First-order Methods for Unconstrained Vector Optimization Problems: A Unified Majorization-Minimization Perspective

Jian Chen[a,b], Jinjie Liu[b], Liping Tang[b], Xinmin Yang[b,*]

[a] *College of Mathematics, Sichuan University, Chengdu 610065, China*
[b] *National Center for Applied Mathematics in Chongqing, Chongqing Normal University, Chongqing 401331, China*

Abstract

In this paper, we develop a unified majorization-minimization scheme and convergence analysis with first-order surrogate functions for unconstrained vector optimization problems (VOPs). By selecting different surrogate functions, the unified method can be reduced to various existing first-order methods. The unified convergence analysis reveals that the slow convergence of the steepest descent method is primarily attributed to the significant gap between the surrogate and objective functions. Consequently, narrowing this surrogate gap can enhance the performance of first-order methods for VOPs. To strike a better trade-off in terms of surrogate gap and per-iteration cost, we reformulate the direction-finding subproblem and elucidate that selecting a tighter surrogate function is equivalent to using an appropriate base of the dual cone in the direction-finding subproblem. Building on this insight, we employ the Barzilai-Borwein method to narrow the surrogate gap and propose a Barzilai-Borwein descent method for VOPs (BBDVO) with polyhedral cones. By reformulating the corresponding subproblem, we provide a novel perspective on the Barzilai-Borwein descent method, bridging the gap between this method and the steepest descent method. Finally, several numerical experiments are presented to validate the efficiency of the BBDVO.

Keywords: Multiple objective programming, Majorization-minimization optimization, Barzilai-Borwein method, Convergence rates, Polyhedral cone
2010 MSC: 90C29, 90C30

1. Introduction

This paper focuses on the following unconstrained vector optimization problem:

$$\min_K F(x), \tag{VOP}$$

where $F : \mathbb{R}^n \to \mathbb{R}^m$ is to be optimized under the partial order induced by a closed, convex, and pointed cone $K \subset \mathbb{R}^m$ with a non-empty interior, defined as follows:

$$y \preceq_K (\text{resp. } \prec_K) y' \ \Leftrightarrow\ y' - y \in K(\text{resp. int}(K)).$$

Let $K^* = \{c^* \in \mathbb{R}^m : \langle c^*, y\rangle \geq 0, \forall y \in K\}$ be the positive polar cone of K, and C be a compact base of K^*, namely, C is a convex set such that $0 \notin \text{cl}C$ and $\text{cone}(C) = K^*$. In vector optimization, it is often impossible to improve all objectives simultaneously with respect to the partial order. Therefore, the concept of optimality is defined as *efficiency* (Jahn, 2011), meaning that there is no better solution for an efficient solution. Specifically, the problem (VOP) corresponds to a multiobjective optimization problem when $K = \mathbb{R}^m_+$, where $\mathbb{R}^m_+$ denotes the non-negative orthant of $\mathbb{R}^m$. Various applications of multiobjective optimization problems (MOPs) can be found in engineering (Marler & Arora, 2004), economics (Tapia & Coello, 2007; Fliege & Werner, 2014), management science (Evans, 1984), environmental analysis (Leschine et al., 1992), machine learning (Sener & Koltun, 2018; Ye et al., 2021), etc. Although many real-world problems reformulated as

[*] Corresponding author.
Email addresses: chenjian_math@163.com (Jian Chen), jinjie.liu@cqnu.edu.cn (Jinjie Liu), tanglipings@163.com (Liping Tang), xmyang@cqnu.edu.cn (Xinmin Yang)

vector-valued problems adhere to the partial order induced by $\mathbb{R}^m_+$, some applications, such as portfolio selection in securities markets (Aliprantis et al., 2004a,b), require partial orders induced by closed convex cones other than the non-negative orthant. Consequently, vector optimization problems (VOPs) have garnered significant attention in recent years.

Over the past two decades, descent methods have received increasing attention within the multiobjective optimization community, primarily due to the seminal work on the steepest descent method proposed by Fliege & Svaiter (2000). Inspired by Fliege and Svaiter's contributions, researchers have extended other numerical algorithms to solve multiobjective optimization problems (MOPs) (see, e.g., Fliege et al., 2009; Qu et al., 2011; Povalej, 2014; Fliege & Vaz, 2016; Carrizo et al., 2016; Mercier et al., 2018; Morovati & Pourkarimi, 2019; Tanabe et al., 2019). To the best of our knowledge, the study of descent methods for unconstrained vector optimization problems can be traced back to the work of Graña Drummond & Svaiter (2005), who extended the steepest descent method for MOPs (SDMO) proposed by Fliege & Svaiter (2000) to VOPs. In this context, the direction-finding subproblem at x^k is formulated as follows:

$$\min_{d\in\mathbb{R}^n}\max_{c^*\in C}\left\langle c^*, JF(x^k)d\right\rangle + \frac{1}{2}\|d\|^2,$$

where $JF(x^k)\in\mathbb{R}^{m\times n}$ is the Jacobian matrix of $F(\cdot)$ at x^k. Similar to MOPs, several standard numerical algorithms have been extended to VOPs, including the Newton method (Graña Drummond et al., 2014), projected gradient method (Graña Drummond & Iusem, 2004), proximal point method (Bonnel et al., 2005), conjugated gradient method (Lucambio Pérez & Prudente, 2018) and conditional gradient method (Chen et al., 2023c).

In recent years, complexity analysis of descent methods for MOPs has been extensively studied. Fliege et al. (2018) and Zeng et al. (2019) established the convergence rates of SDMO under different convexity assumptions. Tanabe et al. (2023) developed convergence results for the multiobjective proximal gradient method. Additionally, Lapucci (2024) studied the complexity of a wide class of multiobjective descent methods with nonconvex assumption. However, Chen et al. (2023b) noted that both theoretical and empirical results indicate that existing multiobjective first-order methods exhibit slow convergence due to objective imbalances. To address this challenge, Chen et al. (2023a) proposed a Barzilai-Borwein descent method for MOPs (BBDMO) that dynamically tunes gradient magnitudes using Barzilai-Borwein's rule (Barzilai & Borwein, 1988) in direction-finding subproblem. From a theoretical perspective, an improved linear convergence rate is confirmed by Chen et al. (2023b), demonstrating that Barzilai-Borwein descent methods can effectively mitigate objective imbalances.

Despite the extensive study of complexity analysis for MOPs, corresponding results have received little attention in VOPs. As described by Chen et al. (2022), the linear convergence rates of first-order descent methods for VOPs are essentially affected by C, which represents the base of K^* in direction-finding subproblem. In general, K^* admits infinitely many possible bases, and the choice of C is therefore not unique. Although this choice critically influences the search direction and ultimately the convergence behavior of the algorithm, the existing literature offers no comprehensive theoretical analysis or guiding principle for selecting such a base. In the classical setting of multiobjective optimization, the subproblem of SDMO can be reformulated as

$$\min_{d\in\mathbb{R}^n}\max_{\lambda\in\Delta_m}\left\langle \lambda, JF(x^k)d\right\rangle + \frac{1}{2}\|d\|^2,$$

where $\Delta_m := \{\lambda \succeq 0 : \sum_{i=1}^m \lambda_i = 1\}$ is a base of $\mathbb{R}^m_+$. While this choice appears natural and convenient from a computational perspective, its theoretical justification and potential advantages over other alternatives remain largely unexplored. Hence, identifying an appropriate and theoretically sound base for K^* constitutes a fundamental yet open problem, which is of both theoretical significance and practical relevance in the design of efficient first-order methods for VOPs. This naturally leads to the following question:

Can we provide a theoretical guidance for the choice of the base? (Q)

To answer the proposed questions, we develop a unified framework and convergence analysis of first-order methods for VOPs from a majorization-minimization perspective. The majorization-minimization principle is a versatile tool for designing novel algorithms, which has been successful employed in nonconvex optimization (Lanza et al., 2017), constrained optimization (Landeros et al., 2023), and incremental optimization (Mairal, 2015; Karimi et al., 2022). The core idea behind the majorization-minimization method is to minimize a difficult optimization problem by iteratively minimizing a simpler surrogate function that majorizes (upper-bounds) the original objective. In this paper, we extend majorization-minimization method to vector optimization, addressing the aforementioned questions. Our work aims to provide a unified framework and convergence analysis of first-order methods for VOPs from a majorization-minimization perspective. The primary contributions of this paper are summarized as follows:

(i) Before presenting the majorization-minimization method and its convergence analysis, we first define the concepts of strong convexity and smoothness for a vector-valued function with respect to a partial order. Leveraging these properties, we extend the notion of the condition number to VOPs, which plays a pivotal role in establishing the linear convergence of first-order methods for VOPs. To the best of our knowledge, it is the first definition of condition number to VOPs.

(ii) We devise a unified majorization-minimization descent method for VOPs and develop its convergence analysis. By selecting different surrogate functions, the unified method can be reduced to several existing first-order methods. It is worth noting that the gap between the surrogate and objective functions significantly affects the performance of descent methods, which plays a central role in majorization-minimization optimization. Specifically, the steepest descent method for VOPs exhibits slow convergence due to the large gap between the surrogate and objective functions. To address this issue, we develop an improved descent method with a tighter surrogate function, resulting in improved linear convergence, and the rate of convergence is determined by the condition number. Interestingly, we show that selecting a tighter surrogate function is equivalent to using an appropriate base in the direction-find subproblem (see Remarks 4.4 and 5.3). This provides a positive answer to the proposed question.

(iii) Theoretical results suggest a tighter surrogate function by using Barzilai-Borwein method, which motivates us to devise a Barzilai-Borwein descent method for VOPs (BBDVO) with polyhedral cones. By reformulating the subproblem, we observe that BBDVO is essentially the steepest descent method with an appropriately chosen base in the direction-finding subproblem (see Remark 6.3). Furthermore, a VOP with a polyhedral cone can be transformed into an MOP using a transform matrix, which is often used to define a specific polyhedral cone. We demonstrate that the performance of BBDVO is insensitive to the choice of the transform matrix, whereas the steepest descent method is highly sensitive to it. From a majorizationminimization perspective, we further elucidate why the line search procedure in VOPs leads to a slower linear convergence rate compared with its counterpart without line search. In contrast, the backtracking strategy preserves the same linear convergence rate as first-order methods for VOPs without line search.

The remainder of this paper is organized as follows. Section 2 introduces the necessary notations and definitions for later use. In Section 3, we present a generic majorization-minimization descent method for VOPs and analyze its convergence rates under various convexity assumptions. Sections 4 and 5 explore the connections between different descent methods from the perspective of majorization-minimization. Section 6 proposes a Barzilai-Borwein descent method for VOPs with a polyhedral cone. Section 7 provides numerical results to demonstrate the efficiency of the BBDVO. Finally, conclusions are presented at the end of the paper.

2. Preliminaries

Throughout this paper, $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$ denote the set of n-dimensional real column vectors and the set of $m\times n$ real matrices, respectively. The space $\mathbb{R}^n$ is equipped with the inner product $\langle\cdot,\cdot\rangle$ and the induced norm $\|\cdot\|$. The interior, boundary and the closure of a set are denoted by $\mathrm{int}(\cdot)$, $\mathrm{bd}(\cdot)$ and $\mathrm{cl}(\cdot)$, respectively. The cone generated by a set is denoted by $\mathrm{cone}(\cdot)$. For simplicity, we denote $[m]:=\{1,2,...,m\}$, $\mathbf{1}_m$ and I_m the all-ones vector in $\mathbb{R}^m$ and identity matrix in $\mathbb{R}^{m\times m}$, respectively.

Since K is a closed convex cone, it follows that $K=K^{**}$ (see (Rockafellar, 1970, Theorem 14.1)),

$$K=\{y\in\mathbb{R}^m:\langle y,c^*\rangle\geq 0,\forall c^*\in K^*\},$$

and

$$\operatorname{int}(K) = \{y \in \mathbb{R}^m : \langle y, c^* \rangle > 0, \forall c^* \in K^* \setminus \{0\}\}.$$

Since $\operatorname{int}(K) \neq \emptyset$, we assume that there exists a compact and convex set C such that

$$0 \notin C, \tag{1}$$
$$\operatorname{cone}(C) = K^*. \tag{2}$$

Therefore

$$K = \{y \in \mathbb{R}^m : \langle y, c^* \rangle \geq 0, \forall c^* \in C\}, \tag{3}$$
$$\operatorname{int}(K) = \{y \in \mathbb{R}^m : \langle y, c^* \rangle > 0, \forall c^* \in C\}. \tag{4}$$

The latter equality, together with the compactness of C, implies that

$$\min_{c^* \in C}\{\langle c^*, y \rangle\} > 0, \quad \forall y \in \operatorname{int}(K). \tag{5}$$

For more details on C, we refer the readers to (Graña Drummond & Svaiter, 2005, pp. 400).

2.1. Vector optimization

In the subsection, we revisit some definitions and results pertinent to VOPs. Firstly, we introduce the concept of efficiency.

Definition 2.1. (Jahn, 2011, Definition 11.3) *A vector $x^* \in \mathbb{R}^n$ is called efficient solution to* (VOP) *if there exists no $x \in \mathbb{R}^n$ such that $F(x) \preceq_K F(x^*)$ and $F(x) \neq F(x^*)$.*

Definition 2.2. (Jahn, 2011, Definition 11.5) *A vector $x^* \in \mathbb{R}^n$ is called weakly efficient solution to* (VOP) *if there exists no $x \in \mathbb{R}^n$ such that $F(x) \prec_K F(x^*)$.*

Definition 2.3. (Graña Drummond & Svaiter, 2005) *A vector $x^* \in \mathbb{R}^n$ is called K-stationary point to* (VOP) *if*

$$\operatorname{range}(JF(x^*)) \cap (-\operatorname{int}(K)) = \emptyset,$$

where $\operatorname{range}(JF(x^*))$ *denotes the range of linear mapping given by the matrix $JF(x^*)$.*

Definition 2.4. (Graña Drummond & Svaiter, 2005) *A vector $d \in \mathbb{R}^n$ is called K-descent direction for $F(\cdot)$ at x if*

$$JF(x)d \in -\operatorname{int}(K).$$

Remark 2.1. *Note that if $x \in \mathbb{R}^n$ is a non-stationary point, then there exists a K-descent direction $d \in \mathbb{R}^n$ such that $JF(x)d \in -\operatorname{int}(K)$.*

Next, we introduce the concept of K-convexity for $F(\cdot)$.

Definition 2.5. (Jahn, 2011, Definition 2.4) *The objective function $F(\cdot)$ is called K-convex if*

$$F(\lambda x + (1-\lambda)y) \preceq_K \lambda F(x) + (1-\lambda)F(y)$$

holds for all $x, y \in \mathbb{R}^n$, $\lambda \in [0,1]$.

By the continuous differentiability of $F(\cdot)$, K-convexity of $F(\cdot)$ is equivalent to

$$JF(x)(y-x) \preceq_K F(y) - F(x)$$

holds for all $x, y \in \mathbb{R}^n$. (see (Jahn, 2011, Theorem 2.20)).

We conclude this section by elucidating the relationship between K-stationary points and weakly efficient solutions.

Lemma 2.1. (Jahn, 2011) *Assume that the objective function $F(\cdot)$ is K-convex, then $x^* \in \mathbb{R}^n$ is a K-stationary point of* (VOP) *if and only if x^* is a weakly efficient solution of* (VOP).

2.2. Strong convexity and smoothness

Strong convexity and smoothness of objective functions play a central role of first-order methods in optimization. This subsection is devoted to strong convexity and smoothness of vector-valued functions under partial order.

Definition 2.6. (Graña Drummond et al., 2014) *The objective function $F(\cdot)$ is called strongly K-convex with $\boldsymbol{\mu} \in K$ if*

$$F(\lambda x + (1-\lambda)y) \preceq_K \lambda F(x) + (1-\lambda)F(y) - \frac{1}{2}\lambda(1-\lambda)\left\|x-y\right\|^2 \boldsymbol{\mu},\ \forall x, y \in \mathbb{R}^n,\ \ \lambda \in [0,1],$$

and the above relation does not hold for any $\hat{\boldsymbol{\mu}}$ with $\hat{\boldsymbol{\mu}} \not\preceq_K \boldsymbol{\mu}$.

