arXiv:2304.00479v2 [math.OC] 4 Apr 2023

广义次模优化的混合整数规划方法及其应用

SimgeK¨u.c¨ukyavuz Qimeng Yu

美国伊利诺伊州埃文斯顿，西北大学工业工程与管理科学系{
simge@northwestern.edu, kim.yu@u.northwestern.edu}

2023年4月6日

摘要。 子模性是整数优化和组合优化中的一个重要概念。经典的次模集合函数刻画了从单个基集中选择同质物品所产生的效用，这类选择可以用二元变量表示。在实践中，许多问题场景涉及从多个基集中选择异质物品，或选择多个同质物品副本，因此需要对次模性进行扩展。我们将与这些广义次模概念相关的优化问题称为广义次模优化（GSO）。GSO广泛应用于基础设施设计、医疗保健、在线营销和机器学习等领域。由于目标函数通常高度非线性（甚至非凸且非凹），且决策空间为混合整数形式，GSO是极具挑战性的混合整数非线性规划问题中的一个广泛子类。本教程首先概述经典子模性，然后介绍GSO的两个子类，并针对这些问题类别所产生的混合整数集合结构，阐述其多面体理论。我们的理论结果催生了高效且通用的精确求解方法，并通过真实世界数据集展示了这些方法在实际问题中的有效性。

关键词 混合整数规划；广义子模性；多面体研究；精确方法

1 引言

复杂系统中的决策通常涉及从给定集合中选择元素，以获得最优效用。例如，在传感器布设中，我们关注的是选择能够实现最大覆盖的传感器位置。在这类决策问题中，我们经常观察到边际收益递减现象。也就是说，已选择的项越多，任何额外元素对总效用的边际贡献就越低。在传感器布设的例子中，传感器布置方案的总覆盖通常呈现边际收益递减：随着安装的传感器越来越多，每新增一个传感器所带来的额外覆盖就越少。边际收益递减 现象在实际应用中无处不在。规模经济、风险厌恶、聚类、覆盖和影响传播都具有边际收益递减性质。

从数学上说，子模性 这一概念对边际收益递减进行了形式化。子模性是集合函数的一种性质，即定义在集合上的函数。由于边际收益递减现象十分普遍，次模集合函数被用于在丰富多样的应用中对效用进行建模，例如设施选址 [19], 图像分割[40], 网络影响传播 [41, 82], 和传感器布设 [44]。相关决策问题被建模为以次模集合函数为目标函数的优化问题，由此形成了次模集合函数优化 （简称次模优化）。在次模优化中，给定一批同质物品（例如传感器布设中的传感器）和一个有限底集（例如候选传感器布设位置）。我们可以将底集视为一组“空箱”，目标是从中确定一个最佳子集，用来放置所提供的同质物品。是否在某个“箱”中放置一个元素，自然可以用一个二元决策变量来建模，因此次模集合函数优化属于离散优化问题。图1以示意图的形式对次模优化进行了高层次概述。

尽管次模集合函数具有广泛的建模能力，实际应用日益需要能够刻画边际收益递减的更通用建模工具。考虑图2所示的两种情形。在左侧的第一种情形中，我们得到了一批 异质的 项。此时，我们的任务不仅是确定要填充的最佳“箱”子集，还要确定适用于所选“箱”的最佳项类型。这就需要子模性的多重集扩展。在右侧的第二种情形中，我们得到一个有限底集，其中包含

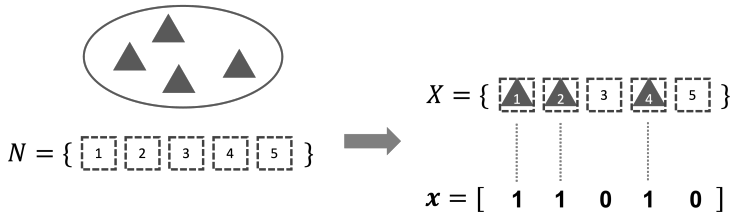

图1：次模集合函数优化的示意图。三角形表示给定的同质物品， N 是一个有限底集。子集 X 表示将物品放入第一个、第二个和第四个“箱”的决策。稍微滥用一下记号，向量$\mathbf{x}$ 以二进制编码的形式表示与 X相同的决策，其中 1 表示放置物品的位置，0 表示其他位置。

“箱”和一组同质的项。不过，现在我们可以在每个“箱”中放入多个，甚至是分数个项。这种情形催生了子模性的混合整数扩展。我们用统一术语广义子模性 (GS)来指称经典次模性及其所有扩展。我们

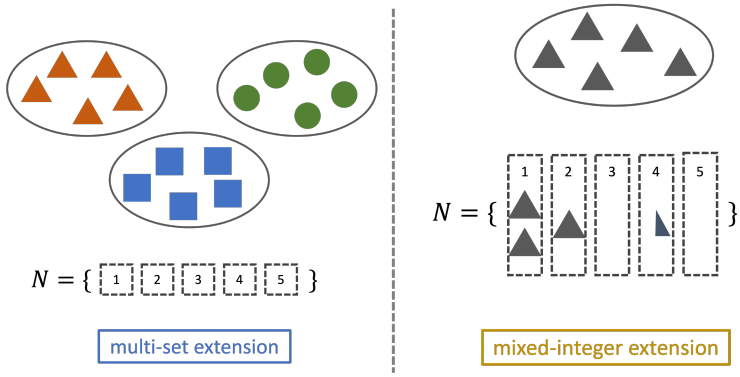

图2：需要推广经典次模性的两种情形。左侧第一种情形需要多重集扩展；右侧第二种情形则引出了混合整数扩展。

总结 t 以下是两个 GS 示例。更多应用将在第4节中详细介绍 5.

多类型传感器布置。实际上，我们可能需要使用多种类型的传感器来监测环境的不同方面（例如温度、湿度和亮度），并且每个候选传感器位置只能放置一个传感器。现在，我们需要决定在哪些位置布设传感器，并进一步决定在每个选定位置安装哪种类型的传感器。子模性的多集合扩展是对这个多类型传感器布置问题进行建模的理想工具。

传感器能量管理。在另一种传感器布设应用中，给定一组监测相同环境因素、但能量水平可调的传感器。能量水平可以是连续范围，也可以是离散刻度盘，我们的目标是设定能实现最优效用的能量水平。此问题属于需要采用混合整数扩展的第二种情形。

本教程介绍广义次模函数（GS）的相关概念，并通过实际应用展示其建模能力。此外，我们还回顾基于混合整数规划（多面体）的方法，这些方法可在有约束或无约束的情况下，将广义次模函数优化至全局最优。多面体方法在求解带复杂约束的经典次模优化问题 [2, 18, 21, 68, 81, 86, 87, 91] 以及一般集合函数问题 [9]并获得全局最优解方面，已取得显著成功。此外，研究人员发现并利用了子模性，以改进混合二元凸二次优化问题和锥优化问题的建模形式 [5, 6, 7, 8, 26, 43]。这些方法在涉及次模结构的随机问题和风险规避问题中也取得了良好成效 [42, 67, 82, 83, 84, 85, 93]。我们将重点探讨如何把这些方法拓展应用于广义次模函数。

本教程其余部分的安排如下。我们首先在第 2 节介绍经典次模性的预备知识。接着在第 3 节说明如何利用经典次模性，帮助求解一类约束次模集合函数最小化问题。在第 4 节和 5节，我们正式阐述次模性的两种推广，并概述其应用。我们将介绍与相关问题对应的

关键混合整数集的多面体结果，并据此介绍这些优化问题的精确求解方法。本教程最后以第 6 节作结。

2 预备知识

本节概述经典次模性。设 $N = \{1, 2, \ldots, n\}$ 为一个有限非空基集， 2^N 为 N的所有子集构成的集合。次模集合函数的形式化定义如下。

