inequality, we distinguish three cases: (a) $\alpha_{\max}^k \leq \mu_{\min}$, (b) $L_{\max} \leq \alpha_{\min}^k$ and (c) $\mu_{\min} < \alpha_{\max}^k$ and $\alpha_{\min}^k < L_{\max}$. If (a) holds, we have $r_k = 1$ and $t_k^{\min} \geq (1 -$

$\sigma)\alpha_{\min}^k/L_{\max}$, it follows that

$$t_k^{\min} r_k \geq (1 - \sigma)\frac{\alpha_{\min}^k}{L_{\max}}.$$

If (b) holds, we derive that $r_k \geq \mu_{\min}/\alpha_{\max}^k$ and $t_k^{\min} \geq (1 - \sigma)$, therefore,

$$t_k^{\min} r_k \geq (1 - \sigma)\mu_{\min}/\alpha_{\max}^k.$$

If (c) holds, we deduce that $r_k > \mu_{\min}/\alpha_{\max}^k$ and $t_k^{\min} > (1 - \sigma)\alpha_{\min}^k/L_{\max}$, then

$$t_k^{\min} r_k \geq (1 - \sigma)\frac{\alpha_{\min}^k \mu_{\min}}{\alpha_{\max}^k L_{\max}}.$$

Consequently, the left hand side of the inequality in (i) holds.

(ii) By distinguishing three cases as in the proof of (i) with $\alpha_{\max}^k = \alpha_{\max}^k = \ell$, the desire result follows.

(iii) If $\alpha_i^k = \mu_i$ for $i \in [m]$, we have $t_k^{\min} = (1 - \sigma)\min_{i \in [m]}\{\mu_i/L_i\}$ and $r_k = 1$. While $\alpha_i^k = L_i$ for $i \in [m]$, we have $t_k^{\min} = (1 - \sigma)$ and $r_k = \min_{i \in [m]}\{\mu_i/L_i\}$. In both cases, we have

$$t_k^{\min} r_k = (1 - \sigma)\min_{i \in [m]}\left\{\frac{\mu_i}{L_i}\right\}.$$

The desire result follows.

Remark 5.1 Theorem 5.2 demonstrates that the scaling parameters $\{\alpha_i^k : i \in [m]\}$ are pivotal in determining the linear convergence rate of Algorithm 5. The primary conclusions can be summarized as follows.

(i) If $\alpha_i^k = \ell > 0$ for $i \in [m]$, then the SPGMO with line search reduces to the PGMO with line search [34]. Consequently, Theorem 5.2 (ii) indicates that the PGMO with line search is unable to address objective imbalances.

(ii) By setting $\alpha_i^k = \mu_i$ for $i \in [m]$ or $\alpha_i^k = L_i$ for $i \in [m]$, we derive the objective imbalances parameter $\zeta = 1$ for scaled problems F^L and F^μ, thereby completely mitigating objective imbalances. In these scenarios, the SPGMO with line search achieves optimal linear convergence and enjoys favorable performance for well-conditioned problems.

(iii) The convergence rate in Theorem 5.2 (iii) is attributable to the global upper and lower bounds employed in (13) and (18), respectively. As a result, setting $\mu_i \leq \alpha_i^k \leq L_i$ leverages the problem's local geometry, whereas $\alpha_i^k = \mu_i$ or $\alpha_i^k = L_i$ is too conservative as the curvature of f_i can be pretty different. Intuitively, the linear convergence rate for Algorithm 5 with $\mu_i \leq \alpha_i^k \leq L_i$ for $i \in [m]$ is, roughly speaking, not less than $1 - \sigma(1 - \sigma)\min_{i \in [m]}\{\mu_i/L_i\}$.

6 Scaled proximal method with Nesterov's acceleration for MCOPs

Tanabe et al. [35] proposed an accelerated proximal gradient method for MCOPs (APGMO), and established the well-known $\mathcal{O}(1/k^2)$ for convex scenarios. The crux of the APGMO lies in

the subproblem:

$$\min_{x \in \mathbb{R}^n} \max_{i \in [m]} \left\{ \langle \nabla f_i(y^k), x - y^k \rangle + g_i(x) + f_i(y^k) - F_i(x^k) + \frac{L_{\max}}{2} \|x - y^k\|^2 \right\}, \tag{19}$$

where the inside of max operator approximates $F_i(x) - F_i(y^k)$ rather that $F_i(x) - F_i(x^k)$, which deeply affects the proof in the multiobjective case. We refer the reader to [35] for more details. It is worth noting that the linear convergence of APGMO for strongly convex cases was not analyzed in [35]. This raises an intriguing question: Does the APGMO with an appropriate momentum parameter exhibit the linear convergence $\mathcal{O}((1 - \sqrt{\mu_{\min}/L_{\max}})^k)$ for strongly convex scenarios?

6.1 Accelerated scaled proximal gradient method for MCOPs

In this subsection, we propose an accelerated scaled proximal method that alleviate the objective imbalances. Inspired by the work of Tanabe et al. [35], the subproblem is defined as follows:

$$\min_{x \in \mathbb{R}^n} \max_{i \in [m]} \left\{ \frac{\langle \nabla f_i(y^k), x - y^k \rangle + g_i(x) + f_i(y^k) - F_i(x^k)}{L_i} + \frac{1}{2} \|x - y^k\|^2 \right\}. \tag{20}$$

Assume that f_i is strongly convex with module $\mu_i \geq 0$ and L_i-smooth for $i \in [m]$, we define

$$\hat{\mu} := \min_{i \in [m]} \left\{ \frac{\mu_i}{L_i} \right\}.$$

It is evident that $0 \leq \hat{\mu} \leq 1$. The accelerated scaled proximal gradient method for MCOPs (ASPGMO) is described as follows.

