

线性约束多目标规划的有效集方法^{*}

马伊婷, 黄应全, 李高西

(重庆工商大学 数学与统计学院, 重庆 400067)

摘要:有效集方法因自身的安全性和可靠性而在实际应用中起着至关重要的作用。本文提出了求解带有线性约束多目标规划的有效集方法。该方法通过一个容易求解的特殊的线性规划问题生成可行方向,并通过有显式解的精确线搜索问题生成步长。在较弱的条件下,证明了上述算法产生的序列收敛到原问题的 Karush-Kuhn-Tucker 稳定点,并在凸性的假设下证明了这一算法的聚点是原问题的弱有效解。最后,通过数值实验验证了该算法在求解多目标优化问题中的有效性与优越性。

关键词:多目标规划;有效集方法;线性约束

中图分类号:O221.6

文献标志码:A

文章编号:1672-6693(2026)03-0020-10

本文研究如下线性约束多目标规划问题:

$$\begin{cases} \min_{x \in \mathbf{R}^n} f(x) = (f_1(x), f_2(x), \dots, f_m(x))^T, j=1, 2, \dots, m; \\ \text{s. t. } g_i(x) = a_i^T x - b_i = 0, i \in \mathcal{E} = \{1, 2, \dots, l\}; \\ g_i(x) = a_i^T x - b_i \geq 0, i \in \mathcal{I} = \{l+1, l+2, \dots, r\}. \end{cases} \quad (1)$$

其中:目标函数 $f(x)$ 的各分量 $f_j(x): \mathbf{R}^n \rightarrow \mathbf{R}, j=1, 2, \dots, m$ 均为二阶连续可微的函数;约束函数 $g_i(x): \mathbf{R}^n \rightarrow \mathbf{R}, i=1, 2, \dots, r$ 均为线性函数且满足 $l < r$ 。假设式(1)的可行域是有界的,记为集合 V ,且有:

$$V = \{x \in \mathbf{R}^n \mid g_i(x) = 0, i \in \mathcal{E}; g_i(x) \geq 0, i \in \mathcal{I}\}.$$

同时,每个目标分量 f_j 的梯度 ∇f_j 在可行域 V 上是 L_j -Lipschitz 连续的。

多目标优化问题在经济、工程、管理等实际领域中具有广泛的应用^[1-9],该类问题的核心是在满足约束条件的前提下寻找能够平衡多个相互冲突的目标的最优解,这类解被定义为 Pareto 最优解或有效解。求解多目标优化问题的一个非常普遍的策略是将标量优化方法扩展到多目标优化方法中,而不是直接使用标量优化方法^[10]。这种策略起源于 2000 年 Fliege 等人^[11]将单个问题的最速下降算法推广到无约束多目标优化问题中。从此之后,越来越多的学者对此策略产生了兴趣,并发现了与该方法相关的一些新性质,且考虑了它的几种变体^[12-17]。2021 年,Assunção 等人^[18]引入了条件梯度方法求解约束多目标优化问题,通过考虑不同步长获取策略,建立了目标函数在无凸性假设和有凸性假设下的渐近收敛性质与迭代复杂度界限。2023 年,Assunção 等人^[19]又使用条件梯度法在同样的步长策略下求解了复合多目标优化问题,其中每个目标函数都是由光滑函数与非光滑函数组成的;Chen 等人^[20]基于文献[18]中的工作,提出了一种用于求解具有非空内部的闭凸锥所诱导的偏序关系下的约束向量优化问题的条件梯度法,并在目标函数满足凸性假设的条件下,保证了方法的收敛性;杨春蓉等人^[21]提出了求解多目标优化问题的非单调对角最速下降算法,并使用非单调的 Armijo 准则求解下降方向,同时还进行了收敛性分析。现有的多目标优化问题求解方法虽已得到较为广泛的应用,但这些方法存在明显局限。例如投影梯度法对约束边界的处理不够灵活,易受约束结构影响导致局部收敛困境^[22]。又如最速下降法依赖单目标函数的梯度方向,而在多目标场景下各目标分量的梯度相互冲突;且该方法不存在统一的“最速下降方向”能同时实现对所有目标的有效改进,需额外调整,由此导致原始“最速”特性丧失;同时该方法易陷入局部非支配点,

* 收稿日期:2025-12-25 修回日期:2026-03-06 网络出版时间:2026-04-10T15:12

资助项目:重庆市自然科学基金面上项目(No. cstc2021jcyj-msxmX0499, No. CSTB2025NSCQ-GPX0817);重庆市研究生导师团队建设项目(No. yds223010)

第一作者简介:马伊婷,女,研究方向为最优化理论与算法,E-mail: mayiting0213@163.com;通信作者简介:黄应全,男,讲师,E-mail: huangyq1110@ctbu.edu.cn

网络出版地址:https://link.cnki.net/urlid/50.1165.N.20260409.1627.010

难以覆盖全局 Pareto 前沿^[11]。而在优化领域中,有效集方法对约束有着高效的处理能力,在单目标带约束优化问题中已发展成熟。有效集方法通过在迭代过程维护“有效约束集”,将复杂的带约束优化问题转化为仅考虑有效约束的简化子问题,明显降低了计算复杂度,尤其在处理线性约束单目标问题时,展现出良好的稳定性和求解效率。多目标优化问题的目标函数的向量特性使得单目标有效集方法难以直接迁移,Halffmann 等人^[23]指出,当前针对线性约束多目标规划的算法研究仍集中于整数变量场景,面向连续型问题的有效集方法尚未形成完善的理论体系和高效算法框架。因此,构造一种适配线性约束特性、兼顾计算效率与优化性能的多目标有效集方法,成为解决此类问题的关键需求。

基于上述研究背景,本文从有效集的核心概念出发,提出一种求解线性约束多目标规划的有效集方法。在迭代过程中,基于当前迭代点的有效集(等式约束与当前起作用的不等式约束的并集),构造易于求解的特殊线性规划子问题以生成可行方向,同时通过具有显式解的精确线搜索确定最优步长,确保算法在满足约束条件的前提下实现目标函数的有效改进。本文的主要贡献体现在以下2个方面:

1) 构造了适用于线性约束多目标规划的有效集方法,通过分解复杂优化问题为简单子问题,降低了计算难度,且迭代过程中动态更新有效集,适配约束结构的变化,实用性强。

2) 完善了算法的收敛性理论,在较弱的假设条件下,证明了算法生成的迭代序列收敛到原问题的 Karush-Kuhn-Tucker(KKT)稳定点,且迭代序列的聚点为原问题的弱有效解,为算法的可靠性提供了理论保障。

