arXiv:2304.02369v1 [math.OC] 5 Apr 2023

基数约束多目标优化：新颖的最优性条件与算法

Matteo Lapucci
全局优化实验室（GOL）信息工程系
佛罗伦萨大学意大利佛罗伦萨 Santa Marta 路 3 号，邮编 50139
matteo.lapucci@unifi.it

Pierluigi Mansueto
全局优化实验室（GOL）信息工程系
佛罗伦萨大学意大利佛罗伦萨 Santa Marta 路 3 号，邮编 50139
pierluigi.mansueto@unifi.it

摘要

本文研究变量向量受稀疏性约束的多目标优化问题。对于这类问题，受稀疏单目标优化中同名必要最优性条件的启发，我们定义了 L-驻点的概念，并分析其与其他现有条件及帕累托最优性概念之间的关系。随后，我们提出两种新颖的算法方法：第一种是旨在寻找单个 L-驻点解的迭代硬阈值方法，第二种则是用于构造整个帕累托前沿近似的两阶段算法。这两种方法均具有收敛性理论性质，可收敛至满足帕累托最优性必要条件的点。此外，我们还报告了数值结果，证实所提出方法的实际有效性。

关键词 多目标优化 · 基数约束 · 最优性条件 · 数值方法 · 帕累托前沿

数学主题分类（***2020***） 90C26 · 90C29 · 90C46

1 引言

多目标优化（MOO）是一种数学工具，在过去25年里受到研究界的广泛关注。事实上，多目标优化问题在各种应用领域都具有重要意义，因为这些领域需要同时考虑彼此冲突的目标（例如，参见 [11, 13, 25, 43, 45]）。

与单目标优化（SOO）相比，多目标优化问题还面临一项额外的复杂性：一般而言，不存在能够同时使所有目标达到最小值的解。因此，需要引入基于帕累托理论的最优性条件，并采用新的、非平凡的优化方案。其中，标量化[24, 42] 方法和进化 [20, 36] 方法都非常流行。然而，尽管这两类算法各有吸引人的特点，也存在重要的局限。具体而言，在选择权重时，标量化方法高度依赖问题的领域和结构；此外，即使在严格的正则性假设下，权重选择不当也可能导致标量化问题无界 [26, 第 7]节。另一方面，进化方法虽然擅长处理极其困难的任务，但其收敛性通常难以证明 [26]。

另一个日益受到关注的多目标优化算法分支是下降算法 [15,22, 26, 27, 34]。这些方法基本上扩展了经典的单目标优化下降方法。对于其中大多数方法，理论收敛性质已得到证明。此类中最早开发的算法只能生成单个帕累托驻点解；为了生成整个帕累托前沿的近似，它们以多起点方式运行多次。近年来，其中一些算法经过扩展，克服了这一局限。这些新方法（例如，参见 [16–19, 33]）能够处理点集序列，

从而以高效且有效的方式生成帕累托前沿近似。受单目标优化相关研究的启发， [38,40],面向前沿的下降方法也与多目标优化遗传算法相结合（例如，参见 [35]）。
优化领域近期研究的另一个主题，是要求解具有较少非零分量的问题 [46]。通常通过直接对变量向量施加基数约束来诱导解的稀疏性。然而，对 ℓ_0 伪范数设置上界会使问题部分具有组合性质，因此属于 $\mathcal{NP}$-困难问题 [8,41]。因此，人们提出了许多旨在近似求解此问题的方法。关于这些方法的全面综述，请参阅 [46] 。不过，文献中也有能够精确处理 ℓ_0 伪范数的算法。尤其是迭代硬阈值 (IHT)算法 [1], 、罚函数分解 (PD)方法 [39] 以及稀疏邻域搜索(SNS)方法 [32], 均可用于最一般的情形，即使不作凸性假设也适用。采用这些方法时，通过连续的局部优化步骤处理问题，并保证收敛到满足必要最优性条件的解。

尽管这两项挑战已分别得到充分研究，但稀疏性与多目标的结合几乎尚未得到探索。基数约束多目标优化的理论基础最近在 [31], 中奠定，该文还针对稀疏多目标优化任务提出了一种罚函数分解方法，并分析了其收敛性。此外， [28]中还介绍了一项理论研究，将 [10] 中的工作拓展到了多目标优化情形。开发高性能方法来处理这类问题，有益于许多现实应用。例如，在机器学习中，促使分类/回归模型具有稀疏性有多种原因（例如可解释性 [4], 稳健性 [47], 轻量性 [12]）。此外，文献中也有一些方法从基于帕累托的多目标视角处理学习任务：拟合质量和模型复杂度就是两个相互冲突的目标，在这类目标间取得良好权衡可能很有价值 [29]。

本文继续 [31], 中开始的理论分析，为具有基数约束的多目标优化问题提出新的最优性条件。具体而言，我们定义了多目标优化中的 L-驻点概念，该概念直接受到稀疏单目标优化任务中同名条件的启发 [1]。随后，我们提出两种用于求解此类问题的新算法；第一种是将IHT方法扩展到多目标情形，旨在求得一个 L-驻点解；我们将此方法称为多目标迭代硬阈值法 (MOIHT)，并证明它确实保证收敛到满足新提出的必要最优性条件的点。另一方面，第二种算法采用两阶段方法，最终目标是近似整个帕累托前沿。我们将此方法称为稀疏前沿最速下降法 (SFSD)；我们也对其进行了理论分析，并通过数值实验表明，它能够很好地重构解集。

本文其余部分安排如下。第 2 节首先回顾主要的多目标优化概念，以及一些关于基数约束多目标优化问题的概念。第 3 节针对所考虑的问题类别定义 L-平稳性，并阐述其与 Pareto 理论及其他现有最优性条件之间的理论关系。第 4 节介绍 MOIHT 算法及其收敛性分析；此外，本节还提出 SFSD 方法。第 5 节报告一些计算实验的结果，展示我们新方法的优越性。最后，第 6 节给出总结性结论。

2 预备知识

本文考虑如下形式的问题:

$$\min_{x\in\mathbb{R}^n} \; F(x) = (f_1(x), \ldots, f_m(x))^\top \text{ s.t. } \|x\|_0 \leq s, \tag{1}$$

其中， $F\colon \mathbb{R}^n \to \mathbb{R}^m$ 是一个连续可微的向量值函数， $\|\cdot\|_0$ 表示 ℓ_0 伪范数，即向量的非零分量数，且 $s \in \mathbb{N}$ 满足 $1 \leq s < n$。下文中，我们用$\|\cdot\|$ 表示 $\mathbb{R}^n$中的欧几里得范数。我们记 $\Omega = \{x \in \mathbb{R}^n \mid \|x\|_0 \leq s\}$ 为由 ℓ_0 伪范数的上界所诱导的闭且非空集合。

为处理多目标情形，我们需要在 $\mathbb{R}^m$ 中定义一个偏序：给定 $u,\ v \in \mathbb{R}^m$，当且仅当对所有 $j \in \{1,\ldots,m\}$, $u_j < v_j$时，我们称$u < v$；对于算子 $\leq,\ >,\ \geq$，也可以作出类似定义。此外，给定目标函数 F，若 x支配 y（相对于 F），即 $F(x) \leq F(y)$ 且 $F(x) \neq F(y)$，则我们称之为支配关系，并记作 $F(x) \lneqq F(y)$。

在多目标优化中，同时使所有目标达到最小值的解通常并不存在。在这种情况下，最优性概念基于帕累托理论。

定义 1. 如果不存在任何 $y \in \Omega$ 使得 $F(y) \lneqq F(\bar{x})$，则点 $\bar{x} \in \Omega$ 是问题 (1) 的帕累托最优点。如果存在一个邻域 $\mathcal{N}(\bar{x})$, 使得该性质在 $\Omega \cap \mathcal{N}(\bar{x})$ 中成立，则 $\bar{x}$是局部帕累托最优点。

帕累托最优性是一项强性质，因此在实践中往往难以实现。于是，可以引入一种较弱但更易实现的条件。

定义 2。 若不存在任何 $y \in \Omega$ 使得$F(y) < F(\bar{x})$，则称点 $\bar{x} \in \Omega$ 是问题 **(1)** 的弱帕累托最优点。若存在一个邻域 $\mathcal{N}(\bar{x})$，使得该性质在 $\Omega \cap \mathcal{N}(\bar{x})$ 中成立，则称 $\bar{x}$是局部弱帕累托最优点。

我们将.....的集合称为 帕累托最优解，称为帕累托集；后者在 F下的像称为帕累托前沿 .

补充记号：给定一个指标集 $S \subseteq \{1,\dots,n\}$，用 $|S|$表示 S的基数，而我们用 $S = \{1,\dots,n\} \setminus S$表示其补集；若 $|S| = 1$，则称 S为单元素集合。令 $x \in \mathbb{R}^n$，用 x_S 表示由 S诱导的 x 的子向量，即由分量 x_i组成的向量，其中 $i \in S$；$S_1(x) = \{i \in \{1,\dots,n\} \mid x_i \neq 0\}$表示 x的支撑集，即对应于 x非零分量的指标集合； $S_0(x) = \{1,\dots,n\} \setminus S_1(x)$ 表示其 $S_1(x)$补集。此外，根据 [2],，若指标集 J满足 $S_1(x) \subseteq J$且 $|J| = s$，则称其为 x的超支持集； x处所有超支持集的集合记为 $\mathcal{J}(x)$，当且仅当 $\|x\|_0 = s$时它是单元素集合。最后，我们用 $\mathbf{1}_N$和 $\mathbf{0}_N$分别表示全 **1** 向量和全 **0** 向量，它们是 $N \in \mathbb{N}^+$维向量。

2.1 多目标优化中的近端算子

关于多目标设置中近端方法的详尽分析可以在文献中找到（参见，例如， [9,44]）。就本工作的范围而言，我们参考 [44], 中进行的讨论，其中考虑的多目标优化问题形式为

$$\min_{x\in\mathbb{R}^n} \left(f_1(x) + g_1(x), \dots, f_m(x) + g_m(x)\right)^\top. \tag{2}$$

对于所有 $j \in \{1,\dots,m\}$， f_j被假定为连续可微的，而 g_j是下半连续的、真凸的，但不一定光滑。

设 $x_k \in \mathbb{R}^n$。在 x_k 处可以根据 $x_{k+1} = x_k + t_k d_k$执行近端步，其中 t_k 是合适的步长，下降方向 d_k 通过求解获得

$$\min_{d\in\mathbb{R}^n} \max_{j\in\{1,\dots,m\}} \left\{\nabla f_j(x_k)^\top d + g_j(x_k + d) - g_j(x_k)\right\} + \frac{L}{2}\|d\|^2, \tag{3}$$

其中 $L > 0$。问题(2)的最优解使得 $\mathbf{0}_n$ 是(3)的解。
有趣的是，与标量情况类似，问题(3)可以看作是定义搜索方向的著名方案的推广：

- 如果对所有 $j \in \{1,\dots,m\}$， $g_j = 0$，则 **(3)** 等同于寻找最速共同下降方向的问题 [27];
- 如果对所有 $j \in \{1,\dots,m\}$， g_j 是凸集 C 的示性函数，则 **(3)** 等价于投影梯度方向问题 [22]。