Algorithm 6: `Accelerated_scale_proximal_gradient_method_for_MCOPs`

Require: $x^{-1} = x^0 \in \mathbb{R}^n$
1: **for** $k = 0, \cdots$ **do**
2: $\quad \gamma_k = \frac{(\theta_k - \hat{\mu})(1 - \theta_{k-1})}{(1 - \hat{\mu})\theta_{k-1}}$
3: $\quad y^k = x^k + \gamma_k(x^k - x^{k-1})$
4: $\quad$ Compute x^{k+1} by solving subproblem (20)
5: $\quad$ **if** $x^{k+1} = y^k$ **then**
6: $\quad\quad$ **return** x^{k+1}
7: $\quad$ **end if**
8: **end for**

Remark 6.1 Algorithm 6 unifies the convex and strongly convex cases by employing distinct values of θ_k in the momentum parameter. When the smooth parameters are unknown, we can update the scaling parameters through backtracking.

6.2 Convergence analysis of ASPGMO

Before proceeding with the complexity analysis, we define the following auxiliary sequences:

$$\sigma_k^L(z) := \min_{i \in [m]} \left\{ \frac{F_i(x^k) - F_i(z)}{L_i} \right\},$$

$$\rho_k(z) := \left\| \frac{1}{\theta_k} x^{k+1} - \frac{1 - \theta_k}{\theta_k} x^k - z \right\|^2.$$

Next, we present two lemmas that will be utilized in the complexity analysis.

Lemma 6.1 *Assume that f_i is strongly convex with modulus $\mu_i \geq 0$ and L_i-smooth for $i \in [m]$. Let $\{x^k\}$ and $\{y^k\}$ be sequences generated by Algorithm 6. Then, we have*

$$\sigma_{k+1}^L(z) \leq \langle x^{k+1} - y^k, z - y^k \rangle - \frac{\hat{\mu}}{2} \left\| y^k - z \right\|^2 - \frac{1}{2} \left\| x^{k+1} - y^k \right\|^2, \tag{21}$$

and

$$\sigma_{k+1}^L(z) - \sigma_k^L(z) \leq \langle x^{k+1} - y^k, x^k - y^k \rangle - \frac{1}{2} \left\| x^{k+1} - y^k \right\|^2 \tag{22}$$

for all $z \in \mathbb{R}^n$ and $k \geq 0$.

Proof By using strong convexity in (21), the assertions can be obtained by using similar arguments as in the proof of [35, Lemma 7].

Lemma 6.2 (See Theorem 9 of [35]) *Let $\{x^k\}$ be sequence generated by Algorithm 6. Then, we have $x^k \in \mathcal{L}_F(F(x^0))$ for all $k \geq 0$.*

Using Lemma 6.1, we establish an auxiliary inequality as follows.

Lemma 6.3 *Assume that f_i is strongly convex with modulus $\mu_i \geq 0$ and L_i-smooth for $i \in [m]$. Let $\{x^k\}$ and $\{y^k\}$ be sequences generated by Algorithm 6 with $\theta_k \geq \hat{\mu}$ for all $k \geq 0$. Then, we have*

$$\sigma_{k+1}^L(z) + \frac{\theta_k^2}{2} \rho_k(z) \leq (1 - \theta_k) \sigma_k^L(z) + \frac{\theta_k(\theta_k - \hat{\mu})}{2} \rho_{k-1}(z) \tag{23}$$

for all $z \in \mathbb{R}^n$ and $k \geq 0$.

Proof Multiplying inequality (21) by θ_k and inequality (22) by $(1 - \theta_k)$ and adding them together, we have

$$\begin{aligned}
&\sigma_{k+1}^L(z) - (1 - \theta_k) \sigma_k^L(z) \\
&\leq -\frac{1}{2} \left\| x^{k+1} - y^k \right\|^2 - \frac{\theta_k \hat{\mu}}{2} \left\| y^k - z \right\|^2 + \langle x^{k+1} - y^k, (1 - \theta_k) x^k + \theta_k z - y^k \rangle \\
&= -\frac{1}{2} \left\| x^{k+1} - y^k \right\|^2 - \frac{\theta_k \hat{\mu}}{2} \left\| y^k - z \right\|^2 + \frac{1}{2} (\left\| x^{k+1} - y^k \right\|^2 + \\
&\qquad \left\| (1 - \theta_k) x^k + \theta_k z - y^k \right\|^2 - \left\| x^{k+1} - (1 - \theta_k) x^k - \theta_k z \right\|^2) \\
&= -\frac{\theta_k \hat{\mu}}{2} \left\| y^k - z \right\|^2 + \frac{\theta_k^2}{2} \left(\left\| \frac{1 - \theta_k}{\theta_k} x^k + z - \frac{1}{\theta_k} y^k \right\|^2 - \rho_k(z) \right),
\end{aligned} \tag{24}$$

where the first inequality is due to the relation $2\langle a,b\rangle \le \|a\|^2 + \|b\|^2 - \|a-b\|^2$ with $a := x^{k+1} - y^k$ and $b := (1-\theta_k)x^k + \theta_k z - y^k$. By reorganizing the terms in $\frac{1-\theta_k}{\theta_k}x^k + z - \frac{1}{\theta_k}y^k$ carefully, we obtain