Remark 2.2. *Comparing with the definition in (Graña Drummond et al., 2014), Definition 2.6 includes the case $\boldsymbol{\mu} \in \mathrm{bd}(K)$, then it reduces K-convexity when $\boldsymbol{\mu} = 0$. Furthermore, tthe final statement in the definition establishes the uniqueness of the parameter $\boldsymbol{\mu}$, which is essential for the convergence analysis.*

Lemma 2.2. (Fliege et al., 2009, Theorem 3.1) *Assume that the objective function $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, then $x^* \in \mathbb{R}^n$ is a K-stationary point of* (VOP) *if and only if x^* is an efficient solution of* (VOP)*.*

By the continuous differentiability of $F(\cdot)$, strong K-convexity of $F(\cdot)$ is equivalent to

$$\frac{1}{2}\left\|x-y\right\|^2 \boldsymbol{\mu} + JF(x)(y-x) \preceq_K F(y) - F(x),\ \forall x, y \in \mathbb{R}^n,$$

it characterizes a quadratic lower-bound of $F(\cdot)$. Intuitively, we use quadratic upper-bound to define the K-smoothness of $F(\cdot)$ under partial order.

Definition 2.7. *The objective function $F(\cdot)$ is called K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$ if*

$$F(y) - F(x) \preceq_K JF(x)(y-x) + \frac{1}{2}\left\|x-y\right\|^2 \boldsymbol{\ell},\ \forall x, y \in \mathbb{R}^n,$$

and the above relation does not hold for any $\hat{\boldsymbol{\ell}}$ with $\boldsymbol{\ell} \not\preceq_K \hat{\boldsymbol{\ell}}$.

Remark 2.3. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in K$ and K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$, then $\boldsymbol{\mu} \preceq_K \boldsymbol{\ell}$.*

Remark 2.4. *Comparing with the smoothness and strong convexity in (Chen et al., 2022, Definitions 7 and 8) with Euclidean distance, i.e., $\omega(\cdot) = \frac{1}{2}\left\|\cdot\right\|^2$, Definitions 2.6 and 2.7 are tighter and do not depend on the reference vector e.*

Next, we characterize the properties of the difference of two vector-valued functions.

Lemma 2.3 (regularity of residual functions)**.** *Let $F, G : \mathbb{R}^n \to \mathbb{R}^m$ be two vector-valued functions. Define $H(\cdot) := G(\cdot) - F(\cdot)$. Then the following statements hold.*

(i) if $G(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$ and $F(\cdot)$ is K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$, where $\boldsymbol{\ell} \preceq \boldsymbol{\mu}$, then $H(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} - \boldsymbol{\ell}$;

(ii) if $G(\cdot)$ is K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$ and $F(\cdot)$ is K-convex, then $H(\cdot)$ is K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$.

(iii) if $G(\cdot)$ K-smooth with $\boldsymbol{\ell} \in \mathrm{int}(K)$ and $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, where $\boldsymbol{\mu} \preceq \boldsymbol{\ell}$, then $H(\cdot)$ is K-smooth with $\boldsymbol{\ell} - \boldsymbol{\mu}$.

Proof. The proof is a consequence of the definition of strong K-convexity and K-smoothness, we omit it here. □

In SOPs, the condition number (the quotient of smoothness parameter and the modulus of strong convexity) plays a key role in the geometric convergence of first-order methods. We end this section with the definition of the condition number of a strongly K-convex function under partial order.

Definition 2.8. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$ and K-smooth with $\boldsymbol{\ell} \in \operatorname{int}(K)$. Then, we denote*

$$\kappa_{F,\preceq_K} := \max_{c^* \in C} \frac{\langle c^*, \boldsymbol{\ell} \rangle}{\langle c^*, \boldsymbol{\mu} \rangle} \tag{6}$$

the condition number of $F(\cdot)$ under partial order $\preceq_K$.

Remark 2.5. *Notice that $0 \notin C$ and $K^* = \operatorname{cone}(C)$, the condition number can be rewritten as follows:*

$$\kappa_{F,\preceq_K} := \max_{c^* \in K^* \setminus \{0\}} \frac{\langle c^*, \boldsymbol{\ell} \rangle}{\langle c^*, \boldsymbol{\mu} \rangle}.$$

In other words, the condition number of $F(\cdot)$ is determined only by K, not C.

In the following, we will show that the condition number can be reduced to that of MOPs (Chen et al., 2023b).

Proposition 2.1. *For any $\boldsymbol{\ell}, \boldsymbol{\mu} \in \mathbb{R}^m_{++}$, we have*

$$\max_{\lambda \in \Delta_m} \frac{\sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i}{\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i} = \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

Proof. Since $\boldsymbol{\ell}, \boldsymbol{\mu} \in \mathbb{R}^m_{++}$, for any $i \in [m]$, we have

$$\boldsymbol{\ell}_i \leq \boldsymbol{\mu}_i \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

Multiply by $\lambda_i \geq 0$ and sum over $i \in [m]$:

$$\sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i \leq \left(\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i \right) \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

Dividing by the positive $\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i$ yields

$$\frac{\sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i}{\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i} \leq \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

Therefore, the relation

$$\max_{\lambda \in \Delta_m} \frac{\sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i}{\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i} \leq \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}$$

holds due to the arbitrary of λ. Let s be an index where the maximum ratio is attained, i.e.

$$\frac{\boldsymbol{\ell}_s}{\boldsymbol{\mu}_s} = \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

Take $\lambda_s = 1$ and $\lambda_i = 0$ for $i \neq s$, we have

$$\max_{\lambda \in \Delta_m} \frac{\sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i}{\sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i} \geq \frac{\boldsymbol{\ell}_s}{\boldsymbol{\mu}_s} = \max_{i \in [m]} \frac{\boldsymbol{\ell}_i}{\boldsymbol{\mu}_i}.$$

This completes the proof. □

Remark 2.6. *If $K = \mathbb{R}^m_+$, then $\kappa_{F,\preceq_K} = \max_{\lambda \in \Delta_m} \sum_{i \in [m]} \lambda_i \boldsymbol{\ell}_i / \sum_{i \in [m]} \lambda_i \boldsymbol{\mu}_i$. By Proposition 2.1, it follows that $\kappa_{F,\preceq_K} = \max_{i \in [m]} \boldsymbol{\ell}_i / \boldsymbol{\mu}_i$. In other words, the condition number for multiobjective optimization is the largest condition number among all objective functions.*

Proposition 2.2. *Assume that $F(\cdot)$ is strongly K_1-convex with $\boldsymbol{\mu}_1 \in \operatorname{int}(K_1)$ and K_1-smooth with $\boldsymbol{\ell}_1 \in \operatorname{int}(K_1)$. Then, for any order cone K_2 satisfied $K_1 \subset K_2$, we have $F(\cdot)$ is strongly K_2-convex with $\boldsymbol{\mu}_2 \in \operatorname{int}(K_2)$ and K_2-smooth with $\boldsymbol{\ell}_2 \in \operatorname{int}(K_2)$. Futhermore, $\boldsymbol{\ell}_2 \preceq_{K_2} \boldsymbol{\ell}_1$, $\boldsymbol{\mu}_1 \preceq_{K_2} \boldsymbol{\mu}_2$, and $\kappa_{F,\preceq_{K_2}} \leq \kappa_{F,\preceq_{K_1}}$.*

Proof. The proof is a consequence of the definitions of strong K-convexity, K-smoothness and condition number, we omit it here. □

Remark 2.7. *Proposition 2.2 shows that, for a fixed vector-valued function, enlarging the underlying order cone effectively simplifies the associated vector optimization problem; see Lemma 4.2 for a formal statement. This observation motivates the use of a larger order cone to accelerate first-order methods for VOPs.*

3. Majorization-minimization with first-order surrogate functions for VOPs

3.1. Majorization-minimization descent method for VOPs

In this section, we present a unified majorization-minimization scheme for minimizing a vector-valued function in the sense of descent.

Algorithm 1: Unified majorization-minimization scheme for VOPs

```
Data: $x^0 \in \mathbb{R}^n$
for $k = 0, 1, ...$ do
    Choose a strongly $K$-convex surrogate function $G_k(\cdot)$ of $F(\cdot) - F(x^k)$ near $x^k$
    Choose a base $C_k$ of dual cone $K^*$
    Update $x^{k+1} := \arg\min_{x \in \mathbb{R}^n} \max_{c^* \in C_k} \langle c^*, G_k(x) \rangle$
    if $x^{k+1} = x^k$ then
        return $K$-stationary point $x^k$
    end
end
```

Remark 3.1. *It is worth noting that we choose a variable base C_k in each iteration, whreas an invariable base C is used in existing descent methods for VOPs, see (Graña Drummond & Svaiter, 2005; Graña Drummond et al., 2014; Graña Drummond & Iusem, 2004; Bonnel et al., 2005; Lucambio Pérez & Prudente, 2018; Chen et al., 2023c).*

The surrogate function $G_k(\cdot)$ plays a central role in the generic majorization-minimization scheme. Intuitively, $G_k(\cdot)$ should well approximate $F(\cdot) - F(x^k)$ near x^k and the related subproblem should be easy to minimize. Therefore, we measure the approximation error by $H_k(\cdot) := G_k(\cdot) - F(\cdot) + F(x^k)$. To characterize surrogates, we introduce a class of surrogate functions, which will be used to establish the convergence results of Algorithm 1.

Definition 3.1. *For $x^k \in \mathbb{R}^n$, we call $G_k(\cdot)$ a first-order surrogate function of $F(\cdot) - F(x^k)$ near x^k when*

(i) $F(x^{k+1}) - F(x^k) \preceq_K G_k(x^{k+1})$, where x^{k+1} is the minimizer of $\min_{x \in \mathbb{R}^n} \max_{c^* \in C_k} \langle c^*, G_k(x) \rangle$, furthermore, when $F(\cdot) - F(x^k) \preceq_K G_k(\cdot)$ for all $x \in \mathbb{R}^n$, we call $G_k(\cdot)$ a majorizing surrogate;

(ii) the approximation error $H_k(\cdot)$ is K-smooth with $\boldsymbol{\ell} \in \operatorname{int}(K)$, $H_k(x^k) = 0$, and $JH_k(x^k) = 0$.

We denote by $\mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$ the set of first-order strongly K-convex surrogate functions with $\boldsymbol{\mu} \in \operatorname{int}(K)$.

Next, we characterize the properties of first-order surrogate functions.

Lemma 3.1. *Let $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$ and x^{k+1} be the minimizer of $\min_{x \in \mathbb{R}^n} \max_{c^* \in C_k} \langle c^*, G_k(x) \rangle$. Then, for all $x \in \mathbb{R}^n$, we have*

(i) $H_k(x) \preceq_K \frac{1}{2} \left\| x - x^k \right\|^2 \boldsymbol{\ell}$;

(ii) $\left\langle c_k^*, F(x^{k+1})\right\rangle + \frac{1}{2}\left\|x^{k+1}-x\right\|^2 \langle c_k^*, \boldsymbol{\mu}\rangle \leq \langle c_k^*, F(x)\rangle + \frac{1}{2}\left\|x^k - x\right\|^2 \langle c_k^*, \boldsymbol{\ell}\rangle$, *where* c_k^* *is a maximizer of* $\max_{c^*\in C_k}\min_{x\in\mathbb{R}^n}\langle c^*, G_k(x)\rangle$.

Proof. Assertion (i) directly follows by the K-smoothness of $H_k(\cdot)$ and the facts that $H_k(x^k) = 0$ and $JH_k(x^k) = 0$. Next, we prove the assertion (ii). By Sion's minimax theorem (Sion, 1958), by denoting c_k^* a maximizer of $\max_{c^*\in C_k}\min_{x\in\mathbb{R}^n}\langle c^*, G_k(x)\rangle$, and x^{k+1} the minimizer of $\min_{x\in\mathbb{R}^n}\max_{c^*\in C_k}\langle c^*, G_k(x)\rangle$, we have $JG_k(x^{k+1})^T c_k^* = 0$. This, together with the strong K-convexity of $G_k(\cdot)$, implies that

$$\left\langle c_k^*, G_k(x^{k+1})\right\rangle + \frac{1}{2}\left\|x^{k+1}-x\right\|^2 \langle c_k^*, \boldsymbol{\mu}\rangle \leq \langle c_k^*, G_k(x)\rangle\,,\ \forall x\in\mathbb{R}^n.$$

We thus use $F(x^{k+1}) - F(x^k) \preceq_K G_k(x^{k+1})$ to get

$$\begin{aligned}\left\langle c_k^*, F(x^{k+1}) - F(x^k)\right\rangle + \frac{1}{2}\left\|x^{k+1}-x\right\|^2 \langle c_k^*, \boldsymbol{\mu}\rangle &\leq \left\langle c_k^*, G_k(x^{k+1})\right\rangle + \frac{1}{2}\left\|x^{k+1}-x\right\|^2 \langle c_k^*, \boldsymbol{\mu}\rangle \\ &\leq \langle c_k^*, G_k(x)\rangle \\ &= \left\langle c_k^*, F(x) - F(x^k)\right\rangle + \langle c_k^*, H_k(x)\rangle \\ &\leq \left\langle c_k^*, F(x) - F(x^k)\right\rangle + \frac{1}{2}\left\|x^k - x\right\|^2 \langle c_k^*, \boldsymbol{\ell}\rangle\,,\end{aligned}$$

where the equality follows by the definition of $H_k(\cdot)$ and the last inequality is due to the assertion (i). This completes the proof. □

3.2. Convergence analysis

In Algorithm 1, it can be observed that it terminates either with a K-stationary point in a finite number of iterations or generates an infinite sequence of non-stationary points. In the subsequent analysis, we will assume that the algorithm produces an infinite sequence of non-stationary points.

3.2.1. Global convergence

Firstly, we establish the global convergence result in the nonconvex setting, the following assumptions are required.

Assumption 3.1. *Assume the following statements hold for* $F(\cdot)$ *and* $G_k(\cdot)$*:*

(i) The level set $\mathcal{L}_F(x^0) := \{x : F(x) \preceq_K F(x^0)\}$ *is bounded;*

(ii) There exist two compact bases C_L *and* $\tilde{C}$ *of* K^* *such that*

$$C_k \subset \tilde{C},\ \textit{and}\ \max_{c^*\in C_k}\langle c^*, y\rangle \geq \max_{c^*\in C_L}\langle c^*, y\rangle$$

hold for all $k \geq 0$ *and* $y \in -K$.

(iii) If C *is a compact base of* K^*, $x^k \to x^*$, $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$ *and* $\min_{y\in\mathbb{R}^n}\max_{c^*\in C}\langle c^*, G_k(y)\rangle \to 0$, *then* x^* *is a* K*-stationary point to (VOP).*

Remark 3.2. *Assumption 3.1(i) is a standard condition for nonconvex cases. Moreover, Assumption 3.1(ii) requires the sequence* $\{C_k\}$ *to be uniformly bounded, which is a mild assumption. Assumption 3.1(iii) holds for* K*-steepest descent method (Graña Drummond & Svaiter, 2005). Specifically,* $\min_{y\in\mathbb{R}^n}\max_{c^*\in C}\langle c^*, G_k(y)\rangle \to 0$ *takes the form of* α_{x_k} *(Graña Drummond & Svaiter, 2005, Definition 3.2), where* α_x *is continuous (Graña Drummond & Svaiter, 2005, Lemma 3.3(3)) and* $\alpha_x = 0$ *if and only if* x *is a* K*-stationary point to (VOP) (Graña Drummond & Svaiter, 2005, Lemma 3.3(1)).*

We are now in a position to establish the global convergence of Algorithm 1.