在下一节中，我们将说明近端算子可用于处理非凸集 Ω，这与标量优化中的相关工作一致 [1]。

3 最优性条件

在目标函数 F 满足可微性假设的条件下，文献 [31] 证明了 Pareto 驻点条件是（局部）最优性的必要条件。下文给出一个略有不同的定义及其性质，并使其适用于问题 **(1)**。

定义 3([31, 定义 3.2]**).** </style}点 $\bar{x} \in \Omega$ 是问题 **(1)** 的 Pareto 驻点，当且仅当

$$\theta(\bar{x}) = \min_{d\in\mathcal{D}(\bar{x})} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top d + \frac{1}{2}\|d\|^2 = 0, \tag{4}$$

其中 $\mathcal{D}(\bar{x}) = \{v \in \mathbb{R}^n \mid \exists \bar{t} > 0 \colon \bar{x} + tv \in \Omega\ \forall t \in [0,\bar{t}]\} = \{v \in \mathbb{R}^n \mid \|v_{S_0(\bar{x})}\|_0 \leq s - \|\bar{x}\|_0\}$ 是 $\bar{x}$ 处的可行方向集。

我们用 $v(\bar{x})$ 表示问题 **(4)** 在 $\bar{x}$ 处的最优解集合。

引理 1 ([31, 命题 3.3]). 设 $\bar{x} \in \Omega$ 是问题 (1) 的局部弱帕累托最优解。则， $\bar{x}$ 是问题 (1) 的帕累托驻点。

第二个引理指出，假设目标函数是凸的，则驻点条件也是局部弱帕累托最优性的充分条件。

引理 2. 假设 F 是分量凸的。设 $\bar{x} \in \Omega$ 是问题 (1)的一个帕累托驻点。那么， $\bar{x}$ 是问题 (1)的局部弱帕累托最优解。

证明。 见附录 A。 □

此外， [31], 针对标量基数约束问题的 Lu-Zhang 一阶最优性条件 [39] 已扩展至多目标优化问题。

定义 4 ([31, 定义 3.6]). 点 $\bar{x} \in \Omega$ 满足问题 (1) 的多目标 *Lu-Zhang* 一阶最优性条件（MOLZ 条件），当且仅当存在一个超支持集 $J \in \mathcal{J}(\bar{x})$，使得

$$\theta_J(\bar{x}) = \min_{d \in \mathbb{R}^n} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top d + \frac{1}{2}\|d\|^2 = 0 \quad \text{s.t.} \quad d_{\bar{J}} = \mathbf{0}_{|\bar{J}|} \tag{5}$$

由于问题 (5) 的目标函数是强凸的，且可行集是凸集，因此其在$\bar{x}$ 处存在唯一的最优解，我们将其记为 $d_J(\bar{x})$。

引理 3([31, 命题 3。7])。 设 $\bar{x} \in \Omega$ 是问题 (1)的一个帕累托驻点。那么， $\bar{x}$ 满足 *MOLZ* 条件。

正如 [31], 所指出的，反向结论并不总是成立；为了使这两个条件等价，我们需要一个更强的要求。

引理 4 ([31, 命题 3。10])。 点 $\bar{x} \in \Omega$ 是问题 (1) 的帕累托驻点，当且仅当它对所有 $J \in \mathcal{J}(\bar{x})$都满足 *MOLZ* 条件。

帕累托驻点条件可视为基数约束 SOO 中基本可行性 概念的直接扩展 [1,2]。因此，标量基本可行性的局限性自然也会延伸到多目标优化情形；具体而言，帕累托驻点条件只是局部最优性条件，无法提供有关当前支撑集质量的信息。MOLZ 条件更加凸显了这一问题，因为它们通常比帕累托驻点条件限制更弱。

基于上述考虑，我们希望将较强的 L-平稳性条件从 [1] 扩展到多目标优化情形。为此，我们将用近端算子重新诠释 L-平稳性。具体而言，我们可以利用问题 (3) 为多目标优化定义 L-平稳性。

考虑如下问题

$$\theta_L(\bar{x}) = \min_{d \in \mathcal{D}_L(\bar{x})} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top d + \frac{L}{2}\|d\|^2, \tag{6}$$

其中 $\mathcal{D}_L(\bar{x}) = \{v \in \mathbb{R}^n \mid \bar{x} + v \in \Omega\}$ ，并记 $v_L(\bar{x})$ 为 $\bar{x}$ 处最优解的集合（由于 Ω 不是凸集，解不一定唯一）。容易看出，问题 (6) 等价于问题 (3)，其中对所有 $j \in \{1,\dots,m\}$， g_j 都是集合 Ω 的示性函数。

引理 5. 设 $\bar{x} \in \Omega$。则以下条件成立：

1. $\theta_L(\bar{x})$和 $v_L(\bar{x})$均定义良好；*2.* $\theta_L(\bar{x}) \leq 0$；*3.* 映射 $\bar{x} \to \theta_L(\bar{x})$是连续的。

证明。 1. 证明是显然的，因为

- 可行集 $\mathcal{D}_L(\bar{x})$是闭且非空的，
- $\max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top d + \frac{L}{2}\|d\|^2$ 在 $\mathcal{D}_L(\bar{x})$上是强凸且连续的。

2. 给定 $\hat{d} = \mathbf{0}_n$，我们有 $\max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top \hat{d} + \frac{L}{2}\|\hat{d}\|^2 = 0$。由于 $\hat{d} \in \mathcal{D}_L(\bar{x})$，因此得证。

3. 证明与 [22] 中命题4的证明相同。集合 $\mathcal{D}_L(\bar{x})$ 非凸并不影响该论证。

□

现在，我们可以在多目标优化中引入 L-驻点性的定义。

定义 5。 若点 $\bar{x} \in \Omega$ 满足 $\theta_L(\bar{x}) = 0$，则称其为问题 (1) 的 L-驻点 。

注 1。 通过简单的代数变换，问题 (6) 可改写为

$$\min_{z\in\Omega} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top (z-\bar{x}) + \frac{L}{2}\|z-\bar{x}\|^2. \tag{7}$$

现在可以看出，若 $m = 1$，定义 5 实际上与标量 L-驻点性一致。事实上，利用 (7)，$\theta_L(\bar{x})$ 等价于

$$\min_{z\in\Omega} \nabla f(\bar{x})^\top (z-\bar{x}) + \frac{L}{2}\|z-\bar{x}\|^2 = \min_{z\in\Omega} \frac{L}{2}\|z-\bar{x}+\frac{1}{L}\nabla f(\bar{x})\|^2 - \frac{1}{2L}\|\nabla f(\bar{x})\|^2.$$

上述问题的最小值在 $z^* \in \Pi_\Omega[\bar{x} - \frac{1}{L}\nabla f(\bar{x})]$,处取得，其中 Π_Ω是到非凸集合 Ω 的（非唯一）欧几里得投影。因此，若 $\frac{L}{2}\|z^*-\bar{x}\|^2 + \nabla f(\bar{x})^\top(z^*-\bar{x}) = 0$，则有 $\theta_L(\bar{x}) = 0$；该条件在 $\bar{x} \in \Pi_\Omega[\bar{x} - \frac{1}{L}\nabla f(\bar{x})]$, 时成立，即 $\bar{x}$根据 [1]是 L-驻点。

在本节余下部分，我们将分析 L-驻点性、帕累托最优性、帕累托驻点性与 MOLZ 条件之间的关系。我们首先证明，对于任意 $L > 0$，每个 L-驻点都是帕累托驻点。

命题 1。 设 $\bar{x} \in \Omega$ 是问题(1) 的 L-驻点，且满足 $L > 0$。则， $\bar{x}$ 是问题(1)的帕累托驻点。

^证明。 反设，假设 $\bar{x}$ 不是问题 (1) 的帕累托驻点，即存在 $d \in \mathcal{D}(\bar{x})$，使得

$$0 > \max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d} + \frac{1}{2}\|\hat{d}\|^2 \geq \max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d}, \tag{8}$$

其中，第二个不等式由范数算子的非负性得到。

~^ ^ 现在定义方向 $d(t) = td$。根据 $\mathcal{D}(\bar{x})$ 的定义以及 d 的可行性，存在 ~ $\bar{t} > 0$ 使得 $\bar{x} + d(t) \in \Omega \ \ \forall t \in [0,\bar{t}]$。因此，根据 $\mathcal{D}_L(\bar{x})$ 的定义，对于所有 $t \in [0,\bar{t}]$， $d(t) \in \mathcal{D}_L(\bar{x})$。定义函数 $\tilde{\theta}_L$: $\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ 为 $\tilde{\theta}_L(x,d) = \max_{j=1,\dots,m} \nabla f_j(x)^\top d + \frac{L}{2}\|d\|^2$. 由 (6) 可得$\theta_L(\bar{x}) = \tilde{\theta}_L(\bar{x},\bar{d^L})$，其中 $\bar{d^L} \in v_L(\bar{x})$，并且

$$\theta_L(\bar{x}) \leq \tilde{\theta}_L(\bar{x},d), \quad \forall d \in \mathcal{D}_L(\bar{x}). \tag{9}$$

结合 $\tilde{d}(t)$ 和 $\tilde{\theta}_L(x,\ d)$ 的定义，可得 $\tilde{\theta}_L(\bar{x},\ \tilde{d}(t)) = t\max_{j=1,\ \dots,\ m} \nabla f_j(\bar{x})^\top \hat{d} + t^2\frac{L}{2}\|\hat{d}\|^2$. 显然,$\theta_L(\bar{x},\ d(t)) < 0$ 当且仅当

$$0 < t < -\frac{2}{L\|\hat{d}\|^2} \max_{j=1,\dots,m} \nabla f_j(\bar{x})^\top \hat{d}, \tag{10}$$

其中，右侧是一个正量，因为 $L > 0$ 且式 (8) 成立。

然后，考虑到 $\tilde{d}(t)$ 的可行性以及式 (9) 和 (10)，我们可以定义一个方向 $\tilde{d}(\hat{t})$，其中$\hat{t} \in \Big(0, \min\{\bar{t}, -\frac{2}{L\|\hat{d}\|^2}\max_{j=1,\ \dots,\ m} \nabla f_j(\bar{x})^\top \hat{d}\}\Big)$，使得 $\tilde{d}(\hat{t}) \in \mathcal{D}_L(\bar{x})$ 且 $\theta_L(\bar{x}) \leq \tilde{\theta}_L(\bar{x},\ \tilde{d}(\hat{t})) < 0$。最后我们得到矛盾，因为根据假设，$\bar{x}$ 是式 (1) 的一个 L-驻点，即 $\theta_L(\bar{x}) = 0$。 □

最后一个结果还揭示了 L-驻点性与 MOLZ 条件之间的关系。正如引理 3 所述，MOLZ 条件是帕累托驻点性的必要条件。我们在下一个推论中将其形式化。

推论 1. 设 $\bar{x} \in \Omega$ 是问题 (1) 的一个 L-驻点，其中 $L > 0$。则， $\bar{x}$ 满足 *MOLZ* 条件。

根据命题 1 和引理 2，我们还可以指出，在 F的凸性假设下， L-驻点性是局部弱帕累托最优性的充分条件。

推论 2. 假设 F 是分量凸的，且令 $\bar{x} \in \Omega$ 为问题 (1) 的一个 L-驻点，其中$L > 0$。那么， $\bar{x}$ 是问题 (1)的局部弱帕累托最优解。

为了继续分析，我们需要引入几个概念。第一个是假设，与 [1], 中用于 L-驻点的假设类似；第二个则是将下降引理 推广到多目标优化。

假设1。对于所有 $j \in \{1,\dots,m\}$，∇f_j 在 $\mathbb{R}^n$ 上是 Lipschitz连续的，其常数为 L（f_j），即，对于所有 x, $y \in \mathbb{R}^n$.