定义 2.1 函数 $f\colon 2^N \to \mathbb{R}$ 称为次模的，如果

$$f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y) \tag{1}$$

对于任意 X、$Y \subseteq N$。

函数 f称为 超模的，如果不等式（1）对所有 X、$Y \subseteq N$反向成立；等价地， $-f$是次模的。此外，函数 f是 单调非递减的，简称 单调，如果 $f(X) \leq f(Y)$对所有 $X \subseteq Y$均成立。对于任意子集 $X \subset N$ 以及任意元素 $i \in N$，定义将 边际收益 的 i 添加到 X 中的收益为

$$\rho_i(X) := f(X \cup \{i\}) - f(X). \tag{2}$$

直观而言，次模性等价于边际收益递减。下面给出的次模集合函数另一种定义清楚地体现了这一点。

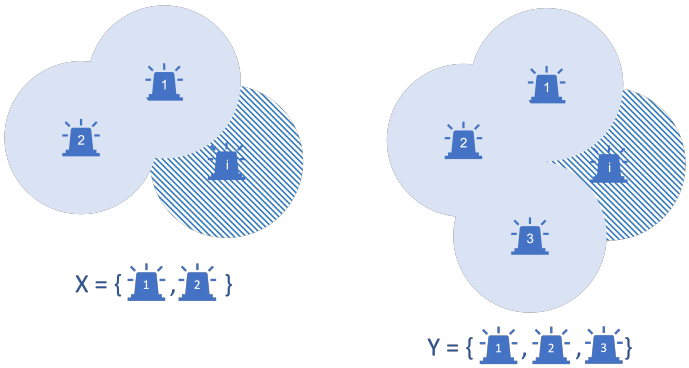

图 3：传感器布设中边际收益递减的示意图。每个圆圈表示一个传感器的覆盖范围。集合 X 和 Y 是我们已安装传感器的两个位置集合。实心区域分别对应布设方案 X 和 Y 的总覆盖范围。条纹区域突出显示了在位置 i 安装一个额外传感器对这两个方案的边际贡献。已安装的传感器越多，再安装更多传感器所能获得的额外覆盖范围就越小。

定义 2.2 函数 $f\colon 2^N \to \mathbb{R}$ 是次模 的，当且仅当

$$\rho_i(X) \geq \rho_i(Y) \tag{3}$$

对于任意 $X \subseteq Y \subseteq N$ 和任意 $i \in N \backslash Y$。

根据定义 2.2，元素 i 的边际收益会随着选择从 X 转变为包含更多元素的选择 Y而递减。图 3 展示了传感器布设中的边际收益递减现象。我们可以利用以下引理判断一个集合函数是否为次模且单调非递减。

引理 2.3 *[81]* 函数 $f\colon 2^N \to \mathbb{R}$ 是次模且单调的，当且仅当对所有 $X, Y \subset N$，

$$f(Y) \leq f(X) + \sum_{i \in Y \backslash X} \rho_i(X).$$

次模集合函数的示例。接下来，我们介绍几个在组合优化中经常出现的次模集合函数示例。回顾一下，$N = \{1, 2, \ldots, n\}$ 表示一个有限且非空的基集。我们考虑以下集合函数 $f: 2^N \to \mathbb{R}$。

- 函数 $f(X) = \sum_{i \in X} a_i$ 是线性的，其中 $\mathbf{a} \in \mathbb{R}^n$，它既是次模的，也是超模的。因此，我们称$f$ 是模的。如果对所有 $i \in N$ 都有 $a_i \geq 0$，那么 f 也是单调的。
- 假设给定一个元素集合 $\mathcal{C}$，以及这些元素的 n 个子集 $A_1, \ldots, A_n \subset \mathcal{C}$。集合覆盖函数 $f(X) = |\bigcup_{i \in X} A_i|$ 是次模的。
- 设 $\mathcal{G} = (N, \mathcal{A})$ 是一个有向图，其顶点集为 N，弧集为 $\mathcal{A}$。此外，每条弧 $(i, j) \in \mathcal{A}$ 的容量为 $c_{ij} \geq 0$。图割 函数 $f(X) = \sum_{(i,j) \in \mathcal{A}: i \in X, j \in N \setminus X} c_{ij}$ 是次模的，但不是单调的。

次模集合函数的运算。以下列出了一些能够保持子模性的运算。注意，下面的函数 $h: 2^N \to \mathbb{R}$ 是定义在 $N = \{1, 2, \ldots, n\}$ 上的集合函数。

- 设 $f: 2^N \to \mathbb{R}$ 是一个子模函数。对于任意常数 $c \in \mathbb{R}$，函数 $h(X) := f(X) - c$ 对所有 $X \subseteq N$ 均成立，且为子模函数。
- 假设 f，$g: 2^N \to \mathbb{R}$ 是两个子模函数。那么对于任意非负常数 α，$\beta \in \mathbb{R}_+$，函数 $h(X) := \alpha f(X) + \beta g(X)$ 对所有 $X \subseteq N$ 均成立，且为子模函数。更一般地，子模函数的任意锥组合都是子模函数。
- 设 $f: \mathbb{R} \to \mathbb{R}$ 是一个凹函数，且 $g: 2^N \to \mathbb{R}$ 是一个非负线性函数。复合函数$h(X) := f(g(X))$ 对所有 $X \subseteq N$ 均成立，且为子模函数。
- 设 f，$g: 2^N \to \mathbb{R}$ 是两个次模集合函数，且 $f - g$ 是单调非递减或单调非递增的。最小值 $h(X) := \min\{f(X), g(X)\}$ 是次模的。

我们给出上述定义和性质，以帮助读者识别应用中的子模性。为说明如何利用子模性在这些应用中获得最优决策，我们进一步总结次模集合函数优化的若干关键结果。

在接下来的讨论中，我们假设 $f(\emptyset) = 0$。这一假设不失一般性，因为否则我们可以等价地考虑 $f'(X) := f(X) - f(\emptyset)$，对所有 $X \subseteq N$均成立。若 f是次模的和/或单调的，则 f'也具有相同性质。我们还假设可以通过值预言机显式获得 f——给定任意 $X \subseteq N$，返回 $f(X)$ 所需的时间关于 n是多项式的。稍微滥用记号，我们将 $X \subseteq N$ 及其二进制表示视为可互换。准确地说，任意 $X \subseteq N$ 都唯一对应于一个二进制向量 $\mathbf{x} \in \{0, 1\}^n$，其中

$$x_i = \begin{cases} 1, & \text{if } i \in X, \\ 0, & \text{otherwise.} \end{cases}$$

现在，集合函数 $f: 2^N \to \mathbb{R}$ 可以重写为定义在二进制向量上的函数，$f: \{0, 1\}^n \to \mathbb{R}$。

2.1 次模集合函数最小化

我们首先考虑无约束 最小化问题

$$\min_{X \subseteq N} f(X), \tag{4}$$

其中 f 是一个次模集合函数。众所周知，这类优化问题是强多项式时间可解的 [20, 27, 35, 36, 37, 47, 49, 57, 65]。事实上，f 还可以通过 *Lov´asz* 扩展进一步推广到连续域 $[0, 1]^n$。[48]。

定义 2.4 设给定任意函数 $f: \{0, 1\}^n \to \mathbb{R}$。$f$ 的Lov´asz 扩展 $f^L: [0, 1]^n \to \mathbb{R}$定义为

$$f^L(\mathbf{x}) = \sum_{i=1}^{n} (x_{\delta(i)} - x_{\delta(i+1)}) f\left(X^{\boldsymbol{\delta}, i}\right), \tag{5}$$

对任意$\mathbf{x} \in [0, 1]^n$均成立。这里，$\boldsymbol{\delta} = (\delta(1)\ \delta(2) \ldots, \delta(n))$ 是 $N = \{1, 2, \ldots, n\}$的一个排列，使得 $x_{\delta(1)} \geq x_{\delta(2)} \geq \cdots \geq x_{\delta(n)}$。令 $x_{\delta(n+1)} = 0$，且 $X^{\boldsymbol{\delta}, i} := \{\delta(1) \ldots, \delta(i)\}$ 对任意 $i \in N$ 均成立。

注意到 (5) 等价于

$$f^L(\mathbf{x}) = \sum_{i=1}^{n} \left[f\left(X^{\boldsymbol{\delta},i}\right) - f\left(X^{\boldsymbol{\delta},i-1}\right) \right] x_{\delta(i)} = \sum_{i=1}^{n} \rho_{\delta(i)}\left(X^{\boldsymbol{\delta},i-1}\right) x_{\delta(i)},$$

其中， $X^{\boldsymbol{\delta},0} = \emptyset$和 $\rho_\cdot(\cdot)$ 的定义见（2）。尽管 f^L 看起来是线性的，但由于排列$\boldsymbol{\delta}$ 取决于$\mathbf{x}$，实际情况未必如此。不过，借助高效的值预言机，可以轻松计算 Lov´asz 扩展。当 f 为次模函数时， f^L 具有以下理想性质。

命题 2。5 *[48]* 函数 f 的 *Lov´asz* 扩展与其凸闭包 f^c 重合，当且仅当 f 是次模集合函数，其中

$$f^c(\mathbf{x}) = \min\left\{ \sum_{S\subseteq N} \alpha_S f(S) : \sum_{S\subseteq N} \alpha_S \mathbf{1}_S = \mathbf{x}, \sum_{S\subseteq N} \alpha_S = 1, \alpha_S \geq 0, \forall\, S \subseteq N \right\},$$