Theorem 3.1. *Suppose that Assumption 3.1 holds, let* $\{x^k\}$ *be the sequence generated by Algorithm 1 with* $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$. *Then,* $\{x^k\}$ *has at least one accumulation point and every accumulation point is a non-stationary point to (VOP).*

Proof. Since $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$, we have

$$F(x^{k+1}) - F(x^k) \preceq_K G_k(x^{k+1}), \tag{7}$$

and

$$G_k(x^{k+1}) \preceq_K G_k(x^k) = H_k(x^k) = 0.$$

Then, we conclude that $\{F(x^k)\}$ is decreasing under partial order $\preceq_K$. It follows by Assumption 3.1(i) and continuity of $F(\cdot)$ that $\{x^k\}$ is bounded and there exists F^* such that

$$F^* \preceq_K F(x^k),\ \forall k \geq 0.$$

The boundedness of $\{x^k\}$ indicates that $\{x^k\}$ has at least one accumulation point. Next, we prove that any accumulation point x^* is a non-stationary point. By summing (7) from 0 to infinity, we have

$$F^* - F(x^0) \preceq_K \sum_{k=0}^{\infty}(F(x^{k+1}) - F(x^k)) \preceq_K \sum_{k=0}^{\infty} G_k(x^{k+1}).$$

It follows that

$$\begin{aligned}
\sum_{k=0}^{\infty} \max_{c^* \in C_k} \left\langle c^*, G_k(x^{k+1}) \right\rangle &\geq \sum_{k=0}^{\infty} \max_{c^* \in C_L} \left\langle c^*, G_k(x^{k+1}) \right\rangle \\
&\geq \max_{c^* \in C_L} \left\langle c^*, \sum_{k=0}^{\infty} G_k(x^{k+1}) \right\rangle \\
&\geq \max_{c^* \in C_L} \left\langle c^*, F^* - F(x^0) \right\rangle \\
&\geq -\infty,
\end{aligned}$$

where the first inequality follows by Assumption 3.1(ii) and $G_k(x^{k+1}) \preceq_K 0$, the second inequality is due to the fact $\max_x f_1(x) + \max_x f_2(x) \geq \max_x\{f_1(x) + f_2(x)\}$. This, together with the fact that $G_k(x^{k+1}) \preceq_K G_k(x^k) = 0$, implies $\max_{c^* \in C_k} \left\langle c^*, G_k(x^{k+1}) \right\rangle \to 0$. A direct calculation gives:

$$0 = \max_{c^* \in \tilde{C}} \left\langle c^*, G_k(x^k) \right\rangle \geq \min_{y \in \mathbb{R}^n} \max_{c^* \in \tilde{C}} \langle c^*, G_k(y) \rangle \geq \min_{y \in \mathbb{R}^n} \max_{c^* \in C_k} \langle c^*, G_k(y) \rangle = \max_{c^* \in C_k} \left\langle c^*, G_k(x^{k+1}) \right\rangle \to 0,$$

where the second inequality follows by $C_k \subset \tilde{C}$. Therefore, $\min_{y \in \mathbb{R}^n} \max_{c^* \in \tilde{C}} \langle c^*, G_k(y) \rangle \to 0$. For the accumulation point x^*, there exists an infinite index set $\mathcal{K}$ such that $x^k \xrightarrow{\mathcal{K}} x^*$. By Assumption 3.1(iii), we conclude that x^* is a K-stationary point. □

3.2.2. Strong convergence

In the following, we establish the strong convergence result of Algorithm 3.1 in K-convex setting.

Theorem 3.2. *Suppose that Assumption 3.1 holds and $F(\cdot)$ is K-convex, let $\{x^k\}$ be the sequence generated by Algorithm 1 with $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$ and $0 \preceq_K \boldsymbol{\ell} = \boldsymbol{\mu}$. Then, the following statements hold:*

(i) $\{x^k\}$ converges to a weakly efficient solution x^* of (VOP);

(ii) $u_0(x^k) \leq \frac{\ell_{\max} R^2}{2k}$, $\forall k \geq 1$, where $\ell_{\max} := \max_{c^* \in \tilde{C}} \langle c^*, \boldsymbol{\ell} \rangle$, $R := \{\|x - y\| : x, y \in \mathcal{L}_F(x^0)\}$, and

$$u_0(x^k) := \max_{x \in \mathbb{R}^n} \min_{c^* \in \tilde{C}} \left\langle c^*, F(x^k) - F(x) \right\rangle$$

is a merit function in the sense of weak efficiency.

Proof. (i) By the similar arguments in the proof of Theorem 3.1, we conclude that $\{x^k\}$ is bounded, and there exists a K-stationary point x^* such that $F(x^*) \preceq_K F(x^k)$. Besides, the K-convexity of $F(\cdot)$ indicates that x^* is a weakly efficient point. From Lemma 3.1(ii), for any $x \in \mathbb{R}^n$ we have

$$\left\langle c_k^*, F(x^{k+1}) - F(x) \right\rangle \leq \frac{1}{2} \left\| x^k - x \right\|^2 \langle c_k^*, \boldsymbol{\ell} \rangle - \frac{1}{2} \left\| x^{k+1} - x \right\|^2 \langle c_k^*, \boldsymbol{\mu} \rangle . \tag{8}$$

Substituting $x = x^*$ into the above inequality, we obtain

$$\left\langle c_k^*, F(x^{k+1}) - F(x^*) \right\rangle \leq \frac{1}{2} \left\| x^k - x^* \right\|^2 \langle c_k^*, \boldsymbol{\ell} \rangle - \frac{1}{2} \left\| x^{k+1} - x^* \right\|^2 \langle c_k^*, \boldsymbol{\mu} \rangle .$$

Recall that $F(x^*) \preceq_K F(x^k)$, it follows that

$$\left\| x^{k+1} - x^* \right\|^2 \langle c_k^*, \boldsymbol{\mu} \rangle \leq \left\| x^k - x^* \right\|^2 \langle c_k^*, \boldsymbol{\ell} \rangle .$$

Furthermore, we use the fact that $0 \preceq_K \boldsymbol{\ell} = \boldsymbol{\mu}$ to get

$$\left\| x^{k+1} - x^* \right\|^2 \leq \left\| x^k - x^* \right\|^2 .$$

Therefore, the sequence $\{\left\| x^k - x^* \right\|\}$ converges. This, together with the fact that x^* is an accumulation point of $\{x^k\}$, implies that $\{x^k\}$ converges to x^*

(ii) Since $0 \preceq_K \boldsymbol{\ell} = \boldsymbol{\mu}$, we use inequality (8) to obtain

$$\left\langle c_k^*, F(x^{k+1}) - F(x) \right\rangle \leq \frac{1}{2} \left\| x^k - x \right\|^2 \langle c_k^*, \boldsymbol{\ell} \rangle - \frac{1}{2} \left\| x^{k+1} - x \right\|^2 \langle c_k^*, \boldsymbol{\mu} \rangle = \frac{\langle c_k^*, \boldsymbol{\ell} \rangle}{2} \left(\left\| x^k - x \right\|^2 - \left\| x^{k+1} - x \right\|^2 \right). \tag{9}$$

Taking the sum of the preceding inequality over 0 to $k-1$, we have

$$\sum_{s=0}^{k-1} \left\langle c_s^*, F(x^{s+1}) - F(x) \right\rangle \leq \sum_{s=0}^{k-1} \frac{\langle c_s^*, \boldsymbol{\ell} \rangle}{2} \left(\| x^s - x \|^2 - \left\| x^{s+1} - x \right\|^2 \right).$$

Notice that $F(x^k) \preceq_K F(x^{s+1})$ for all $s \leq k-1$, it leads to

$$\sum_{s=0}^{k-1} \left\langle c_s^*, F(x^{k}) - F(x) \right\rangle \leq \sum_{s=0}^{k-1} \frac{\langle c_s^*, \boldsymbol{\ell} \rangle}{2} \left(\| x^s - x \|^2 - \left\| x^{s+1} - x \right\|^2 \right).$$

Denote $\hat{c}_k^* := \sum_{s=0}^{k-1} c_s^*/k$. It follows from the convexity of $\tilde{C}$ and the fact that $c_s^* \in \tilde{C}$ that $\hat{c}_k^* \in \tilde{C}$. Therefore, we conclude that

$$\left\langle \hat{c}_k^*, F(x^{k}) - F(x) \right\rangle \leq \sum_{s=0}^{k-1} \frac{\langle c_s^*, \boldsymbol{\ell} \rangle}{2k} \left(\| x^s - x \|^2 - \left\| x^{s+1} - x \right\|^2 \right).$$

Select $y^k \in \arg\max_{x \in \mathbb{R}^n} \min_{c^* \in \tilde{C}} \left\langle c^*, F(x^k) - F(x) \right\rangle$, it holds that

$$\begin{aligned} u_0(x^k) &= \max_{x \in \mathbb{R}^n} \min_{c^* \in \tilde{C}} \left\langle c^*, F(x^k) - F(x) \right\rangle = \min_{c^* \in \tilde{C}} \left\langle c^*, F(x^k) - F(y^k) \right\rangle \\ &\leq \left\langle \hat{c}_k^*, F(x^k) - F(y^k) \right\rangle \leq \sum_{s=0}^{k-1} \frac{\langle c_s^*, \boldsymbol{\ell} \rangle}{2k} \left(\left\| x^s - y^k \right\|^2 - \left\| x^{s+1} - y^k \right\|^2 \right). \end{aligned}$$

By the definition of y^k, we deduce that $y^k \in \{x : F(x) \preceq_K F(x^k)\} \subset \mathcal{L}_F(x^s)$ for all $s \leq k-1$. Substituting this relation into (9), we have $\left\| x^s - y^k \right\|^2 - \left\| x^{s+1} - y^k \right\|^2 \geq 0$ for all $s \leq k-1$. Therefore,

$$u_0(x^k) \leq \frac{\ell_{\max}}{2k} \sum_{s=0}^{k-1} \left(\left\| x^s - y^k \right\|^2 - \left\| x^{s+1} - y^k \right\|^2 \right) \leq \frac{\ell_{\max} \left\| x^0 - y^k \right\|^2}{2k}.$$

Recall that $y^k \in \mathcal{L}_F(x^0)$, the desired result follows. □

3.2.3. Linear convergence

By further assuming that $F(\cdot)$ is strongly K-convex, the linear convergence result of Algorithm 1 can be derived as follows.

Theorem 3.3. *Suppose that Assumption 3.1(ii) holds and $F(\cdot)$ is strongly K-convex, let $\{x^k\}$ be the sequence generated by Algorithm 1 with $G_k(\cdot) \in \mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\mu}}(F, x^k)$ and $0 \preceq_K \boldsymbol{\ell} \prec_K \boldsymbol{\mu}$. Then, the following statements hold:*

(i) *$\{x^k\}$ converges to an efficient solution x^* of (VOP);*

(ii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{\max\limits_{c^* \in C_k} \frac{\langle c^*, \boldsymbol{\ell}\rangle}{\langle c^*, \boldsymbol{\mu}\rangle}} \left\|x^k - x^*\right\|, \ \forall k \geq 0.$

Proof. (i) Since $F(\cdot)$ is strongly K-convex, then Assumption 3.1(i) holds and every weakly efficient solution is actually an efficient solution. Therefore, assertion (i) is a consequence of Theorem 3.2(i).

(ii) By substituting $x = x^*$ into inequality (8), we have

$$\left\langle c_k^*, F(x^{k+1}) - F(x^*)\right\rangle \leq \frac{1}{2} \left\|x^k - x^*\right\|^2 \langle c_k^*, \boldsymbol{\ell}\rangle - \frac{1}{2} \left\|x^{k+1} - x^*\right\|^2 \langle c_k^*, \boldsymbol{\mu}\rangle .$$

It follows by $F(x^*) \preceq_K F(x^{k+1})$ that

$$\left\|x^{k+1} - x^*\right\| \leq \sqrt{\frac{\langle c_k^*, \boldsymbol{\ell}\rangle}{\langle c_k^*, \boldsymbol{\mu}\rangle}} \left\|x^k - x^*\right\| .$$

The desired result follows . □

Remark 3.3. *It seems that the convexity of $F(\cdot)$ plays no role in the proof of Theorems 3.2 and 3.3. However, it can indeed be shown that $F(\cdot)$ is necessarily K-convex if $\boldsymbol{\ell} = \boldsymbol{\mu}$ and strongly K-convex with $\boldsymbol{\mu} - \boldsymbol{\ell}$ if $\boldsymbol{\ell} \prec_K \boldsymbol{\mu}$. In the next section, we will give some examples where such a condition holds.*

Remark 3.4. *Note that $0 \notin C_k$ and cone$(C_k) = K^*$, we have*

$$\max_{c^* \in C_k} \frac{\langle c^*, \boldsymbol{\ell}\rangle}{\langle c^*, \boldsymbol{\mu}\rangle} = \max_{c^* \in K^* \setminus \{0\}} \frac{\langle c^*, \boldsymbol{\ell}\rangle}{\langle c^*, \boldsymbol{\mu}\rangle}.$$

Therefore, the linear convergence rate is related to $\{G_k(\cdot)\}$, not $\{C_k\}$, which confirms that the rate of convergence can be improved by choosing a tighter surrogate.

4. First-order methods for VOPs with majorizing surrogate functions

It is worth noting that Remark 3.4 may suggest that the choice of the base in the subproblem is inessential, which could make the question (Q) raised in the introduction seem trivial. However, as will be demonstrated in this section, the selection of such a base not only influences the convergence rate but also determines the computational complexity of the subproblem. Both aspects are of central importance in the framework of majorization-minimization optimization.

In what follows, we first revisit the classical steepest descent method for VOPs (SDVO) (Graña Drummond & Svaiter, 2005) and establish its connection with Algorithm 1. To mitigate the slow convergence of SDVO, we then investigate, from a majorization-minimization perspective, how an appropriate choice of the base can effectively accelerate first-order methods for VOPs.

4.1. K-steepest descent method for VOPs without line search

For $x \in \mathbb{R}^n$, recall that d^k, the K-steepest descent direction (Graña Drummond & Svaiter, 2005) at x^k, is defined as the optimal solution of

$$\min_{d \in \mathbb{R}^n} \max_{c^* \in C} \left\langle c^*, JF(x^k)d\right\rangle + \frac{1}{2}\|d\|^2. \tag{10}$$

Select a vector $e \in \operatorname{int}(K)$, and denote $C_e = \{c^* \in K^* : \langle c^*, e\rangle = 1\}$. If we set $C = C_e$ in (10), then the K-steepest descent direction can be reformulated as the optimal solution of

$$\min_{d\in\mathbb{R}^n} \max_{c^*\in C_e} \left\langle c^*, JF(x^k)d + \frac{1}{2}\|d\|^2 e \right\rangle. \tag{11}$$

Remark 4.1. *If $K = \mathbb{R}^m_+$, and $C_e = \Delta_m$, then $e = \mathbf{1}_m$ and the subproblem (11) reduces to that of steepest descent method for MOPs (Fliege & Svaiter, 2000). In what follows, we refer to subproblems of the forms (10) and (11) as **seperate** and **coupled** subproblems, respectively.*

From now on, we assume that $F(\cdot)$ is K-smooth with $\boldsymbol{\ell} \in \operatorname{int}(K)$, denote

$$L_{\max} := \max_{c^*\in C_e} \langle c^*, \boldsymbol{\ell}\rangle.$$

Let us revisit the K-steepest descent method without line search:

Algorithm 2: K-steepest descent method for VOPs

```
Data: $x^0 \in \mathbb{R}^n, L \geq L_{\max}$
for $k = 0, 1, ...$ do
    Update $x^{k+1} := \arg\min_{x\in\mathbb{R}^n} \max_{c^*\in C_e} \left\langle c^*, JF(x^k)(x - x^k)\right\rangle + \frac{L}{2}\|x - x^k\|^2$
    if $x^{k+1} = x^k$ then
        return $K$-stationary point $x^k$
    end
end
```

We consider the following surrogate:

$$G_{k,Le}(x) := JF(x^k)(x - x^k) + \frac{L}{2}\left\|x - x^k\right\|^2 e. \tag{12}$$

It is obvious that Algorithm 2 is a special case of Algorithm 1 with $C_k = C_e$ and $G_k(\cdot) = G_{k,Le}(\cdot)$. As described in Remark 3.4, the peformance of Algorithm 2 is mainly attributed to $G_{k,Le}(\cdot)$. The following results show that $G_{k,Le}(\cdot)$ is a majorizing surrogate function of $F(\cdot) - F(x^k)$ near x^k.

Proposition 4.1. *Let $G_{k,Le}(\cdot)$ be defined as (12). Then, the following statements hold.*

(i) For any $L \geq L_{\max}$, $G_{k,Le}(\cdot)$ is a majorizing surrogate of $F(\cdot) - F(x^k)$, i.e., $F(\cdot) - F(x^k) \preceq_K G_{k,Le}(\cdot)$.

(ii) If $F(\cdot)$ is K-convex, then $G_{k,Le}(\cdot) \in \mathcal{S}_{Le,Le}(F, x^k)$ for all $L \geq L_{\max}$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, then $G_{k,Le}(\cdot) \in \mathcal{S}_{Le-\boldsymbol{\mu},Le}(F, x^k)$ for all $L \geq L_{\max}$.

Proof. By the definition of $L_{\max}$, we have $\boldsymbol{\ell} \preceq_K L_{\max}e$, it follows from the K-smoothness of $F(\cdot)$ that assertion (i) holds. Notice that $G_{k,Le}(\cdot)$ is strongly K-convex and K-smooth with Le, and $\boldsymbol{\mu} \preceq_K Le$, then we obtain assertion (ii) and (iii) by Lemma 2.3 (ii) and (iii), respectively. □

Note that for a strongly K-convex objective function, $G_{k,Le}(\cdot) \in \mathcal{S}_{Le-\boldsymbol{\mu},Le}(F, x^k)$ for all $L \geq L_{\max}$. We are now in a position to present the rate of linear convergence for SDVO.

Lemma 4.1. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 2. Then, the following statements hold:*

(i) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(ii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - \mu_{\min}/L}\left\|x^k - x^*\right\|$, $\forall k \geq 0$, where $\mu_{\min} := \min_{c^*\in C_e} \langle c^*, \boldsymbol{\mu}\rangle$.