$$
\begin{aligned}
\frac{\theta_k^2}{2}&\left\|\frac{1-\theta_k}{\theta_k}x^k + z - \frac{1}{\theta_k}y^k\right\|^2\\
&= \frac{\theta_k^2}{2}\left\|\frac{\hat{\mu}}{\theta_k}(z-y^k) + \frac{\theta_k-\hat{\mu}}{\theta_k}\left(z + \frac{1-\theta_k}{\theta_k-\hat{\mu}}x^k - \frac{1-\hat{\mu}}{\theta_k-\hat{\mu}}y^k\right)\right\|^2\\
&\le \frac{\theta_k\hat{\mu}}{2}\left\|z-y^k\right\|^2 + \frac{\theta_k(\theta_k-\hat{\mu})}{2}\left\|z + \frac{1-\theta_k}{\theta_k-\hat{\mu}}x^k - \frac{1-\hat{\mu}}{\theta_k-\hat{\mu}}y^k\right\|^2\\
&= \frac{\theta_k\hat{\mu}}{2}\left\|z-y^k\right\|^2 + \frac{\theta_k(\theta_k-\hat{\mu})}{2}\rho_{k-1}(z),
\end{aligned}
$$

where the inequality is due to the facts that $\theta_k > \hat{\mu}$ and the convexity of $\|\cdot\|^2$, and the last equality follows by the definition of y^k in Algorithm 6. Substituting the above inequality into (24), we derive

$$
\sigma_{k+1}^L(z) - (1-\theta_k)\sigma_k^L(z) \le \frac{\theta_k(\theta_k-\hat{\mu})}{2}\rho_{k-1}(z) - \frac{\theta_k^2}{2}\rho_k(z).
$$

Hence, the inequality (23) holds for all $z \in \mathbb{R}^n$ and $k \ge 0$.

We define a Lyapunov function for $\mu_i = 0,\ i \in [m]$:

$$
\mathcal{E}_{k+1}(z) := \frac{\sigma_{k+1}^L(z)}{\theta_k^2} + \frac{1}{2}\rho_k(z). \tag{25}
$$

We will show that $\mathcal{E}_{k+1}(z) \le \mathcal{E}_k(z)$ for all $k \ge 0$ and establish the convergence rate in the following theorem.

Theorem 6.1 *Assume that f_i is convex and L_i-smooth for $i \in [m]$. Let $\theta_k = 2/(k+2)$ in Algorithm 6, i.e., $\gamma_k = (k-1)/(k+2)$. Then, we have*

$$
\mathcal{E}_{k+1}(z) \le \mathcal{E}_k(z)
$$

for all $k \ge 0$. If the level set $\mathcal{L}_F(F(x^0))$ is bounded, then

$$
u_0^L(x^k) \le \frac{2R^2}{(k+1)^2}
$$

for all $k \ge 0$, where $R := \max\left\{\|x-y\| : x,y \in \mathcal{L}_F(x^0)\right\}$.

Proof In this case, we have $\hat{\mu} = 0$, thus $\theta_k \ge \hat{\mu}$ holds for all $k \ge 0$. By substituting $\hat{\mu} = 0$ into (23), it follows that

$$
\sigma_{k+1}^L(z) + \frac{\theta_k^2}{2}\rho_k(z) \le (1-\theta_k)\sigma_k^L(z) + \frac{\theta_k^2}{2}\rho_{k-1}(z).
$$

Dividing both sides of the above inequality by θ_k^2, we have

$$\frac{\sigma_{k+1}^L(z)}{\theta_k^2} + \frac{1}{2}\rho_k(z) \le \frac{1-\theta_k}{\theta_k^2}\sigma_k^L(z) + \frac{1}{2}\rho_{k-1}(z).$$

Recall that $\theta_k = \frac{2}{k+2}$, then we can deduce $\frac{1-\theta_k}{\theta_k^2} \le \frac{1}{\theta_{k-1}^2}$. This together with the above inequality implies

$$\mathcal{E}_{k+1}(z) \le \mathcal{E}_k(z).$$

Notice that $\rho_{k-1}(z) \ge 0$ and $x^0 = x^{-1}$, the above inequality implies

$$\sigma_k^L(z) \le \theta_{k-1}^2 \left(\frac{\sigma_0^L(z)}{\theta_{-1}^2} + \frac{1}{2}\left\|x^0 - z\right\|^2 \right).$$

Setting $1/\theta_{-1}^2 = 0$, and selecting $x_*^k \in \arg\max_{z\in\mathbb{R}^n} \min_{i\in[m]} \left\{ (F_i(x^k) - F_i(z))/L_i \right\}$, the previous inequality implies

$$u_k^L(x^k) \le \frac{2\left\|x^0 - x_*^k\right\|^2}{(k+1)^2}.$$

Then, the desired result follows by $F(x_*^k) \preceq F(x^k)$ and $x^k \in \mathcal{L}_F(F(x^0))$.

We also define a Lyapunov function for $\mu_i > 0$, $i \in [m]$:

$$\mathcal{E}_{k+1}^\mu(z) := \frac{1}{(1-\sqrt{\hat{\mu}})^{k+1}} \left(\sigma_{k+1}^L(z) + \frac{\hat{\mu}}{2}\rho_k(z) \right). \tag{26}$$

We will show that $\mathcal{E}_{k+1}^\mu(z) \le \mathcal{E}_k^\mu(z)$ for all $k \ge 0$ and establish the convergence rate in the following theorem.