且 $\mathbf{1}_S$ 是一个二进制向量，在 $i \in S$ 处取值为 *1*，在所有其他分量处取值为 *0*。

换言之，对次模函数 f 的凸闭包进行最小化，可得到二进制最优解。因此，无约束次模集合函数最小化等价于最小化一个凸函数，这表明该问题可以高效求解。

无约束次模集合函数最小化的复杂度也可由优化与分离在多项式时间上的等价性推出。接下来，我们从多面体视角讨论这一类最小化问题。令

$$\mathcal{Q}_f := \{(\mathbf{x}, w) \in \{0,1\}^n \times \mathbb{R} : w \geq f(\mathbf{x})\}$$

是 f 的上图。问题（4）可等价地重写为:

$$\min\{w : (\mathbf{x}, w) \in \operatorname{conv}(\mathcal{Q}_f)\}. \tag{6}$$

上图的凸包 conv $(\mathcal{Q}_f)$ 由平凡不等式 $\mathbf{0} \leq \mathbf{x} \leq \mathbf{1}$ 以及所有扩展拟阵多面体不等式 （EPI） [8, 9, 48]描述。每个 EPI 都与 N 的一个排列相关联，定义如下。

定义 2.6 给定 N 的任意排列 $\boldsymbol{\delta}$ ，相应的扩展拟阵多面体不等式（EPI）为

$$w \geq \sum_{i=1}^{n} \rho_{\delta(i)}\left(X^{\boldsymbol{\delta},i-1}\right) x_{\delta(i)}, \tag{7}$$

其中 $X^{\boldsymbol{\delta},\cdot} = \{\delta(1)\ldots,\delta(i)\}$ 对任意 $i \in N$， $X^{\boldsymbol{\delta},0} = \emptyset$，且 $\rho_\cdot(\cdot)$ 如 (2)中所定义 .

公式（$\mathcal{Q}_f$）中的凸包 conv 6由所有 EPI 以及 $\mathbf{0} \leq \mathbf{x} \leq \mathbf{1}$定义，因此该公式是一个具有指数级数量约束的线性规划。

延迟约束生成框架。 对于约束数量庞大的优化问题，自然可以采用 延迟约束生成（*DCG*）框架。以重写式（6）为例。我们不将这个包含指数级数量 EPI 的完整主问题提交给求解器，而是迭代求解松弛主问题，这些问题包含较少的约束，并在每次迭代中添加必要的约束。松弛主问题的形式为

$$\begin{aligned} \min \quad & w \\ \text{s.t.} \quad & (\mathbf{x}, w) \in \mathcal{C}, \end{aligned} \tag{8}$$

其中， $\mathcal{C}$ 仅由问题 (6) 中定义 conv $(\mathcal{Q}_f)$ 的约束的一个子集构造。算法 1 概述了 DCG 算法。项 (UB − LB)/UB 衡量最优性间隙，其中 LB 和 UB 分别表示目标值的下界和上界。在该算法中，只要最优性间隙超过预先设定的容差参数 ϵ，我们就求解松弛主问题 (8)，以获得其最优解及目标值对 (x w)。由于 (8) 是原问题 (6) 的松弛版本， w 构成最优目标值的下界。函数值 f(x) 是一个上界，因为 x 是可行的。如果 w 与f(x)不匹配，我们就在 $\mathcal{C}$ 中加入一个被违反的有效不等式（第 7 行）。确定此类不等式的问题称为分离问题。理想情况下，我们希望找到违反程度最大的有效不等式，它是最有力的

约束，可在 DCG 的下一次迭代中将 $(\mathbf{x}\ w)$ 排除在外。此外，通过使违反程度最大化，我们还能够判断是否不存在违反。我们求解更新后的松弛主问题，并重复此过程，直到最优性间隙足够小。

算法 1： 延迟约束生成

输入 初始 $\mathcal{C}$、LB $= -\infty$、UB $= \infty$;
当 (UB $-$LB)/UB $> \epsilon$ **时**
  求解当前松弛主问题（8），得到（$\mathbf{x}\ w$）；
  LB $\leftarrow \overline{w}$;
  计算 $f(\mathbf{x})$;
  如果 $w < f(\mathbf{x})$ **则**
    确定一个被违反的有效不等式 $w \geq \pi^\top x$ 并将其添加到更新 $\mathcal{C}$中;
  结束如果
  如果 上界 $> f(\mathbf{x})$ **则**
    上界 $\leftarrow f(\overline{\mathbf{x}})$；
  结束如果
  将当前最优解更新为$\mathbf{x}$;
</style)结束循环
输出 w，$\overline{\mathbf{x}}$。

如椭球法所示，非正式地说，如果可以在多项式时间内解决 conv $(\{\boldsymbol{c}^\top \mathbf{x} : \mathbf{x} \in \mathcal{X}\}$ 的分离问题，那么优化问题 min$\mathcal{X})$ 就可以在多项式时间内求解。问题 $(\mathcal{Q}_f)$ 中 conv (6) 的分离问题恰好是“容易”的。对于任意 $\mathbf{x} \in [0,1]^n$，都可以通过 $\mathcal{O}(n\log\ n$ 贪心算法找到最违反 EPI 的约束，该算法由 [21]提出。

与无约束情形不同，约束次模最小化（例如，带有基数约束）通常是 NP难的 [76]。当问题满足某些特殊性质时，也存在例外，例如，在基数约束下最小化凹函数与非负仿射函数的复合 [31, 56]。第3节讨论针对这一特定问题类别的混合整数规划方法。为了获得一般约束次模最小化问题的精确最优解，我们可以在 DCG 框架中使用额外的线性可表示约束实现重构式（6）（例如，参见 [8] ）。

2.2 次模集合函数最大化

即使在无约束情形下，次模最大化也是 NP难问题。鉴于其复杂性，研究者提出了许多启发式算法和近似算法 [16, 46, 53, 58, 75]。特别是， [54] 提出了一种著名的（$1 - 1/e$）近似算法，用于求解基数约束最大化问题：

$$\max_{X:|X|\leq k} f(X),$$

当 f 为单调次模函数时。这里， e 是自然对数的底数。该算法采用贪心方法，依次选取边际收益最高的 k 个元素。

在许多应用中，尤其是涉及设施选址和传感器网络设计等高资本投入的应用，人们希望求得精确的最优解，而非采用次优解。这是因为精确的最优解能够显著提高效用并降低成本。在[81],中，作者研究了次模最大化问题，即

$$\max_{X\subseteq N} f(X), \tag{9}$$

采用整数规划方法。需要指出的是，次模目标函数 f 不必是单调的。令 $f^*(X) := f(X) - \sum_{j\in X} \rho_j(N \setminus \{j\})$ 对所有 $X \subseteq N$均成立。作者提出了以下等价的混合整数规划表述：

$$\max_{(\mathbf{x},w)\in\{0,1\}^n\times\mathbb{R}} \{w : (\mathbf{x}, w) \in \mathcal{Q}_f\}, \tag{10}$$

其中

$$\mathcal{Q}_f := \{(\mathbf{x}, w) \in \mathbb{R}^n_+ \times \mathbb{R} : w \leq f^*(S) + \sum_{j\in N\setminus S} [f^*(S \cup \{j\}) - f^*(S)]x_j + \sum_{j\in N} \rho_j(N \setminus \{j\})x_j, \forall\ S \subseteq N\}$$

是将 f 限制为二进制向量时的上图。每个不等式

$$w \leq f^*(S) + \sum_{j \in N \setminus S} [f^*(S \cup \{j\}) - f^*(S)]x_j + \sum_{j \in N} \rho_j(N \setminus \{j\})x_j \tag{11}$$

称为次模不等式与 $S \subseteq N$相关联。公式（10）包含指数级数量的约束，可通过 DCG 求解。混合整数规划方法能够处理额外的约束，并且公式（10）已针对约束次模最大化进一步加强，见 [2, 68, 86]。

3 基数约束次模最小化

为说明如何利用子模性促进决策，以及如何应用混合整数规划方法，本节聚焦于一类约束次模集合函数最小化问题。

如上一节所述，单变量凹函数与非负线性函数的复合是一个次模集合函数。也就是说，对于任意$\mathbf{a} \in \mathbb{R}^n_+$以及任意凹函数 f: $\mathbb{R} \to \mathbb{R}$,$F\ X\ = f\left(\sum_{i \in X} a_i\right)$ $X \subseteq N$ F $F\ = f\ \sum_{i=1}^n a_i x_i$ ( ): 对所有，均为次模函数。或者，也可以写成（$\mathbf{x}$）( )，对所有$\mathbf{x} \in \{0,1\}^n$均成立。