Proof. Since $F(\cdot)$ is strongly K-convex, it follows that $G_{k,Le}(\cdot) \in \mathcal{S}_{Le-\boldsymbol{\mu},Le}(F, x^k)$ and Assumption 3.1 holds in this case. By setting $C_k = C_e$, Theorem 3.3 (i) and (ii) reduce to the assertions (i) and (ii), respectively. □

Remark 4.2. *If $K=\mathbb{R}^m_+$ and $e=\mathbf{1}_m$, then $C_e=\Delta_m$ is a base of $\mathbb{R}^m_+$, the convergence rate in Lemma 4.1 reduces to that of (Tanabe et al., 2023, Theorem 5.3) with $g(\cdot)=0$. Specifically, the linear convergence rate is worse than $\mathcal{O}((\sqrt{1-\mu_{\min}/L_{\max}})^k)$ (setting $L=L_{\max}$), where $L_{\max}=\max_{i\in[m]}\{\boldsymbol{\ell}_i\}$ and $\mu_{\min}=\min_{i\in[m]}\{\boldsymbol{\mu}_i\}$. Therefore, even each of objective functions is well-conditioned ($\max_{i\in[m]}\{\boldsymbol{\ell}_i/\boldsymbol{\mu}_i\}$ is relative small), the linear convergence rate can be very slow due to objective imbalances ($L_{\max}/\mu_{\min}$ can be extremely large). It is worth noting that the rate of convergence is related to C_e, since in the seperate subproblem the surrogate function is inherently determined by C_e. To the best of our knowledge, apart from Δ_m, it remains an open problem for the better choice of the base in MOPs.*

4.2. Improved K-steepest descent method for VOPs without line search

As detailed in Remark 4.2, the linear convergence rate can be very slow with imbalanced objectives, this is mainly due to the large gap between $F(\cdot)-F(x^k)$ and $G_{k,Le}(\cdot)$ from a majorization-minimization perspective. To reduce this gap, one natural strategy is to construct a tighter surrogate function that better approximates the local behavior of $F(\cdot)-F(x^k)$. Notice that $\boldsymbol{\ell}\preceq_K L_{\max}e$, we denote the following tighter majorizing surrogate:

$$G_{k,\boldsymbol{\ell}}(x):=JF(x^k)(x-x^k)+\frac{1}{2}\left\|x-x^k\right\|^2\boldsymbol{\ell}. \tag{13}$$

The properties of $G_{k,\boldsymbol{\ell}}(\cdot)$ is presented as follows.

Proposition 4.2. *Let $G_{k,\boldsymbol{\ell}}(\cdot)$ be defined as (13). Then, the following statements hold.*

(i) $G_{k,\boldsymbol{\ell}}(\cdot)$ is a tight majorizing surrogate of $F(\cdot)-F(x^k)$, i.e., $F(\cdot)-F(x^k)\preceq_K G_{k,\boldsymbol{\ell}}(\cdot)$, and the relation does not hold for any $G_{k,\hat{\boldsymbol{\ell}}}(\cdot)$ such that $\boldsymbol{\ell}\not\preceq_K\hat{\boldsymbol{\ell}}$.

(ii) If $F(\cdot)$ is K-convex, then $G_{k,\boldsymbol{\ell}}(\cdot)\in\mathcal{S}_{\boldsymbol{\ell},\boldsymbol{\ell}}(F,x^k)$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu}\in\operatorname{int}(K)$, then $G_{k,\boldsymbol{\ell}}(\cdot)\in\mathcal{S}_{\boldsymbol{\ell}-\boldsymbol{\mu},\boldsymbol{\ell}}(F,x^k)$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 4.1. □

By using the tighter surrogate, we devise the following improved K-steepest descent method with coupled subproblems for VOPs.

Algorithm 3: improved K-steepest descent method for VOPs with coupled subproblems

```
Data: $x^0\in\mathbb{R}^n$
for $k=0,1,...$ do
    Update $x^{k+1}:=\arg\min_{x\in\mathbb{R}^n}\max_{c^*\in C_e}\left\langle c^*,JF(x^k)(x-x^k)+\frac{1}{2}\|x-x^k\|^2\boldsymbol{\ell}\right\rangle$
    if $x^{k+1}=x^k$ then
        return $K$-stationary point $x^k$
    end
end
```

Lemma 4.2. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu}\in\operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 3. Then, the following statements hold:*

(i) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(ii) $\left\|x^{k+1}-x^*\right\|\leq\sqrt{1-1/\kappa_{F,\preceq_K}}\left\|x^k-x^*\right\|,\ \forall k\geq 0$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Lemma 4.1. □

Remark 4.3. *If $K=\mathbb{R}^m_+$, and $e=\mathbf{1}_m$, the convergence rate in Lemma 4.2 reduces to that of (Chen et al., 2023b, Corollary 4.3) with $g(\cdot)=0$. Notice that $1/\kappa_{F,\preceq_K}\geq\mu_{\min}/L$, which indicates that Algorithm 3 enjoys faster linear convergence than Algorithm 2. Furthermore, by Remark 3.4, we conclude that the improved linear convergence does not depend on the choice of C_e.*

4.3. Trade-off between surrogate gap and per-iteration cost

Although Algorithms 3 exhibits improved linear convergence by using a tighter surrogate function, the per-iteration cost is more expensive than that of Algorithm 2 due to coupled subproblems. Details on solving these subproblems will be provided in Section 6 (see Remark 6.2). In the spirit of majorization-minimization optimization, a direct question arises: how to strike a better trade-off in terms of surrogate gap and per-iteration cost?

Recall that the linear convergence rate of Algorithm 3 does not depend on the choice of C_e, which is mainly due to coupled subproblems. By denoting

$$C_{\boldsymbol{\ell}} := \{c^* \in K^* : \langle c^*, \boldsymbol{\ell} \rangle = 1\},$$

we propose the following improved K-steepest descent method with seperate subproblems for VOPs.

Algorithm 4: improved K-steepest descent method for VOPs with seperate subproblems

```
Data: $x^0 \in \mathbb{R}^n$
for $k = 0, 1, \ldots$ do
    Update $x^{k+1} := \arg\min_{x\in\mathbb{R}^n} \max_{c^*\in C_{\boldsymbol{\ell}}} \left\langle c^*, JF(x^k)(x - x^k) \right\rangle + \frac{1}{2}\|x - x^k\|^2$
    if $x^{k+1} = x^k$ then
        return $K$-stationary point $x^k$
    end
end
```

Using the definition of $C_{\boldsymbol{\ell}}$, the seperate subproblem in Algorithm 4 can be rewritten equivalently as the following coupled form:

$$\min_{x\in\mathbb{R}^n} \max_{c^*\in C_{\boldsymbol{\ell}}} \left\langle c^*, JF(x^k)(x - x^k) + \frac{1}{2}\|x - x^k\|^2 \boldsymbol{\ell} \right\rangle.$$

Therefore, Algorithm 4 enjoys the same improved linear convergence as that of Algorithm 3.

Lemma 4.3. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 4. Then, the following statements hold:*

(i) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(ii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - 1/\kappa_{F,\preceq_K}} \left\|x^k - x^*\right\|$, $\forall k \geq 0$.

Remark 4.4. *If $K = \mathbb{R}^m_+$, and $\Delta^{\boldsymbol{\ell}}_m := \{c^* \in \mathbb{R}^m_{++} : \langle c^*, \boldsymbol{\ell} \rangle = 1\}$, the Algorithm 4 reduces to (Chen et al., 2023b, Algorithm 5) with $g(\cdot) = 0$. Interestingly, the relations between Algorithms 2, 3 and 4 depend solely on the choice of C_e. If $C_e = C_{\boldsymbol{\ell}}$, Algorithms 2, 3 and 4 are equivalent. Consequently, regarding the open problem of selecting a better C_e mentioned in Remark 4.2, we provide a theoretical answer by setting $C_e = C_{\boldsymbol{\ell}}$. Furthermore, we can summarize that choosing a tighter surrogate function is equivalent to selecting an appropriate base in the seperate subproblem.*

Remark 4.5. *Although Algorithms 3 and 4 both exhibit similar improved linear convergence, the computational cost of solving seperate subproblems is generally lower in Algorithm 4. Details on solving these subproblems will be provided in Section 6 (see Remark 6.2).*

5. First-order methods for VOPs with non-majorizing surrogate functions

In the previous section, the majorization-minimization optimization methods were developed using majorizing surrogate functions; however, these surrogates may be overly conservative due to reliance on global upper bounds. From the perspective of majorization-minimization, selecting a non-majorizing surrogate function could potentially enhance performance.

5.1. K-steepest descent method with line search

Firstly, we revisit K-steepest descent method for VOPs with line search.

Algorithm 5: K-steepest descent method for VOPs with line search

```
Data: $x^0 \in \mathbb{R}^n, \gamma \in (0,1)$
for $k = 0, 1, ...$ do
    Update $d^k := \arg\min_{d\in\mathbb{R}^n} \max_{c^*\in C_e} \left\langle c^*, JF(x^k)d \right\rangle + \frac{1}{2}\|d\|^2$
    if $d^k = 0$ then
        return $K$-stationary point $x^k$
    else
        Compute the stepsize $t_k \in (0,1]$ in the following way:
            $t_k := \max\left\{\gamma^j : j \in \mathbb{N}, F(x^k + \gamma^j d^k) - F(x^k) \preceq_K \gamma^j \left(JF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 e\right)\right\}$
        $x^{k+1} := x^k + t_k d^k$
    end
end
```

The stepsize has the following lower bound.

Proposition 5.1. The stepsize generated in Algorithm 5 satisfies $t_k \geq t_{\min} := \min\left\{\frac{\gamma}{L_{\max}}, 1\right\}$.

Proof. By the line search condition in Algorithm 5, we have

$$F(x^k + \frac{t_k}{\gamma}d^k) - F(x^k) \npreceq_K \frac{t_k}{\gamma}\left(JF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 e\right).$$

Then there exists $c_1^* \in C_e$ such that

$$\left\langle c_1^*, F(x^k + \frac{t_k}{\gamma}d^k) - F(x^k)\right\rangle > \left\langle c_1^*, \frac{t_k}{\gamma}\left(JF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 e\right)\right\rangle. \tag{14}$$

On the other hand, the K-smoothness of $F(\cdot)$ implies

$$F(x^k + \frac{t_k}{\gamma}d^k) - F(x^k) \preceq_K \frac{t_k}{\gamma}JF(x^k)d^k + \frac{1}{2}\left\|\frac{t_k}{\gamma}d^k\right\|^2 \boldsymbol{\ell}.$$

Therefore, we have

$$\left\langle c_1^*, F(x^k + \frac{t_k}{\gamma}d^k) - F(x^k)\right\rangle \leq \left\langle c_1^*, \frac{t_k}{\gamma}JF(x^k)d^k\right\rangle + \frac{1}{2}\left\|\frac{t_k}{\gamma}d^k\right\|^2 \left\langle c_1^*, \boldsymbol{\ell}\right\rangle.$$

This, together with inequality (14), yields

$$t_k \geq \frac{\gamma}{\left\langle c_1^*, \boldsymbol{\ell}\right\rangle}.$$

Then the desired result follows. □

We consider the following surrogate:

$$G_{k,e/t_{\min}}(x) := JF(x^k)(x - x^k) + \frac{1}{2t_{\min}}\left\|x - x^k\right\|^2 e. \tag{15}$$

The following results show that $G_{k,e/t_{\min}}(\cdot)$ is a non-majorizing surrogate function of $F(\cdot) - F(x^k)$ near x^k.

Proposition 5.2. *Let $G_{k,e/t_{\min}}(\cdot)$ be defined as (15). Then, the following statements hold.*

(i) $F(x^{k+1}) - F(x^k) \preceq_K G_{k,e/t_{\min}}(x^{k+1})$.

(ii) If $F(\cdot)$ is K-convex, then $G_{k,e/t_{\min}}(\cdot) \in \mathcal{S}_{e/t_{\min},e/t_{\min}}(F, x^k)$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, then $G_{k,e/t_{\min}}(\cdot) \in \mathcal{S}_{e/t_{\min}-\boldsymbol{\mu},e/t_{\min}}(F, x^k)$.

Proof. By the line search condition, we have

$$F(x^{k+1})-F(x^k) \preceq_K JF(x^k)(x^{k+1}-x^k)+\frac{1}{2t_k}\left\|x^{k+1}-x^k\right\|^2 e \preceq_K JF(x^k)(x^{k+1}-x^k)+\frac{1}{2t_{\min}}\left\|x^{k+1}-x^k\right\|^2 e.$$

Then, the assertion (i) holds. The assertions (ii) and (iii) can be obtained by using the similar arguments as in the proof of Proposition 4.1. □

Lemma 5.1. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 5. Then, the following statements hold:*

(i) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(ii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - t_{\min}\mu_{\min}}\left\|x^k - x^*\right\|$, $\forall k \geq 0$, where $\mu_{\min} := \min_{c^* \in C_e} \langle c^*, \boldsymbol{\mu}\rangle$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Lemma 4.1. □

Remark 5.1. *If $K = \mathbb{R}^m_+$, and $e = \mathbf{1}_m$, i.e., $C_e = \Delta_m$, the convergence rate in Lemma 5.1(ii) reduces to those established in (Fliege et al., 2018, Theorem 4.2) and (Zeng et al., 2019, Theorem 5.6).*

5.2. Generic first-order method for VOPs with line search

To reduce the gap between $F(\cdot) - F(x^k)$ and $G_{k,e/t_{\min}}(\cdot)$, we select $e_k \in \operatorname{int}(K)$ and devise the following generic first-order method:

Algorithm 6: Generic first order method for VOPs with line search

```
Data: $x^0 \in \mathbb{R}^n, \gamma \in (0,1)$
for $k = 0, 1, ...$ do
    Select $e_k \in \operatorname{int}(K)$
    Update $d^k := \arg\min_{d\in\mathbb{R}^n} \max_{c^*\in C_e} \langle c^*, JF(x^k)d + \frac{1}{2}\|d\|^2 e_k \rangle$
    if $d^k = 0$ then
        return $K$-stationary point $x^k$
    else
        Compute the stepsize $t_k \in (0,1]$ in the following way:
            $t_k := \max\left\{\gamma^j : j \in \mathbb{N}, F(x^k + \gamma^j d^k) - F(x^k) \preceq_K \gamma^j \left(JF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 e_k\right)\right\}$
        $x^{k+1} := x^k + t_k d^k$
    end
end
```

It is worth noting that we don't specify how to select e_k in Algorithm 6. This naturally raises the question: what role does e_k play in determining the convergence rate? Firstly, we derive the lower bound of stepsize in each iteration.

Proposition 5.3. *The stepsize generated in Algorithm 6 satisfies $t_k \geq t_k^{\min} := \min\left\{\min\limits_{c^*\in C_e} \frac{\gamma\langle c^*, e_k\rangle}{\langle c^*, \boldsymbol{\ell}\rangle}, 1\right\}$.*

Proof. The result can be obtained by using the similar arguments as in the proof of Proposition 5.1. □

We consider the following surrogate:

$$G_{k,e_k/t_k}(x) := JF(x^k)(x - x^k) + \frac{1}{2t_k^{\min}}\left\|x - x^k\right\|^2 e_k. \tag{16}$$

We can show that $G_{k,e_k/t_k^{\min}}(\cdot)$ is a non-majorizing surrogate function of $F(\cdot) - F(x^k)$ near x^k.

Proposition 5.4. *Let $G_{k,e_k/t_k^{\min}}(\cdot)$ be defined as (16). Then, the following statements hold.*

(i) $F(x^{k+1}) - F(x^k) \preceq_K G_{k,e_k/t_k^{\min}}(x^{k+1})$..

(ii) If $F(\cdot)$ is K-convex, then $G_{k,e_k/t_k^{\min}}(\cdot) \in \mathcal{S}_{e_k/t_k^{\min},e_k/t_k^{\min}}(F, x^k)$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, then $G_{k,e_k/t_k^{\min}}(\cdot) \in \mathcal{S}_{e_k/t_k^{\min}-\boldsymbol{\mu},e_k/t_k^{\min}}(F, x^k)$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.2. □

The following results show that e_k plays a significant role in the convergence rate of Algorithm 6.

Lemma 5.2. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 6. Then, the following statements hold:*

(i) $\{x^k\}$ converges to an efficient solution x^* of (VOP).

(ii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - \min\limits_{c^* \in C_e} \frac{\langle c^*, \boldsymbol{\mu}\rangle}{\langle c^*, e_k/t_k^{\min}\rangle}} \left\|x^k - x^*\right\|, \ \forall k \geq 0.$

(iii) If $e_k = e$, we have

$$\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - t_{\min}\mu_{\min}} \left\|x^k - x^*\right\|, \ \forall k \geq 0,$$

where $\mu_{\min} := \min_{c^* \in C_e} \langle c^*, \boldsymbol{\mu}\rangle$.