Theorem 6.2 *Assume that f_i is strongly convex with module $\mu_i > 0$ and L_i-smooth for $i \in [m]$. Let $\theta_k = \sqrt{\hat{\mu}}$ in Algorithm 6, i.e., $\gamma_k = (1 - \sqrt{\hat{\mu}})/(1 + \sqrt{\hat{\mu}})$. Then, we have*

$$\mathcal{E}_{k+1}^\mu(z) \le \mathcal{E}_k^\mu(z),$$

and

$$u_0^L(x^k) \le \left(1 - \sqrt{\hat{\mu}}\right)^k \left(u_0^L(x^0) + \frac{\hat{\mu}}{2}R^2 \right)$$

for all $k \ge 0$, where $R := \max \left\{ \|x - y\| : x, y \in \mathcal{L}_F(F(x^0)) \right\}$.

Proof Since $\theta_k = \sqrt{\hat{\mu}}$ and $\hat{\mu} \le 1$, thus $\theta_k \ge \hat{\mu}$ holds for all $k \ge 0$. By substituting $\theta_k = \sqrt{\hat{\mu}}$ into (23), it follows that

$$\sigma_{k+1}^L(z) + \frac{\hat{\mu}}{2}\rho_k(z) \le (1 - \sqrt{\hat{\mu}})\sigma_k^L(z) + \frac{\sqrt{\hat{\mu}}(\sqrt{\hat{\mu}} - \hat{\mu})}{2}\rho_{k-1}(z).$$

Dividing both sides of the above inequality by $(1 - \sqrt{\hat{\mu}})^{k+1}$, we have

$$\mathcal{E}_{k+1}^\mu(z) \le \mathcal{E}_k^\mu(z).$$

Notice that $\rho_{k-1}(z) \geq 0$ and $x^0 = x^{-1}$, the above inequality implies

$$\sigma_k^L(z) \leq \left(1 - \sqrt{\hat{\mu}}\right)^k \left(\sigma_0^L(z) + \frac{\hat{\mu}}{2}\left\|x^0 - z\right\|^2\right).$$

Selecting $x_*^k \in \arg\max_{z\in\mathbb{R}^n}\min_{i\in[m]}\left\{(F_i(x^k) - F_i(z))/L_i\right\}$, the previous inequality implies

$$\begin{aligned}
u_0^L(x^k) &\leq \left(1 - \sqrt{\hat{\mu}}\right)^k \left(\sigma_0^L(x^*) + \frac{\hat{\mu}}{2}\left\|x^0 - x_*^k\right\|^2\right) \\
&\leq \left(1 - \sqrt{\hat{\mu}}\right)^k \left(u_0^L(x^0) + \frac{\hat{\mu}}{2}\left\|x^0 - x_*^k\right\|^2\right),
\end{aligned}$$

where the second inequality is due to the definition of u_0^L. Notice that f_i is strongly convex and g_i is convex for $i \in [m]$, we conclude that $\mathcal{L}_F(x^0)$ is bounded. Then, the desired result follows by the facts that $F(x_*^k) \preceq F(x^k)$ and $x^k \in \mathcal{L}_F(x^0)$.

Remark 6.2 By setting $\gamma_k = (1 - \sqrt{\mu_{\min}/L_{\max}})/(1 + \sqrt{\mu_{\min}/L_{\max}})$ in APGMO, the method demonstrates a linear convergence rate of $\mathcal{O}((1 - \sqrt{\mu_{\min}/L_{\max}})^k)$ for strongly convex cases.

7 Numerical experiments

In this section, we present numerical results to illustrate the performance of SPGMO across various problems. All numerical experiments were implemented in Python 3.7 and executed on a personal computer with an Intel Core i7-11390H, 3.40 GHz processor, and 16 GB of RAM. For the SPGMO with line search, we selected scaling parameters using the Barzilai-Borwein method [10]. The subproblems of the tested algorithms were solved via dual problems (see [35] for more details), which can be efficiently solved by Frank-Wolfe method. We set $\sigma = 10^{-4}$ in the line search procedure and employed stopping criteria of $\|d^k\| \leq 10^{-4}$ or $\|x^{k+1} - y^k\| \leq 10^{-4}$ for all tested algorithms to ensure termination after a finite number of iterations. Additionally, we set the maximum number of iterations to 500.

7.1 Common examples

In this subsection, we compare PGMO, APGMO, SPGMO and ASPGMO with line search across several common examples. Each objective function in the tested problems consists of two components, where $g_i = \frac{1}{n}\|x\|_1$ for $i \in [m]$, and the details of f are provided in Table 1. The recorded averages from 200 runs include the number of iterations, the number of function evaluations, and the CPU time.

Table 2 presents the average number of iterations (iter), average number of function evaluations (feval) and average CPU time (time (ms)) for each tested algorithm across various problems with line search. In both the non-accelerated and accelerated cases, the numerical results indicate that SPGMO consistently outperforms PGMO. Additionally, PGMO demonstrates poor performance on problems such as DD1, Deb, FDS, imbalance2, and WIT1-3, which exhibit imbalanced

Table 1: Description of all test problems used in numerical experiments.

Problem	n	m	x_L	x_U	Reference
DD1	5	2	(-20,...,-20)	(20,...,20)	[11]
Deb	2	2	(0.1,0.1)	(1,1)	[12]
Far1	2	2	(-1,-1)	(1,1)	[21]
FDS	5	3	(-2,...,-2)	(2,...,2)	[14]
FF1	2	2	(-1,-1)	(1,1)	[21]
Hil1	2	2	(0,0)	(1,1)	[20]
Imbalance1	2	2	(-2,-2)	(2,2)	[10]
Imbalance2	2	2	(-2,-2)	(2,2)	[10]
VU1	2	2	(-3,-3)	(3,3)	[21]
WIT1	2	2	(-2,-2)	(2,2)	[39]
WIT2	2	2	(-2,-2)	(2,2)	[39]
WIT3	2	2	(-2,-2)	(2,2)	[39]

Table 2: Number of average iterations (iter), number of average function evaluations (feval) and average CPU time (time (ms)) of test algorithms on different test problems **with line search**.