下文中，我们用 $L(F) \in \mathbb{R}^m$ 表示 *Lipschitz* 常数向量，即 $L(F) = (L(f_1)\dots, L(f_m))^\top$。引理*6*（[3, 命题 *A.* 24]）。设 f_j, $j = 1,\dots,m$为满足假设*1*的连续可微函数。那么，对于所有 $L \geq L(f_j)$以及任意 $x, d \in \mathbb{R}^n$，有 $f_j(x+d) \leq f_j(x) + \nabla f_j(x)^\top d + \frac{L}{2}\|d\|^2$。

我们现在可以证明，对于特定的 L 取值，L-平稳性条件是弱帕累托最优性的必要条件。

命题2。设假设*1*成立，$\bar{x} \in \Omega$ 是问题(1) 的弱帕累托最优点，且 $L >$最大值$_{j=1,\dots,m} L(f_j)$。那么，$\bar{x}$ 是(1)的 L-平稳点。此外，我们有 $v_L(\bar{x}) = \{\mathbf{0}_n\}$，即集合 $v_L(\bar{x})$是单元素集合。

证明。反证法，假设 $\bar{x}$ 不是(1)的 L-驻点，或者 $v_L(\bar{x}) \setminus \{\mathbf{0}_n\} \neq \emptyset$。那么，存在一个方向 $d \in \mathcal{D}_L(\bar{x})$，使得 $d \neq \mathbf{0}_n$ 并且

$$\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d} + \frac{L}{2}\|\hat{d}\|^2 \leq 0. \tag{11}$$

根据引理6，对于所有 $h \in \{1,\dots,m\}$，我们有

$$f_h\left(\bar{x}+\hat{d}\right) \leq f_h(\bar{x}) + \nabla f_h(\bar{x})^\top \hat{d} + \frac{L(f_h)}{2}\|\hat{d}\|^2. \tag{12}$$

由等式(11)可得 $\nabla f_h(\bar{x})^\top \hat{d} \leq \max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d} \leq -\frac{L}{2}\|\hat{d}\|^2$，其中，第一个不等式来自最大值算子的定义。结合关于 L 的假设以及范数的非负性，将(12)与上一个结果合并，可得

$$f_h\left(\bar{x}+\hat{d}\right) \leq f_h(\bar{x}) + \frac{L(f_h)-L}{2}\|\hat{d}\|^2 < f_h(\bar{x}) + \frac{L(f_h) - \max_{j\in\{1,\dots,m\}} L(f_j)}{2}\|\hat{d}\|^2.$$

因此，对于所有 $h \in \{1,\dots,m\}$，我们都有 $f_h(\bar{x}+\hat{d}) - f_h(\bar{x}) < \frac{\|d\|^2}{2}(L(f_h) - \max_{j\in\{1,\dots,m\}} L(f_j)) \leq 0$, 由此得出结论：我们找到了一个点 $\bar{x}+d \in \Omega$，使得 $F(\bar{x}+d) < F(\bar{x})$。这与假设矛盾，因为根据假设，$\bar{x}$ 对于 (1) 是弱帕累托最优的。因此，命题得证。 □

对 L-驻点性的分析表明，L 值的选择可能至关重要：如果 L 过小，L-驻点性可能不是必要的最优性条件；另一方面，如果 L 过大，所有帕累托驻点也会成为 L-驻点。通过一个例子可以更清楚地看出这一现象。

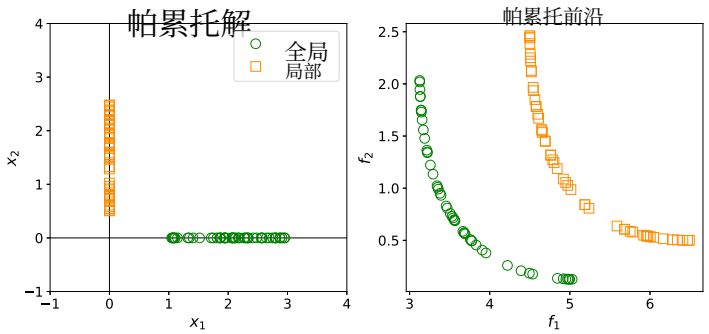


图1：例1问题的帕累托最优解和帕累托前沿。

示例1。考虑以下优化问题：

$$\min_{x\in\mathbb{R}^2} \quad \frac{1}{2}\left((x_1-3)^2 + (x_2-2.5)^2, (x_1-1)^2 + (x_2-0.5)^2\right)^\top \text{ s.t. } \|x\|_0 \leq 1.$$

两个目标函数 f_j 的梯度的Lipschitz常数均为 $L(f_j) = 1$。图1绘出了帕累托最优解和帕累托前沿：该问题的全局最优解对应于满足$x_1 \neq 0$的点；局部最优解则由第二个分量 $x_2 \neq 0$刻画。由引理1-3可知，所有考察的点都是帕累托驻点，并满足 MOLZ 条件。图2展示了在 L取四种不同值时，哪些帕累托解是 L-驻点。若 L 取值过小（图2a），则某些全局帕累托解并非 L-驻点。如命题2所述，当 L 值大于Lipschitz常数时， L-驻点条件是帕累托最优性的必要条件（图2b中 $L = 1.01$）。另一方面，取值过高会使该条件相当宽松：在图2c（$L = 1.25$）中，甚至一些局部帕累托解也是 L-驻点。图2d进一步凸显了这一情况，其中 $L = 2$，且所有帕累托驻点也都是 L-驻点。

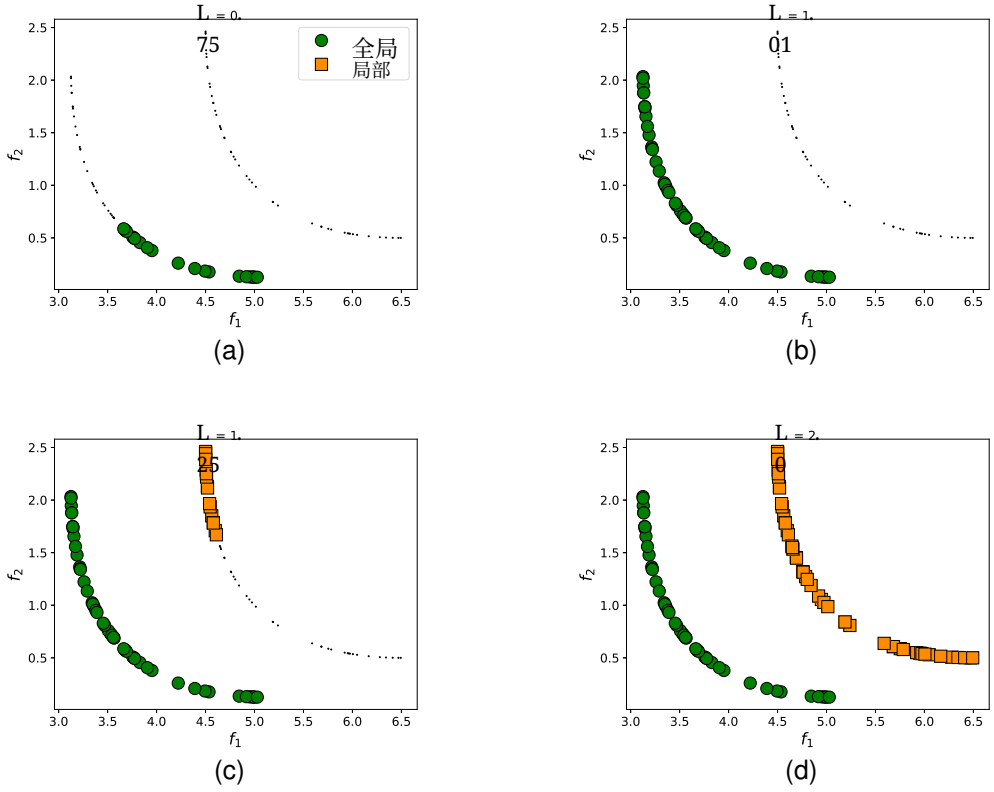


图 2： L-驻点　　例1问题的帕累托前沿上的点（$L(F) = [1, 1]^\top$），对应不同的 L取值　　.

4 稀疏多目标优化问题的新算法方法

本节提出两种求解基数约束多目标优化问题的过程。第一种可视为对 迭代硬阈值（IHT）算法 [1] 的扩展；第二种是前沿近似方法，以候选解作为输入，这些解可能对应不同的支撑集，然后对与这些支撑集相关的帕累托前沿部分进行展布。我们将在不同的小节中介绍它们的方案并讨论其性质。

4.1 多目标迭代硬阈值

我们首先介绍多目标迭代硬阈值 算法。在本文其余部分中，我们将其称为 MOIHT。该方法的步骤见算法 1。在 MOIHT 的每次迭代中，通过求解问题 (13) 来更新当前解 x_k。执行过程持续进行，直至找到问题 (1) 的 L-驻点。

注记 2。 在每次迭代 k中，MOIHT 生成的解 x_{k+1} 对于 (1) 是可行的。事实上，根据 $\mathcal{D}_L(x_k)$的定义，很容易得出其可行性。

注记 3。 需要特别强调的是，步骤 4 是一项实用操作，在一般情况下可以有效实施。实际上，问题 (13) 可以通过混合整数规划技术等方法求得全局最优解（参见例如 [6,7]）。定义一个足够大的标量 $M > 0$后，该问题可以等价地改写为

$$\min_{t,d,\delta} \; t + \frac{L}{2}\|d\|^2 \quad \text{s.t.} \quad \nabla f_j(\bar{x})^\top d \le t \quad \forall j = 1,\dots,m, \qquad \mathbf{1}_n^\top \delta \le s,$$
$$-M\delta \le \bar{x} + d \le M\delta, \qquad t \in \mathbb{R},\, d \in \mathbb{R}^n,\, \delta \in \{0,1\}^n.$$

算法1： 多目标迭代硬阈值法

输入： $x_0 \in \Omega$, $L > \max_{j\in\{1,\dots,m\}} L(f_j)$.$k$
$= 0$
当 x_k 不是 L问题 (1) 的驻点时**执行**计
  算
$$v_L(x_k) = \arg\min_{d\in\mathcal{D}_L(x_k)} \max_{j\in\{1,\dots,m\}} \nabla f_j(x_k)^\top d + \frac{L}{2}\|d\|^2 \tag{13}$$
  令 $d_k^L \in v_L(x_k)$
  $x_{k+1} = x_k + d_k^L$
  令 $k = k+1$
返回 x_k