(iv) For any $e_k \in \operatorname{int}K$, we have

$$\min_{c^* \in C_e} \frac{\langle c^*, \boldsymbol{\mu}\rangle}{\langle c^*, e_k/t_k^{\min}\rangle} \leq \frac{\gamma}{\kappa_{F,\preceq_K}}. \tag{17}$$

Moreover, the equality holds with $e_k = \boldsymbol{\mu}$ or $e_k = \boldsymbol{\ell}$. In these cases, we have

$$\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - \gamma/\kappa_{F,\preceq_K}} \left\|x^k - x^*\right\|, \ \forall k \geq 0. \tag{18}$$

Proof. The assertions (i) and (ii) can be obtained by using the similar arguments as in the proof of Lemma 4.1. By substituting $e_k = e$ into (ii), we can obtain the assertion (iii). Next, we prove the assertion (iv). Since C_e is a compact set, there exists a vector $c_0^* \in C_e$ such that $1/\kappa_{F,\preceq_K} = \langle c_0^*, \boldsymbol{\mu}\rangle / \langle c_0^*, \boldsymbol{\ell}\rangle$. On the other hand, by the definition of $t_k^{\min}$ we can deduce

$$\min_{c^* \in C_e} \frac{\langle c^*, \boldsymbol{\mu}\rangle}{\langle c^*, e_k/t_k^{\min}\rangle} \leq \frac{\langle c_0^*, \boldsymbol{\mu}\rangle}{\langle c_0^*, e_k\rangle} \frac{\gamma \langle c_0^*, e_k\rangle}{\langle c_0^*, \boldsymbol{\ell}\rangle} = \frac{\gamma}{\kappa_{F,\preceq_K}}.$$

Then the relation (17) follows. The equality can be obtain by substituting $e_k = \boldsymbol{\mu}$ or $e_k = \boldsymbol{\ell}$ into the left-hand side of (17) . Moreover, The equality leads to the relation (18). □

Remark 5.2. *As described in Lemma 5.2(iv), by setting $e_k = \boldsymbol{\mu}$ or $e_k = \boldsymbol{\ell}$, we can derive the optimal linear convergence rate for Algorithm 6, and the convergence rate reduces to that of Lemma 4.3(ii) with constant γ. Intuitively, to explore the local curvature information of $F(\cdot)$, we can devise a tighter local surrogate $G_{k,e_k/t_k}(\cdot)$ with $\boldsymbol{\mu} \preceq_K e_k \preceq_K \boldsymbol{\ell}$. In this case, the performance of Algorithm 6 can be further improved by using a tighter local surrogate $G_{k,e_k/t_k}(\cdot)$.*

To narrow the surrogate gap and better capture the local curvature information, we compute e_k by Barzilai-Borwein method, namely, we set

$$e_k := \frac{\left\langle JF(x^k) - JF(x^{k-1}), x^k - x^{k-1}\right\rangle}{\left\|x^k - x^{k-1}\right\|^2}. \tag{19}$$

Lemma 5.3. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, let $\{x^k\}$ be the sequence generated by Algorithm 6, where e_k is defined as in (19). Then, the following statements hold:*

(i) $\boldsymbol{\mu} \preceq_K e_k \preceq_K \boldsymbol{\ell}$;

(ii) $t_k \geq \min_{c^* \in C_e} \{\gamma \langle c^*, e_k \rangle / \langle c^*, \boldsymbol{\ell} \rangle\}$;

Proof. Assertion (i) follows by the strong k-convexity and K-smoothness of $F(\cdot)$, and the definition of e_k. We can obtain the assertion (ii) by using the similar arguments as in the proof of Proposition 5.1. □

Remark 5.3. *To reduce per-iteration cost, we set* $e = e_k$, *i.e.,* $C_e = C_{e_k}$, *so that the coupled subproblem in Algorithm 6 can be reformulated into the following seperate form:*

$$\min_{d \in \mathbb{R}^n} \max_{c^* \in C_{e_k}} \left\langle c^*, JF(x^k)d \right\rangle + \frac{1}{2}\|d\|^2.$$

Hence, we conclude that using a variable C_{e_k} *serves as an appropriate choice of base in SDVO, which provides a theoretical answer to (Q). For* $K = \mathbb{R}^2_+$, *Fig. 1 illustrates the choice of the base under the assumption of strong convexity. It suggests that bases should be adaptively selected from the pink region according to (19).*

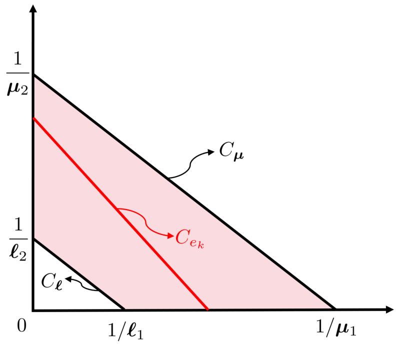


Fig. 1. Illustration of the C_{e_k} of $K = \mathbb{R}^2_+$.

6. First-order methods for VOPs with polyhedral cones

In this section, we consider the VOPs that K is a polyhedral cone with nonempty interior. Without loss of generality, for the polyhedral cone K, there exists a transform matrix $A \in \mathbb{R}^{l \times m}$ with $m \leq l$ such that

$$K := \{x \in \mathbb{R}^m : 0 \preceq Ax\}.$$

In this case, for any $a, b \in \mathbb{R}^m$, $a \preceq_K b$ can be equivalently represented as $Aa \preceq Ab$. Denote A_i the i-th row vector of A. For the polyhedral cone K, we denote the set of transform matrices as follows:

$$\mathcal{A} := \left\{A \in \mathbb{R}^{l \times m} : AK = \mathbb{R}^l_+\right\}.$$

Notably, in practice, the polyhedral cone K is often defined by a specific transform matrix A. This raises a crucial question: does the transform matrix A affect the performance of descent methods for VOPs with polyhedral cones?

6.1. Steepest descent method for VOPs with polyhedral cones

By using a transform matrix $A \in \mathcal{A}$, the steepest descent direction subproblem for VOPs with polyhedral cones is formulated as follows:

$$\min_{d \in \mathbb{R}^n} \max_{\lambda \in \Delta_l} \left\langle \lambda, AJF(x^k)d \right\rangle + \frac{1}{2}\|d\|^2. \tag{20}$$

The complete K-steepest descent method for VOPs with polyhedral cones is described as follows:

Algorithm 7: K-steepest descent method for VOPs with polyhedral cones

```
Data: $x^0 \in \mathbb{R}^n, \gamma \in (0,1)$
Select a transform matrix $A \in \mathcal{A}$
for $k = 0, 1, ...$ do
    Update $d^k$ as the minimizer of (20)
    if $d^k = 0$ then
        return $K$-stationary point $x^k$
    else
        Compute the stepsize $t_k \in (0,1]$ in the following way:
            $$t_k := \max\left\{\gamma^j : j \in \mathbb{N}, A(F(x^k + \gamma^j d^k) - F(x^k)) \preceq \gamma^j \left(AJF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 \mathbf{1}_l\right)\right\}$$
        $x^{k+1} := x^k + t_k d^k$
    end
end
```

We consider the following surrogate:

$$G_{k,A,\mathbf{1}_l/t_k}(x) := AJF(x^k)(x - x^k) + \frac{1}{2t_k}\left\|x - x^k\right\|^2 \mathbf{1}_l. \tag{21}$$

We can show that $G_{k,A,\mathbf{1}_l/t_k}(\cdot)$ is a non-majorizing surrogate function of $A(F(\cdot) - F(x^k))$ near x^k.

Proposition 6.1. *Let $G_{k,A,\mathbf{1}_l/t_k}(\cdot)$ be defined as (21). Then, the following statements hold.*

(i) $A(F(x^{k+1}) - F(x^k)) \preceq G_{k,A,\mathbf{1}_l/t_k}(x^{k+1})$.

(ii) If $F(\cdot)$ is K-convex, then $G_{k,A,\mathbf{1}_l/t_k}(\cdot) \in \mathcal{S}_{\mathbf{1}_l/t_k,\mathbf{1}_l/t_k}(AF, x^k)$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, then $G_{k,A,\mathbf{1}_l/t_k}(\cdot) \in \mathcal{S}_{\mathbf{1}_l/t_k - A\boldsymbol{\mu},\mathbf{1}_l/t_k}(AF, x^k)$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.2. □

Lemma 6.1. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, where $K = \{x \in \mathbb{R}^m : 0 \preceq Ax\}$. Let $\{x^k\}$ be the sequence generated by Algorithm 7. Then, the following statements hold:*

(i) $t_k \geq \min\{\min_{i\in[l]}\{\gamma/\langle A_i, \boldsymbol{\ell}\rangle\}, 1\}$;

(ii) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(iii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - t_k \min_{i\in[l]}\langle A_i, \boldsymbol{\mu}\rangle}\left\|x^k - x^*\right\|, \ \forall k \geq 0.$

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.1 and Lemma 4.1. □

Remark 6.1. *The subproblem (20) can be reformulated as*

$$\min_{d\in\mathbb{R}^n}\max_{c^*\in C}\left\langle c^*, JF(x^k)d\right\rangle + \frac{1}{2}\left\|d\right\|^2,$$

where $C := conv\{A_i, i \in [l]\}$ is a base of K^*. In other words, selecting a transform matrix in (20) is equivalent to selecting a base of K^*. Consequently, the linear convergence rate of Algorithm 7 is sensitive to the choice of A. When $K = \mathbb{R}^m_+$, the steepest descent method (Fliege & Svaiter, 2000) fixs $A = I_m$, i.e., $C = \Delta_m$.

6.2. Barzilai-Borwein descent method for VOPs with polyhedral cones

In general, for the e_k defined as (19) we have $e_k \preceq_K \boldsymbol{\ell}$, which can be written as $Ae_k \preceq A\boldsymbol{\ell}$. We denote $\alpha^k \in \mathbb{R}^l_{++}$ as follows:

$$\alpha_i^k = \begin{cases} \max\left\{\alpha_{\min}, \min\left\{\dfrac{\langle s_{k-1}, y_i^{k-1}\rangle}{\|s_{k-1}\|^2}, \alpha_{\max}\right\}\right\}, & \langle s_{k-1}, y_i^{k-1}\rangle > 0, \\ \max\left\{\alpha_{\min}, \min\left\{\dfrac{\|y_i^{k-1}\|}{\|s_{k-1}\|}, \alpha_{\max}\right\}\right\}, & \langle s_{k-1}, y_i^{k-1}\rangle < 0, \\ \alpha_{\min}, & \langle s_{k-1}, y_i^{k-1}\rangle = 0, \end{cases} \tag{22}$$

for all $i \in [l]$, where $s_{k-1} = x^k - x^{k-1}$, y_i^{k-1} is the i-th row vector of $A(JF(x^k) - JF(x^{k-1}))$, $\alpha_{\max}$ is a sufficient large positive constant and $\alpha_{\min}$ is a sufficient small positive constant. The Barzilai-Borwein descent direction is defined as the minimizer of

$$\min_{d\in\mathbb{R}^n}\max_{\lambda\in\Delta_l}\left\langle \lambda, AJF(x^k)d + \frac{1}{2}\|d\|^2\alpha^k\right\rangle. \tag{23}$$

Alternatively, we use the similar strategy in Algorithm 4, the Barzilai-Borwein descent direction subproblem can be rewritten equivalently as the following coupled form:

$$\min_{d\in\mathbb{R}^n}\max_{\lambda\in\Delta_l^{\alpha^k}}\left\langle \lambda, AJF(x^k)d\right\rangle + \frac{1}{2}\|d\|^2, \tag{24}$$

where $\Delta_l^{\alpha^k} := \{c^* \in \mathbb{R}^l_+ : \langle c^*, \alpha^k\rangle = 1\}$. The subproblem can be reformulated as follows:

$$\min_{d\in\mathbb{R}^n}\max_{\lambda\in\Delta_l}\left\langle \lambda, \Lambda^k AJF(x^k)d\right\rangle + \frac{1}{2}\|d\|^2, \tag{25}$$

where

$$\Lambda^k := \begin{bmatrix} \frac{1}{\alpha_1^k} & & \\ & \ddots & \\ & & \frac{1}{\alpha_l^k} \end{bmatrix}.$$

By Sion's minimax theorem (Sion, 1958), the minimizer of (25) can be written as

$$d^k = -(\Lambda^k AJF(x^k))^T\lambda^k,$$

where $\lambda^k \in \Delta_l$ is a solution of the following dual problem:

$$\min_{\lambda\in\Delta_l}\frac{1}{2}\left\|(\Lambda_k AJF(x^k))^T\lambda\right\|^2. \tag{DP}$$

Remark 6.2. *In general, the dual problem (DP) is a lower dimensional quadratic programming with unit simplex constraint (the vertices of unit simplex constraint are known), then it can be solved by Frank-Wolfe/conditional gradient method efficiently (see, e.g., Sener & Koltun, 2018; Chen et al., 2023a). However, dual problem of (23) reads as*

$$\min_{\lambda\in\Delta_l}\frac{1}{2}\frac{\left\|(AJF(x^k))^T\lambda\right\|^2}{\sum\limits_{i\in[l]}\lambda_i\alpha_i^k},$$

which is not easy to solve.

Remark 6.3. *If $K = \mathbb{R}^m_+$ and $A = I_m$, the subproblem (25) reduces to that of the BBDMO (Chen et al., 2023a). Consequently, transforming (23) into (25) gives a new insight into BBDMO from a majorization-minimization perspective. Comparing the subproblems (20) and (24), it turns out that the differences between the directions of the steepest descent and the Barzilai-Borwein descent lie in the choice of base for K^*.*

The complete K-Barzilai-Borwein descent method for VOPs with polyhedral cones is described as follows:

Algorithm 8: K-Barzilai-Borwein descent method for VOPs with polyhedral cones

```
Data: $x^0 \in \mathbb{R}^n, \gamma \in (0,1)$
Select a transform matrix $A \in \mathcal{A}$
Choose $x^{-1}$ in a small neighborhood of $x^0$
for $k = 0, 1, ...$ do
    Update $\alpha^k$ as (22)
    Update $d^k$ as the minimizer of (25)
    if $d^k = 0$ then
        return $K$-stationary point $x^k$
    else
        Compute the stepsize $t_k \in (0,1]$ in the following way:
            $t_k := \max\left\{\gamma^j : j \in \mathbb{N}, A(F(x^k + \gamma^j d^k) - F(x^k)) \preceq \gamma^j \left(AJF(x^k)d^k + \frac{1}{2}\left\|d^k\right\|^2 \alpha^k\right)\right\}$
        $x^{k+1} := x^k + t_k d^k$
    end
end
```

We consider the following surrogate:

$$G_{k,A,\alpha^k/t_k}(x) := AJF(x^k)(x - x^k) + \frac{1}{2t_k}\left\|x - x^k\right\|^2 \alpha^k. \tag{26}$$

We can show that $G_{k,A,\alpha^k/t_k}(\cdot)$ is a non-majorizing surrogate function of $A(F(\cdot) - F(x^k))$ near x^k.

Proposition 6.2. *Let $G_{k,A,\alpha^k/t_k}(\cdot)$ be defined as (26). Then, the following statements hold.*

(i) $A(F(x^{k+1}) - F(x^k)) \preceq G_{k,A,\alpha^k/t_k}(x^{k+1})$.

(ii) If $F(\cdot)$ is K-convex, then $G_{k,A,\alpha^k/t_k}(\cdot) \in \mathcal{S}_{\alpha^k/t_k,\alpha^k/t_k}(AF, x^k)$.

(iii) If $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, then $G_{k,A,\alpha^k/t_k}(\cdot) \in \mathcal{S}_{\alpha^k/t_k - A\boldsymbol{\mu},\alpha^k/t_k}(AF, x^k)$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.2. □

Lemma 6.2. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \mathrm{int}(K)$, where $K = \{x \in \mathbb{R}^m : 0 \preceq Ax\}$. Let $\{x^k\}$ be the sequence generated by Algorithm 8. Then, the following statements hold:*

(i) $A\boldsymbol{\mu} \preceq \alpha^k \preceq A\boldsymbol{\ell}$;

(ii) $t_k \geq \min_{i\in[l]}\{\gamma\alpha_i^k / \langle A_i, \boldsymbol{\ell}\rangle\}$;

(iii) $\{x^k\}$ converges to an efficient solution x^* of (VOP);

(iv) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - t_k \min_{i\in[l]}\{\langle A_i, \boldsymbol{\mu}\rangle/\alpha_i^k\}}\left\|x^k - x^*\right\|, \ \forall k \geq 0.$

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.1 and Lemma 4.1. □

A large stepsize may speed up the convergence of Algorithm 8. Accordingly, the Armijo line search can be applied, namely, compute the stepsize $t_k \in (0,1]$ in the following way:

$$t_k := \max\left\{\gamma^j : j \in \mathbb{N}, A(F(x^k + \gamma^j d^k) - F(x^k)) \preceq \sigma\gamma^j AJF(x^k)d^k\right\}, \tag{27}$$

where $\sigma \in (0,1)$.