Problem	PGMO			APGMO			SPGMO			ASPGMO		
	iter	feval	time	iter	feval	time	iter	feval	time	iter	feval	time
DD1	41.21	70.65	135.31	67.75	132.46	400.16	**4.52**	**4.90**	**31.02**	6.43	7.27	65.31
Deb	26.52	253.17	83.67	14.39	30.44	58.75	5.97	9.86	**28.83**	**5.15**	**8.04**	31.80
Far1	**6.06**	21.09	16.88	8.06	17.75	29.38	6.76	**7.86**	**9.92**	7.97	14.41	19.53
FDS	175.75	778.54	1350.16	55.15	107.88	737.89	**3.44**	**3.82**	**2.16**	7.36	12.00	54.53
FF1	3.41	3.55	2.42	4.13	5.44	3.05	**2.10**	**2.26**	**2.27**	2.93	4.00	5.55
Hil1	8.71	18.62	27.50	**7.02**	15.81	30.07	7.49	**8.26**	13.91	7.88	12.24	25.00
Imbalance1	2.92	5.63	42.81	7.79	13.80	134.53	**2.44**	**3.11**	34.69	3.82	5.35	39.14
Imbalance2	83.56	589.13	1180.39	21.09	42.18	149.84	**1.00**	**1.00**	**1.79**	**1.00**	**1.00**	2.65
VU1	8.52	8.59	**1.72**	6.20	10.47	9.69	2.08	2.15	3.36	3.61	5.31	7.42
WIT1	27.63	145.06	118.98	20.05	41.14	51.64	**2.94**	**3.24**	**7.11**	4.83	6.71	15.73
WIT2	48.07	286.20	193.67	19.22	39.65	83.36	**3.14**	**3.34**	**9.84**	4.91	6.49	19.69
WIT3	18.58	79.78	74.30	35.51	71.72	124.68	**3.92**	**4.13**	**14.22**	5.33	7.64	17.89

objective functions. Given the performance of SPGMO on these problems, it is well-suited for addressing such challenges. Table 2 also shows that APGMO surpasses PGMO, validating the effectiveness of Nesterov's acceleration technique in MOPs. However, we note that Nesterov's acceleration technique slightly hampers the performance of SPGMO. We infer that the primary reasons for this may be the Barzilai-Borwein method utilized in SPGMO and the simplicity of the problems tested.

7.2 Quadratic examples

In the following, we compare different test problem with known smooth parameters. Consider a series of quadratic problems defined as follows:

$$f_i(x) = \frac{1}{2} \langle x, A_i x \rangle + \langle b_i, x \rangle, \ g_i(x) = \frac{1}{n} \|x\|_1, \ i = 1, 2,$$

where $A_i \in \mathbb{R}^{n \times n}$ is a positive definite matrix. Here we use, $A_i = H_i D_i H_i^T$ where H_i is a orthogonal matrix and D_i is a diagonal matrix with positive components. More details of f are provided in Table 1, where κ and ζ denote the condition number and objective imbalances parameter,

respectively. QPa is a well-conditioned problem without objective imbalances, while QPb-f exhibit objective imbalances with a parameter value of 100. Additionally, QPd and QPf display significant ill-conditioning, which poses challenges for first-order methods. For each problem, we conducted 200 computations using the same initial points across the different tested algorithms. The initial points were randomly selected within the specified lower and upper bounds. The recorded averages from the 200 runs include the number of iterations and the CPU time.

Table 3: Description of quadratic problems

Problem	n	(κ, ζ)	x_L	x_U
QPa	10	$(10, 1)$	10[-1,...,-1]	10[1,...,1]
QPb	10	$(10, 10^2)$	10[-1,...,-1]	10[1,...,1]
QPc	10	$(10^2, 10^2)$	10[-1,...,-1]	10[1,...,1]
QPd	10	$(10^4, 10^2)$	10[-1,...,-1]	10[1,...,1]
QPe	100	$(10^2, 10^2)$	100[-1,...,-1]	100[1,...,1]
QPf	100	$(10^3, 10^2)$	100[-1,...,-1]	100[1,...,1]

Table 4: Number of average iterations (iter) and average CPU time (time (ms)) of tested algorithms implemented on different test problems **with known smooth parameters**.

Problem	PGMO		APGMO		APGMO-sc		SPGMO		ASPGMO		ASPGMO-sc	
	iter	time	iter	time	iter	time	iter	time	iter	time	iter	time
QPa	43.07	94.92	33.42	99.14	**20.56**	60.47	43.07	100.31	33.42	105.16	**20.56**	**59.53**
QPb	–	–	–	–	196.03	477.42	48.44	89.21	34.30	82.03	**21.21**	**62.81**
QPc	–	–	–	–	–	–	367.21	210.78	149.33	144.53	**68.97**	**74.22**
QPd	–	–	–	–	–	–	–	–	–	–	**422.72**	**25.70**
QPe	–	–	–	–	–	–	326.31	368.20	186.47	320.54	**81.66**	219.61
QPf	–	–	–	–	–	–	–	–	–	–	**262.87**	**351.25**