4.1.1 收敛性分析

本节从理论角度分析该方法。在证明主要收敛性结果之前，我们需要对 F 陈述一个额外假设，并证明一个技术引理。

假设 2. F在多目标意义下具有有界水平集，即对于任意 $z \in \mathbb{R}^m$，集合 $\mathcal{L}_F(z) = \{x \in \Omega \mid F(x) \le z\}$ 有界。

引理 7. 设假设 *1–2* 成立，且 $\{x_k\}$ 是由算法 *1* 在常数 L >最大值$_{j\in\{1,\dots,m\}} L(f_j)$下生成的序列。则：

1. 对所有 k, $F(x_k) - F(x_{k+1}) \ge \frac{1}{2}\|x_k - x_{k+1}\|^2(\mathbf{1}_m L - L(F))$;
2. 对所有 k，若 $x_k \ne x_{k+1}$，则 $F(x_{k+1}) < F(x_k)$；*3.* 对所有 $j \in \{1,\dots,m\}$，序列 $\{f_j(x_k)\}$ 单调不增；*4.* 序列 $\{F(x_k)\}$ 收敛；*5.* 极限$_{k\to\infty} \|x_k - x_{k+1}\|^2 = 0$。

证明。 1. 该论 断可通过采用类似于命题2的论证并注意到以下事实来证明 即$x_{k+1} = x_k + d_k^L$, 其中 $d_k^L \in v_L(x_k)$（算法 1 的第 6 步）。

2. 这直接 由第 1 点可得；回想一下，对所有 $j \in \{1,\dots,m\}$ 和 $\|x_k - x_{k+1}\| > 0$, $L > L(f_j)$。

3. 根据第 1 点及关于 L 的假设，对所有 k 和 $j \in \{1,\dots,m\}$，有 $f_j(x_{k+1}) \le f_j(x_k)$。因此，对所有 $j \in \{1,\dots,m\}$，序列 $\{f_j(x_k)\}$ 单调不增。

4. 由假设 2 和第 3 点可得。

5. 根据第 1 点，对于所有 k和 $j \in \{1,\dots,m\}$, $\frac{L-L_{(f_j)}}{2}\|x_k - x_{k+1}\|^2 \le f_j(x_k) - f_j(x_{k+1})$. 由于$f_j$ 连续，我们可以对不等式两边关于 $k \to \infty$取极限：$\lim_{k\to\infty} \frac{L-L_{(f_j)}}{2}\|x_k - x_{k+1}\|^2 \le \lim_{k\to\infty} f_j(x_k) - f_j(x_{k+1}) = 0$, 其中等式由第 4 点得出。根据 L的定义以及范数的非负性，命题得证。

□

命题 3. 设假设 *1-2* 成立，且 $\{x_k\}$ 为由常数 $L > \max_{j\in\{1,\dots,m\}} L(f_j)$下的算法 *1* 生成的序列。则该序列存在聚点，且每个聚点都是问题 (1)的 L-驻点。

证明。 首先，我们证明该序列存在极限点。由引理 7 可得，对所有 k, $F(x_k) \leq F(x_{k-1}) \leq \dots \leq F(x_0)$。此外，如注2所述，对于所有 k， $x_k \in \Omega$。因此，我们有 $x_k \in \mathcal{L}_F(F(x_0))\ \forall k$。由于假设 2 成立，我们得出该序列 $\{x_k\}$ 有界，因此存在极限点。现在，反证法，假设存在一个子序列 $K \subseteq \{0,1,\dots\}$，使得

$$\lim_{\substack{k\to\infty \\ k\in K}} x_k = \bar{x} \tag{14}$$

且 $\bar{x}$ 对于问题 (1) 不是 L-驻点。于是，存在 $\hat{\varepsilon} > 0$，使得对于所有 $k \in K$， $\theta_L(x_k) \leq -\hat{\varepsilon} < 0$。按照该算法的指示，在每次迭代 k 中， $d_k^L \in v_L(x_k)$ 求解问题 (13)。此外，根据步骤 6，$d_k^L = x_{k+1} - x_k$。因此，$\max_{j\in\{1,\dots,m\}} \nabla f_j(x_k)^\top (x_{k+1} - x_k) + \frac{L}{2}\|x_{k+1} - x_k\|^2 = \theta_L(x_k) \leq -\hat{\varepsilon}$，这可以改写为

$$\max_{j\in\{1,\dots,m\}} \nabla f_j(x_k)^\top (x_{k+1} - x_k) \leq -\hat{\varepsilon} - \frac{L}{2}\|x_{k+1} - x_k\|^2 . \tag{15}$$

根据引理6和算法的第6步，我们有，给定 $h \in \{1,\dots,m\}$,

$$f_h(x_{k+1}) \leq f_h(x_k) + \nabla f_h(x_k)^\top (x_{k+1} - x_k) + \frac{L(f_h)}{2}\|x_{k+1} - x_k\|^2 . \tag{16}$$

根据最大值算子的定义，我们可以结合 (15)-(16) 得到 $f_h(x_{k+1}) \leq f_h(x_k) - \hat{\varepsilon} + L(f_h) - L\|x_{k+1} - x_k\|^2$，因此，

$$f_h(x_k) - f_h(x_{k+1}) \geq \hat{\varepsilon} + \frac{L - L(f_h)}{2}\|x_{k+1} - x_k\|^2 . \tag{17}$$

根据引理7的第3点，我们知道，对所有 $k \geq 1$， $f_h(x_{k-1}) \geq f_h(x_k)$。结合后一个结果和 (17)，对于满足 $k \geq 1$的 $k \in K$，我们得到

$$f_h(x_{k-1}) - f_h(x_{k+1}) \geq \hat{\varepsilon} + \frac{L - L(f_h)}{2}\|x_{k+1} - x_k\|^2 . \tag{18}$$

递归地 应用 (18) 中使用的推理，我们可以得出结论：对于所有满足 $k_1 < k$ 的 $k_1, k_2 \in K$ ，

$$f_h(x_{k_1}) - f_h(x_{k_2+1}) \geq \hat{\varepsilon} + \frac{L - L(f_h)}{2}\|x_{k_2+1} - x_{k_2}\|^2 . \tag{19}$$

现在，考虑到引理7的第(v)点、(14)以及范数的连续性，我们得到 $\lim_{k\to\infty, k\in K} x_{k+1} = \bar{x}$。因此，对 $k \to \infty, k \in K$ 在 (19) 中取极限，再次回顾引理7的第(v)点、(14)以及 f_h 的连续性，我们得出 $0 = f_h(\bar{x}) - f_h(\bar{x}) \geq \hat{\varepsilon}$.。由于 $\hat{\varepsilon} > 0$，这就产生了矛盾。因此，命题得证。 □

注4。 根据命题 3 和 θ_L （引理 5）的连续性，我们可以保证，对于任意 $\varepsilon > 0$，算法 1 都会在有限次迭代内产生一个满足 $\theta_L(x_k) > -\varepsilon$ 的点 x_k。因此，我们可以有效地将此条件用作 MOIHT 程序的实用停止准则。

4.2 稀疏前沿最速下降

接下来，我们将介绍并分析稀疏前沿最速下降 (SFSD)方法。该算法可视为一种两阶段方法，其依据如下：对于形式如（1）的问题，帕累托前沿通常是不规则集合，由若干彼此分离的光滑部分构成；前沿上的每个这类优良部分通常都是一组具有相同结构的解的像，即与同一支撑集相关联。因此，所提出算法的基本思路是先定义一组初始解；理想情况下，这些解的支撑集应具有多样性，并界定一个包含部分帕累托集的子空间。随后，可从这组初始解出发运行前沿最速算法 [17,33] 的一种改进，以穷尽式地遍历前沿。据我们所知，SFSD 是首个用于基数约束多目标优化的前沿导向方法。

4.2.1 PhaseOne：初始化

SFSD 算法的第一阶段旨在识别一组初始解。最直接的方法或许是穷举所有超支持集，并为每个超支持集选取一个解。然而，可能的支撑集数量庞大，增长速度高达 $\binom{n}{s}$，但只有很小一部分会对帕累托前沿有所贡献。因此，这种策略效率低下，对于规模不小的问题甚至完全不切实际。

完全随机的初始化似乎也可以作为一种可行方案，但基于与上述类似的原因，最终只会变成完全依赖运气的操作。因此，我们建议利用单点求解器来求取帕累托驻点解。事实上，借助这类算法的机制，得到的点不仅是驻点，而且从全局优化的角度来看，通常也是较好的解。我们探讨了以下几种方案（并非穷尽）：

- 以多起点方式使用第 4.1 节讨论的 MOIHT。由于该算法能够找到 L-驻点解，因此优化过程应避免采用“较差”的支撑集。
- 以多起点方式使用 [31] 中的多目标稀疏罚函数分解（MOSPD）方法；简而言之，在 MOSPD 的每次迭代 k 中，都会找到一对 $(x_{k+1},\ y_{k+1})$，使得 x_{k+1} 对于罚函数 $Q_{\tau_k}(x,\ y_{k+1}) = F(x) + \frac{\tau_k}{2}\mathbf{1}_m\|x - y_{k+1}\|^2$，是（近似）帕累托驻点，其中 $\tau_k \to \infty$ 对应于 $k \to \infty$。这对 $(x_{k+1},\ y_{k+1})$通过一种交替最小化 方案得到。更多细节请参见 [31]。已证明，MOSPD 收敛到满足问题 (1) 的多目标 Lu-Zhang 条件的点；这些条件甚至弱于帕累托驻点条件。然而，已有研究表明，在标量 [30] 和多目标 [31] 情形下，罚函数分解方法均能求得从全局优化角度来看较好的解。
- 结合前述两种策略：对于多起点随机初始化中的每个点，我们可以先运行 MOSPD 过程，利用其探索能力；然后再串联使用 MOIHT，从而最终逃离那些不满足 L-驻点条件的不良 Lu-Zhang 点。我们将这种方法称为 MOHyb。
- 针对不同的权衡参数求解标量化的单目标问题。

通过上述任一策略获得初始解集后，还需进一步执行一个步骤。具体而言，我们需要为每个解关联一个超支持集。如果某个解具有满支撑，那么其超支持集唯一，不存在歧义。然而，有些解的支撑可能不完整；这些解可能不是帕累托驻点（例如通过 MOSPD 获得的解），此时我们应沿最陡可行下降方向执行一步下降；另一方面，如果我们实际得到的是一个支撑不完整的帕累托驻点，则应将其与任一超支持集相关联。

显然，我们可以通过筛选操作完成第一阶段，其中被支配的解将被舍弃。总之，算法第一阶段的结果提供了一组初始解，每个解都与一个超支持集相关联。

4.2.2 第二阶段：前沿最速下降

算法2给出了所提出算法框架（SFSD）的方案。

该方法从初始化步骤（即算法第一阶段）得到的初始解集开始运行；所得集合 $\mathcal{X}^0$ 定义如下：

$$\mathcal{X}^0 = \{(x, J_x) \mid J_x \in \mathcal{J}(x)\},$$

也就是说，这些解与相应的超支持集相关联。对于任意一对 $(x,\ J_x)$，$(y,\ J_y) \in \mathcal{X}^0$，其中 $J_x = J_y$，我们假设 x 和 y 关于 F 互不支配。