The following result shows that the convergence rate of Algorithm 8 is not sensitive to the choice of transform matrix. More specifically, the descent direction d^k and stepsize t_k of Algorithm 8 are invariant for some $A \in \mathcal{A}$.

Proposition 6.3. *(Affine Invariance) Let $A^1, A^2 \in \mathcal{A}$, d_1^k, t_k^1 and d_2^k, t_k^2 be the descent directions and stepsize generated by Algorithm 8 with A^1 and A^2, respectively. If $\alpha_{\min} < \alpha_i^{k,1}, \alpha_i^{k,2} < \alpha_{\max}$, $i \in [l]$, we have $d_1^k = d_2^k$ and $t_k^1 = t_k^2$.*

Proof. Denote A_i^1 and A_i^2 the the i-th row vector of A^1 and A^2, respectively. Before presenting the main results, we rewritten the subproblem (25) as follows:

$$\min_{d \in \mathbb{R}^n} \max_{i \in [l]} \left\langle \frac{A_i}{\alpha_i^k}, JF(x^k)d \right\rangle + \frac{1}{2} \|d\|^2 .$$

Recall that $A^1, A^2 \in \mathcal{A}$, there exists a vector $a \in \mathbb{R}^l_{++}$ such that

$$\left\{A_i^1 : i \in [l]\right\} = \left\{a_i A_i^2 : i \in [l]\right\} . \tag{28}$$

We claim the following assertion:

$$\left\{\frac{A_i^1}{\alpha_i^{k,1}} : i \in [l]\right\} = \left\{\frac{A_i^2}{\alpha_i^{k,2}} : i \in [l]\right\} . \tag{29}$$

This, together with the reformulated subproblem, implies that $d_1^k = d_2^k$. Therefore, $t_k^1 = t_k^2$ is a consequence of (29).

Next, we prove that assertion (29) holds. For any $i \in [l]$, it follows by (28) that there exist $j \in [l]$ such that $A_i^1 = a_j A_j^2$. Notice that $\alpha_{\min} < \alpha_i^{k,1}, \alpha_i^{k,2} < \alpha_{\max}$, $i \in [l]$, we distinguish two cases:

$$\alpha_i^{k,1} = \left\|A_i^1(JF(x^k) - JF(x^{k-1}))\right\| / \|s_{k-1}\| \text{ and } \alpha_j^{k,2} = \left\|A_j^2(JF(x^k) - JF(x^{k-1}))\right\| / \|s_{k-1}\|,$$

and

$$\alpha_i^{k,1} = \left\langle A_i^1(JF(x^k) - JF(x^{k-1})), s_{k-1}\right\rangle / \|s_{k-1}\|^2 \text{ and } \alpha_i^{k,2} = \left\langle A_i^1(JF(x^k) - JF(x^{k-1})), s_{k-1}\right\rangle / \|s_{k-1}\|^2.$$

Since $A_i^1 = a_j A_j^2$ $(a_j > 0)$, it is easy to verify that

$$\frac{A_i^1}{\alpha_i^{k,1}} = \frac{A_j^2}{\alpha_j^{k,2}}$$

holds in both cases. Thus, we have

$$\left\{\frac{A_i^1}{\alpha_i^{k,1}} : i \in [l]\right\} \subseteq \left\{\frac{A_i^2}{\alpha_i^{k,2}} : i \in [l]\right\} .$$

The relation

$$\left\{\frac{A_i^2}{\alpha_i^{k,2}} : i \in [l]\right\} \subseteq \left\{\frac{A_i^1}{\alpha_i^{k,1}} : i \in [l]\right\}$$

follows the similar arguments, this concludes the proof. □

Remark 6.4. *Assume $\hat{\mathcal{A}} \subset \mathcal{A}$ is bounded and A_i is bounded away from 0 for all $A \in \hat{\mathcal{A}}$. Then, there exists $\alpha_{\min}$ and $\alpha_{\max}$ such that the assumption $\alpha_{\min} < \alpha_i^k < \alpha_{\max}$, $i \in [l]$ holds for all $A \in \hat{\mathcal{A}}$ with $\left\langle s_{k-1}, A_i(JF(x^k) - JF(x^{k-1}))\right\rangle \neq 0$. For the case with linear objective, $\left\langle s_{k-1}, A_i(JF(x^k) - JF(x^{k-1}))\right\rangle = 0$ may hold. As illustrated in (Chen et al., 2023a, Example 3), for any $A_1, A_2 \in \hat{\mathcal{A}}$, we have $d_1^k \approx d_2^k$ with sufficient small $\alpha_{\min}$. As a result, we conclude that the performance of Algorithm 8 is not sensitive to the choice of A.*

6.3. Backtracking method for VOPs with polyhedral cones

As described in Lemma 6.2, we apply Barzilai-Borwein method to ensure $A\boldsymbol{\mu} \preceq \alpha^k \preceq A\boldsymbol{\ell}$, which in turn is expected to obtain $A\boldsymbol{\mu} \preceq \alpha^k/t_k \preceq A\boldsymbol{\ell}$, thereby achieving a fast linear convergence rate in practice. However, this strategy actually yields an even slower convergence rate.

Proposition 6.4. *Assume that $F(\cdot)$ is strongly K-convex with $\boldsymbol{\mu} \in \operatorname{int}(K)$, where $K = \{x \in \mathbb{R}^m : 0 \preceq Ax\}$. Let $\{x^k\}$ be the sequence generated by Algorithm 8 and x^* be the efficient solution satisfies $F(x^*) \preceq_K F(x^k)$ for all $k \geq 0$. Then*

$$\|x^{k+1} - x^*\| \leq \sqrt{1 - \gamma \min_{i \neq j, i,j \in [l]} \left\{ \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \frac{\langle A_j, \boldsymbol{\mu}\rangle}{\langle A_j, \boldsymbol{\ell}\rangle} \right\}} \|x^k - x^*\|$$

holds for all $k \geq 0$.

Proof. By Lemma 6.2(i), we have $A\boldsymbol{\mu} \preceq \alpha^k \preceq A\boldsymbol{\ell}$. For any $i \neq j$, consider the choice $\alpha_i^k = \langle A_i, \boldsymbol{\mu}\rangle$ and $\alpha_j^k = \langle A_j, \boldsymbol{\ell}\rangle$, then we have

$$\min_{i \in [l]} \frac{\alpha_i^k}{\langle A_i, \boldsymbol{\ell}\rangle} \min_{i \in [l]} \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\alpha_i^k} \leq \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \frac{\langle A_j, \boldsymbol{\mu}\rangle}{\langle A_j, \boldsymbol{\ell}\rangle}.$$

The arbitrary of i and j with $i \neq j$ yields

$$\min_{i \in [l]} \frac{\alpha_i^k}{\langle A_i, \boldsymbol{\ell}\rangle} \min_{i \in [l]} \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\alpha_i^k} \leq \min_{i \neq j, i,j \in [l]} \left\{ \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \frac{\langle A_j, \boldsymbol{\mu}\rangle}{\langle A_j, \boldsymbol{\ell}\rangle} \right\}.$$

Conversely, notice that

$$\min_{i \in [l]} \tfrac{\alpha_i^k}{\langle A_i, \boldsymbol{\ell}\rangle} \geq \min_{i \in [l]} \tfrac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \quad \text{and} \quad \min_{i \in [l]} \tfrac{\langle A_i, \boldsymbol{\mu}\rangle}{\alpha_i^k} \geq \min_{i \in [l]} \tfrac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle},$$

and no single α^k can simultaneously satisfy both equilities unless there exists two distinct indices $i_0 \neq j_0$ such that

$$\min_{i \in [l]} \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} = \frac{\langle A_{i_0}, \boldsymbol{\mu}\rangle}{\langle A_{i_0}, \boldsymbol{\ell}\rangle} = \frac{\langle A_{j_0}, \boldsymbol{\mu}\rangle}{\langle A_{j_0}, \boldsymbol{\ell}\rangle},$$

which yields

$$\min_{i \in [l]} \frac{\alpha_i^k}{\langle A_i, \boldsymbol{\ell}\rangle} \min_{i \in [l]} \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\alpha_i^k} \geq \min_{i \neq j, i,j \in [l]} \left\{ \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \frac{\langle A_j, \boldsymbol{\mu}\rangle}{\langle A_j, \boldsymbol{\ell}\rangle} \right\}.$$

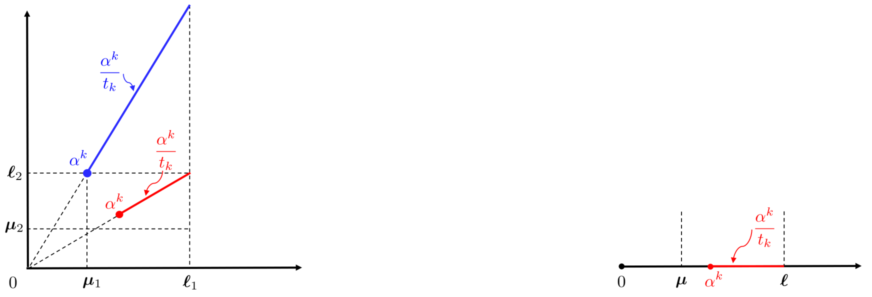
Hence,

$$\min_{i \in [l]} \frac{\alpha_i^k}{\langle A_i, \boldsymbol{\ell}\rangle} \min_{i \in [l]} \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\alpha_i^k} = \min_{i \neq j, i,j \in [l]} \left\{ \frac{\langle A_i, \boldsymbol{\mu}\rangle}{\langle A_i, \boldsymbol{\ell}\rangle} \frac{\langle A_j, \boldsymbol{\mu}\rangle}{\langle A_j, \boldsymbol{\ell}\rangle} \right\}.$$

The desired result follows by Lemma 6.2(ii) and (iv). □

In contrast to the line search method for VOPs, the corresponding method for SOPs preserves the same linear convergence rate as its counterpart without line search. This discrepancy arises because the line search in VOPs is imposed with respect to a **strict dominance relation** by the underlying partial order. We clarify the distinction with the following example.

Example 6.1. *Let $K = \mathbb{R}^2_+$ and $A = I_2$. Consider first the case where $\alpha^k = (\boldsymbol{\mu}_1, \boldsymbol{\ell}_2)^T$. As illustrated in Fig. 2(a), the ratio α^k/t_k lies on the blue line determined by the strict dominance relation. In this situation, the line search results in a slower linear convergence rate, with the worst-case rate being $\mathcal{O}\left(\left(\sqrt{1 - \gamma \frac{\boldsymbol{\mu}_1 \boldsymbol{\mu}_2}{\boldsymbol{\ell}_1 \boldsymbol{\ell}_2}}\right)^k\right)$.*



(a) Line search for VOPs

(b) Line search for SOPs

Fig. 2. Differences in line search for VOPs and SOPs.

In contrast, if α^k falls on the red point in Fig. 2(a), then α^k/t_k lies on the red line. In this case, line search for VOPs potentially improves the practical linear convergence rate, with the worst-case rate coinciding with that of the method without line search. As illustrated in Fig. 2(b), line search for SOPs tends to improve linear convergence rate, and the worst case rate is the same as that of its counterpart without line search.

The remaining question is how to preserve the linear convergence rate of descent methods for VOPs without line search when the smoothness parameter $\boldsymbol{\ell}$ is unknown. To address this, we revisit the backtracking method for VOPs, which was first proposed in (Chen et al., 2023b).

Algorithm 9: Backtracking method for VOPs with polyhedral cones

```
Data: $x^0 \in \mathbb{R}^n,\ A \in \mathcal{A},\ 0 \prec \boldsymbol{\ell}^0 \preceq A\boldsymbol{\ell},\ \tau > 1$
for $k = 0, 1, ...$ do
    Update $\alpha^k := \boldsymbol{\ell}^k$
    Update $x^{k+1} := \arg\min_{x \in \mathbb{R}^n} \max_{\lambda \in \Delta_l^{\alpha^k}} \left\{ \langle \lambda, AJF(x^k)(x - x^k) \rangle + \frac{1}{2} \left\| x - x^k \right\|^2 \right\}$
    if $x^{k+1} = x^k$ then
        return $K$-stationary point $x^k$
    else
        $s_i = 0, i \in [l]$
        repeat
            Update $\alpha_i^k = \tau^{s_i} \boldsymbol{\ell}_i^k, i \in [l]$
            Update $x^{k+1} := \arg\min_{x \in \mathbb{R}^n} \max_{\lambda \in \Delta_l^{\alpha^k}} \left\{ \langle \lambda, AJF(x^k)(x - x^k) \rangle + \frac{1}{2} \left\| x - x^k \right\|^2 \right\}$
            for $i = 1, \cdots, l$ do
                if $\langle A_i, F(x^{k+1}) - F(x^k) \rangle > \langle A_i, JF(x^k)(x^{k+1} - x^k) \rangle + \frac{\alpha_i^k}{2} \|x^{k+1} - x^k\|^2$ then
                    Update $s_i = s_i + 1$
                end
            end
        until $\langle A_i, F(x^{k+1}) - F(x^k) \rangle \leq \langle A_i, JF(x^k)(x^{k+1} - x^k) \rangle + \frac{\alpha_i^k}{2} \|x^{k+1} - x^k\|^2,\ i \in [l]$;
    end
    Update $\boldsymbol{\ell}^{k+1} := \alpha^k / \tau$
end
```

We consider the following surrogate:

$$G_{k,A,\alpha^k}(x) := AJF(x^k)(x - x^k) + \frac{1}{2} \left\| x - x^k \right\|^2 \alpha^k. \tag{30}$$

We can show that $G_{k,A,\alpha^k}(\cdot)$ is a non-majorizing surrogate function of $A(F(\cdot) - F(x^k))$ near x^k.

Proposition 6.5. *Let $G_{k,A,\alpha^k}(\cdot)$ be defined as (30). Then, the following statements hold.*

(i) $A(F(x^{k+1}) - F(x^k)) \preceq G_{k,A,\alpha^k}(x^{k+1})$.

(ii) If $F(\cdot)$ *is* K*-convex, then* $G_{k,A,\alpha^k}(\cdot) \in \mathcal{S}_{\alpha^k,\alpha^k}(AF, x^k)$.

(iii) If $F(\cdot)$ *is strongly* K*-convex with* $\boldsymbol{\mu} \in \operatorname{int}(K)$*, then* $G_{k,A,\alpha^k}(\cdot) \in \mathcal{S}_{\alpha^k - A\boldsymbol{\mu},\alpha^k}(AF, x^k)$.

Proof. The assertions can be obtained by using the similar arguments as in the proof of Proposition 5.2. □

Lemma 6.3. *Assume that* $F(\cdot)$ *is strongly* K*-convex with* $\boldsymbol{\mu} \in \operatorname{int}(K)$*, where* $K = \{x \in \mathbb{R}^m : 0 \preceq Ax\}$*. Let* $\{x^k\}$ *be the sequence generated by Algorithm 8. Then, the following statements hold:*

(i) $\alpha^k \prec \tau A\boldsymbol{\ell}$*;*

(ii) $\{x^k\}$ *converges to an efficient solution* x^* *of (VOP);*

(iii) $\left\|x^{k+1} - x^*\right\| \leq \sqrt{1 - \tau \min_{i\in[l]} \langle A_i, \boldsymbol{\mu}\rangle / \langle A_i, \boldsymbol{\ell}\rangle} \left\|x^k - x^*\right\|, \ \forall k \geq 0.$

Proof. (i) Suppose, to the contrary, that $\alpha_i^k \geq \tau \langle A_i, \boldsymbol{\ell}\rangle$ holds for some $i \in [l]$. Then the backtracking procedure would be triggered only when $\alpha_i^k \geq \langle A_i, \boldsymbol{\ell}\rangle$, which contradicts the backtracking condition. Assertions (ii) and (iii) can be obtained by using the similar arguments as in the proof of Lemma 4.1. □

In contrast to the line search method for VOPs, the backtracking method preserves the same linear convergence rate as its counterpart without line search. This discrepancy arises because backtracking for VOPs is imposed with respect to a **weak dominance relation** induced by the underlying partial order. We clarify the distinction with the following example.

Example 6.2. *Let* $K = \mathbb{R}^2_+$ *and* $A = I_2$*. Consider the case where* $\boldsymbol{\ell}^k = (\boldsymbol{\mu}_1, \boldsymbol{\ell}_2)^T$*.*

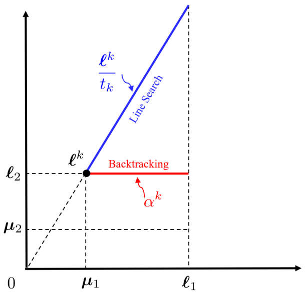


Fig. 3. Differences between line search and backtracking for VOPs.