此类函数常用于对涉及风险厌恶或规模经济的问题中的效用进行建模，包括均值-风险优化 [7, 8] 和凹成本设施选址 [23, 28]。均值-风险优化是金融应用中的一种强大建模工具，包括投资组合管理。在这些应用中，我们再次观察到边际收益递减。例如，在投资组合管理中，基于边际收益递减的直觉，可以使用上述函数对投资者的风险感知进行建模：随着投资者的投资组合变得更加激进，她往往会觉得任何额外金融产品所带来的风险更温和。所选金融产品的数量受基数上界 k限制。因此，考虑以下问题具有实际意义：

$$\min \left\{ f\left(\sum_{i=1}^n a_i x_i\right) : \mathbf{x} \in \{0,1\}^n, \sum_{i=1}^n x_i \leq k \right\}. \tag{12}$$

问题（12）可改写为以下形式：

$$\min \{w : (\mathbf{x}, w) \in \mathcal{P}_k^m\},$$

其中

$$\mathcal{P}_k^m = \left\{ (\mathbf{x}, w) \in \times\{0,1\}^n \times \mathbb{R} : w \geq f\left(\sum_{i=1}^n a_i x_i\right), \sum_{i=1}^n x_i \leq k \right\} \tag{13}$$

表示 f在基数约束下的上图。上标 $m \in \{1, 2, \ldots, n\}$ 表示 $\mathbf{a}$中不同值的数量。问题（12）可在多项式时间内求解 [31, 56], ，尽管一般的约束次模最小化是 NP难问题。这一复杂性结果表明，完整刻画 $\mathcal{P}_k^m$的凸包或许是可行的。这样的刻画值得研究，因为它能导出一种灵活的分支切割法，以处理基数约束之外的更多约束。然而，如何可处理地刻画 conv（$\mathcal{P}_k^m$）仍是一个开放问题。

回顾第2 节，EPI（7）描述了不含基数约束的上图集凸包。文献 [87] 考虑了基数约束，并给出了 conv $\left(\mathcal{P}_k^1\right)$ 的完整描述（即，当$a_i = \alpha \geq 0$ 对所有 $i \in N$成立时）。作者提出了一类称为分离不等式（*SI*）的强有效线性不等式。

定义 3.1 对于 N 的任意排列 $\boldsymbol{\delta} = (\delta(1)\ \delta(2) \ldots, \delta(n))$ 以及固定整数参数 $0 \leq i_0 \leq k-1$，一个分离不等式（SI）定义如下：

$$w \geq \sum_{i=1}^{i_0} \rho_{\delta(i)} x_{\delta(i)} + \sum_{i=i_0+1}^{n} \psi x_{\delta(i)}, \tag{14}$$

其中 ψ: $= {}^{f\ \underline{(k\alpha)\ -f\ (i_0\alpha)\ k-i_0}}$，且 $\rho_{\delta(i)}$: $= f\ (i\alpha)\ - f\ (\ (i-1)\ \alpha)$ 。

作者证明，SI、基数约束和 $\mathbf{0} \leq \mathbf{x} \leq \mathbf{1}$ 完全刻画了 conv $(\mathcal{P}_k^1)$。此外，他们根据定义 3.2，针对任意排列 $\boldsymbol{\delta}$ 给出的 N，将 N 重新编号，使得 $\boldsymbol{\delta}$ 按自然顺序排列（$\mathbf{1}$，2，……，n）。因此，下文的记号中省略 $\boldsymbol{\delta}$。

定义 3.2 近似提升不等式（即 *ALIs*）形式如下：

$$w \geq \sum_{i=1}^{k} \rho_i x_i + \sum_{i=k+1}^{n} \phi_i x_i.$$

系数 ρ_i 对于 $i \in \{1, \ldots, k\}$ 与 (14) 中的系数一致。对于每个 $i > k$，令 T 满足 $|T| = k-1$ 为 $\{1, \ldots, i-1\}$ 的一个子集，使其权重之和尽可能大，并且 $\phi_i = f(a_i + \sum_{j \in T} a_j)$ − $f \sum_{j \in T} a_j$ ()。

针对一般情形（即任意 m），也有人提出了更强的有效不等式，但没有给出系数的显式形式。

在 [91], 中，我们进一步研究了 $\mathbf{a}$ 至多包含两个不同取值的情形，并展示了将其结果应用于一般情形（即任意 $m \geq 1$）的多种方法。当 $m = 2$时，我们分别用 a_L 和 a_H 表示 $\mathbf{a}$ 中的两个权重，其中 $a_L \leq a_H$。我们还定义 $\mathcal{I}_L = \{i \in N\colon\ a_i = a_L\}$ 和 $\mathcal{I}_H = \{i \in N\colon\ a_i = a_H\}$，分别作为 $\mathbf{a}$ 中对应较低值和较高值的指标集。接下来描述的结果适用于 N 的任意排列 $\boldsymbol{\delta}$。对基集 N 重新编号，使 $\boldsymbol{\delta}$ 按自然顺序排列，因此在记号中省略。考虑任意一个基数为 k 的子集 $S \subseteq N$。不失一般性，通过重新编号，假设 $S = \{1, \ldots, k\}$。

现在我们分享两个重要的观测值。首先，考虑集合

$$\mathcal{P}_k^2(S) = \left\{ (\mathbf{x}, w) \in \{0,1\}^k \times \mathbb{R} : w \geq f\left(\sum_{i=1}^{k} a_i x_i \right) \right\},$$

其恰好是 $\mathcal{P}_k^2$ 其中 x_i 对所有 $i \in N \setminus S$ 固定为零。基数约束显然成立，因为至多有 k 个变量可以取非零值。我们观察到，对于与 7）相关的每个 EPI（$w \geq \sum_{i=1}^{k} \rho_i x_i$ 的任意排序，S 都是 $\mathcal{P}_k^2$（S）的刻面定义不等式（即最强的有效不等式之一）。接下来，考虑集合 $\mathcal{P}_k^1$（$\mathcal{I}_L$）和 $\mathcal{P}_k^1$（$\mathcal{I}_H$），其中

$$\mathcal{P}_k^1(\mathcal{I}_L) = \left\{ (w, x) \in \mathbb{R} \times \{0,1\}^{\mathcal{I}_L} : w \geq f\left(a_L \sum_{i \in \mathcal{I}_L} x_i \right), \sum_{i \in \mathcal{I}_L} x_i \leq k \right\}, \tag{15}$$

and

$$\mathcal{P}_k^1(\mathcal{I}_H) = \left\{ (w, x) \in \mathbb{R} \times \{0,1\}^{\mathcal{I}_H} : w \geq f\left(a_H \sum_{i \in \mathcal{I}_H} x_i \right), \sum_{i \in \mathcal{I}_H} x_i \leq k \right\}. \tag{16}$$

在 $\mathcal{P}_k^1(\mathcal{I}_L)$ 中，所有变量 x_i 以及 $i \in \mathcal{I}_H$ 都固定为零，而 $\mathcal{P}_k^1(\mathcal{I}_H)$ 对 $i \in \mathcal{I}_L$ 施加相同的限制。我们注意到，SI 对这两个集合都是刻面定义的。这两个观测值意味着，我们可以通过精确提升获得 $\mathcal{P}_k^2$ 的刻面定义不等式。

提升是一种强化有效不等式、从而改进混合整数规划表述的技术。我们通过一个提升 EPI 的例子来说明这一技术。在提升之前，刻面定义不等式 $w \geq \sum_{i=1}^{k} \rho_i x_i$ 仅涉及 x_i 中的变量 $i \in S$。我们的目标是构造以下形式的最强不等式

$$w \geq \sum_{i=1}^{k} \rho_i x_i + \sum_{i=k+1}^{n} \xi_i x_i. \tag{17}$$

对于 $\mathcal{P}_k^2$ 有效。为实现这一目标，我们采用序列相关提升过程，逐步将变量 x_i、$i \in N \setminus S$ 以强系数加入该不等式。在每个中间步骤中，我们提升该不等式

，其中 x_j 对应 $j \in \{k+1, \ldots, n\}$ 通过求解 j-th 提升问题（18）。

$$\begin{aligned} \xi_j := \min \ & w - \sum_{i=1}^{k} \rho_i x_i - \sum_{i=k+1}^{j-1} \xi_i x_i \\ \text{s.t.} \ & w \geq f\left(a_j + \sum_{i=1}^{j-1} a_i x_i\right), \\ & \sum_{i=1}^{j-1} x_i \leq k-1, \\ & x \in \{0,1\}^{j-1}. \end{aligned} \tag{18}$$

最优目标值 ξ_j 恰好是多面体的凸包的刻面定义不等式 $w \geq \sum_{i=1}^{k} \rho_i x_i + \sum_{i=k+1}^{j} \xi_i x_i$ 中的提升系数

$$\mathcal{P}_k^2(\{1, \ldots, j\}) = \left\{(w, x) \in \mathbb{R} \times \{0,1\}^j : w \geq f\left(\sum_{i=1}^{j} a_i x_i\right), \sum_{i=1}^{j} x_i \leq k\right\}.$$