基本上，SFSD 算法采用前沿最速下降算法 [17]，的指令，并按照 [33]，中的建议进行修改，分别处理与不同超支持集相关联的点。

具体而言，对于当前帕累托前沿近似中的任意非支配点 x_c，沿支撑集 J_{x_c} 对应的子空间执行一个共同下降步，并进行标准的 Armijo 型线搜索 [27]。换言之，根据式 (5)，搜索方向由 $d_{J_{x_c}}(x_c)$ 给出。随后，只要所得点未被集合 y 中满足 $J_y = J_{x_c}$ 的任何其他点支配，就从该点出发，针对目标子集进一步进行搜索。这些额外探索沿当前点参考子空间中的部分下降方向 [16, 17] 进行。将 $I \subseteq \{1, \ldots, m\}$ 视为目标索引的子集，我们定义$\theta_J^I(\bar{x}) = \min_{d\in\mathbb{R}^n} \max_{j\in I} \nabla f_j(\bar{x})^\top d + \frac{1}{2}\|d\|^2$使得 $d_{\bar{J}} = \mathbf{0}_{|\bar{J}|}$。与式 (5) 类似，该问题有唯一解，记为 $d_J^I(\bar{x})$。

算法2： 稀疏前沿最速下降

输入： $F\colon \mathbb{R}^n \to \mathbb{R}^m$, $\alpha_0 > 0$, $\delta \in (0,1)$, $\gamma \in (0,1)$.
$\mathcal{X}^0 =$ 初始化(F)
$k = 0$
当未满足停止准则时 **执行**
  $\widehat{\mathcal{X}}^k = \mathcal{X}^k$
  对所有 $(x_c, J_{x_c}) \in \mathcal{X}^k$ **执行**
    如果$(x_c, J_{x_c}) \in \widehat{\mathcal{X}}^k$ **则**
      如果 $\theta_{J_{x_c}}(x_c) < 0$**然后**
        $\alpha_c^k =$最大$_{h\in\mathbb{N}} \{\alpha_0\delta^h \mid F(x_c + \alpha_0\delta^h d_{J_{x_c}}(x_c)) \leq F(x_c) + \mathbf{1}_m\gamma\alpha_0\delta^h\theta_{J_{x_c}}(x_c)\}$
      else
        $\alpha_c^k = 0$
      $z_c^k = x_c + \alpha_c^k d_{J_{x_c}}(x_c)$
      $\widehat{\mathcal{X}}^k = \widehat{\mathcal{X}}^k \setminus \left\{(y, J_y) \in \widehat{\mathcal{X}}^k \mid J_y = J_{x_c},\ F(z_c^k) \lneq F(y)\right\} \cup \left\{(z_c^k, J_{x_c})\right\}$
      对所有 $I \subseteq \{1,\dots,m\}$ 使得 $\theta^I_{J_{x_c}}(z_c^k) < 0$ **执行**
        如果$(z_c^k, J_{x_c}) \in \mathcal{X}^k$ **则**
          $\alpha_c^I =$
          $\max_{h\in\mathbb{N}} \{\alpha_0\delta^h \mid \forall (y, J_y) \in \widehat{\mathcal{X}}^k,\ J_y = J_{x_c},\ \exists j \in \{1,\dots,m\} \text{ s.t. } f_j(z_c^k + \alpha_0\delta^h d^I_{J_{x_c}}(z_c^k)) < f_j(y)\}$
          $\hat{z} = z_c^k + \alpha_c^I d^I_{J_{x_c}}(z_c^k)$
          $\widehat{\mathcal{X}}^k = \widehat{\mathcal{X}}^k \setminus \left\{(y, J_y) \in \widehat{\mathcal{X}}^k \mid J_y = J_{x_c},\ F(\hat{z}) \lneq F(y)\right\} \cup \{(\hat{z}, J_{x_c})\}$
  $\mathcal{X}^{k+1} = \widehat{\mathcal{X}}^k$
  $k = k + 1$
返回 $\mathcal{X}^k$

由于只有对应于同一超支持集的解才会进行比较，因此在 SFSD 中，不同超支持集所诱导的子空间会分别进行探索。因此，我们基本上得到相互独立的帕累托前沿近似，每个近似对应一个超支持集。SFSD 执行结束后，可以对所有点进行比较，并最终筛除被支配的点，从而得到问题 (1) 的最终帕累托前沿近似。

需要注意的是，从概念上讲，该算法可以看作多次独立运行前沿最速下降法，每次都将优化过程限制在某个特定子空间内；然而，SFSD 会在不同支撑上“并行”探索，从而均匀地构建前沿近似，避免将全部计算预算都耗费在对最先考虑的少数支撑进行优化上。

注 5。 由于每个点仅在其关联超支持集所诱导的子空间中进行搜索步骤，因此很容易得出，每个新解都将满足 (1) 的可行性要求。

4.2.3 算法理论分析

本节陈述 SFSD 方法的收敛性质。关于 SFSD 直接继承自前沿最速下降法的性质证明，请参阅 [33] 。

在证明收敛结论之前，我们需要引入集合 $X_J^k = \left\{x \mid \exists (x, J) \in \mathcal{X}^k\right\}$， 其中 J 表示一个超支持集，并回顾关联序列的定义。关联序列最初由 [37] 引入。

定义6. 设 $\left\{X_J^k\right\}$ 为算法2生成的、与超支持集 J相关联的非支配点集合序列。我们将满足以下条件的序列 $\left\{x_{j_k}^J\right\}$ 定义为关联序列：对于所有 k，点 $x_{j_k}^J \in X_J^k$ 由点 $x_{j_{k-1}}^J \in X_J^{k-1}$ 在算法2的第 $k-1$ 次迭代中生成。

命题4。 假设 X_J^0 是一组彼此非支配的点，且存在 $x_0^J \in X_J^0$，使得集合 $\widehat{\mathcal{L}}_F\left(x_0^J\right) = \bigcup_{j=1}^m \{x \in \Omega \mid f_j(x) \leq f_j(x_0^J)\}$ 是紧的。此外，设 $\left\{X_J^k\right\}$ 为集合序列

由算法*2*生成，与超支持集 J相关联的非支配点构成，且 $\left\{x_{j_k}^J\right\}$ 为一个链接序列。那么，后者存在聚点，且每个聚点都满足问题(1)的 *MOLZ* 条件。

证明。 根据算法的指示，每个关联序列 $\left\{x_{j_k}^J\right\}$ 都可以看作是通过对最小化 $F(x)$ 且满足 $x_{\bar{J}}=0$的问题，从 [33] 开始应用前向最速下降算法所生成的关联序列。因此，我们可以遵循 [33, 命题 3.4] 的证明来说明，每个关联序列 $\left\{x_{j_k}^J\right\}$的聚点 $\bar{x}$ 都满足 $\theta_J(\bar{x})=0$，即 $\bar{x}$ 满足问题 (1) 的 MOLZ 条件。

□

5 计算实验

本节报告一些计算实验的结果，这些实验旨在评估所提方法的数值潜力。实验代码使用 Python3 编写，并在以下配置的计算机上运行：Ubuntu 22.04、Intel Xeon Processor E5-2430 v2（6 核，2.50 GHz）、16 GB RAM。为求解形如 (4)-(5)-(13) 的问题实例，使用了 Gurobi 优化器（9.1 版）。

5.1 实验设置

在数值实验中，我们考虑了两类问题：基数约束二次问题和稀疏逻辑回归任务。

二次多目标优化问题常被用作优化领域的有效测试基准，其形式为

$$\min_{x\in\mathbb{R}^n} \quad \tfrac{1}{2}\left(x^\top Q_1 x - c_1^\top x, x^\top Q_2 x - c_2^\top x\right)^\top \quad \text{s.t.} \quad \|x\|_0 \le s,$$

其中 Q_1，$Q_2\in\mathbb{R}^{n\times n}$ 是随机生成的半正定矩阵，c_1，$c_2\in\mathbb{R}^n$ 是取值在 $[-1,1)$ 范围内随机采样的向量。在实验中，我们改变了以下问题参数：规模$n\in\{10,25,50\}$；矩阵的条件数 $\kappa\in\{1,10,100\}$；基数上界 s。具体而言，后者按以下方式设置：对于 $n=10$，$s\in\{2,5,8\}$；对于 $n=25$，$s\in\{5,10,20\}$；对于 $n=50$，$s\in\{5,15,30\}$。此外，我们使用了 3 个不同的伪随机数生成器种子，因此总共得到 81 个二次问题。对于每个实例，Q_1 和 Q_2 具有相同的条件数，即$L(f_1)=L(f_2)=\kappa$。

对于稀疏逻辑回归问题 [5,14],，它是机器学习和统计学习中的一项重要任务。给定一个包含 N 个样本和 n 个特征的数据集 $R=(r_1,\dots,r_N)^\top\in\mathbb{R}^{N\times n}$，以及属于 $\{-1,1\}$ 的 N 个对应标签 $\{t_1,\dots,t_N\}$，正则化稀疏逻辑回归问题 可表示为：

$$\min_{w\in\mathbb{R}^n} \ \tfrac{1}{N}\sum_{i=1}^{N}\log\left(1+\exp\left(-t_i\left(w^\top r_i\right)\right)\right)+\tfrac{\lambda}{2}\|w\|^2 \quad \text{s.t.} \quad \|w\|_0\le s,$$

其中 $\lambda\ge 0$。逻辑损失旨在拟合训练数据，而正则化项有助于避免过拟合。这两个函数显然相互对立。在我们的实验中，我们采用了 [31] 中考虑的多目标重构形式：

$$\min_{w\in\mathbb{R}^n} \quad \left(\tfrac{1}{N}\sum_{i=1}^{N}\log(1+\exp(-t_i(w^\top r_i))),\ \tfrac{1}{2}\|w\|^2\right)^\top \quad \text{s.t.} \quad \|w\|_0\le s.$$

对于这个问题，$L(F)=(\|R^\top R\|/N,1)^\top$。我们考虑的数据集套件由来自 UCI 机器学习资料库的 7 个二元分类数据集组成 [23] （表 1）。我们在具有 $s\in\{2,5,8,12,20\}$ 的问题实例上测试了这些算法。对于每个数据集，均删除了含缺失值的样本。此外，分类变量采用独热编码，其他变量则标准化为均值为零、标准差为一。

为了评估各算法之间的性能，我们采用了性能剖面 [21]。简而言之，性能剖面展示了在某个问题上，一种方法所得指标值与所有参与比较的求解器所取得的最佳值相比，在因子 $\tau\in\mathbb{R}$ 范围内的概率。更多详情请参阅 [21] 。我们采用了多目标优化文献中的一些经典指标：纯度， Γ–分布度， Δ–分布度 [19]和超体积 [48]。请注意，由于纯度 和超体积 的值越大，解越好，因此基于所得值的倒数绘制了相应的性能剖面。

请注意，对于以下两类问题，我们还将考虑基于标量化的求解方法，即求解问题 $\min_{x\in\Omega}\ f_1(x)+\lambda f_2(x)$，其中 $\lambda\geq 0$。对于二次问题，可以利用 MIQP 重构，借助 Gurobi 等商业求解器求解。对于逻辑回归问题，我们则使用贪心稀疏单纯形（GSS）算法 [1]。请注意，与二次问题中的 MIQP 方法不同，GSS 不保证能得到帕累托最优解。
正如第 4.2 节.1 所述，我们的 SFSD 方法采用上述单点方法生成的解作为起始解进行了测试，即 MOIHT、MOSPD、MOHyb、MIQP 和 GSS。与 [33], 中的做法类似，我们采用了一种策略来限制用于部分下降搜索的点数，以提高 SFSD 的效率，并避免产生过多彼此非常接近的解。具体而言，我们添加了一个基于拥挤距离 [20] 的条件，用于判断一个点在完成常规下降步骤后是否应继续探索。