1 预备知识

设 $\mathbf{R}$ 是实数集。在 n 维欧氏空间 $\mathbf{R}^n$ 中,内积和范数分别用 $\langle \cdot, \cdot \rangle$ 和 $\| \cdot \|$ 表示。设 $J := \{1, 2, \dots, m\}$, $L := \max_{j \in J} L_j$ 。这里 L_j 是 ∇f_j 的 Lipschitz 常数。函数 f 在 $\mathbf{x} \in \mathbf{R}^n$ 上的 Jacobi 矩阵用 $\mathbf{J}_f$ 表示为含有以下元素的 $m \times n$ 矩阵:

$$[\mathbf{J}_f]_{i,j} = \frac{\partial f_i}{\partial x_j}(\mathbf{x}).$$

通过求解子问题式(2)来寻找式(1)的解:

$$\begin{cases} \min_{\mathbf{x} \in \mathbf{R}^n} \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))^\top, j=1, 2, \dots, m; \\ \text{s. t. } g_i(\mathbf{x}) = \mathbf{a}_i^\top \mathbf{x} - b_i = 0, i \in S(\mathbf{x}_k). \end{cases} \quad (2)$$

记 $\mathcal{I}(\mathbf{x}_k) = \{i \mid \mathbf{a}_i^\top \mathbf{x}_k - b_i = 0, i \in \mathcal{I}\}$, 在点 $\mathbf{x}$ 处的有效集为 $S(\mathbf{x}_k) = \mathcal{E} \cup \mathcal{I}(\mathbf{x}_k)$ (也即等式约束与当前起作用的不等式约束的并集)。

定义 1^[24] 如果不存在 $\mathbf{y} \in V$ 使得 $\mathbf{f}(\mathbf{y}) \leq \mathbf{f}(\mathbf{x}_k)$ 且 $\mathbf{f}(\mathbf{y}) \neq \mathbf{f}(\mathbf{x}_k)$ ($\mathbf{f}(\mathbf{y}) < \mathbf{f}(\mathbf{x}_k)$), 则称点 $\mathbf{x}_k \in V$ 是式(1)的有效(弱有效)解。此时, $\mathbf{f}(\mathbf{x}_k)$ 被称为非支配(弱非支配)点。

弱有效解的判定可通过一阶条件精准刻画:当目标与可行集为凸时,点 $\mathbf{x}^* \in V$ 是式(1)的弱有效解的充要条件是 $\mathbf{x}^*$ 是 Pareto 临界(稳定)的,也即:

$$\max_{j=1, \dots, m} \langle \nabla f_j(\mathbf{x}^*), \mathbf{y} - \mathbf{x}^* \rangle \geq 0, \forall \mathbf{y} \in V. \quad (3)$$

定义 2^[24] 如果存在一个邻域 $W \subseteq V$ 使得点 $\mathbf{x}_k$ 是 f 在 W 上的有效(弱有效)解,则称点 $\mathbf{x}_k \in V$ 是式(1)的局部有效(弱有效)解。此时, $\mathbf{f}(\mathbf{x}_k)$ 被称为局部非支配(弱非支配)点。

实际上,在本文基于方向的算法中,每次迭代时都需要一个同时改进所有目标函数的可行方向。

定义 3^[24] 如果存在 $\alpha_0 > 0$ 使得 $\forall \alpha \in (0, \alpha_0]$ 和 $j \in 1, 2, \dots, m$ 有 $f_j(\mathbf{x}_k + \alpha \mathbf{d}) < f_j(\mathbf{x}_k)$, 则称 $\mathbf{d} \in \mathbf{R}^n$ 是式(1)在点 $\mathbf{x}_k \in V$ 处的改进方向。

定义 3 中提及的改进方向被认为是所有目标函数同时改进的方向。

定义 4^[24] 如果存在 $\alpha_0 > 0$ 使得 $\forall \alpha \in (0, \alpha_0]$ 和 $i \in S(\mathbf{x}_k)$ 有 $\mathbf{a}_i^\top \mathbf{d} = 0$, 则称 $\mathbf{d} \in \mathbf{R}^n$ 是式(1)在点 $\mathbf{x}_k \in V$ 处的可行方向。

从理论和计算的角度来看,凸性是优化中的一个有用的假设。在存在凸性的情况下,一个问题的全局解可以由它的局部解导出。

定义 5^[24] 设 U 为 $\mathbf{R}^n$ 上的一个凸集,如果它的所有分量都是凸的,则向量函数 $f: U \rightarrow \mathbf{R}^m$ 被称为 $\mathbf{R}^m$ -凸 ($\mathbf{R}^m$ -严格凸)函数。

引理 1^[24] 设 U 为 $\mathbf{R}^n$ 上的一个凸集, $f: U \rightarrow \mathbf{R}^p$ 是一个 $\mathbf{R}^p$ -凸函数, 则式(1)的任何局部有效解都是有效解。

由于本文所考虑问题的目标函数 $f(x)$ 是具有凸性的, 且可行域为凸集, 根据引理 1, 式(1)的任何局部有效解都是全局有效解, 因此本文中提到的解均为全局解。

2 多目标问题的有效集方法

对一个给定点 $x_k \in V$, 定义搜索方向为 $d(x_k) := p(x_k) - x_k$, 有:

$$p(x_k) \in \arg \min_{p \in V} \max_{j \in J} \{ \nabla f_j(x_k)^T (p - x_k) : a_i^T (p - x_k) = 0, i \in S(x_k) \}. \quad (4)$$

因为可行域 V 是由线性约束组成的有界集合, 因此 V 是紧集, 式(4)有解, 且 $p(x_k)$ 定义良好。记式(4)的最优值为 $\gamma(x_k)$, 即:

$$\gamma(x_k) := \max_{j \in J} \{ \nabla f_j(x_k)^T (p(x_k) - x_k) : a_i^T (p(x_k) - x_k) = 0, i \in S(x_k) \}. \quad (5)$$

接下来给出约束多目标优化式(1)临界点的定义。

定义 6 如果 $\forall i \in S(x_k), \forall d \in \mathbf{R}^n$ 满足 $a_i^T d = 0$, 存在 $j \in J$ 使得 $\nabla f_j(x_k)^T d \geq 0$, 则称解 $x_k \in V$ 为式(1)的临界点。

利用上述临界点的定义, 可以给出寻找改进可行方向的一个直接结果。

推论 1 设 $x_k \in V$ 为式(1)的一个非临界点, 则在点 x_k 处存在改进方向 d 使得:

$$\begin{cases} \nabla f_j(x_k)^T d < 0, \forall j \in J; \\ a_i^T d = 0, \forall i \in S(x_k). \end{cases}$$

不可微问题式(4)的解可以通过求解下面的约束线性规划问题得到:

$$\begin{cases} \min_{(\tau, p)} \tau; \\ \text{s. t. } \nabla f_j(x_k)^T (p - x_k) \leq \tau; \\ a_i^T (p - x_k) = 0, i \in S(x_k); \\ \tau \in \mathbf{R}, p \in V. \end{cases} \quad (6)$$

下面给出关于函数 $\gamma(\cdot)$ 的性质的一个引理, 用于说明函数 $\gamma(\cdot)$ 和式(1)的弱有效解之间的关系, 也是本文算法的停机准则。

引理 2 关于式(1), 下列命题成立: 1) $\forall x \in V, \gamma(x) \leq 0$; 2) $\gamma(\cdot)$ 是连续函数; 3) $\gamma(x) = 0$ 当且仅当 $x \in V$ 是式(1)的稳定点。