在 [91], 中，我们给出了 EPI 和 SI 提升问题的闭式最优解和目标值。我们提出了三类强有效不等式，分别称为提升的 EPI（LEPI）、低阶 SI（LSI）和高阶 SI（HSI）。有关这些不等式的具体形式，请参阅 [91] 。

示例 *3.3 LEPI* 与 *ALI* 的比较 *[91])* 设 $N = [6]$ 且 $k = 2$。给定一个 $= [4, 100, 100, 100, 4, 4]$ 和凹函数 $f(\mathbf{a}^\top \mathbf{x})$ $= \sqrt{\mathbf{a}^\top \mathbf{x}}$。排列为 $\boldsymbol{\delta} = (5、2、3、1、4、6)$ 的近似提升不等式 *[87]* 为

$$w \geq 0.198x_1 + 8.198x_2 + 4.142x_3 + 4.142x_4 + 2x_5 + 0.198x_6.$$

由 $S = \{2, 5\}$ 和 $\boldsymbol{\delta}$对应的 *EPI* 提升得到的提升的 *EPI* 为

$$w \geq 0.828x_1 + 8.198x_2 + 5.944x_3 + 5.944x_4 + 2x_5 + 0.828x_6.$$

提升的 *EPI* 支配近似提升不等式。一般而言，提升的 *EPI* 至少与近似提升不等式*[87]*一样强。

基数约束均值—风险最小化。 现在，我们将展示如何在分支切割算法中使用所提出的不等式，以提升计算性能。首先，我们通过投资组合管理中的应用来说明均值-风险最小化问题。假设给定一个集合 N，其中包含 n 个投资选项。我们的预算有限，最多只能选择 k 个选项。设这些投资选项的损失为正态随机变量 $\tilde{\boldsymbol{p}}$，其均值为 $\boldsymbol{\mu}$，协方差为 Q。我们的目标是为投资组合选择至多 k 个选项，使得我们以足够高的概率获得最大收益（或等价地，使损失最小）。基于这一目标，我们得到以下形式:

$$\min_{\mathbf{x} \in \{0,1\}^n} \left\{w : \mathbb{P}\left(\tilde{\boldsymbol{p}}^\top \mathbf{x} \leq w\right) \geq \beta, \sum_{i=1}^{n} x_i \leq k\right\}. \tag{19}$$

每个二元决策变量 x_i、 $i \in N$表示是否选择相应的选项。辅助变量 w 表示投资组合的总损失，我们希望将其最小化。参数 $\beta \in$ （0.5，1）表示我们的风险厌恶程度。若我们倾向于降低风险，同时允许利润可能较低，则可以将 β 设得更高，反之亦然。事实上，上述优化问题称为风险价值最小化，并可改写为如下 [5, 7, 8, 13]:

$$\min_{\mathbf{x} \in \{0,1\}^n} \left\{-\boldsymbol{\mu}^\top \mathbf{x} + \Omega\sqrt{\mathbf{x}^\top Q \mathbf{x}} : \sum_{i=1}^{n} x_i \leq k\right\}. \tag{20}$$

这里，Ω 是一个常数参数 $\Phi^{-1}(\beta)$，其中 Φ 是标准正态累积分布函数。我们注意到 Q 是一个半正定矩阵。问题 (20) 是一个基数约束均值—风险最小化问题，基数约束均值—风险最小化 其中

相关随机变量 [5, 7]。我们用记号 diag($\boldsymbol{y}$) 表示以向量 $\boldsymbol{y}$ 为对角元素的对角矩阵。协方差矩阵 Q 是 $Q - \mathrm{diag}(\mathbf{a})$ 与 diag(a) 之和，其中 $\mathbf{a} \in \mathbb{R}^n_+$ 满足 $Q - \mathrm{diag}(\mathbf{a}) \succeq 0$。这种分解的优点在于，可分离二次项 $\mathbf{x}^\top \mathrm{diag}(\mathbf{a})\mathbf{x}$ 可以线性化为 $\mathbf{a}^\top \mathbf{x}$。因此，问题 (20) 有一个等价的表述形式（二阶锥规划）：

$$\begin{aligned}\min_{(w,y,z,\mathbf{x})\in\mathbb{R}^3_+\times\{0,1\}^n} \{-\boldsymbol{\mu}^\top \mathbf{x} + \Omega z : w \geq \sqrt{\sum_{i\in N} a_i x_i},\ \sum_{i=1}^n x_i \leq k,\\ y \geq \sqrt{\mathbf{x}^\top (Q - \mathrm{diag}(\mathbf{a}))\mathbf{x}}, z^2 \geq w^2 + y^2\},\end{aligned} \tag{21}$$

这些问题可以由 Gurobi 等优化求解器处理。我们注意到，由问题 (21) 的一部分约束定义的集合 $\{(w, \mathbf{x}) \in \overline{\mathbb{R} \times \{0,1\}}^n : w \geq \sqrt{\sum_{i\in N} a_i x_i},\ \sum_{i=1}^n x_i \leq k\}$，与 (13) 中给出的 $\mathcal{P}^m_k$一致。基于这一观察，我们可以在分支切割算法中使用 [87, 91]的有效不等式，以提升计算性能。为应用 [91], 提出的 LEPI 和 LSI割，我们将 a 分解为$^{\mathrm{two}} + \mathbf{a}^{\mathrm{res}}$，使得 $\mathbf{a}^{\mathrm{two}}$、$\mathbf{a}^{\mathrm{res}} \in \mathbb{R}^n_+$和 $\mathbf{a}^{\mathrm{two}}$ 包含两个不同的权重。于是，问题 (21) 可以改写为

$$\begin{aligned}\min_{(v,w,y,z,\mathbf{x})\in\mathbb{R}^4_+\times\{0,1\}^n} \{-\boldsymbol{\mu}^\top \mathbf{x} + \Omega z : w \geq \sqrt{\sum_{i\in N} a_i^{\mathrm{two}} x_i}, \sum_{i=1}^n x_i \leq k,\\ v \geq \sqrt{\sum_{i\in N} a_i^{\mathrm{res}} x_i}, y \geq \sqrt{\mathbf{x}^\top (Q - \mathrm{diag}(\mathbf{a}))\mathbf{x}},\\ z^2 \geq v^2 + w^2 + y^2\}.\end{aligned}$$

表 1 总结了采用包含 LEPI 和 LSI割的分支切割算法（BC-LEPI-LSI）、采用 ALI 的分支切割算法（BC-ALI），以及直接求解问题 (20)（SOCP）时的计算性能。所有测试实例均为随机生成。实验在 Linux 服务器上使用单线程运行，该服务器配备 Intel Haswell E5-2680 处理器，主频为 2.5GHz，内存为 128GB。所有求解方法均使用 Python 3.6 和 Gurobi Optimizer 9.5.1 实现。表 1 的第一列报告风险容忍参数$\beta \in \{0.95, 0.975, 0.99\}$。第二列显示每个测试用例中的基数上界 $k \in \{5, 10, 15\}$。第四列给出平均运行时间（秒）。下一列列出平均最终间隙。最终间隙按 (UB−LB)/UB $\times$100% 计算，其中 UB 和 LB 分别是目标函数的最佳上界和下界。最后两列报告访问的分支定界节点平均数和添加的不等式平均数。在所有测试用例中，BC-LEPI-LSI 算法均优于 BC-ALI 和 SOCP；除一个测试用例外，BC-LEPI-LSI 均能求得最优解，且该测试用例满足 $\beta = 0.99$ 和 $k = 15$。在这一特定情况下，BC-LEPI-LSI 达到较小的最终间隙 0.6%。BC-ALI 和 SOCP 的平均运行时间均长于 BC-LEPI-LSI，且求解失败的实例更多。

前几节展示了次模集合函数优化在决策问题建模与计算中的强大作用。接下来，我们将详细介绍广义子模性的两种概念，即多重集扩展和混合整数扩展，并说明如何将前文介绍的技术应用于这些情形。

4 子模性的多集合扩展

本节讨论这样的情形：给定异质物品，我们需要确定要填充的“箱”的最佳子集，以及放入所选“箱”中的物品类型（见图2）。具体而言，我们将讨论子模性的多重集扩展，称为 k-子模性。这里，$k \geq 1$ 是表示物品类型数量的正整数。