每次运行的时间限制为 4 分钟。具体而言，每种单点方法均以多起点方式进行测试：它必须在 2 分钟内尽可能多地处理输入点；剩余时间则用于运行 MOSD 程序进行细化，从每个返回点出发，并固定其零值变量，以确保满足基数约束。对于 SFSD，算法的两个阶段的时间限制均设为 2 分钟。对于 MOIHT、MOSPD 和 MOHyb，我们考虑了从一个盒状区域中随机采样的 $2n$ 个初始解（$[-2,2]^n$ 用于二次问题；$[0,1]^n$ 用于逻辑回归）。为了使每个初始点可行，首先将其投影到 Ω 上。这些算法使用伪随机数生成器的不同种子运行 5 次，以降低随机初始化带来的敏感性。随后，根据 纯度 指标比较生成的五个前沿，并仅选取其中最好和最差的前沿进行比较。基于标量化的方法运行一次，取 $2n$ 个 λ 值，即 $\lambda\in\{2^{i+\frac{1}{2}}\mid i\in\mathbb{Z},\ i\in[-n,n)\}$，并以初始解 $\mathbf{0}_n\in\Omega$ 为起点。

数据集	N	n
心脏病（Statlog）	270	25
威斯康星州乳腺癌（预后）	194	33
QSAR 生物降解性	1055	41
SPECTF 心脏	267	44
垃圾邮件数据集	4601	57
手写数字的光学识别	3823	62
Madelon	2000	500

表 1：用于稀疏逻辑回归实验的数据集。特征数量已计入分类变量的独热编码。

5.2 二次问题

本节报告基数约束二次问题的实验结果。关于算法参数，根据一些为简洁起见未在此报告的初步实验，我们设置：MOIHT 使用 $\varepsilon=10^{-7}, L=1.1\kappa$；MOSPD 使用 $\tau_{k+1}=1.5\tau_k, \varepsilon_0=10^{-2}, \varepsilon_{k+1}=0.9\varepsilon_k$，并将 $\|x_{k+1}-y_{k+1}\|\leq 10^{-3}$ 作为停止条件；MOSD 的帕累托驻点近似度为 $\varepsilon=10^{-7}$；所有 Armijo 型线搜索均使用 $\alpha_0=1, \delta=0.5$ 和 $\gamma=10^{-4}$。MOSPD 参数 τ_0 的可能取值将在下一节讨论。MOIHT 和 MOSPD 的参数设置也用于 MOHyb。

5.2.1 MOIHT、MOSPD和MOHyb 的初步评估

我们首先在一组选定的二次问题上分析 MOIHT、MOSPD 和 MOHyb 的有效性，并在图 3 中对它们进行比较。为了尽可能清晰地展示各算法之间的差异，本实验中我们仅进行一次运行，并让各方法使用相同的 25 个随机抽取的初始点作为输入。此外，我们不设时间限制，以便所有算法都能针对每个初始解运行，直至满足各自的停止准则。

我们考察了 MOSPD 和 MOHyb 在 $\tau_0\in\{1,100\}$ 取不同值时的性能（$\tau_0=100$ 的结果显示在图的左列，$\tau_0=1$ 的结果显示在右列）。黑点表示参考前沿：该前沿由运行 SFSD 并采用所有提出的初始化策略所得到的前沿合并而成，同时剔除了受支配解。

在良态问题（$\kappa=1$）中，MOIHT 算法表现良好，所得解均属于参考前沿。至于 MOSPD，当 $\tau_0=100$ 时，其结果较差，得到的解远离参考前沿。随着问题维数 n 增大，这种情况更加明显。这是合理的，因为

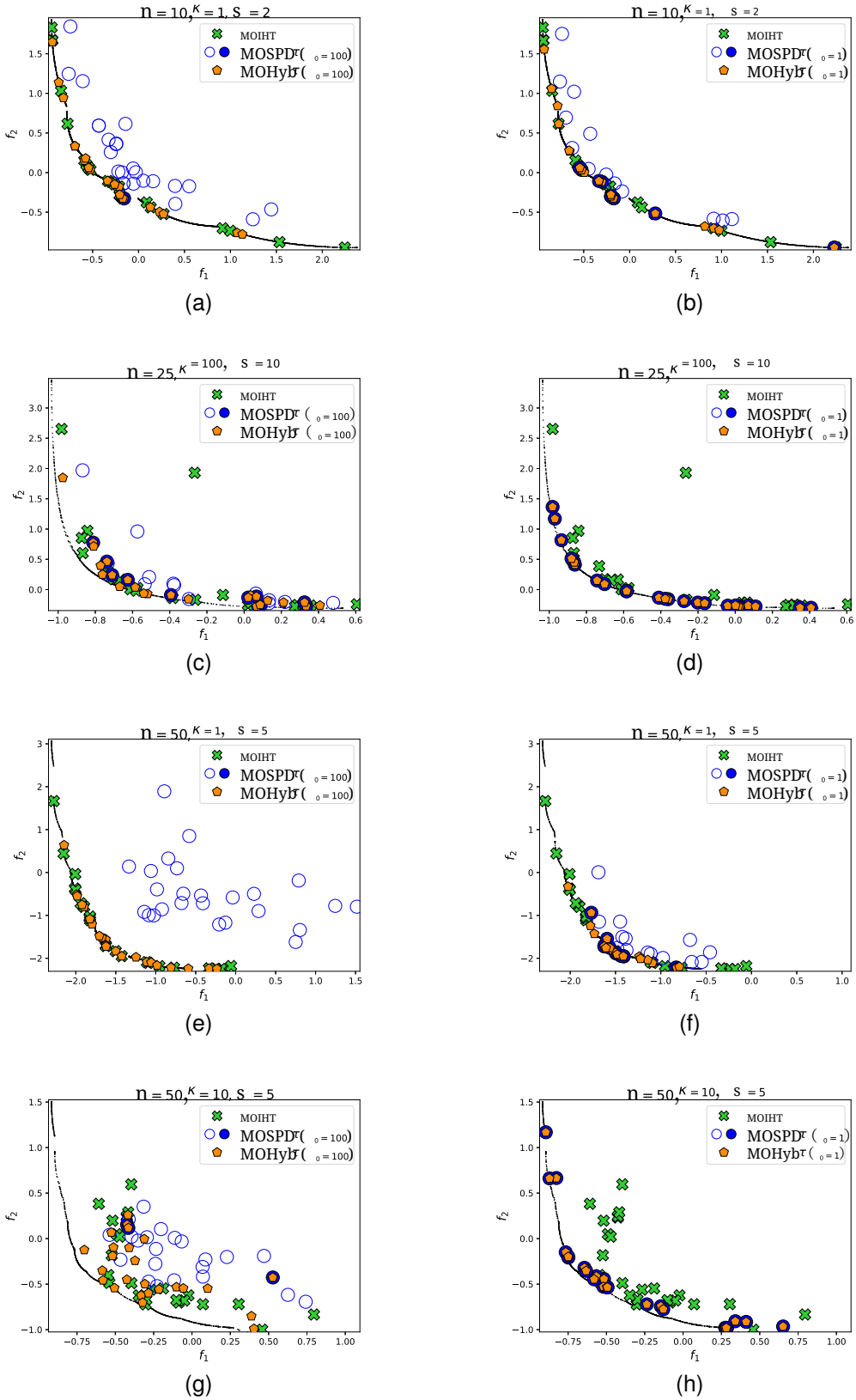
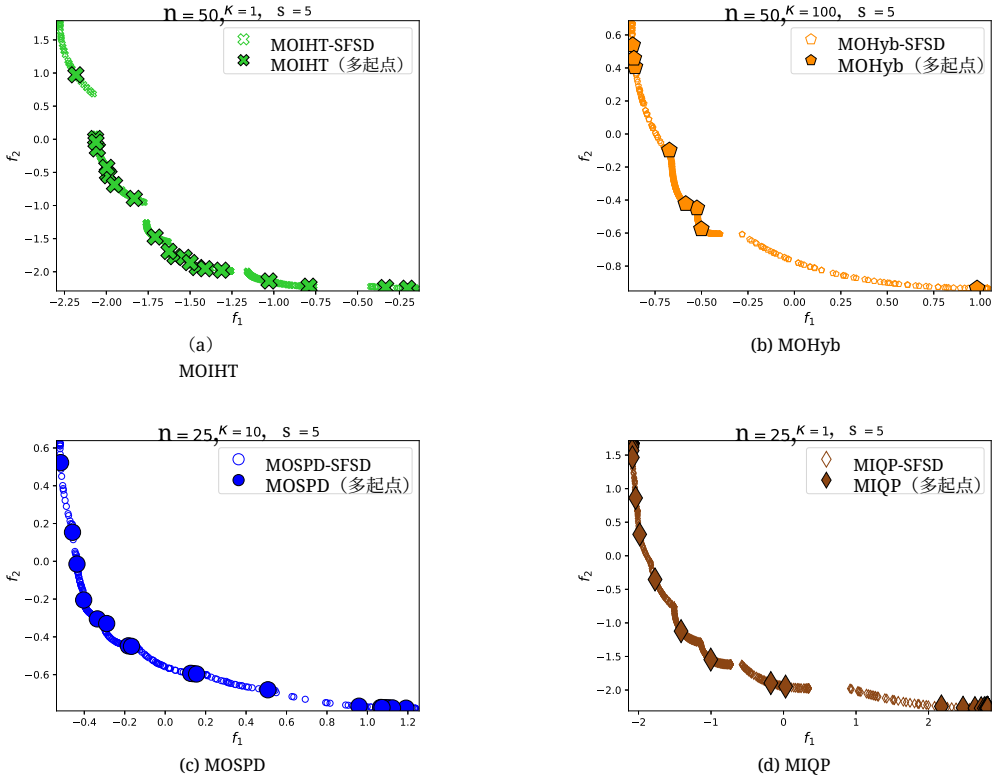


图 3：MOIHT、MOSPD 和 MOHyb 在 $\tau_0 \in \{1, 100\}$ 下的结果，从 25 个随机初始解出发，在选定的二次问题上进行测试。实心标记表示 L-驻点解（$L = 1.1\kappa$）。小黑点构成参考前沿。



(a) MOIHT　(b) MOHyb

(c) MOSPD　(d) MIQP

图 4：SFSD 第二阶段的结果与对第一阶段获得的解进行简单 MOSD 精细化的结果对比。对于每种考虑的多起点/第一阶段策略，我们展示了一个实例。

在罚函数分解方案中设置 $\tau_0 = 100$ 会使变量被限制在初始可行解附近，因此无法发挥 MOSPD 的探索能力。更好的 τ_0 （$\tau_0 = 1$）选择可以提高算法的性能，不过 MOIHT 的表现仍然更好。这个结果在一定程度上符合理论：MOIHT 生成 L-驻点，而 MOSPD 收敛得到的解仅能保证满足（较弱的）MOLZ 条件。在这种情况下，MOHyb 继承了 MOIHT 的有效性：无论 τ_0 取何值，它都成功获得了参考前沿上的解。