证明 1) 因为 $x \in V$, 根据式(4)和式(5), 有 $\gamma(x) \leq \max_{j \in J} \nabla f_j(x)^T (x - x) = 0$ 。

2) 设 $x \in V$, 考虑序列 $\{x_k\} \subset V$ 使得 $\lim_{k \rightarrow \infty} x_k = x$ 。一方面, 因为 $p(x_k) \in V$, 根据式(4)和式(5)有 $\gamma(x_k) \leq \max_{j \in J} \nabla f_j(x_k)^T (p(x_k) - x_k)$, 又因为函数 f_j 是二阶连续可微的且 $\mathbf{R}^m \ni p \mapsto \max_{j \in J} p_j$ 是连续的, 故有:

$$\limsup_{k \rightarrow \infty} \gamma(x_k) \leq \gamma(x). \quad (7)$$

另一方面, 因为 $p(x_k) \in V$, 有:

$$\begin{aligned} \gamma(x) &\leq \max_{j \in J} \nabla f_j(x)^T (p(x_k) - x) = \max_{j \in J} \nabla f_j(x)^T (p(x_k) - x_k + x_k - x) \leq \\ &\max_{j \in J} \nabla f_j(x)^T (p(x_k) - x_k) + \max_{j \in J} \nabla f_j(x)^T (x_k - x). \end{aligned}$$

因此, 在上述不等式两侧取极限 $\liminf_{k \rightarrow \infty}$, 使用连续性论证, 有:

$$\begin{aligned} \gamma(x) &\leq \liminf_{k \rightarrow \infty} \max_{j \in J} \nabla f_j(x)^T (p(x_k) - x_k) = \\ \liminf_{k \rightarrow \infty} (\gamma(x_k) + \max_{j \in J} \nabla f_j(x)^T (p(x_k) - x_k) - \max_{j \in J} \nabla f_j(x_k)^T (p(x_k) - x_k)) &\leq \\ \liminf_{k \rightarrow \infty} (\gamma(x_k) + \|J_f(x_k) - J_f(x)\| \|p(x_k) - x_k\|), \end{aligned}$$

这里的第 2 个不等式成立是因为 $\mathbf{R}^m \ni p \mapsto \max_{j \in J} p_j$ 是 Lipschitz 连续的且常数为 1。因为函数 f_j 是二阶连续可微的, 因此 ∇f_j 也是连续函数, 又已知 V 是紧集, 且 $\|p(x_k) - x_k\| \in V$, 有:

$$\gamma(x) \leq \liminf_{k \rightarrow \infty} \gamma(x_k). \quad (8)$$

结合式(7)和式(8)有 $\lim_{k \rightarrow \infty} \gamma(x_k) = \gamma(x)$ 。

3) 假设 $x \in V$ 是式(1)的平稳点,也即 $\forall p \in V$, 有 $\max_{j \in J} \nabla f_j(x)^T(p-x) \geq 0$ 。因为 $p(x) \in V$, 故有 $\gamma(x) = \max_{j \in J} \nabla f_j(x)^T(p(x)-x) \geq 0$, 结合引理 2 命题 1) 中 $\gamma(x) \leq 0$, 可得 $\gamma(x) = 0$ 。反过来, 假设 $\gamma(x) = 0$ 。根据式(5)中 $\gamma(\cdot)$ 的定义, 有: $0 = \gamma(x) \leq \max_{j \in J} \nabla f_j(x)^T(p-x)$, $\forall p \in V$ 。这表示点 x 满足式(3), 故点 x 是式(1)的稳点。

证毕

基于上述内容, 本文提出了用于求解式(1)的有效集方法, 算法具体内容如下。

算法 1 多目标规划的有效集算法。

步骤 0(初始化), 选取任意初始点 $x_0 \in \mathbf{R}^n$, 令 $k=0, \epsilon > 0$ 。

步骤 1(计算搜索方向), 在点 x_k 处确定相应的有效集 $S(x_k) = \mathcal{E} \cup \mathcal{I}(x_k)$, 并根据式(4)和式(5)求解子问题式(6), 得到 $\gamma(x_k)$ 和 $p(x_k)$ 。定义搜索方向为 $d(x_k) := p(x_k) - x_k$ 。

步骤 2(终止准则), 若 $|\gamma(x_k)| < \epsilon$, 停止计算, 输出 x_k 作为解。

步骤 3(计算步长和新迭代), 令 $\alpha_k = \min\{\alpha_k^{\max}, \bar{\alpha}_k\}$, 其中 $\alpha_k^{\max} := 1, \bar{\alpha}_k = \min_{i \in \mathcal{I} \setminus S(x_k)} \left\{ \frac{b_i - a_i^T x_k}{a_i^T d(x_k)} \mid a_i^T d(x_k) < 0 \right\}$; 令 $x_{k+1} := x_k + \alpha_k d(x_k)$ 。

步骤 4(更新系数和活动集), 若 $\alpha_k = 1$, 令 $S(x_{k+1}) := S(x_k)$ 。否则, 若 $\alpha_k < 1$, 令 $S(x_{k+1}) := S(x_k) \cup \{i_k\}$, 其中 i_k 满足 $\bar{\alpha}_k = \frac{b_{i_k} - a_{i_k}^T x_k}{a_{i_k}^T d(x_k)}$ 。令 $k := k+1$, 回到步骤 1。

注 1 如果算法在第 k 次迭代时停止, 则引理 2 中的命题 3) 意味着 x_k 是式(1)的弱有效解。

注 2 在步骤 3 中, 下降最好的可行点通过线搜索方法求出。注意到目标函数是凸函数, 那么这一可行点应在可行域的边界上达到。因此, 只要求出满足可行条件的最大步长 α_k 即可。

当 $i \in S(x_k)$ 时, $\forall \alpha_k \geq 0$, 都有 $\langle \nabla g_i(x_k), d(x_k) \rangle = a_i^T d(x_k) = 0$ 和 $a_i^T(x_k + \alpha_k d(x_k)) = a_i^T x_k = b_i$, 此时, $\alpha_k \geq 0$ 不受限制。当 $i \notin S(x_k)$ 时, 即第 i 个约束是严格的不等式约束, 此时要求 α_k 满足 $a_i^T(x_k + \alpha_k d(x_k)) - b_i \geq 0$, 即:

$$\alpha_k a_i^T d(x_k) \geq b_i - a_i^T x_k, i \notin S(x_k)。$$

注意到上式右端非正, 故当 $a_i^T d(x_k) \geq 0$ 时, 上式恒成立, 而当 $a_i^T d(x_k) < 0$ 时, 由上式可解得:

$$\alpha_k \leq \frac{b_i - a_i^T x_k}{a_i^T d(x_k)}。$$

故有:

$$\alpha_k = \bar{\alpha}_k = \min \left\{ \frac{b_i - a_i^T x_k}{a_i^T d(x_k)} \mid a_i^T d(x_k) < 0 \right\}。$$

因此, 为了使步长更加保守, 在算法中设置 $\alpha_k = \min\{\alpha_k^{\max} := 1, \bar{\alpha}_k\}$ 。