As illustrated in Fig. 3, the ratio $\boldsymbol{\ell}^k/t_k$ *lies on the blue line determined by the strict dominance relation. In this situation, the line search yields a slower linear convergence rate, with the worst-case rate given by* $\mathcal{O}\left(\left(\sqrt{1 - \gamma\frac{\boldsymbol{\mu}_1\boldsymbol{\mu}_2}{\boldsymbol{\ell}_1\boldsymbol{\ell}_2}}\right)^k\right)$*. In contrast, the* α^k *lies on the red line determined by the weak dominance relation. In this case, the backtracking procedure for VOPs preserves the same linear convergence rate as its counterpart without line search.*

Remark 6.5. *A notable advantage of backtracking is its ability to adapt to the local smoothness and to preserve the same linear convergence rate as its counterpart without line search. However, this benefit comes at the price of increased per-iteration computational cost, since backtracking requires repeatedly solving the associated subproblems.*

7. Numerical Results

In this section, we present numerical results to demonstrate the performance of Barzilai-Borwein descent methods for VOPs (BBDVO) with polyhedral cones. We also compare BBDVO with steepest descent method for VOPs (SDVO) and equiangular direction method (Katrutsa et al., 2020) for VOPs (EDVO). 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.

7.1. Implementation details

For all tested algorithms, we used Armijo line search (27) with $\sigma = 10^{-4}$ and $\gamma = 0.5$. The test algorithms were executed on several test problems, and the problem illustration is given in Table 1. The dimensions of variables and objective functions are presented in the second and third columns, respectively. x_L and x_U represent lower bounds and upper bounds of variables, respectively.

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

Problem	n	m	x_L	x_U	Reference
BK1	2	2	(-5,-5)	(10,10)	(Huband et al., 2006)
DD1	5	2	(-20,...,-20)	(20,...,20)	(Das & Dennis, 1998)
Deb	2	2	(0.1,0.1)	(1,1)	(Deb, 1999)
FF1	2	2	(-1,-1)	(1,1)	(Huband et al., 2006)
Hil1	2	2	(0,0)	(1,1)	(Hillermeier, 2001)
Imbalance1	2	2	(-2,-2)	(2,2)	(Chen et al., 2023a)
JOS1a	50	2	(-2,...,-2)	(2,...,2)	(Jin et al., 2001)
LE1	2	2	(-5,-5)	(10,10)	(Huband et al., 2006)
PNR	2	2	(-2,-2)	(2,2)	(Preuss et al., 2006)
WIT1	2	2	(-2,-2)	(2,2)	(Witting, 2012)

For the tested problems, the partial order are induced by polyhedral cones $\mathbb{R}^2_+$, K_1, and K_2, respectively, where

$$K_1 := \{x \in \mathbb{R}^2 : 5x_1 - x_2 \geq 0,\ \ -x_1 + 5x_2 \geq 0\} \subseteq \mathbb{R}^2_+,$$

and

$$K_2 := \{x \in \mathbb{R}^2 : 5x_1 + x_2 \geq 0,\ x_1 + 5x_2 \geq 0\} \supseteq \mathbb{R}^2_+.$$

The polyhedral cones are illustrated in Fig. 4.

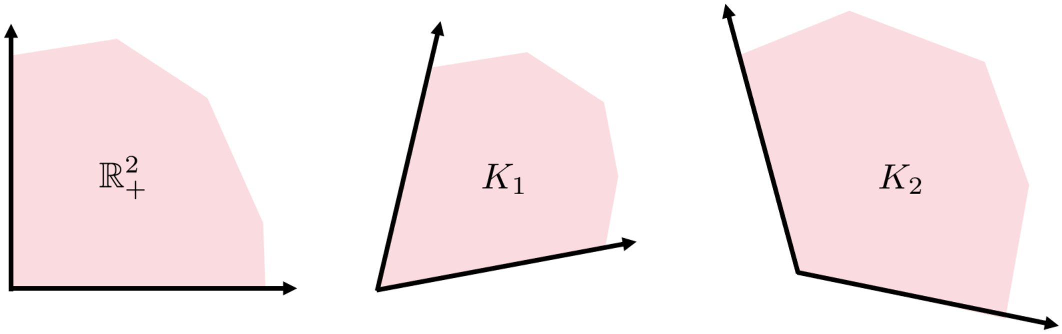


Fig. 4. Illustration of the polyhedral cones.

For each problem, we used the same initial points for different tested algorithms. The initial points were randomly selected within the specified lower and upper bounds. Dual subproblems of different algorithms were efficiently solved by Frank-Wolfe method. To guarantee a fair comparison, we decided to let the algorithms run until one of the following stopping conditions was satisfied:

- the current solution satisfies $\left\|\bar{d}^k\right\| \leq 10^{-6}$, where $\bar{d}$ is the steepest descent direction (20) with $A = \bar{A}$ and $\bar{A} \in \mathcal{A}$ with row vectors $\left\|\bar{A}_i\right\| = 1$ for $i = 1, ..., l$;
- the number of iterations reaches 500.

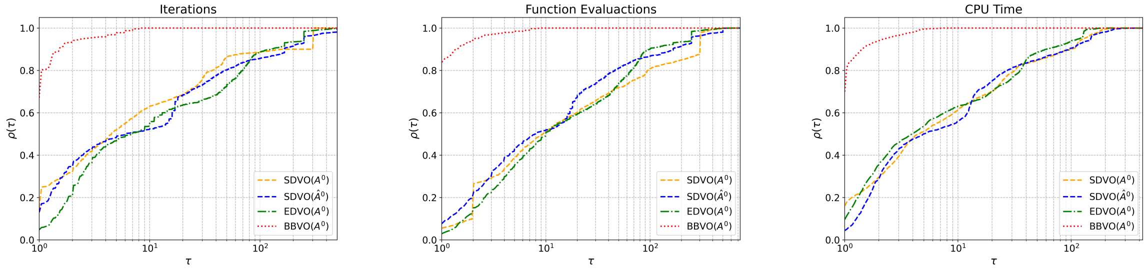
The recorded averages from the 200 runs include the number of iterations, the number of function evaluations, and the CPU time. The performance profiles (Dolan & Moré, 2002) in terms of iterations, function evaluations and CPU time are used to illustrate the overall performance of the 200 runs.

7.2. Numerical results for VOPs with $K = \mathbb{R}^2_+$

In this case, we denote the set of transform matrices $\mathcal{A}_0 := \{A : A\mathbb{R}^2_+ = \mathbb{R}^2_+\}$. For SDVO, we choose $A^0 = I_2 \in \mathcal{A}_0$ and $\hat{A}^0 \in \mathcal{A}_0$ in subproblem, respectively, where $\hat{A}^0 := \{A : A_i = A^0_i / \max\{1, \left\|\nabla F_i(x^0)\right\|_\infty\},\ i = 1, 2\}$[1]. For EDVO, normalization is applied for each of gradients in the transformed subproblem, which implies that EDVO is also not sensitive to the choice of transform matrix. As a result, we choose $A = A^0$ in subproblems of EDVO and BBDVO.

Table 2: Number of average iterations (iter), number of average function evaluations (feval), and average CPU time (time(ms)) of SDVO, EDVO and BBDVO implemented on different test problems with $K = \mathbb{R}^2_+$

Problem	SDVO with $A = A^0$			SDVO with $A = \hat{A}^0$			EDVO with $A = A^0$			BBDVO with $A = A^0$		
	iter	feval	time	iter	feval	time	iter	feval	time	iter	feval	time
BK1	**1.00**	2.00	0.24	36.93	40.15	4.03	30.12	34.07	2.67	**1.00**	**1.00**	**0.21**
DD1	70.95	199.40	12.14	248.33	250.81	30.41	376.96	376.96	38.11	**7.33**	**8.51**	**1.24**
Deb	57.79	376.02	13.67	5.67	13.15	1.36	10.48	26.39	1.43	**4.51**	**6.67**	**0.79**
FF1	28.95	29.11	3.93	8.22	9.15	1.38	8.68	9.51	1.03	**4.68**	**5.90**	**0.84**
Hil1	24.89	82.53	5.15	15.62	36.52	3.05	20.09	51.60	3.06	**11.42**	**12.25**	**2.07**
Imbalance1	88.23	178.31	12.35	387.91	388.32	43.47	420.04	420.99	42.01	**2.62**	**3.60**	**0.50**
JOS1a	302.72	302.72	35.80	16.51	18.03	3.40	74.13	74.44	9.29	**1.00**	**1.00**	**0.26**
LE1	22.44	52.57	3.72	7.98	29.78	1.70	11.80	29.93	1.47	**4.65**	**7.13**	**0.82**
PNR	11.24	48.40	2.83	12.32	16.09	1.85	14.37	23.24	1.75	**4.28**	**4.77**	**0.71**
WIT1	73.30	461.37	19.93	139.88	146.58	15.15	10.64	17.97	1.33	**3.59**	**3.68**	**0.62**



(a) Iterations (b) Function Evaluations (c) CPU Time

Fig. 5. Performance profiles on the test problems in Table 2 with $K = \mathbb{R}^2_+$.

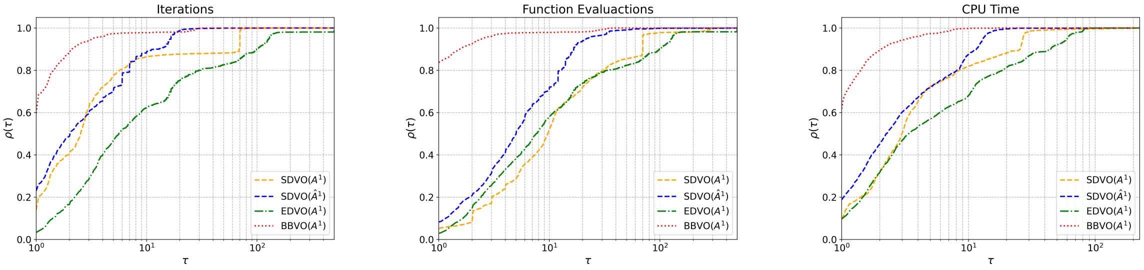
7.3. Numerical results for VOPs with $K = K_1$

In this case, we denote the set of transform matrices $\mathcal{A}_1 := \{A : AK_1 = \mathbb{R}^2_+\}$. For SDVO, we choose $A^1 = \begin{pmatrix} 5 & -1 \\ -1 & 5 \end{pmatrix} \in \mathcal{A}_0$ and $\hat{A}^1 \in \mathcal{A}_1$ in subproblem, respectively, where $\hat{A}^1 := \{A : A_i = A^1_i / \max\{1, \left\|\nabla F_i(x^0)\right\|_\infty\},\ i = 1, 2\}$.

[1] The scale strategy is initially proposed in (Gonçalves et al., 2022) due to numerical reasons.

Table 3: Number of average iterations (iter), number of average function evaluations (feval), and average CPU time (time(ms)) of SDVO, EDVO and BBDVO implemented on different test problems with $K = K_1$

Problem	SDVO with $A = A^1$			SDVO with $A = \hat{A}^1$			EDVO with $A = A^1$			BBDVO with $A = A^1$		
	iter	feval	time	iter	feval	time	iter	feval	time	iter	feval	time
BK1	**1.00**	2.65	**0.21**	7.51	9.87	1.21	48.30	51.56	4.07	**1.00**	**1.00**	0.22
DD1	92.40	373.32	18.42	118.35	120.30	15.34	496.99	496.99	53.13	**41.85**	**47.12**	**6.39**
Deb	102.21	876.19	27.68	**27.82**	146.28	**6.29**	81.45	161.00	9.44	35.69	**71.88**	6.48
FF1	33.19	105.29	5.85	17.23	56.82	3.16	45.35	65.10	4.73	**16.00**	**16.95**	**2.61**
Hil1	28.28	151.71	7.15	19.41	86.98	4.71	23.09	56.07	3.51	**17.74**	**18.35**	**3.10**
Imbalance1	77.34	309.32	13.55	397.37	397.44	46.97	500.00	500.00	50.14	**28.78**	**31.18**	**4.49**
JOS1a	69.86	69.86	6.82	6.57	17.06	2.55	117.30	117.43	13.78	**1.00**	**1.00**	**0.26**
LE1	14.02	69.28	2.88	12.37	55.82	2.46	15.99	39.60	1.92	**6.31**	**7.49**	**1.14**
PNR	27.10	157.72	5.71	12.76	37.65	2.15	21.80	27.54	2.41	**9.78**	**10.97**	**1.64**
WIT1	244.44	2084.16	111.19	199.03	245.92	24.42	414.29	416.78	42.52	**158.78**	**164.91**	**22.80**



(a) Iterations (b) Function Evaluations (c) CPU Time

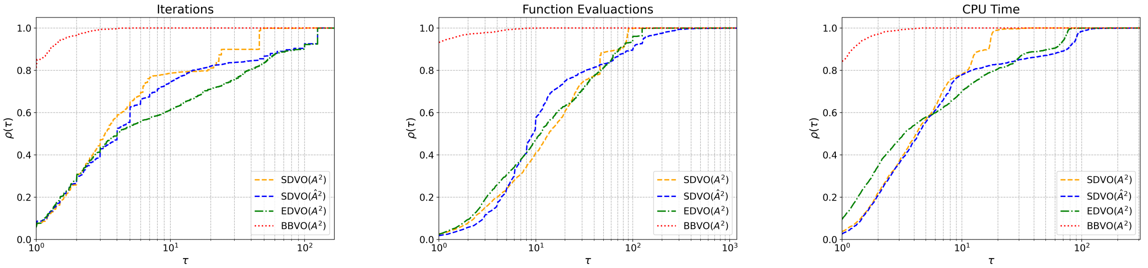
Fig. 6. Performance profiles on the test problems in Table 2 with $K = K_1$.

7.4. Numerical results for VOPs with $K = K_2$

We denote the set of transform matrices $\mathcal{A}_2 := \{A : AK_2 = \mathbb{R}^2_+\}$. For SDVO, we choose $A^2 = \begin{pmatrix} 5 & 1 \\ 1 & 5 \end{pmatrix} \in \mathcal{A}_0$ and $\hat{A}^2 \in \mathcal{A}_2$ in subproblem, respectively, where $\hat{A}^2 := \{A : A_i = A^2_i / \max\{1, \left\|\nabla F_i(x^0)\right\|_\infty\},\ i = 1, 2\}$.

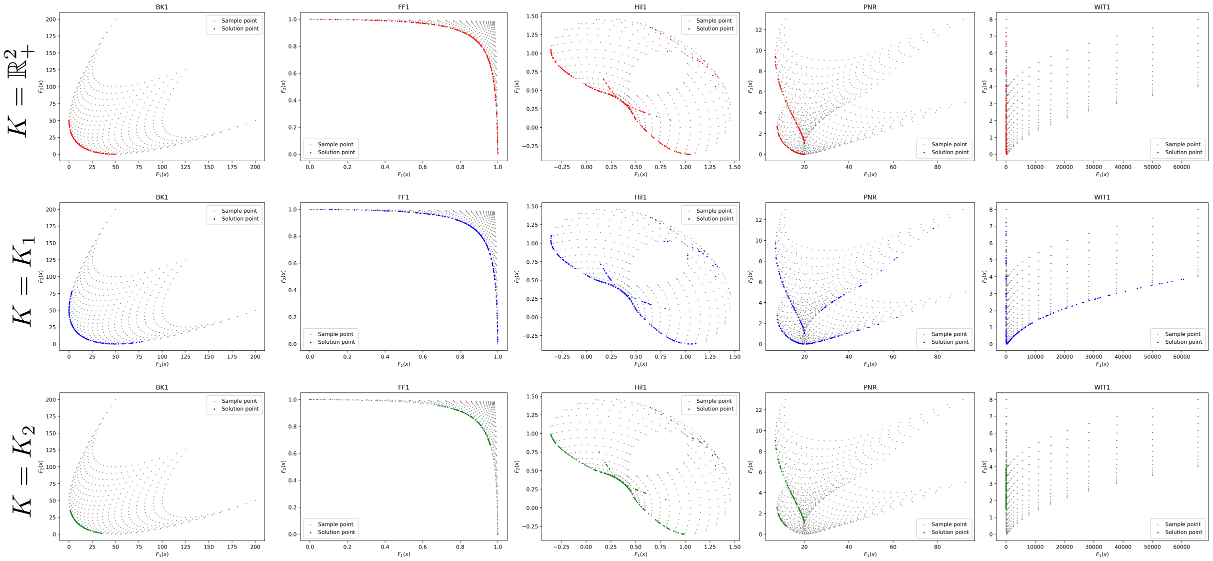
Table 4: Number of average iterations (iter), number of average function evaluations (feval), and average CPU time (time(ms)) of SDVO, EDVO and BBDVO implemented on different test problems with $K = K_2$

Problem	SDVO with $A = A^2$			SDVO with $A = \hat{A}^2$			EDVO with $A = A^2$			BBDVO with $A = A^2$		
	iter	feval	time	iter	feval	time	iter	feval	time	iter	feval	time
BK1	22.29	85.27	2.52	6.67	11.79	1.25	20.69	26.53	1.92	**1.00**	**1.00**	**0.21**
DD1	17.16	62.14	3.37	46.38	47.71	5.03	133.89	134.00	13.33	**4.84**	**5.29**	**0.83**
Deb	19.48	164.15	5.43	18.22	135.69	4.63	21.95	141.27	4.40	**9.47**	**48.94**	**2.02**
FF1	13.52	18.80	1.96	14.76	67.80	3.04	17.68	122.48	4.04	**4.72**	**5.74**	**0.91**
Hil1	14.69	75.19	3.86	17.60	82.61	4.44	16.61	42.27	2.65	**8.38**	**9.24**	**1.52**
Imbalance1	25.76	105.22	4.60	500.00	610.90	61.64	500.00	500.04	48.91	**4.35**	**5.76**	**0.73**
JOS1a	46.11	46.11	4.47	4.58	10.16	1.77	48.35	51.67	6.34	**1.00**	**1.00**	**0.26**
LE1	9.64	47.62	2.05	14.79	79.00	3.08	11.40	54.09	1.95	**7.58**	**43.00**	**1.63**
PNR	17.22	103.38	3.60	13.30	50.20	2.41	11.12	36.16	1.69	**6.80**	**8.83**	**1.04**
WIT1	23.12	140.31	5.32	311.78	975.91	44.38	10.46	29.67	1.52	**8.66**	**9.95**	**1.27**



(a) Iterations (b) Function Evaluations (c) CPU Time

Fig. 7. Performance profiles on the test problems in Table 2 with $K = K_2$.