在病态问题（$\kappa > 1$）中，MOIHT 的性能会变差：该方法难以达到参考前沿。这或许可以用这些问题中必须采用较大的 L 值（$L = 1.1\kappa$）来解释。随着 L 的增大， L-驻点条件无法提供足够的信息来判断解的支撑集质量。因此，MOIHT 最终可能落入许多具有“较差”支撑集的 L-驻点，也就是说，这些点远离问题的实际帕累托前沿。在这些情况下，采用 $\tau_0 = 1$ 的 MOSPD 获得了更好的解。使用较小的 τ_0 值时，该方法起初可以搜索使 F 最小化的良好点：这一特性对于避免大量“较差”的 L-驻点、并达到参考前沿上的解，可能至关重要。MOHyb（$\tau_0 = 1$）在这些情况下也证明了其有效性。在这些病态情况下，混合方法利用了 MOSPD 的探索能力，达到了相同的解。不过，与条件良好的情况一样，MOHyb 也证明自己对 τ_0 的取值不如 MOSPD 敏感，并利用 MOIHT 的机制至少获得 L-驻点（当 $\tau_0 = 100$ 时）。

5.2.2 SFSD方法评估

我们通过图 4 开始分析 SFSD 算法在二次问题上的性能。图中展示了前沿下降阶段如何改进第一阶段所对应的基础多起点方法的结果，即 MOIHT、MOSPD、MOHyb 和 MIQP。根据第 5.2.1 节的结果，对于 MOSPD 和 MOHyb，我们设置 $\tau_0 = 1$。

正如第 4.2 节所预期的，在基数约束多目标优化中，帕累托前沿通常是不规则的，由若干光滑部分组成。图中的曲线完美地反映了这些特征：每一段前沿确实可以

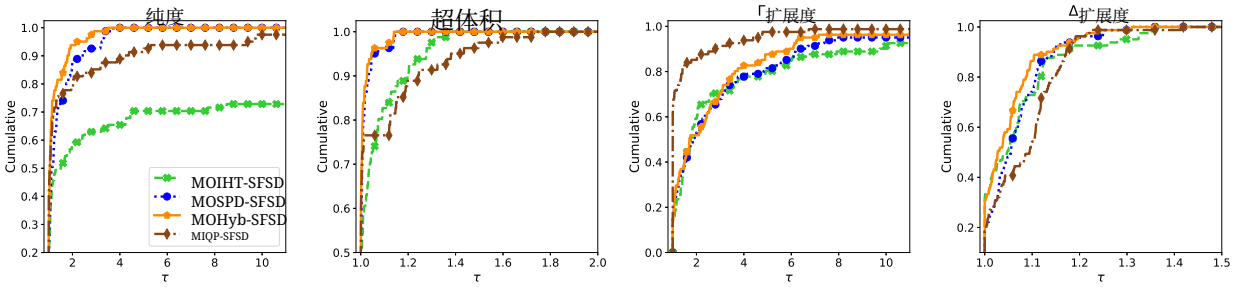


图 5：在二次问题上，SFSD 使用不同初始化策略（即 MOIHT、MOSPD、MOHyb（按纯度指标选取的最佳运行结果）和 MIQP）的性能剖面。

与特定支撑集相关联。从单点方法生成的解出发，SFSD 方法证明能够有效地穷尽遍历与支撑集相关联的每一部分。这一特点使我们新颖的前沿导向方法能够识别帕累托前沿中原本很难通过多起点策略覆盖的区域。

如第 4.2 节所述，据我们所知，SFSD 是首个用于基数约束多目标优化的前沿导向方法。由于缺少其他专用算法，难以对我们算法的潜力进行定量评估，因此需要通过目视检查这些解。在本节余下部分，我们将重点考察 SFSD 算法第一阶段的不同选项，并比较不同初始化策略下的性能。我们在整个二次问题基准集上，借助性能剖面（图 5）进行比较。

从纯度 和超体积 指标来看，与 MOIHT 和 MOSPD 相比，以 MOHyb 作为初始化策略时，SFSD 的表现更为稳健。这些结果反映了图 3 中展示的三种单点算法的表现：MOIHT 在良态问题上效果更好；MOSPD 在为 τ_0 选择合适数值时，则在（规模更大的）病态问题集合上表现更佳；MOHyb 融合了两者的机制，因此在这两类问题上都取得了良好结果。对于 Γ–扩展度，MIQP 在生成目标空间极端区域的解方面优于其他单点方法，因此 SFSD 能够重构出覆盖范围更广的前沿。在这一情形下，第一阶段采用 MOHyb、MOIHT 和 MOSPD 的前沿导向方法，性能相当。对于 Δ–扩展度指标，即帕累托前沿近似的均匀性，所有初始化策略的结果都相近。

5.3 逻辑回归

在本节最后，我们分析 SFSD 在逻辑回归问题上的性能剖面（图 6），并考察算法第一阶段的各种可能选择。算法参数的取值同样依据初步实验确定；为简洁起见，这些实验结果不予报告。具体而言，我们设定： $L = 1.1\max\{L(f_1)\ L(f_2)\}$ 用于 MOIHT； $\varepsilon = 10^{-7}$ 用于 MOIHT 和 MOSD； $\tau_0 = 1$、 $\tau_{k+1} = 1.3\tau_k$、 $\varepsilon_0 = 10^{-5}$、 $\varepsilon_{k+1} = 0.9\varepsilon_k$ $\|x_{k+1} - y_{k+1}\| \leq 10^{-3}$ $\alpha_0 = 1$ $\delta = 0.5$ $\gamma = 10^{-4}$ 以及作为 MOSPD 的停止条件；所有 Armijo 型线搜索均采用 , 和 。同样，MOIHT 和 MOSPD 的参数也用于 MOHyb。最后，由于目标函数的尺度各不相同，与 [31], 中的做法类似，在计算分布度 指标时，我们对 f_2 值取以 10 为底的对数，然后重新缩放 f_1 和 f_2，使其取值落在 $[0, 1]$ 中。

就纯度 和超体积指标而言，采用 MOIHT、MOSPD 和 MOHyb 作为初始化策略时，SFSD 表现得更稳健，其中 MOHyb 略占优势。对于 Γ–分布度 指标，GSS 是 SFSD 第一阶段的最佳算法。不过，尽管采用这一设置的 SFSD 能够有效到达目标空间中的远端区域，但它难以获得均匀的帕累托前沿近似，因此也难以取得良好的Δ–分布度 值。对于这一最后的指标，采用 MOIHT 作为初始化策略是更好的选择。

注记 6。 在前面的章节中，我们考察了采用 MOIHT、MOSPD 和 MOHyb 的 SFSD 的最佳运行结果，并将其与第一阶段采用 MIQP/GSS 时，我们面向前沿的算法所产生的确定性输出进行了比较。因此，为了完整起见，图7报告了基于纯度 指标、通过考察最差运行结果得到的性能剖面。将这些性能剖面与图5a-6a中的结果进行比较后，我们发现非确定性策略的性能仅略有下降。

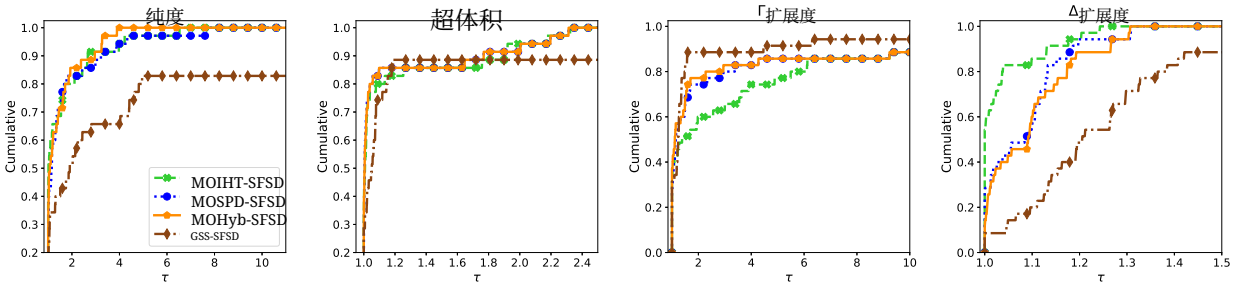


图6：SFSD 在采用不同初始化策略（即 MOIHT、MOSPD、MOHyb（按纯度计的最佳执行结果）和 GSS）时，在 35 个逻辑回归问题上的性能剖面。

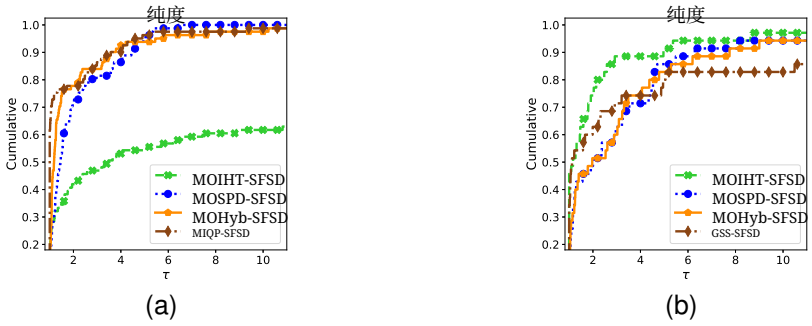


图7：性能剖面：SFSD 采用不同初始化策略，即 MOIHT、MOSPD、MOHyb（按纯度计的最佳执行结果）和 MIQP/GSS。（a）二次问题；（b）逻辑回归问题。

6 结论

本文研究了基数约束多目标优化问题。受稀疏单目标优化中同名条件的启发，我们定义了 MOO 的 L-平稳性概念，并分析了它与主要帕累托最优性概念和条件之间的关系。

随后，我们针对所考虑的问题类别提出了两种新算法。第一种是将迭代硬阈值 方法 [1] 扩展到多目标优化情形，称为 MOIHT：与原方法一样，它旨在生成一个 L-驻点。第二种算法称为稀疏前沿最速下降 （SFSD）。据我们所知，这是首个用于基数约束多目标优化的前沿导向方法。作为前沿最速算法 [17, 33], 的一种改编，SFSD 旨在逼近所研究问题的帕累托前沿（通常不规则且碎片化）。该方法依赖适当的初始化策略，例如采用 MOIHT 或 MOSPD [31] 算法进行多起点搜索、将两者混合化，或采用标量化方法。从理论角度来看，我们证明了对于 MOIHT，点序列收敛到 L-驻定解；而对于 SFSD，我们则陈述了其全局收敛到满足多目标 Lu-Zhang 最优性条件的点。

通过数值实验，我们还评估了所提出的方法在二次问题和逻辑回归问题基准测试中的表现。结果表明，SFSD 方法能够全面覆盖帕累托前沿，而多起点混合 MOSPD-MOIHT 过程是最有希望用于算法第一阶段的方案。

至于未来工作，我们可能会将理论结果扩展到处理基数约束以外的其他约束。此外，进一步的研究可能会集中于算法在更广泛问题集上的性能评估，这些问题具有更一般且可能非凸的目标函数。