3 收敛性分析

在本节中, 对算法 1 进行收敛性分析。约束优化问题的最优性分析常依赖对偶理论的支撑, Farkas 引理^[25]作为连接原始问题与对偶问题的核心工具, 为后续的推导提供了关键理论依据。以下首先给出 Farkas 引理的一个等价描述方式。

引理 3 设 A_1 为 $n \times s$ 矩阵, A_2 为 $n \times t$ 矩阵, b 为 n 维向量, 则下述 2 个系统恰有 1 个解: 1) $A_1^T x = 0, A_2^T x \geq 0, b^T x < 0$; 2) $A_1 u + A_2 v = b, v \geq 0$ 。

下述定理揭示了 $\gamma(x) = 0$ 与 KKT 点的内在联系。

定理 1 设 $x_k \in V$, 则 x_k 是式(1)KKT 点的充要条件是子问题式(5)中 $\gamma(x_k) = 0$ 。

证明 1) 必要性。假设 x_k 是式(1)的 KKT 点, 即存在非零权重向量 $\lambda \geq 0$ 和乘子 μ 使得:

$$\sum_{j \in J} \lambda_j \nabla f_j(x_k) = \sum_{i \in \mathcal{E}} \mu_i a_i + \sum_{i \in \mathcal{I}(x_k)} \mu_i a_i, \mu_i \geq 0, i \in \mathcal{I}(x_k)。 \quad (9)$$

考虑任意满足 $a_i^T d = 0 (i \in S(x_k))$ 的可行方向 d , 将式(9)两边与 d 做内积, 可得:

$$\sum_{j \in J} \lambda_j \nabla f_j(x_k)^T d = \sum_{i \in S(x_k)} \mu_i a_i^T d = 0。$$

由于 $\lambda_j > 0$, 若存在某个 j 使得 $\nabla f_j(\mathbf{x}_k)^T \mathbf{d} < 0$, 则需存在其他 j' , 使得 $\nabla f_{j'}(\mathbf{x}_k)^T \mathbf{d} > 0$, 从而抵消加权和; 因此有 $\max_{j \in J} \nabla f_j(\mathbf{x}_k)^T \mathbf{d} \geq 0$ 。结合式(5)中 $\gamma(\mathbf{x})$ 的定义和引理 2 中的命题 1), 可得 $\gamma(\mathbf{x}_k) = 0$ 。

2) 充分性。假设 $\gamma(\mathbf{x}_k) = 0$, 证明 $\mathbf{x}_k$ 是式(1)的 KKT 点。

由式(5)中 $\gamma(\mathbf{x})$ 的定义, 有:

$$\gamma(\mathbf{x}_k) = \min_{\mathbf{p} \in V} \max_{j \in J} \{ \nabla f_j(\mathbf{x}_k)^T (\mathbf{p} - \mathbf{x}_k) : \mathbf{a}_i^T (\mathbf{p} - \mathbf{x}_k) = 0, i \in S(\mathbf{x}_k) \}。$$

令 $\mathbf{d} = \mathbf{p} - \mathbf{x}_k$, 则上式等价于 $\gamma(\mathbf{x}_k) = \min_{\mathbf{d} \in D} \max_{j \in J} \nabla f_j(\mathbf{x}_k)^T \mathbf{d}$, 其中:

$$D = \{ \mathbf{d} : \mathbf{a}_i^T \mathbf{d} = 0, i \in S(\mathbf{x}_k), \mathbf{x}_k + \mathbf{d} \in V \}。$$

利用最大函数的对偶表示:

$$\max_{j \in J} \nabla f_j(\mathbf{x}_k)^T \mathbf{d} = \max_{\lambda \in \Delta} \sum_{j \in J} \lambda_j \nabla f_j(\mathbf{x}_k)^T \mathbf{d},$$

其中: $\Delta = \{ \lambda \in \mathbf{R}^m : \lambda \geq 0, \sum \lambda_j = 1 \}$, 因此可将 $\gamma(\mathbf{x}_k)$ 改写为:

$$\gamma(\mathbf{x}_k) = \min_{\mathbf{d} \in D} \max_{\lambda \in \Delta} \sum_{j \in J} \lambda_j \nabla f_j(\mathbf{x}_k)^T \mathbf{d}。$$

根据 Minimax 定理, 由于 Δ 和 D 均为紧凸集且目标函数为双线性函数, 可交换最小和最大运算顺序, 于是有:

$$\gamma(\mathbf{x}_k) = \max_{\lambda \in \Delta} \min_{\mathbf{d} \in D} \sum_{j \in J} \lambda_j \nabla f_j(\mathbf{x}_k)^T \mathbf{d}。$$

由 $\gamma(\mathbf{x}_k) = 0$ 可知存在 $\lambda^* \in \Delta$ 使得 $\min_{\mathbf{d} \in D} \sum_{j \in J} \lambda_j^* \nabla f_j(\mathbf{x}_k)^T \mathbf{d} = 0$, 即 $\forall \mathbf{d} \in D$, 有 $\sum_{j \in J} \lambda_j^* \nabla f_j(\mathbf{x}_k)^T \mathbf{d} \geq 0$ 。

考虑系统: $-\sum_{j \in J} \lambda_j^* \nabla f_j(\mathbf{x}_k)^T \mathbf{d} < 0 (j \in J), \mathbf{a}_i^T \mathbf{d} = 0 (i \in S(\mathbf{x}_k)), \mathbf{x}_k + \mathbf{d} \in V$ 。该系统显然无解(否则将与 $\sum_{j \in J} \lambda_j^* \nabla f_j(\mathbf{x}_k)^T \mathbf{d} \geq 0$ 矛盾)。

根据引理 3, 上述系统无解当且仅当存在 $\mu_i (i \in S(\mathbf{x}_k))$ 使得:

$$-\sum_{j \in J} \lambda_j^* \nabla f_j(\mathbf{x}_k) = \sum_{i \in S(\mathbf{x}_k)} \mu_i \mathbf{a}_i,$$

其中: 对于 $i \in \mathcal{I}(\mathbf{x}_k)$, 有 $\mu_i \geq 0$ 。对于 $i \in \mathcal{I} \setminus \mathcal{I}(\mathbf{x}_k)$ (非活跃的不等式约束), 取 $\mu_i = 0$, 则得到完整的 KKT 条件。因此, $\mathbf{x}_k$ 是式(1)的 KKT 点。证毕

在此基础上, 以下定理将进一步证明算法 1 产生的迭代序列必然存在聚点, 且聚点同时满足 KKT 条件与弱有效解的要求。

定理 2 针对式(1), 假设算法 1 生成的序列为 $\{\mathbf{x}_k\}$ 。如果目标函数 $f(\mathbf{x})$ 是 $\mathbf{R}^m$ -凸(严格 $\mathbf{R}^m$ -凸)且二阶连续可微的, 水平集 $\{\mathbf{x} \in V | f_j(\mathbf{x}) \leq f_j(\mathbf{x}_1)\}$ 是有界集, 约束函数 $g_i(\mathbf{x}) (i \in \{1, 2, \dots, r\})$ 是线性的, 那么, 序列 $\{\mathbf{x}_k\}$ 至少存在 1 个聚点 $\mathbf{x}^*$, 且任何 $\{\mathbf{x}_k\}$ 的聚点都是满足 KKT 条件的弱有效解(或有效解)。