在正式定义 k-子模性之前，我们先介绍一些记号。回顾一下，$N = \{1, 2, \ldots, n\}$ 是一个有限且非空的基集。我们称 k个由 N 的两两不相交子集组成的元组为一个 k-集。图4 给出了一个 3-集的示例：给定三种物品类型，$\mathbf{X}$ 是一个 3-集。其每个分量 X_i、$i \in \{1, 2, 3\}$都是$\mathbf{X}$ 的组成部分，包含放置类型为 i 的物品的“箱”。每个“箱”只能容纳一个物品，因此这些分量两两不相交。我们使用

$$(k+1)^N = \{(S_1, S_2, \ldots, S_k) : S_q \subseteq N, S_q \cap S_{q'} = \emptyset, \text{ for all } q, q' \in \{1, 2, \ldots, k\} \text{ with } q \neq q'\}$$

表示所有 k-集合的集合。为便于记号表示，令 $\mathbf{S} = (S_1, S_2, \ldots\ldots, S_k) \in (k+1)^N$。当 $k = 1$ 时，$(k+1)^N = 2^N$ 恰好是 N 的幂集。

表 1： BC-LEPI-LSI、BC-ALI 和 SOCP 在问题（20）上的 ^{i} 表示在 i 个实例中有五个在一小时的时间限制内得到求解。对于 BC-LEPI-LSI，割的总数均值表示为 $\mathrm{m}^{\mathrm{LEPI}}+\mathrm{m}^{\mathrm{LSI}}=\mathrm{m}$，其中 $\mathrm{m}^{\mathrm{LEPI}}$ 是五次试验中添加的 LEPI 均值，$\mathrm{m}^{\mathrm{LSI}}$ 是 LSI 割的均值。

β	k	方法	时间（秒）	最终间隙	节点数	割数
0.95	5	BC-LEPI-LSI	171.9^5	0.0%	6295.2	394.6+140.2=534.8
		BC-ALI	244.1^5	0.0%	10723.2	1071.4
		SOCP	1702.1^5	0.0%	173698.4	N/A
	10	BC-LEPI-LSI	577.5^5	0.0%	15612.2	1251.0+248.6=1499.6
		BC-ALI	958.3^5	0.0%	26572.0	2656.6
		SOCP	$-^0$	5.8%	82144.0	N/A
	15	BC-LEPI-LSI	124.3^5	0.0%	3414.6	258.0+73.0=331.0
		BC-ALI	217.2^5	0.0%	5907.4	590.0
		SOCP	2789.6^2	1.7%	79930.6	N/A
0.975	5	BC-LEPI-LSI	827.4^5	0.0%	20236.2	1176.2+705.2=1881.4
		BC-ALI	1254.1^5	0.0%	41768.4	4175.6
		SOCP	3349.5^1	31.1%	166471.6	N/A
	10	BC-LEPI-LSI	838.9^5	0.0%	21996.8	1498.6+648.8=2147.4
		BC-ALI	1187.4^4	1.2%	40896.2	4088.8
		SOCP	$-^0$	12.9%	74527.4	N/A
	15	BC-LEPI-LSI	988.6^5	0.0%	19225.8	1575.0+324.2=1899.2
		BC-ALI	1912.7^3	0.8%	43951.8	4393.8
		SOCP	$-^0$	6.9%	52940.8	N/A
0.99	5	BC-LEPI-LSI	650.1^5	0.0%	20280.0	1175.6+715.4=1891.0
		BC-ALI	947.5^5	0.0%	38703.6	3869.4
		SOCP	3355.9^1	72.3%	259287.8	N/A
	10	BC-LEPI-LSI	1855.7^5	0.0%	38376.0	2946.4+842.4=3788.8
		BC-ALI	3068.4^1	4.7%	88635.6	8862.6
		SOCP	$-^0$	30.8%	75435.8	N/A
	15	BC-LEPI-LSI	1354.1^4	0.6%	24718.6	2269.6+184.4=2454.0
		BC-ALI	2434.1^2	1.8%	70845.8	7083.4
		SOCP	$-^0$	10.5%	65965.4	N/A

定义 4.1 对于任意正整数 k，若函数 $f:(k+1)^N\to\mathbb{R}$ 满足以下条件，则称其为 k-子模的：对于任意 $\mathbf{X}=(X_1,X_2,\ldots,X_k)$、$\mathbf{Y}=(Y_1,Y_2,\ldots,Y_k)\in(k+1)^N$，

$$f(\mathbf{X})+f(\mathbf{Y})\geq f(\mathbf{X}\sqcap\mathbf{Y})+f(\mathbf{X}\sqcup\mathbf{Y}),$$

其中

$$\mathbf{X}\sqcap\mathbf{Y}=(X_1\cap Y_1,X_2\cap Y_2,\ldots,X_k\cap Y_k),$$

and

$$\mathbf{X}\sqcup\mathbf{Y}=\left((X_1\cup Y_1)\backslash\bigcup_{q=2}^{k}(X_q\cup Y_q),\ldots,(X_k\cup Y_k)\backslash\bigcup_{q=1}^{k-1}(X_q\cup Y_q)\right).$$

定义 4.1 与次模集合函数的定义 2.1类似。二者的区别在于，定义4.1 中的交运算符和并运算符 k按分量作用于 -元组。特别地， $\sqcup$ 运算符会排除额外元素，以确保所得 -集合两两不相交。当 $k=1$时，满足定义 4.1 的函数恰好是次模集合函数。当 $k=2$ 时，这些函数称为 双次模函数。

4.1 k-子模性的应用

接下来, 我们介绍几个 k-次模性的应用，以展示其建模能力。

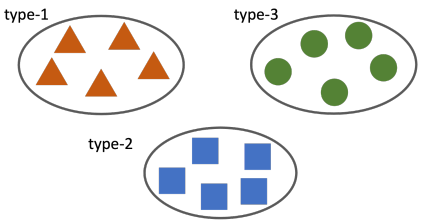

图 4：一个 k-集的示例。

4.1.1 多类型传感器布置。

物联网（IoT）使传感器网络能够监测智能家居 [25], 、智能电网 [1]和智慧城市 [92]等系统。在这些应用中，我们通常会使用不止一种类型的传感器，来测量环境的不同方面（例如温度、湿度和亮度）。假设给定 $k \geq 2$ 种类型的传感器，以及由 $N = \{1, 2, \ldots, n\}$ 个候选位置组成的集合，且每个位置至多放置一个传感器。设类型为 i 的传感器可布设的位置为 S_i。那么，每个 k-集合 $\mathrm{S} = (S_1, \ldots, S_k) \in (k+1)^N$ 都唯一表示一种 k类型的传感器布设方案。任何布设方案的有效性都可以用以下熵函数 $H(X_{\mathrm{S}}) = -\sum_{x \in \mathcal{X}_{\mathrm{S}}} \mathbb{P}(X_{\mathrm{S}} = x) \log \mathbb{P}(X_{\mathrm{S}} = x)$ 来量化，其中 X_{S} 是一个随机变量，表示在 S 处安装的相应类型传感器可能获得的观测值；集合 $\mathcal{X}_{\mathrm{S}}$ 包含所有可能的观测值[55]。直观来说，当分别位于 S_1、……、S_k 处的 k 种类型的测量值难以预测时， X_{S} 的熵较高。对于所有 $\mathrm{S} \in (k+1)^N$，函数 $f(\mathrm{S}) = H(X_{\mathrm{S}})$ 都是 k-次模的 [55]。

4.1.2 多主题影响力传播。

社交媒体平台一直是重要的信息来源。社交网络中的信息传播机制可以用一个次模集合函数来建模，该函数将初始信息传播者集合映射为受这些信息影响的网络用户总数的期望值 [41]。该模型对于病毒式营销等应用尤其有帮助。该模型经过推广，可支持 $k \geq 2$ 种类型的影响力 [55]。在社交网络 $G = (N, A)$中， $N = \{1, 2, \ldots, n\}$ 表示网络用户，弧集 A 对用户之间的交互进行建模。对于任意 i、 $j \in N$和 $q \in \{1, 2, \ldots, k\}$， $p^q(i,j)$表示 i 在主题 q上影响 j 的概率。在病毒式营销中， $p^q(i,j)$可解释为 i 说服 j 购买产品 q的可能性。设 $S_q \subseteq N$ 为被选中推广类型为q 的产品的影响者。此外，设 $A_q(S_q)$表示在 [41]提出的随机模型下，初始传播者 S_q 围绕产品 q 所影响的个体。函数$f(\mathrm{S}) = \mathbb{E}[|\bigcup_{q=1}^{k} A_q(S_q)|]$ 用于评估给定初始影响者S时，受影响个体总数的期望值，并已证明该函数是单调的且具有 k-次模性 [55]。