Table 4 provides the average number of iterations (iter) and average CPU time (time (ms)) for each tested algorithm across the various problems with known smooth parameters, where "–" indicates that the tested algorithm failed to meet the stopping criteria within 500 iterations. As noted in Table 4, we conclude that PGMO exhibits slow convergence on imbalanced problems in both the non-accelerated and accelerated scenarios. The subpar performance of APGMO on imbalanced problems further confirms that Nesterov's acceleration technique is inadequate for addressing objective imbalances. In comparing the performances of SPGMO and ASPGMO, we conclude that the performance of SPGMO with known smooth parameters is enhanced by the incorporation of Nesterov's acceleration technique. Remarkably, even for the highly ill-conditioned problems QPd and QPf, the ASPGMO with strongly convex momentum (ASPGMO-sc) shows commendable performance.
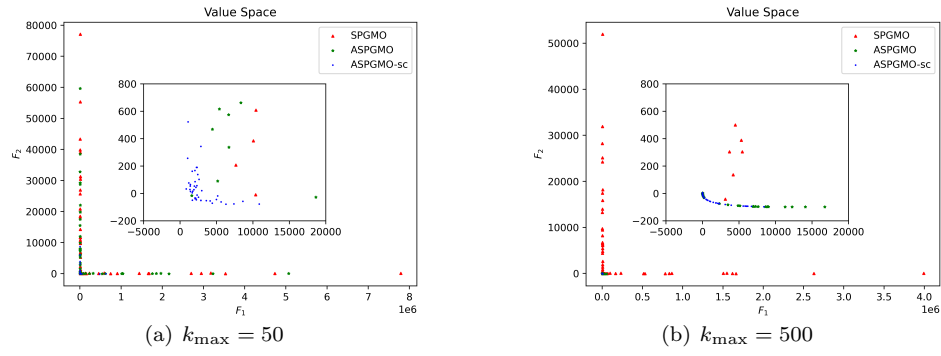
(a) $k_{\max} = 50$ (b) $k_{\max} = 500$

Fig. 1: Value space for different maximum numbers of iterations $k_{\max} = 50, 500$ for QPd.
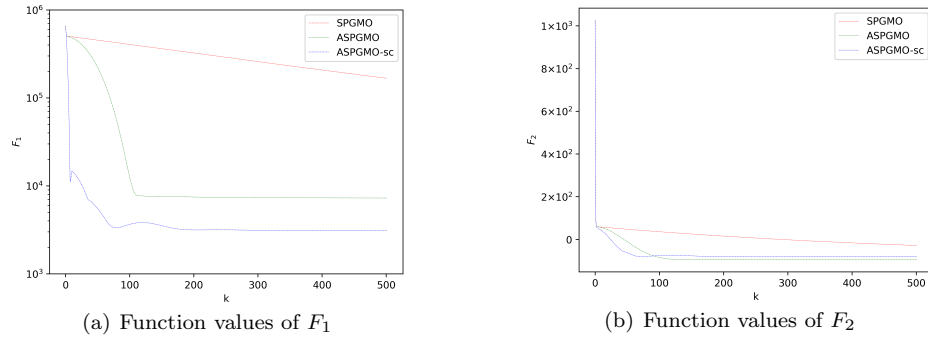
(a) Function values of F_1 (b) Function values of F_2
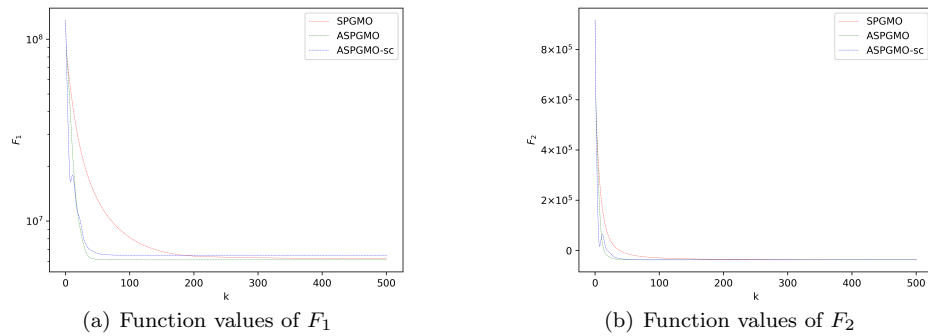
Fig. 2: Function values of iterations for QPd.

(a) Function values of F_1 (b) Function values of F_2

Fig. 3: Function values of iterations for QPf.

(a) $\left\| x^{k+1} - y^k \right\|$ for problem QPd (b) $\left\| x^{k+1} - y^k \right\|$ for problem QPf
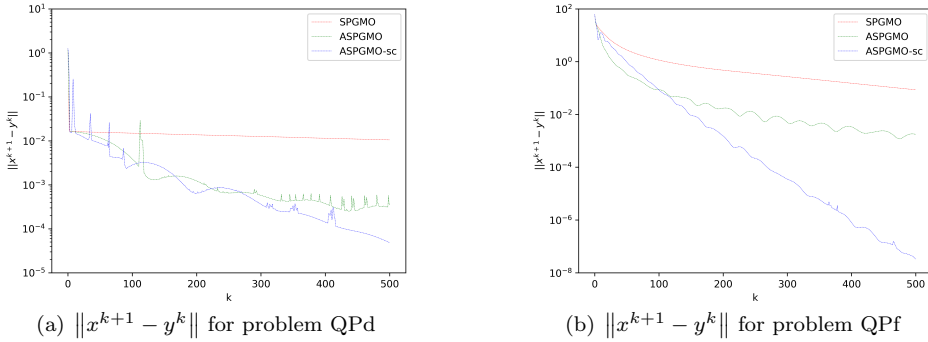
Fig. 4: $\left\| x^{k+1} - y^k \right\|$ for problems QPd and QPf.