证明 由算法设计可知, $\mathbf{d}(\mathbf{x}_k) = \mathbf{p}(\mathbf{x}_k) - \mathbf{x}_k$ 是改进方向(定义 3), 结合精确线搜索的步长选择, 则 $\forall j \in J$ 均有 $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$, 因此目标函数序列 $\{f(\mathbf{x}_k)\}$ 严格单调递减。

又因水平集 $\{\mathbf{x} \in V | f_j(\mathbf{x}) \leq f_j(\mathbf{x}_1)\}$ 有界, 且 $\{\mathbf{x}_k\}$ 始终包含于该水平集, 故 $\{\mathbf{x}_k\}$ 是有界序列。根据 Bolzano-Weierstrass 定理(有界序列必有聚点), $\{\mathbf{x}_k\}$ 至少存在 1 个聚点 $\mathbf{x}^*$, 即存在子序列 $\{\mathbf{x}_{k_t}\} \subseteq \{\mathbf{x}_k\}$, 使得 $\lim_{t \rightarrow \infty} \mathbf{x}_{k_t} = \mathbf{x}^*$ 。

由引理 2 命题 1) 可知, $\forall \mathbf{x} \in V, \gamma(\mathbf{x}) \leq 0$, 因此 $\{\gamma(\mathbf{x}_k)\}$ 是非递增且有上界的序列。根据单调有界定理, $\{\gamma(\mathbf{x}_k)\}$ 存在极限 $\gamma_0 = \lim_{k \rightarrow \infty} \gamma(\mathbf{x}_k) \leq 0$ 。若 $\gamma_0 < 0$, 则存在 $\delta > 0$, 使得对所有充分大的 k , 有 $\gamma(\mathbf{x}_k) \leq -\delta$ 。由 $\gamma(\mathbf{x}_k)$ 的定义可知, 子问题式(4)始终存在非零改进方向 $\mathbf{d}(\mathbf{x}_k)$, 使得 $\max_{j \in J} \nabla f_j(\mathbf{x}_k)^T \mathbf{d}(\mathbf{x}_k) = \gamma(\mathbf{x}_k) \leq -\delta < 0$, 即 $\mathbf{d}(\mathbf{x}_k)$ 是严格改进方向。结合精确线搜索的步长 α_k , 目标函数序列 $\{f(\mathbf{x}_k)\}$ 会持续严格下降, 与 $\{f(\mathbf{x}_k)\}$ 单调递减且有下界(因可行域 V 有界, $f(\mathbf{x})$ 连续故有界)矛盾。因此 $\gamma_0 < 0$ 不成立, 故有 $\gamma_0 = 0$, 也即 $\lim_{k \rightarrow \infty} \gamma(\mathbf{x}_k) = 0$ 。

接下来证明聚点 $\mathbf{x}^*$ 是弱有效解(若 $f(\mathbf{x})$ 是严格 $\mathbf{R}^m$ -凸则为有效解)。

用反证法。假设 $\mathbf{x}^*$ 不是弱有效解。根据定义 1, 存在不同于点 $\mathbf{x}^*$ 的点 $\mathbf{y} \in V$, 使得 $\forall j \in J$ 有 $f_j(\mathbf{y}) \leq f_j(\mathbf{x}^*)$, 且至少存在 1 个 $h \in J$ 使得 $f_h(\mathbf{y}) < f_h(\mathbf{x}^*)$ 。由于 $\lim_{t \rightarrow \infty} \mathbf{x}_{k_t} = \mathbf{x}^*$ 且 $f(\mathbf{x})$ 连续, 故 $\lim_{t \rightarrow \infty} f_h(\mathbf{x}_{k_t}) = f_h(\mathbf{x}^*)$, 结合 $f_h(\mathbf{y}) < f_h(\mathbf{x}^*)$ 可知, 存在充分大的 t 使得:

$$f_h(\mathbf{y}) < f_h(\mathbf{x}_{k_t}). \quad (10)$$

令 $\mathbf{d} = \mathbf{y} - \mathbf{x}_{k_t}$, 因为 V 是凸集(线性约束界定的可行域必为凸集), 且 $\forall \alpha \in (0, 1]$, 有:

$$\mathbf{x}_{k_t} + \alpha \mathbf{d} = (1 - \alpha)\mathbf{x}_{k_t} + \alpha \mathbf{y} \in V,$$

故 $\mathbf{d}$ 是可行方向。

又因 $f_h(\mathbf{x})$ 可微, 由式(10)及可微函数的单调性判定, 有:

$$\nabla f_h(\mathbf{x}_{k_t})^T \mathbf{d} = \lim_{\alpha \rightarrow 0^+} \frac{f_h(\mathbf{x}_{k_t} + \alpha \mathbf{d}) - f_h(\mathbf{x}_{k_t})}{\alpha} < 0,$$

这表明 $\mathbf{d}$ 是 $\mathbf{x}_{k_t}$ 处的改进方向, 然而根据算法设计, $\mathbf{d}(\mathbf{x}_{k_t})$ 是子问题式(4)的最优改进方向, 即目标函数沿着 $\mathbf{d}(\mathbf{x}_{k_t})$ 的下降程度优于所有其他改进方向, 因此有:

$$f_h(\mathbf{x}_{k_t+1}) \leq f_h(\mathbf{x}_{k_t} + \alpha \mathbf{d}), \forall \alpha \in (0, \alpha_{k_t}].$$

但式(10)表明, 存在 $\mathbf{y} \in V$ 使得 $f_h(\mathbf{y}) < f_h(\mathbf{x}_{k_t})$, 即沿方向 $\mathbf{d}$ 可获得更大幅度的下降, 与 $\mathbf{d}(\mathbf{x}_{k_t})$ 的最优性矛盾。因此假设不成立, $\mathbf{x}^*$ 是式(1)的弱有效解。

由引理2中的命题2), 并结合上述结论 $\lim_{k \rightarrow \infty} \gamma(\mathbf{x}_k) = 0$ 及聚点的定义 $\lim_{t \rightarrow \infty} \mathbf{x}_{k_t} = \mathbf{x}^*$, 有:

$$\gamma(\mathbf{x}^*) = \gamma(\lim_{t \rightarrow \infty} \mathbf{x}_{k_t}) = \lim_{t \rightarrow \infty} \gamma(\mathbf{x}_{k_t}) = 0.$$

此时, 由引理2中的命题3)也再次印证了 $\mathbf{x}^*$ 是式(1)的弱有效解。进一步地, 根据定理1, $\mathbf{x}^*$ 满足式(1)的 KKT 条件, 是式(1)的 KKT 点。