4.1.3 多类特征选择。

特征选择是许多研究领域中的一项重要技术，例如机器学习 [66], 生物信息学 [63], 和数据挖掘 [61]。这一过程能够提高模型的可解释性，并避免维数灾难。多类特征选择是对常规特征选择问题的推广。在 k 类特征选择中，给定 k 个互不相关的预测变量和一个特征集合，我们的目标是选出信息量最大的特征，并确定与所选特征相对应的预测变量。当 $k = 2$ 时，该问题称为耦合特征选择问题 [70]。具体而言， C_1, C_2 是两个预测变量， N 是一个特征集合。对于任意 $\mathrm{S} = (S_1, S_2) \in 3^N$， S_i 包含为预测 $C_i, i \in \{1, 2\}$ 而选取的特征。假设给定 $C = \{C_1, C_2\}$ 时， S_1, S_2 条件相互独立。耦合特征选择的信息量由其互信息 $I(\mathrm{S}; C) = H(S_1 \cup S_2) - \sum_{i \in S_1} H(i \mid C_1) - \sum_{j \in S_2} H(j \mid C_2)$ 衡量，其中 $H(X) = -\sum_{x \in \mathcal{X}} \mathbb{P}(X = x) \log \mathbb{P}(X = x)$，且 $H(X \mid Y) = -\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \mathbb{P}(X = x, Y = y) \log \frac{\mathbb{P}(X=x, Y=y)}{\mathbb{P}(Y=y)}$。

较高的互信息表明，特征双集合对预测任务提供了更多信息。函数 $f(\mathbf{S}) = I(\mathbf{S};\ C)$ 是单调且双子模的 [70]。

4.1.4 药物相互作用检测。

某些药物同时服用时，会在人体内发生反应。这些反应称为药物相互作用（DDI）。药物相互作用至少占药物不良事件的 30%，每年造成 770,000 人受伤或死亡 [59, 77]。 [33] 的作者提出了一个函数，用于建模药物组合与不良反应之间的相关性；他们进一步证明该函数是双子模的，并利用双子模性，根据在线医疗论坛数据确定药物相互作用。

4.2 k-子模优化的精确求解方法

上一节说明了 k-子模函数如何在实际应用中对效用进行建模。得到此类模型后，下一步是优化 k-子模效用函数，以确定最优决策。本节介绍 k-子模函数最大化和最小化的精确求解方法。在本节中，不失一般性地假设 $f(\emptyset) = 0$，其中 $\emptyset = (\emptyset,\ \ldots\ldots,\ \emptyset)$。

4.2.1 k-子模最大化。

k-子模最大化　　问题假设如下　　形式：

$$\max_{\mathbf{X}\in\mathcal{X}} f(\mathbf{X}), \tag{22}$$

其中，f 是 k-子模的，且 $\mathcal{X} \subseteq (k+1)^N$ 包含所有可行的 k-集。当问题（22）无约束时，$\mathcal{X} = (k+1)^N$。我们还可以加入基数约束和背包约束等线性约束，使$\mathcal{X} \subsetneq (k+1)^N$。

次模集合函数最大化是 k-子模最大化的一个特例。鉴于次模最大化是NP难的，k-子模最大化同样是NP难的。针对 k-子模最大化提出的大多数方法都是近似算法。一类双次模函数可以通过常数因子近似算法 [70]进行最大化。包括 [38, 79, 80] 在内的研究为无约束双子模最大化中的随机贪心算法建立了 1/2-近似保证。当 $k \geq 3$时，[80] 实现了$\max(1/3,\ 1/(1+\max(1,\ \sqrt{(k-1)/4})))$的近似保证；[39]将这一结果改进至 $1/2$ 。其他研究则考虑在各种约束（如基数约束 [55],拟阵约束 [64]）下最大化非负单调 k-子模函数。

在本教程中，我们重点介绍从混合整数规划方法推导出的精确求解方法。稍微滥用一下记号，我们引入以下 k-集的二进制表示。也就是说，每个 $\mathbf{X} \in (k+1)^N$都唯一映射为$\mathbf{x} = [\mathbf{x}^1, \ldots, \mathbf{x}^k]^\top \in \{0,1\}^{kn}$，其中$\sum_{q=1}^{k} x_i^q \leq 1$对所有 $i \in N$均成立，且

$$x_i^q = \begin{cases} 1, & \text{if } i \in X_q, \\ 0, & \text{otherwise.} \end{cases}$$

线性不等式 $\sum_{q=1}^{k} x_i^q \leq 1$ 对所有 $i \in N$ 均成立，确保对应k-集中的分量两两不相交。我们用 $\mathcal{X}$ 表示可行的 k-集以及可行的二元关联向量。于是，问题（22）可改写为

$$\max\{w : (\mathbf{x}, w) \in \mathcal{T}_f, \mathbf{x} \in \mathcal{X}\}, \tag{23}$$

其中

$$\mathcal{T}_f = \left\{ (\mathbf{x}, w) \in \{0,1\}^{kn} \times \mathbb{R} : w \leq f(\mathbf{x}), \sum_{q=1}^{k} x_i^q \leq 1 \text{ for all } i \in N \right\}.$$

令 $\rho_{q,i}(\mathbf{X}) = f(X_1, \ldots, X_q \cup \{i\}, \ldots, X_k) - f(\mathbf{X})$ 表示将元素 $i \in N$添加到$\mathbf{X} \in (k+1)$ 的第 q 个分量所产生的边际贡献。

定义 *4.2* 若对任意满足 $X_q \subseteq Y_q$ 的 $\boldsymbol{X}$、$\boldsymbol{Y} \in (k+1)^N$，$k$-次模函数 $f : (k+1)^N \to \mathbb{R}$ 满足 $f(\boldsymbol{Y}) \geq f(\boldsymbol{X})$，则称其为单调非递减函数，或简称单调函数。等价地，若对任意 $\boldsymbol{X} \in (k+1)^N$、$i \in N \setminus \bigcup_{p=1}^{k} X_p$ 和 $q \in \{1, \ldots, k\}$，f 满足 $\rho_{q,i}(\boldsymbol{X}) \geq 0$，则称 f 为单调函数。

给定一个非单调的 k-次模函数 f，可以如下构造其单调对应函数。对于每个 $i \in N$ 和 $q \in \{1, \ldots, k\}$，令

$$\xi_i^q = \min \left\{ \rho_{q,i}(\mathbf{S}) : \mathbf{S} \in (k+1)^{N \backslash \{i\}}, \bigcup_{p=1}^{k} S_p = N \backslash \{i\} \right\}.$$

引理 4。3 *[89]* 对于任意非单调的 k-次模函数 f，

$$f^*(\mathbf{X}) := f(\mathbf{X}) - \sum_{q=1}^{k} \sum_{i \in X_q} \xi_i^q$$

是 k-次模且单调的。

有关这一结果的更多细节，请参阅 [89] 的引理 3.2。

定义 4.4 给定任意 k-次模函数 f以及任意 $\mathbf{S} \in (k+1)^N$，与 $\mathbf{S}$相关联的 k-次模不等式为

$$w \leq f(\mathbf{S}) + \sum_{q=1}^{k} \sum_{i \notin \bigcup_{r=1}^{k} S_r} \rho_{q,i}(\mathbf{S}) x_i^q + \sum_{q=1}^{k} \sum_{p \in \{1,\ldots,k\} \backslash \{q\}} \sum_{i \in S_p} \rho_{q,i}(\emptyset) x_i^q - \sum_{q=1}^{k} \sum_{i \in S_q} \xi_i^q (1 - x_i^q). \tag{24}$$

命题 4。5 *[89]* 对于任意 k-次模函数 f 以及任意 $\mathbf{S} \in (k+1)^N$，与 $\mathbf{S}$ 相关联的 k-次模不等式对 $\mathcal{T}_f$均成立。 (24)

有关该命题的证明，请参阅 [89] 中的命题 4.2。我们注意到，由 [81] 提出的次模不等式（11）是 k-次模不等式在 $k = 1$时的特例。-次模不等式为目标函数 f提供了分段线性表示，据此我们在下一个定理中推导出 k-次模最大化的另一种等价形式。

定理 4。6 *[89]* 问题 (23) 等价于

$$\begin{aligned} \max_{(\mathbf{x},w) \in \{0,1\}^{kn} \times \mathbb{R}} \{ w : w \leq f(\mathbf{S}) &+ \sum_{q=1}^{k} \sum_{i \notin \bigcup_{r=1}^{k} S_r} \rho_{q,i}(\mathbf{S}) x_i^q + \sum_{q=1}^{k} \sum_{p \in \{1,\ldots,k\} \backslash \{q\}} \sum_{i \in S_p} \rho_{q,i}(\emptyset) x_i^q \\ &- \sum_{q=1}^{k} \sum_{i \in S_q} \xi_i^q (1 - x_i^q), \forall\, \mathbf{S} \in (k+1)^N, \\ &\sum_{q=1}^{k} x_i^q \leq 1, \forall\, i \in N, \mathbf{x} \in \mathcal{X}. \} \end{aligned} \tag{25}$$