附录 A 引理 2 的证明

证明。Si nce $\bar{x}$ 是问题(1)的Pareto平稳点，根据定义3，我们有 $\theta(\bar{x}) = 0$，即 ,

$$\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top d + \frac{1}{2}\|d\|^2 \geq 0, \qquad \forall d \in \mathcal{D}(\bar{x}). \tag{A1}$$

ˆ 反设存在一个方向 $d \in \mathcal{D}(\bar{x})$，使得

$$\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d} < 0. \tag{A2}$$

由于 (A1) 成立，我们由此推得 $\frac{1}{2}\|\hat{d}\|^2 \geq |\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d}| > 0$. 现在，我们引入函数 $\tilde{\theta}:\mathbb{R}^n\times\mathbb{R}^n\times[0,1]\to\mathbb{R}$，定义为 $\tilde{\theta}(x,d,t) = \max_{j\in\{1,\dots,m\}} \nabla f_j(x)^\top(td) + \frac{1}{2}\|td\|^2 = t\max_{j\in\{1,\dots,m\}} \nabla f_j(x)^\top d + \frac{t^2}{2}\|d\|^2$.ˆ ˆ ˜ˆ 根据 (4) 以及 d的可行性可知，对于所有 $t\in[0,1]$， $td \in \mathcal{D}(\bar{x})$ 和 $\theta(\bar{x})$，均有 $\leq \theta(\bar{x},d,t)$。容易看出，如果 $0 < t < -(2/\|\hat{d}\|^2)\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top \hat{d}$，则 $\tilde{\theta}(\bar{x},\hat{d},t) < 0$；其中，右侧是正值，这是由˜ ˆ 式 (A2) 得出的。但在这种情况下，我们将得到 $\theta(\bar{x}) \leq \theta(\bar{x},d,t) < 0$，这与 $\bar{x}$ 的帕累托驻点性矛盾。因此，我们证明了：若 $\bar{x}$ 是问题 (1) 的帕累托驻点，则 $\max_{j\in\{1,\dots,m\}} \nabla f_j(\bar{x})^\top d \geq 0$, $\forall d \in \mathcal{D}(\bar{x})$。从这里开始，我们可以沿用 [31] 中命题 3 的证明。

□

利益冲突

作者声明不存在利益冲突。

经费来源

本研究未获得公共部门、商业部门或非营利部门资助机构的任何专项资助。

数据可用性声明

本研究分析的所有数据集均可在 UCI 机器学习资料库 [23].https://archive.ics.uci.edu/ml/datasets.php 获取。

参考文献

[1] A. Beck 和 Y. C. Eldar. 稀疏性约束非线性优化：最优性条件与算法。*SIAM Journal on Optimization*, 23(3):1480–1509, 2013.[2] A. Beck 和 N. Hallak. 关于稀疏对称集上的最小化：投影、最优性条件与算法。*Mathematics of Operations Research*, 41(1):196–223, 2016.[3] D. P. Bertsekas. 非线性规划. Athena Scientific, Belmont, MA, 第2版, 1999.[4] D. Bertsimas, A. Delarue, P. Jaillet 和 S. Martin. 可解释性的代价。*arXiv* 预印本 *arXiv:1907.03419*, 2019.[5] D. Bertsimas 和 A. King. 逻辑回归：从艺术走向科学。*Statistical Science*, 32(3):367–384, 2017.[6] D. Bertsimas, A. King 和 R. Mazumder. 以现代优化视角进行最佳子集选择。*The Annals ofStatistics*, 44(2):813 – 852, 2016.[7] D. Bertsimas, J. Pauphilet 和 B. Van Parys. 稀疏回归：可扩展算法与实证性能。*Statistical Science*, 35(4):555–578, 2020.[8] D. Bienstock. 一类混合整数二次规划问题的计算研究。*MathematicalProgramming*, 74(2):121–140, 1996年8月.[9] H. Bonnel, A. N. Iusem 和 B. F. Svaiter. 向量优化中的近端方法。*SIAM Journal on Optimization*, 15(4):953–970, 2005.

[10] O. P. Burdakov, C. Kanzow 和 A. Schwartz. 带基数约束的数学规划：通过互补型条件进行重构及一种正则化方法。SIAM Journal on Optimization, 26(1):397–425, 2016.[11] E. F. Campana, M. Diez, G. Liuzzi, S. Lucidi, R. Pellegrini, V. Piccialli, F. Rinaldi 和 A. Serani. 用于船体优化的多目标直接算法。Computational Optimization and Applications, 71(1):53–72, 2018年9月.[12] M. A. Carreira-Perpiñán 和 Y. Idelbayev. 用于神经网络剪枝的“学习-压缩”算法。载于 IEEE 计算机视觉与模式识别会议（CVPR）论文集，2018年6月.[13] E. Carrizosa 和 J. B. G. Frenk. 凸函数的支配集及其若干应用。Journal of Optimization Theory and Applications, 96(2):281–295, 1998年2月.[14] E. Civitelli, M. Lapucci, F. Schoen 和 A. Sortino. 一种基于信息准则的逻辑回归特征子集选择有效方法。Computational Optimization and Applications, 80(1):1–32, 2021年9月.[15] G. Cocchi 和 M. Lapucci. 一种用于多目标优化的增广拉格朗日算法。Computational Optimization and Applications, 77(1):29–56, 2020年9月.[16] G. Cocchi, M. Lapucci 和 P. Mansueto. 通过多目标增广拉格朗日方法进行帕累托前沿近似。EURO Journal on Computational Optimization, 9:100008, 2021.[17] G. Cocchi, G. Liuzzi, S. Lucidi 和 M. Sciandrone. 多目标优化中最速下降方法的收敛性。Computational Optimization and Applications, 77(1):1–27, 2020年9月.[18] G. Cocchi, G. Liuzzi, A. Papini 和 M. Sciandrone. 一种用于带盒约束的无导数多目标优化的隐式过滤算法。Computational Optimization and Applications, 69(2):267–296, 2018年3月.[19] A. L. Custódio, J. F. A. Madeira, A. I. F. Vaz 和 L. N. Vicente. 用于多目标优化的直接多重搜索。SIAM Journal on Optimization, 21(3):1109–1140, 2011.[20] K. Deb, A. Pratap, S. Agarwal 和 T. Meyarivan. 一种快速且精英化的多目标遗传算法：NSGA-II。IEEE Transactions on Evolutionary Computation, 6(2):182–197, 2002.[21] E. D. Dolan 和 J. J. Moré. 使用性能剖面评测优化软件。Mathematical Programming, 91(2):201–213, 2002.[22] L. G. Drummond 和 A. N. Iusem. 一种用于向量优化问题的投影梯度方法。Computational Optimization and Applications, 28(1):5–29, 2004年4月.[23] D. Dua 和 C. Graff. UCI 机器学习库，2017.[24] G. Eichfelder. 一种多目标优化中的自适应标量化方法。SIAM Journal on Optimization, 19(4):1694–1718, 2009.[25] J. Fliege. Olaf——一种用于评估和优化空气污染设施选址的通用建模系统。OR-Spektrum, 23(1):117–136, 2001年2月.[26] J. Fliege, L. M. G. Drummond 和 B. F. Svaiter. 多目标优化中的牛顿法。SIAM Journal on Optimization, 20(2):602–626, 2009.[27] J. Fliege 和 B. F. Svaiter. 多准则优化中的最速下降方法。Mathematical Methods of Operations Research, 51(3):479–494, 2000年8月.[28] R. Garmanjani, E. Krulikovski 和 A. Ramos. 带基数约束的多目标优化问题的驻点条件与约束资格条件，2022.[29] Y. Jin 和 B. Sendhoff. 基于帕累托的多目标机器学习：综述与案例研究。IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 38(3):397–415, 2008.[30] C. Kanzow 和 M. Lapucci. 带几何约束优化问题的非精确罚函数分解方法。Computational Optimization and Applications, 0(0):1–35, 2023.[31] M. Lapucci. 一种用于多目标基数约束优化问题的罚函数分解方法。Optimization Methods and Software, 0(0):1–33, 2022.[32] M. Lapucci, T. Levato, F. Rinaldi 和 M. Sciandrone. 稀疏性约束优化的统一框架。arXiv 预印本 arXiv:2104.13244, 2021.[33] M. Lapucci 和 P. Mansueto. 用于多目标优化的改进型前沿最速下降法。Operations Research Letters, 2023.

[34] M. Lapucci 和 P. Mansueto。面向多目标优化的有限内存拟牛顿方法。计算优化与应用，2023年3月。[35] M. Lapucci、P. Mansueto 和 F. Schoen。用于全局多目标优化的模因算法。数学规划计算，2022年11月。[36] M. Laumanns、L. Thiele、K. Deb 和 E. Zitzler。在进化多目标优化中结合收敛性与多样性。进化计算，10(3):263–282，2002年。[37] G. Liuzzi、S. Lucidi 和 F. Rinaldi。一种用于约束多目标非光滑优化的无导数方法。*SIAM* 优化期刊，26(4):2744–2774，2016年。[38] M. Locatelli、M. Maischberger 和 F. Schoen。基于局部搜索的差分进化方法。计算机与运筹研究，43:169–180，2014年。[39] Z. Lu 和 Y. Zhang。通过惩罚分解方法进行稀疏近似。*SIAM* 优化期刊，23(4):2448–2478，2013年。[40] P. Mansueto 和 F. Schoen。用于聚类问题的模因差分进化方法。模式识别，114:107849，2021年。[41] B. K. Natarajan。线性系统的稀疏近似解。*SIAM* 计算期刊，24(2):227–234，1995年。[42] A. Pascoletti 和 P. Serafini。向量优化问题的标量化。优化理论与应用期刊，42(4):499–524，1984年4月。[43] R. Pellegrini、E. Campana、M. Diez、A. Serani、F. Rinaldi、G. Fasano、U. Iemma、G. Liuzzi、S. Lucidi 和 F. Stern。将无导数多目标算法应用于真实海洋环境中的高速双体船可靠性鲁棒设计优化1。优化工程 *IV-Rodrigues* 等（编），第15页，2014年。[44] H. Tanabe、E. H. Fukuda 和 N. Yamashita。用于多目标优化及其应用的近端梯度方法。计算优化与应用，72(2):339–361，2019年3月。[45] M. Tavana。使用多准则决策方法对 NASA 火星载人探索的备选任务架构进行主观评估。计算机与运筹研究，31(7):1147–1164，2004年。[46] A. M. Tillmann、D. Bienstock、A. Lodi 和 A. Schwartz。基数最小化、约束与正则化：综述。*arXiv* 预印本 *arXiv:2106.09606*，2021年。[47] J. Weston、A. Elisseeff、B. Schölkopf 和 M. Tipping。零范数在线性模型和核方法中的应用。机器学习研究期刊，3:1439–1461，2003年。[48] E. Zitzler 和 L. Thiele。使用进化算法进行多目标优化——一项比较案例研究。载于 A. E. Eiben、T. Bäck、M. Schoenauer 和 H.-P. Schwefel（编），并行问题求解（自然）——*PPSN V*，第292–301页，柏林、海德堡，1998年。Springer Berlin Heidelberg。