综上所述, 序列 $\{\mathbf{x}_k\}$ 的任一聚点 $\mathbf{x}^*$ 均为式(1)满足 KKT 条件的弱有效解(或有效解)。

证毕

4 数值实验

本节应用算法1来求解线性约束多目标规划问题, 并验证这一算法的性能。所有的数值实验均在配备 MATLAB R2020b 的笔记本电脑(2.10 GHz CPU、16.0 GB RAM)上完成。

4.1 数值实验1

给出如下简单示例:

$$\begin{aligned} \min_{\mathbf{x}} \quad & (x_1^2 + 2x_2^2, (x_1 - 3)^2 + (x_2 - 2)^2); \\ \text{s. t.} \quad & x_1 + x_2 = 3; \\ & 1 \leq x_1 \leq 3; \\ & x_2 \geq 1. \end{aligned}$$

这个问题包含1个等式约束与3个不等式约束, 算法的收敛精度设置为 10^{-9} , 最大迭代次数为20, 给定初始点 $\mathbf{x}_0 = [1.2, 1.8]$, 求解过程如图1~3所示。

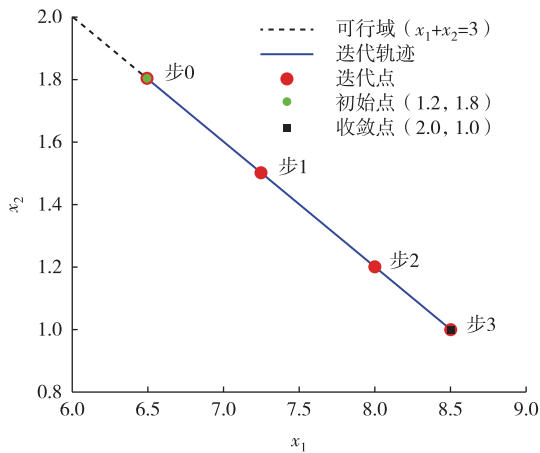


图1 决策空间中的迭代轨迹

Fig. 1 Iterative trajectories in decision space

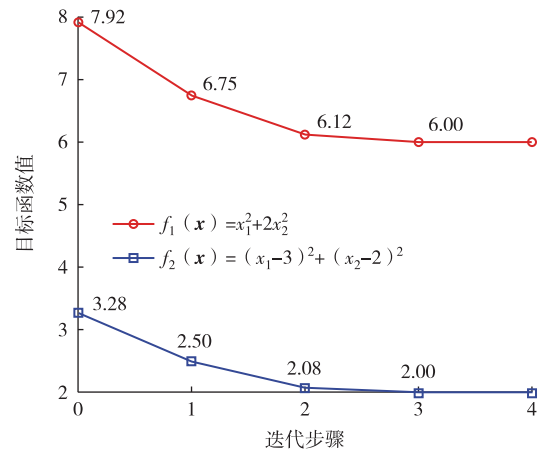


图2 目标函数值的变化

Fig. 2 Variation in objective function value

由图 1 可见,从初始点出发,迭代轨迹逐步沿可行域移动,最终收敛到 Pareto 最优点 $[2.0,1.0]$ 。由图 2 可以看出,迭代过程中 2 个目标被同步改进,直到收敛至无进一步优化空间。由图 3 可以看出,迭代轨迹(绿色实线)从初始点逐步逼近 Pareto 前沿,最终收敛点 $[6.00,2.00]$ 正好落在 Pareto 前沿上。

4.2 数值实验 2

通过对 ZDT2(前沿非凸)和 DTLZ1(前沿凸)测试问题额外添加合适的等式约束(表 1)将本文提出的多目标规划的有效集算法与多目标优化的加权求和算法进行了对比实验。设置目标函数的数量分别为 2(ZDT2 问题)、3(DTLZ1 问题)、4(DTLZ1 问题)和 5(DTLZ1 问题)个,决策变量($\mathbf{x}$)的维数为 30,收敛精度为 10^{-6} ,最大迭代次数为 1 000,加权求和方法求解时随机生成权重。对比了 2 种方法的 CPU 耗时、非支配解的数量、超体积指标(归一化)、约束违反度 4 个参数,如图 4~7 所示。

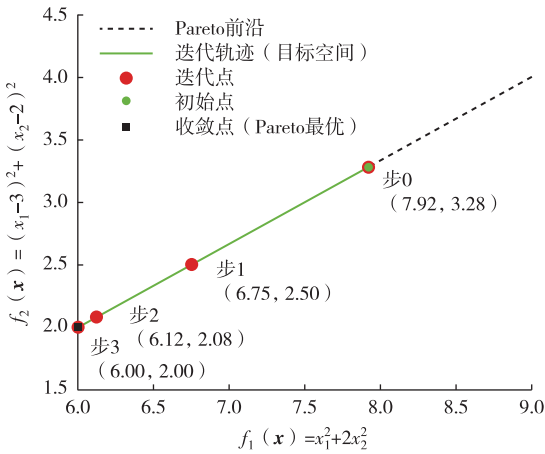


图 3 Pareto 前沿上的迭代轨迹
Fig. 3 Iterative trajectory on the Pareto front

表 1 多目标测试问题
Tab. 1 Multi-objective test problems

问题	目标个数	目标函数	约束条件	文献来源	凸性
ZDT2	2	$f_1(\mathbf{x}) = x_1,$ $f_2(\mathbf{x}) = g(\mathbf{x}) \cdot \left[1 - \left(\frac{x_1}{g(\mathbf{x})} \right)^2 \right],$ $g(\mathbf{x}) = 1 + 9 \cdot \frac{\sum_{i=2}^{30} x_i}{29}$	原始约束: $x_i \in [0,1](i=1,2,\cdots,30)$ 额外添加: $2x_1 + \sum_{i=2}^{10} x_i = 4$	[26]	非凸
DTLZ1	3	$f_1(\mathbf{x}) = \frac{1}{2}x_1x_2(1+g(\mathbf{x})),$ $f_2(\mathbf{x}) = \frac{1}{2}x_1(1-x_2)(1+g(\mathbf{x})),$ $f_3(\mathbf{x}) = \frac{1}{2}(1-x_1)(1+g(\mathbf{x})),$ $g(\mathbf{x}) = 100 \cdot \left[28 + \sum_{i=3}^{30} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))) \right]$	原始约束: $x_i \in [0,1](i=1,2,\cdots,30)$ 额外添加: $x_1 + x_2 + x_3 = 1.2$	[27]	凸
DTLZ1	4	$f_1(\mathbf{x}) = \frac{1}{2}x_1x_2x_3(1+g(\mathbf{x})),$ $f_2(\mathbf{x}) = \frac{1}{2}x_1x_2(1-x_3)(1+g(\mathbf{x})),$ $f_3(\mathbf{x}) = \frac{1}{2}x_1(1-x_2)(1+g(\mathbf{x})),$ $f_4(\mathbf{x}) = \frac{1}{2}(1-x_1)(1+g(\mathbf{x})),$ $g(\mathbf{x}) = 100 \cdot \left[27 + \sum_{i=4}^{30} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))) \right]$	原始约束: $x_i \in [0,1](i=1,2,\cdots,30)$ 额外添加: $x_1 + x_2 + x_3 + x_4 = 1.0$	[27]	凸