(a) BK1 (b) FF1 (c) Hil1 (d) PNR (e) WIT1

Fig. 8. Numerical results in value space obtained by BBDVO for problems BK1, FF1, Hil1, PNR and WIT1 with $K = \mathbb{R}^2_+$, $K = K_1$ and $K = K_2$, respectively.

For test problems with different partial orders, the number of average iterations (iter), number of average function evaluations (feval), and average CPU time (time(ms)) of the different algorithms are listed in Tables 2, 3 and 4, respectively. We conclude that BBDVO outperforms SDVO and EDVO, especially for problems DD1 and Imbalance1. For SDVO, its performance is sensitive to the choice of transform matrix, changing the transform matrix in subproblem cannot improve the performance on all test problems. Naturally, a question arises that how to choose an appropriate transform matrix for a specific test problem in SDVO. It is worth noting that BBDVO can be viewed as SDVO with variable transform matrices ($\Lambda^k A$ is a transform matrix of K) and thus enjoys promising performance on these test problems. This provides a positive answer to the question. EDVO can also be viewed as SDVO with variable transform matrices, it generates descent directions with norm less than 1 (the minimizer of subproblem is the minimal norm element of the convex hull of some unit vectors), decelerating the convergence in large-scale problems (the initial point may be far from the Pareto set). Figs. 5, 6 and 7 present the performance profiles based on iterations, function evaluations, and CPU time. The results confirm that the proposed BBDVO significantly outperforms SDVO and EDVO.

Fig. 8 plots the final points obtained by BBDVO on problems BK1, FF1, Hil1, PNR and WIT1 with $K = \mathbb{R}^2_+$, $K = K_1$ and $K = K_2$, respectively. We can observe that enlarging the partial order cone reduces the number of obtained Pareto critical points, especially in the long tail regions, where improving one objective function slightly can sacrifice the others greatly. As a result, we can use an order cone containing the non-negative orthant in real-world MOPs to obtain Pareto points with a better trade-off.

8. Conclusions

In this paper, we develop a unified framework and convergence analysis of descent methods for VOPs from a majorization-minimization perspective. We emphasize that the convergence rate of a descent method can be improved by narrowing the surrogate functions. By changing the base in subproblems, we elucidate that choosing a tighter surrogate function is equivalent to selecting an appropriate base of the dual cone. From the majorization-minimization perspective, we employ Barzilai-Borwein method to narrow the local surrogate functions and propose a Barzilai-Borwein descent method for VOPs polyhedral cone. The proposed method is not sensitive to the choice of transform matrix, which affects the performance of SDVO. Numerical experiments confirm the efficiency of the proposed method.

From a majoriztion-minimization perspective, we also rediscover the preconditioned Barzilai-Borwein method for MOPs. This highlights the versatility of the majorization-minimization principle as a powerful framework for designing novel algorithms in vector optimization. In future work, it is worth analyzing proximal gradient methods and high-order methods for VOPs within the majorization-minimization framework. Additionally, exploring solution methods for VOPs with non-polyhedral cones presents an intriguing avenue for further research.

Acknowledgement

This work was funded by the National Key Research and Development Program of China [grant number 2023YFA1011504]; the Major Program of the National Natural Science Foundation of China [grant numbers 11991020, 11991024]; the Key Program of the National Natural Science Foundation of China [grant number 12431010]; the General Program of the National Natural Science Foundation of China [grant number 12171060]; NSFC-RGC (Hong Kong) Joint Research Program [grant number 12261160365]; the Team Project of Innovation Leading Talent in Chongqing [grant number CQYC20210309536]; the Natural Science Foundation of Chongqing [grant numbers ncamc2022-msxm01, CSTB2024NSCQ-LZX0140]; Major Project of Science and Technology Research Rrogram of Chongqing Education Commission of China [grant number KJZD-M202300504]; the Chongqing Postdoctoral Research Project Special Grant [grant number 2024CQB-SHTB1007]; the Science and Technology Research Program of Chongqing Education commission of China [grant number KJQN202400520] and Foundation of Chongqing Normal University [grant numbers 22XLB005, 22XLB006].

Data Availability

The data that support the findings of this study are available from the first author, [Jian Chen], upon reasonable request.

References

Aliprantis, C., Florenzano, M., da Rocha, V. M., & Tourky, R. (2004a). Equilibrium analysis in financial markets with countably many securities. *Journal of Mathematical Economics*, *40*, 683–699. URL: https://doi.org/10.1016/j.jmateco.2003.06.003.

Aliprantis, C. D., Florenzano, M., & Tourky, R. (2004b). General equilibrium analysis in ordered topological vector spaces. *Journal of Mathematical Economics*, *40*, 247–269. URL: https://doi.org/10.1016/j.jmateco.2003.11.004.

Barzilai, J., & Borwein, J. M. (1988). Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, *8*, 141–148. URL: https://doi.org/10.1093/imanum/8.1.141.

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

Carrizo, G. A., Lotito, P. A., & Maciel, M. C. (2016). Trust region globalization strategy for the nonconvex unconstrained multiobjective optimization problem. *Mathematical Programming*, *159*, 339–369. URL: https://doi.org/10.1007/s10107-015-0962-6.

Chen, J., Tang, L. P., & Yang, X. M. (2022). Convergence rates analysis of interior Bregman gradient method for vector optimization problems. *arXiv preprint arXiv:2206.10070*, .

Chen, J., Tang, L. P., & Yang, X. M. (2023a). A Barzilai-Borwein descent method for multiobjective optimization problems. *European Journal of Operational Research*, *311*, 196–209. URL: https://https://doi.org/10.1016/j.ejor.2023.04.022.

Chen, J., Tang, L. P., & Yang, X. M. (2023b). Barzilai-Borwein proximal gradient methods for multiobjective composite optimization problems with improved linear convergence. *arXiv preprint arXiv:2306.09797v2*, . URL: https://arxiv.org/pdf/2306.09797v2.pdf.

Chen, W., Yang, X. M., & Zhao, Y. (2023c). Conditional gradient method for vector optimization. *Computational Optimization and Applications*, *85*, 857–896.

Das, I., & Dennis, J. E. (1998). Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, *8*, 631–657. URL: https://doi.org/10.1137/S1052623496307510.

Deb, K. (1999). Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation*, *7*, 205–230. URL: https://doi.org/10.1162/evco.1999.7.3.205.

Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, *91*, 201–213.

Graña Drummond, L. M., Raupp, F. M. P., & Svaiter, B. F. (2014). A quadratically convergent Newton method for vector optimization. *Optimization*, *63*, 661–677.

Graña Drummond, L. M., & Svaiter, B. F. (2005). A steepest descent method for vector optimization. *Journal of Computational and Applied Mathematics*, *175*, 395–414.

Evans, G. (1984). Overview of techniques for solving multiobjective mathematical programs. *Management Science*, *30*, 1268–1282. URL: https://doi.org/10.1287/mnsc.30.11.1268.

Fliege, J., Graña Drummond, L. M., & Svaiter, B. F. (2009). Newton's method for multiobjective optimization. *SIAM Journal on Optimization*, *20*, 602–626. URL: https://doi.org/10.1137/08071692X.

Fliege, J., & Svaiter, B. F. (2000). Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, *51*, 479–494. URL: https://doi.org/10.1007/s001860000043.

Fliege, J., & Vaz, A. I. F. (2016). A method for constrained multiobjective optimization based on SQP techniques. *SIAM Journal on Optimization*, *26*, 2091–2119. URL: https://doi.org/10.1137/15M1016424.

Fliege, J., Vaz, A. I. F., & Vicente, L. N. (2018). Complexity of gradient descent for multiobjective optimization. *Optimization Methods and Software*, *34*, 949–959. URL: https://doi.org/10.1080/10556788.2018.1510928.

Fliege, J., & Werner, R. (2014). Robust multiobjective optimization & applications in portfolio optimization. *European Journal of Operational Research*, *234*, 422–433. URL: https://doi.org/10.1016/j.ejor.2013.10.028.

Gonçalves, M. L. N., Lima, F. S., & Prudente, L. F. (2022). A study of Liu-Storey conjugate gradient methods for vector optimization. *Applied Mathematics and Computation*, *425*, 127099. URL: https://doi.org/10.1016/j.amc.2022.127099.

Graña Drummond, L. M., & Iusem, A. N. (2004). A projected gradient method for vector optimization problems. *Computational Optimization and Applications*, *28*, 5–29. URL: https://doi.org/10.1023/B:COAP.0000018877.86161.8b.

Hillermeier, C. (2001). Generalized homotopy approach to multiobjective optimization. *Journal of Optimization Theory and Applications*, *110*, 557–583. URL: https://doi.org/10.1023/A:1017536311488.

Huband, S., Hingston, P., Barone, L., & While, L. (2006). A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, *10*, 477–506. URL: https://doi.org/10.1109/TEVC.2005.861417.

Jahn, J. (2011). *Vector optimization: theory, applications and extensions*. Berlin: Springer.

Jin, Y., Olhofer, M., & Sendhoff, B. (2001). Dynamic weighted aggregation for evolutionary multi-objective optimization: Why does it work and how? In *Proceedings of the Genetic and Evolutionary Computation Conference* (pp. 1042–1049).

Karimi, B., Wai, H.-T., Moulines, E., & Li, P. (2022). Minimization by incremental stochastic surrogate optimization for large scale nonconvex problems. In S. Dasgupta, & N. Haghtalab (Eds.), *Proceedings of The 33rd International Conference on Algorithmic Learning Theory* (pp. 606–637). PMLR volume 167 of *Proceedings of Machine Learning Research*. URL: https://proceedings.mlr.press/v167/karimi22a.

html.

Katrutsa, A., Merkulov, D., Tursynbek, N., & Oseledets, I. (2020). Follow the bisector: a simple method for multi-objective optimization. *arXiv preprint arXiv:2007.06937*, . URL: https://arxiv.org/pdf/2007.06937.

Landeros, A., Xu, J., & Lange, K. (2023). Mm optimization: Proximal distance algorithms, path following, and trust regions. *Proceedings of the National Academy of Sciences*, *120*, e2303168120. doi:10.1073/pnas.2303168120.

Lanza, A., Morigi, S., Selesnick, I., & Sgallari, F. (2017). Nonconvex nonsmooth optimization via convexnonconvex majorizationminimization. *Numerische Mathematik*, *136*.

Lapucci, M. (2024). Convergence and complexity guarantees for a wide class of descent algorithms in nonconvex multi-objective optimization. *Operations Research Letters*, *54*, 107115. URL: https://doi.org/10.1016/j.orl.2024.107115.

Leschine, T. M., Wallenius, H., & Verdini, W. A. (1992). Interactive multiobjective analysis and assimilative capacity-based ocean disposal decisions. *European Journal of Operational Research*, *56*, 278–289. URL: https://doi.org/10.1016/0377-2217(92)90228-2.

Lucambio Pérez, L. R., & Prudente, L. F. (2018). Nonlinear conjugate gradient methods for vector optimization. *SIAM Journal on Optimization*, *28*, 2690–2720. URL: https://doi.org/10.1137/17M1126588.

Mairal, J. (2015). Incremental majorization-minimization optimization with application to large-scale machine learning. *SIAM Journal on Optimization*, *25*, 829–855. URL: https://doi.org/10.1137/140957639.

Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, *26*, 369–395. URL: https://doi.org/10.1007/s00158-003-0368-6.

Mercier, Q., Poirion, F., & Désidéri, J. A. (2018). A stochastic multiple gradient descent algorithm. *European Journal of Operational Research*, *271*, 808–817. URL: https://doi.org/10.1016/j.ejor.2018.05.064.

Morovati, V., & Pourkarimi, L. (2019). Extension of Zoutendijk method for solving constrained multiobjective optimization problems. *European Journal of Operational Research*, *273*, 44–57. URL: https://doi.org/10.1016/j.ejor.2018.08.018.

Povalej, Ž. (2014). Quasi-Newton's method for multiobjective optimization. *Journal of Computational and Applied Mathematics*, *255*, 765–777. URL: https://doi.org/10.1016/j.cam.2013.06.045.

Preuss, M., Naujoks, B., & Rudolph, G. (2006). Pareto set and EMOA behavior for simple multimodal multiobjective functions. In T. P. Runarsson, H.-G. Beyer, E. Burke, J. J. Merelo-Guervós, L. D. Whitley, & X. Yao (Eds.), *Parallel Problem Solving from Nature - PPSN IX* (pp. 513–522). Berlin, Heidelberg: Springer Berlin Heidelberg.

Qu, S. J., Goh, M., & Chan, F. T. (2011). Quasi-Newton methods for solving multiobjective optimization. *Operations Research Letters*, *39*, 397–399. URL: https://doi.org/10.1016/j.orl.2011.07.008.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press.

Sener, O., & Koltun, V. (2018). Multi-task learning as multi-objective optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc. volume 31. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/432aca3a1e345e339f35a30c8f65edce-Paper.pdf.

Sion, M. (1958). On general minimax theorems. *Pacific Journal of Mathematics*, *8*, 171–176. URL: https://doi.org/10.2140/PJM.1958.8.171.

Tanabe, H., Fukuda, E. H., & Yamashita, N. (2019). Proximal gradient methods for multiobjective optimization and their applications. *Computational Optimization and Applications*, *72*, 339–361. URL: https://doi.org/10.1007/s10589-018-0043-x.

Tanabe, H., Fukuda, E. H., & Yamashita, N. (2023). Convergence rates analysis of a multiobjective proximal gradient method. *Optimization Letters*, *17*, 333–350. URL: https://doi.org/10.1007/s11590-022-01877-7.

Tapia, M. G. C., & Coello, C. A. C. (2007). Applications of multi-objective evolutionary algorithms in economics and finance: A survey. In *2007 IEEE Congress on Evolutionary Computation* (pp. 532–539). URL: https://doi.org/10.1109/CEC.2007.4424516.

Witting, K. (2012). *Numerical algorithms for the treatment of parametric multiobjective optimization problems and applications*. Ph.D. thesis Paderborn, Universität Paderborn, Diss., 2012.

Ye, F. Y., Lin, B. J., Yue, Z. X., Guo, P. X., Xiao, Q., & Zhang, Y. (2021). Multi-objective meta learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, & J. W. Vaughan (Eds.), *Advances in Neural Information Processing Systems* (pp. 21338–21351). Curran Associates, Inc. volume 34. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/b23975176653284f1f7356ba5539cfcb-Paper.pdf.

Zeng, L. Y., Dai, Y. H., & Huang, Y. K. (2019). Convergence rate of gradient descent method for multi-objective optimization. *Journal of Computational Mathematics*, *37*, 689–703. URL: https://doi.org/10.4208/jcm.1808-m2017-0214.