In Figure 1, we utilize the same 50 random starting points for SPGMO, ASPGMO, and ASPGMO-sc. For these experiments, we apply different maximal numbers of iterations $k_{\max}$. As shown in Figure 1 (a), the points generated by ASPGMO-sc progress significantly faster than those produced by SPGMO and ASPGMO in the initial stages. In Figure 1 (b), we observe that the points generated by SPGMO are consistently outperformed by both ASPGMO and ASPGMO-sc. Although SPGMO and ASPGMO do not meet the stopping criteria within 500 iterations on the QPe and QPf problems, Figures 2 and 3 indicate that the function values produced by ASPGMO decrease more rapidly at the outset compared to SPGMO. Moreover, from Figures 2 and 3, we note that the function values of ASPGMO and ASPGMO-sc exhibit a slower decrease after 100 iterations and converge to similar values. However, as illustrated in Figure 4, both ASPGMO and ASPGMO-sc converge more rapidly than the non-accelerated SPGMO, despite some oscillations. Additionally, ASPGMO-sc demonstrates superior performance compared to ASPGMO, successfully reaching the stopping criteria within 500 iterations for the QPd and QPf problems.

8 Conclusions

We develop a scaled method to mitigate objective imbalances that lead to the slow convergence of existing multiobjective first-order methods. It is proven that the SPGMO converges linearly at a rate of $\sqrt{1 - \min_{i \in [m]} \{\mu_i / L_i\}}$, whereas the linear convergence rate of PGMO is $\sqrt{1 - \mu_{\min} / L_{\max}}$. We also establish improved linear convergence in both linear and accelerated cases. To the best of our knowledge, the improved linear convergence is the first theoretical result that bridges the theoretical gap between first-order methods for SOPs and MOPs. Notably, scaling serves as a flexible manipulation that opens new avenues for exploring efficient descent methods in multiobjective optimization.

For future work, several points warrant consideration:

– Objective imbalances are intrinsic to MOPs that reflect differences among objectives. This paper quantifies objective imbalances by (5) for first-order methods in strongly convex cases.

It would be interesting to study objective imbalances in non-convex scenarios as well as in high-order methods.

– The line search method described in Remark 5.1 (iii) suggests selecting appropriate scaling parameters to capture the problem's local geometry. However, line search methods often impose significant computational burdens. By extending the fully adaptive method introduced in [23] to determine scaling parameters, developing an adaptive SPGMO could be an interesting avenue for exploration.

– Numerical results indicate that the ASPGMO performs well for ill-conditioned MOPs when the strongly convex parameters are known. However, these parameters are frequently unknown and challenging to estimate. It would be worthwhile to develop efficient ASPGMO algorithms that do not require knowledge of the strongly convex parameters.

References

1. M. A. T. Ansary, *A Newton-type proximal gradient method for nonlinear multi-objective optimization problems*, Optimization Methods and Software, 38 (2023), pp. 570–590.

2. M. A. T. Ansary and G. Panda, *A globally convergent SQCQP method for multiobjective optimization problems*, SIAM Journal on Optimization, 31 (2021), pp. 91–113, https://doi.org/10.1137/18M1182152.

3. P. Assunção, O. Ferreira, and L. Prudente, *A generalized conditional gradient method for multiobjective composite optimization problems*, Optimization, (2023), pp. 1–31.

4. H. Attouch, G. Garrigos, and X. Goudou, *A dynamic gradient approach to pareto optimization with nonsmooth convex objective functions*, Journal of Mathematical Analysis and Applications, 422 (2015), pp. 741–771.

5. J. Barzilai and J. M. Borwein, *Two-point step size gradient methods*, IMA Journal of Numerical Analysis, 8 (1988), pp. 141–148.

6. Y. Bello-Cruz, J. G. Melo, and R. V. Serra, *A proximal gradient splitting method for solving convex vector optimization problems*, Optimization, 71 (2022), pp. 33–53.

7. H. Bonnel, A. N. Iusem, and B. F. Svaiter, *Proximal methods in vector optimization*, SIAM Journal on Optimization, 15 (2005), pp. 953–970, https://doi.org/10.1137/S1052623403429093.

8. R. I. Boţ and S. M. Grad, *Inertial forward-backward methods for solving vector optimization problems*, Optimization, 67 (2018), pp. 959–974.

9. G. A. Carrizo, P. A. Lotito, and M. C. Maciel, *Trust region globalization strategy for the nonconvex unconstrained multiobjective optimization problem*, Mathematical Programming, 159 (2016), pp. 339–369.

10. J. Chen, L. P. Tang, and X. M. Yang, *A Barzilai-Borwein descent method for multiobjective optimization problems*, European Journal of Operational Research, 311 (2023), pp. 196–209.

11. I. DAS AND J. E. DENNIS, *Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems*, SIAM Journal on Optimization, 8 (1998), pp. 631–657, https://doi.org/10.1137/S1052623496307510.

12. K. DEB, *Multi-objective genetic algorithms: Problem difficulties and construction of test problems*, Evolutionary Computation, 7 (1999), pp. 205–230.

13. G. EVANS, *Overview of techniques for solving multiobjective mathematical programs*, Management Science, 30 (1984), pp. 1268–1282.

14. J. FLIEGE, L. M. GRAÑA DRUMMOND, AND B. F. SVAITER, *Newton's method for multiobjective optimization*, SIAM Journal on Optimization, 20 (2009), pp. 602–626, https://doi.org/10.1137/08071692X.