续表 1

问题	目标 个数	目标函数	约束条件	文献 来源	凸性
DTLZ1	5	$f_1(\mathbf{x}) = \frac{1}{2}x_1x_2x_3x_4(1+g(\mathbf{x})),$	原始约束: $x_i \in [0,1](i=1,2,\cdots,30)$ 额外添加: $x_1+x_2+x_3+x_4+x_5 = 1.0$	[27]	凸
		$f_2(\mathbf{x}) = \frac{1}{2}x_1x_2x_3(1-x_4)(1+g(\mathbf{x})),$			
		$f_3(\mathbf{x}) = \frac{1}{2}x_1x_2(1-x_3)(1+g(\mathbf{x})),$			
		$f_4(\mathbf{x}) = \frac{1}{2}x_1(1-x_2)(1+g(\mathbf{x})),$			
		$f_5(\mathbf{x}) = \frac{1}{2}(1-x_1)(1+g(\mathbf{x})),$			
		$g(\mathbf{x}) = 100 \cdot \left[26 + \sum_{i=5}^{30} ((x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5))) \right]$			

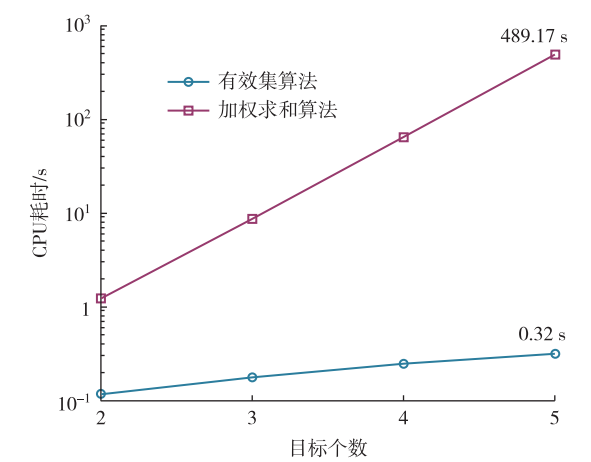


图 4 CPU 耗时对比

Fig. 4 Comparison of CPU time consuming

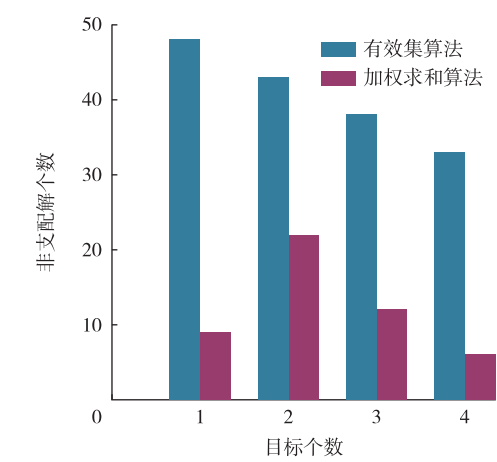


图 5 生成非支配解数量对比

Fig. 5 Comparison of the number of generated non-dominated solutions

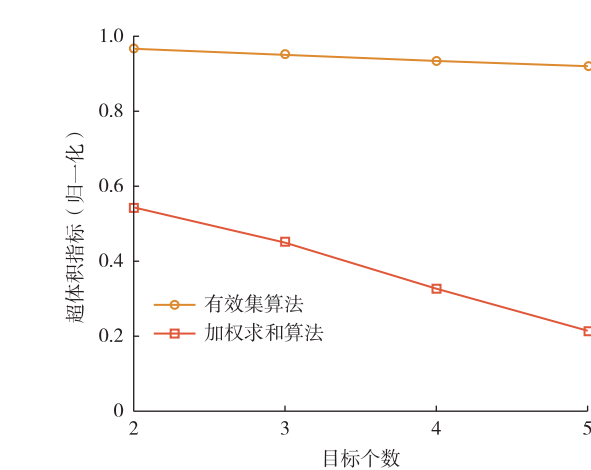


图 6 超体积指标(归一化)对比

Fig. 6 Comparison of hypervolume index (normalized)

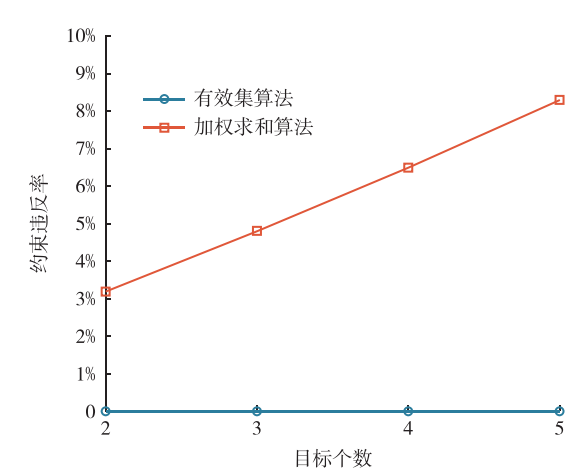


图 7 约束违反率对比

Fig. 7 Comparison of constraint violation rate

图 4 显示:有效集算法的 CPU 耗时比加权求和算法少;且当目标函数的维度增大时,有效集算法的 CPU 耗
时增长速率也比加权求和算法低得多。同时,通过有效集算法生成的非支配解的数量更多(图 5),且该算法的超
体积指标更大(图 6);因此相比于加权求和算法,经有效集算法所得解的质量更好。从图 7 则可以看出:有效集

算法几乎不产生违反约束的解;而加权求和算法在求解问题时依赖 `fmincon` 求解器,优先降低目标函数值,其次才考虑满足约束,所以这一算法在求解问题时必然会存在约束违反率。

综合上述结果可知,本文提出的多目标规划的有效集算法相较于多目标优化的加权求和算法在求解效率、解的覆盖性、解的质量、约束安全性等 4 个维度上都体现了优越性;该算法不仅在求解非凸前沿(ZDT2)和凸前沿(DTLZ1)问题时均能稳定工作,而且它的优势随着目标函数个数增加而更加明显,还解决了加权求和方法在多目标场景下效率低、解覆盖不足、约束违反等问题,从而为线性约束多目标规划提供了更可靠的求解方案。

5 结束语

本文提出了求解带有线性约束多目标优化问题的有效集方法。通过一个易于求解的线性规划问题生成可行方向,并使用精确线搜索生成步长。基于假设条件,证明了算法生成序列的聚点都是满足 KKT 条件的弱有效解(或有效解)。最后,将本文提出的算法与多目标优化的加权求和方法进行了对比实验,结果表明本文提出的算法性能更高。本文的研究思路在今后可扩展至带有更加复杂约束的场景,并进行可行性分析。

参考文献:

- [1] Shen Z M, Daskin M S. Trade-offs between customer service and cost in integrated supply chain design[J]. *Manufacturing & Service Operations Management*, 2005, 7 (3): 188-207.
- [2] Eschenauer H, Koski J, Osyczka A. Multicriteria design optimization: procedures and applications[M]. Heidelberg: Springer, 1990.
- [3] Carrizosa E, Frenk J B G. Dominating sets for convex functions with some applications[J]. *Journal of Optimization Theory and Applications*, 1998, 96 (2): 281-295.
- [4] Carrizosa E, Conde E, Muñoz-Marquez M, et al. Planar point-objective location problems with nonconvex constraints: a geometrical construction[J]. *Journal of Global Optimization*, 1995, 6 (1): 77-86.
- [5] Coelho F, Costa M, Verleysen M, et al. LASSO multi-objective learning algorithm for feature selection[J]. *Soft Computing*, 2020, 24 (17): 13209-13217.
- [6] 刘朗亭, 刘学文, 李文霞. 回流焊工艺中炉温分布的多目标优化[J]. *重庆师范大学学报(自然科学版)*, 2025, 42 (2): 88-99.
Liu Langting, Liu Xuewen, Li Wenxia. Multi-objective optimization for furnace temperature profile in reflow soldering process[J]. *Journal of Chongqing Normal University (Natural Science)*, 2025, 42 (2): 88-99.
- [7] 葛建军, 陈鑫, 温玉玺. 异构并行车间多目标排产优化粒子群算法研究[J]. *重庆师范大学学报(自然科学版)*, 2025, 42 (4): 24-33.
Ge Jianjun, Chen Xin, Wen Yuxi. Multi-objective scheduling for heterogeneous parallel workshops with a particle swarm optimization algorithm[J]. *Journal of Chongqing Normal University (Natural Science)*, 2025, 42 (4): 24-33.
- [8] 李翰章, 马瞰. 基于 NSGA-II 的物流末端人员配置的多目标决策[J]. *重庆师范大学学报(自然科学版)*, 2024, 41 (2): 100-108.
Li Hanzhang, Ma Jian. Multi-objective decision in making of logistics terminal personnel configuration based on NSGA-II[J]. *Journal of Chongqing Normal University (Natural Science)*, 2024, 41 (2): 100-108.
- [9] 譙露, 舒勤, 赵克全. 油库选址问题的多目标优化方法[J]. *重庆师范大学学报(自然科学版)*, 2023, 40 (1): 82-87.
Qiao Lu, Shu Qin, Zhao Kequan. Multi-objective optimization method for oil depot location problem[J]. *Journal of Chongqing Normal University (Natural Science)*, 2023, 40 (1): 82-87.
- [10] Geoffrion A M. Proper efficiency and the theory of vector maximization[J]. *Journal of Mathematical Analysis and Applications*, 1968, 22 (3): 618-630.
- [11] Fliege J, Svaiter B F. Steepest descent methods for multicriteria optimization[J]. *Mathematical Methods of Operations Research*, 2000, 51 (3): 479-494.
- [12] Bello Cruz J Y. A subgradient method for vector optimization problems[J]. *SIAM Journal on Optimization*, 2013, 23 (4): 2169-2182.
- [13] Bello Cruz J Y, Allende G B. A steepest descent-like method for variable order vector optimization problems[J]. *Journal of Optimization Theory and Applications*, 2014, 162 (2): 371-391.
- [14] Fliege J, Vaz A I F, Vicente L N. Complexity of gradient descent for multiobjective optimization[J]. *Optimization*

- Methods and Software, 2019, 34 (5): 949-959.
- [15] Fukuda E H, Drummond L M G. On the convergence of the projected gradient method for vector optimization[J]. Optimization, 2011, 60 (8/9): 1009-1021.
- [16] Drummond L M G, Iusem A N. A projected gradient method for vector optimization problems[J]. Computational Optimization and Applications, 2004, 28 (1): 5-29.
- [17] Drummond L M G, Svaiter B F. A steepest descent method for vector optimization[J]. Journal of Computational and Applied Mathematics, 2005, 175 (2): 395-414.
- [18] Assunção P B, Ferreira O P, Prudente L F. Conditional gradient method for multiobjective optimization [J]. Computational Optimization and Applications, 2021, 78 (3): 741-768.
- [19] Assunção P B, Ferreira O P, Prudente L F. A generalized conditional gradient method for multiobjective composite optimization problems[PP/OL]. V1. (2023-02-24) [2025-12-18]. <https://arXiv.org/abs/2302.12912>.
- [20] Chen W, Yang X M, Zhao Y. Conditional gradient method for vector optimization[J]. Computational Optimization and Applications, 2023, 85 (3): 857-896.
- [21] 杨春蓉, 谭豫琳, 赵克全. 多目标优化问题的非单调对角最速下降算法[J]. 重庆师范大学学报(自然科学版), 2023, 40 (1): 114-122.
- Yang Chunrong, Tan Yuling, Zhao Kequan. Nonmonotonic diagonal steepest descent algorithm for multi-objective optimization[J]. Journal of Chongqing Normal University (Natural Science), 2023, 40 (1): 114-122.
- [22] 刘雪, 周犁文. 解决向量优化问题的一种非单调投影梯度算法[J]. 应用数学进展, 2023, 12 (5): 2327-2339.
- Liu Xue, Zhou Liwen. A nonmonotone projected gradient algorithm for solving vector optimization problems [J]. Advances in Applied Mathematics, 2023, 12 (5): 2327-2339.
- [23] Halffmann P, Schäfer L E, Dächert K, et al. Exact algorithms for multiobjective linear optimization problems with integer variables: a state of the art survey[J]. Journal of Multi-Criteria Decision Analysis, 2022, 29 (5/6): 341-363.
- [24] Ehrgott M. Multicriteria optimization[M]. Berlin, Heidelberg: Springer, 2000.
- [25] Farkas J. Theorie der einfachen ungleichungen[J]. Journal für Die Reine und Angewandte Mathematik (Crelles Journal), 1902, 1902 (124): 1.
- [26] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II [J]. IEEE Transactions on Evolutionary Computation, 2002, 6 (2): 182-197.
- [27] Li H, Deb K, Zhang Q F, et al. Comparison between MOEA/D and NSGA-III on a set of novel many and multi-objective benchmark problems with challenging difficulties[J]. Swarm and Evolutionary Computation, 2019, 46: 104-117.

Operations Research and Cybernetics

The Effective Set Method for Linearly Constrained Multi-Objective Programming

MA Yiting, HUANG Yingquan, LI Gaoxi

(School of Mathematics and Statistics, Chongqing Technology and Business University, Chongqing 400067, China)

Abstract: The effective set method plays a crucial role in practical applications due to its inherent safety and reliability. This paper proposes an effective set method for solving multi-objective programming problems with linear constraints. This method generates feasible directions through a special linear programming problem that is easy to solve and generates step sizes through an exact line search problem with an explicit solution. Under relatively weak conditions, this paper proves that the sequence generated by the above algorithm converges to a Karush-Kuhn-Tucker stable point of the original problem, and under the assumption of convexity, it is proved that the accumulation point of this algorithm is a weakly efficient solution to the original problem. Finally, the effectiveness and superiority of the algorithm in solving multi-objective optimization problems are verified through numerical experiments.

Keywords: multi-objective programming; effective set method; linear constraint

(责任编辑 方 兴)