15. J. FLIEGE AND B. F. SVAITER, *Steepest descent methods for multicriteria optimization*, Mathematical Methods of Operations Research, 51 (2000), pp. 479–494.

16. J. FLIEGE AND A. I. F. VAZ, *A method for constrained multiobjective optimization based on SQP techniques*, SIAM Journal on Optimization, 26 (2016), pp. 2091–2119, https://doi.org/10.1137/15M1016424.

17. J. FLIEGE, A. I. F. VAZ, AND L. N. VICENTE, *Complexity of gradient descent for multiobjective optimization*, Optimization Methods and Software, 34 (2019), pp. 949–959.

18. J. FLIEGE AND R. WERNER, *Robust multiobjective optimization & applications in portfolio optimization*, European Journal of Operational Research, 234 (2014), pp. 422–433.

19. L. M. GRAÑA DRUMMOND AND A. N. IUSEM, *A projected gradient method for vector optimization problems*, Computational Optimization and Applications, 28 (2004), pp. 5–29.

20. C. HILLERMEIER, *Generalized homotopy approach to multiobjective optimization*, Journal of Optimization Theory and Applications, 110 (2001), pp. 557–583.

21. S. HUBAND, P. HINGSTON, L. BARONE, AND L. WHILE, *A review of multiobjective test problems and a scalable test problem toolkit*, IEEE Transactions on Evolutionary Computation, 10 (2006), pp. 477–506.

22. L. R. LUCAMBIO PÉREZ AND L. F. PRUDENTE, *Nonlinear conjugate gradient methods for vector optimization*, SIAM Journal on Optimization, 28 (2018), pp. 2690–2720, https://doi.org/10.1137/17M1126588.

23. Y. MALITSKY AND K. MISHCHENKO, *Adaptive gradient descent without descent*, in Proceedings of the 37th International Conference on Machine Learning, H. D. III and A. Singh, eds., vol. 119, 2020, pp. 6702–6712, http://proceedings.mlr.press/v119/malitsky20a/malitsky20a.pdf.

24. H. MARKOWITZ, *Portfolio selection*, Journal of Finance, 7 (1952), pp. 77–91.

25. R. T. MARLER AND J. S. ARORA, *Survey of multi-objective optimization methods for engineering*, Structural and Multidisciplinary Optimization, 26 (2004), pp. 369–395.

26. Q. MERCIER, F. POIRION, AND J. A. DÉSIDÉRI, *A stochastic multiple gradient descent algorithm*, European Journal of Operational Research, 271 (2018), pp. 808–817.

27. V. MOROVATI AND L. POURKARIMI, *Extension of zoutendijk method for solving constrained multiobjective optimization problems*, European Journal of Operational Research, 273 (2019), pp. 44–57.

28. H. MUKAI, *Algorithms for multicriterion optimization*, IEEE Transactions on Automatic Control, 25 (1980), pp. 177–186.

29. Ž. POVALEJ, *Quasi-Newton's method for multiobjective optimization*, Journal of Computational and Applied Mathematics, 255 (2014), pp. 765–777.

30. S. QU, M. GOH, AND F. T. CHAN, *Quasi-Newton methods for solving multiobjective optimization*, Operations Research Letters, 39 (2011), pp. 397–399.

31. O. SENER AND V. KOLTUN, *Multi-task learning as multi-objective optimization*, in Advances in Neural Information Processing Systems, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds., vol. 31, Curran Associates, Inc., 2018, https://proceedings.neurips.cc/paper_files/paper/2018/file/432aca3a1e345e339f35a30c8f65edce-Paper.pdf.

32. K. SONNTAG AND S. PEITZ, *Fast convergence of inertial multiobjective gradient-like systems with asymptotic vanishing damping*, SIAM Journal on Optimization, 34 (2024), pp. 2259–2286, https://doi.org/10.1137/23M1588512.

33. K. SONNTAG AND S. PEITZ, *Fast multiobjective gradient methods with Nesterov acceleration via inertial gradient-like systems*, Journal of Optimization Theory and Applications, 201 (2024), pp. 539–582.

34. H. TANABE, E. H. FUKUDA, AND N. YAMASHITA, *Proximal gradient methods for multiobjective optimization and their applications*, Computational Optimization and Applications, 72 (2019), pp. 339–361.

35. H. TANABE, E. H. FUKUDA, AND N. YAMASHITA, *An accelerated proximal gradient method for multiobjective optimization*, Computational Optimization and Applications, 86 (2023), pp. 421–455.

36. H. TANABE, E. H. FUKUDA, AND N. YAMASHITA, *Convergence rates analysis of a multiobjective proximal gradient method*, Optimization Letters, 17 (2023), pp. 333–350.

37. H. TANABE, E. H. FUKUDA, AND N. YAMASHITA, *New merit functions for multiobjective optimization and their properties*, Optimization, (2023), pp. 1–38.

38. M. G. C. TAPIA AND C. A. C. COELLO, *Applications of multi-objective evolutionary algorithms in economics and finance: A survey*, in 2007 IEEE Congress on Evolutionary Computation, 2007, pp. 532–539.

39. K. WITTING, *Numerical algorithms for the treatment of parametric multiobjective optimization problems and applications*, PhD thesis, Ph.D. thesis, Universität Paderborn, Paderborn, 2012, 2012.

40. L. ZENG, Y. H. DAI, AND Y. K. HUANG, *Convergence rate of gradient descent method for multi-objective optimization*, Journal of Computational Mathematics, 37 (2019), pp. 689–703.