我们使用 DCG 框架（参见算法 1）来克服在混合 0-1 线性规划重构（25）中存在指数数量的 k-次模不等式这一挑战。第 4.3节还包含一项关于多类型传感器布置问题的计算研究。

4.2.2 k-次模最小化

本节总结最小化 k-子模函数方面的研究成果。特别地，我们着重讨论 $k = 2$的情形。研究 [60] 证明了 Lov´asz 扩展（见第2节）在双次模函数中的类似结果。这表明，可以使用椭球法在多项式时间内求解无约束双子模最小化问题。针对无约束双子模最小化问题，也已提出弱多项式算法和强多项式算法 [24, 50]。子模和双次模最小化的极小极大定理已推广至 k-子模情形，其中 $k \geq 3$ [34]，但一般 k-子模最小化问题的复杂度仍是一个开放问题。在 [88], 中，我们完整刻画了任意双次模函数的上图集凸包；这一多面体刻画带来了有效的割平面算法，可将约束双子模最小化问题求解至全局最优。

现在我们考虑约束双子模最小化问题:

$$\begin{aligned} \min \quad & f(S_1, S_2) \\ \text{s.t.} \quad & (S_1, S_2) \in \mathcal{S} \subseteq 3^N, \end{aligned} \tag{26}$$

并总结我们在 [88]中提出的混合整数规划方法。不失一般性，我们再次假设 $f(\emptyset, \emptyset) = 0$。此外， $\mathcal{S} \subseteq 3^N$是可能纳入任何线性可表示约束后得到的可行解集合。这是一类关于双集合的非线性优化问题，无法直接输入优化求解器。

为了将问题（26）重新表述为混合整数线性规划，我们首先将双集合转换为三元特征向量。更具体地说，对于任意（S_1， S_2） $\in 3^N$，存在唯一的向量$\mathbf{x} \in \{0, \pm 1\}^n$，使得对每个 $i \in \{1, 2, \ldots, n\}$,

$$x_i := \begin{cases} 1, & i \in S_1, \\ -1, & i \in S_2, \\ 0, & \text{otherwise.} \end{cases}$$

反过来，每个$\mathbf{x} \in \{0, \pm 1\}^n$ 都唯一对应一个双集合。至此，我们已成功将问题（$\mathcal{S}$）中可行双集的集合转换为整数向量$\mathbf{x} \in \mathcal{X}$，其中 $\mathcal{X}$包含所有$\mathbf{x} \in \{0, \pm 1\}^n$，且这些向量还满足所有附加线性约束。

下一步是将非线性目标函数 f 改写为优化求解器能够处理的形式。在 [88], 中，我们提出了 f的分段线性表示。该表示依赖于极值多双拟阵不等式，这些不等式定义如下。

定义 4.7 给定任意排列 $\boldsymbol{\delta} = (\delta_1, \delta_2, \ldots, \delta_n)$ 属于 N以及任意 $\boldsymbol{\sigma} \in \{\pm 1\}^n$。与和相关联的极值多双拟阵不等式为 $w \geq \boldsymbol{\pi}^\top \mathbf{x}$，其中 $\boldsymbol{\pi} \leftarrow \texttt{Signed Greedy}(\boldsymbol{\delta}, \boldsymbol{\sigma})$ *(见算法 2)*。 -

算法 2: Signed Greedy [3, 50]

输入一个排列 $\boldsymbol{\delta} = \{\delta_1, \delta_2, \ldots, \delta_n\}$ 属于 N，一个向量 $\boldsymbol{\sigma} \in \{\pm 1\}^n$; $\boldsymbol{\pi} \leftarrow \mathbf{0}$, $S_1 \leftarrow \emptyset$, $S_2 \leftarrow \emptyset$; 对于 $i = 1, 2, \ldots,$ n，执行 如果 $\sigma_{\delta_i} = 1$ ，则 $\pi_{\delta_i} \leftarrow f(S_1 \cup \{\delta_i\}, S_2) - f(S_1, S_2)$; $S_1 \leftarrow S_1 \cup \{\delta_i\}$; 否则 $\pi_{\delta_i} \leftarrow -f(S_1, S_2 \cup \{\delta_i\}) + f(S_1, S_2)$; $S_2 \leftarrow S_2 \cup \{\delta_i\}$; 结束 if 结束 for 输出 $\boldsymbol{\pi} \in \mathbb{R}^n$。

定理 4.8正式确立了极值多双拟阵不等式为 f提供分段线性表示。关于定理 [88] 的证明，请参阅 4.8。

定理 4.8 *[88]* f的上图的凸包 *conv* $((\mathbf{x}\ w) \in \{0, \pm 1\}^n \times \mathbb{R} : \quad w \geq f(\mathbf{x}))$，完全由所有极值多双拟阵不等式 $w \geq \boldsymbol{\pi}^\top \mathbf{x}$以及平凡不等式 $-1 \leq x_i \leq 1$ 描述，其中对所有$i \in \{1, 2, \ldots, n\}$均成立。

由定理 4.8可知，约束双子模最小化问题（26）具有一个等价的重新表述:

$$\begin{aligned} \min \quad & w \\ \text{s.t.} \quad & (\mathbf{x}, w) \in \mathcal{C}, \\ & \mathbf{x} \in \mathcal{X}. \end{aligned} \tag{27}$$

多面体集合 $\mathcal{C}$ 由所有极值多双拟阵不等式定义。集合 $\mathcal{X}$ 包含所有与可行双集对应的三元特征向量。鉴于问题 (27) 含有指数多条极值多双拟阵不等式，因此宜采用 DCG 框架（见算法1）。

在任意分数解 $\mathbf{x} \in \mathbb{R}^n$ 处分离一条极值多双拟阵不等式的问题，可以通过 $\mathcal{O}(n\log n)$ 贪心算法高效求解（算法 3）。

算法3： 广义贪心算法 [15]

输入 $\mathbf{x} \in \mathbb{R}^n$;
将 $\mathbf{x}$ 中的元素排序，得到排列 $\boldsymbol{\delta}$，使得 $|x_{\delta_1}| \geq |x_{\delta_2}| \geq \cdots \geq |x_{\delta_n}|$;
$\boldsymbol{\pi} \leftarrow \mathbf{0}$, $S_1, S_2 \leftarrow \emptyset$;
对于 $i = 1, 2, \cdots, n$ **do**
  如果 $x_{\delta_i} \geq 0$，**则**
    $\pi_{\delta_i} \leftarrow f(S_1 \cup \{\delta_i\}, S_2) - f(S_1, S_2)$;
    $S_1 \leftarrow S_1 \cup \{\delta_i\}$;
  else
    $\pi_{\delta_i} \leftarrow -f(S_1, S_2 \cup \{\delta_i\}) + f(S_1, S_2)$;
    $S_2 \leftarrow S_2 \cup \{\delta_i\}$;
  结束条件判断
循环结束
输出 一个极值多多拟阵不等式 $w \geq \boldsymbol{\pi}^\top \mathbf{x}$。

现在我们已经介绍了求解 k-次模最大化和最小化问题的精确方法，接下来将在多类型传感器布置问题及其鲁棒变体中展示所提出的 DCG 算法。

4.3 多类型传感器布置的深入示例

我们将深入探讨第 4.1.1 节中描述的多类型传感器布置问题，以展示 k-次模优化在实践中的应用。有关该应用中用于建模效用的熵函数数值示例，请参阅 [88] 中的示例 5.1。回顾一下，给定 k 种传感器类型和n 个候选位置。我们希望确定一个最优的 k-类型传感器布设方案 $\boldsymbol{S} \in (k+1)^N$，使熵最大化。此外，我们的预算有限，因此可用传感器数量受到限制——每种类型最多可安装 B_q 个传感器 q，其中 $q \in \{1, 2, \ldots, k\}$。这些限制可表述为以下基数约束：$\sum_{i=1}^{n} x_i^q \leq B_q$ 对于每个 $q \in \{1, 2, \ldots, k\}$。

由于目标函数具有高度非线性，该问题不存在紧凑的混合整数线性规划形式。因此，我们将 DCG 算法与 k-次模不等式进行比较，并以唯一的精确求解方法——穷举搜索（ES）作为对照。我们使用 Intel Berkeley 研究实验室数据集 [14], 随机生成测试实例，该数据集包含实验室 54 个位置处的亮度、温度和湿度传感器读数。对于 $k = 2$ 的一组实验，我们旨在为亮度和温度传感器寻找最佳布设方案。当 $k = 3$时，我们的目标是确定针对全部三种测量值的最优布设方案。我们的实验在一台 Linux 服务器上使用两个线程运行，该服务器配备 Intel Haswell E5-2680 处理器，主频为 2.5GHz，内存为 128GB。我们的算法使用 Python 3.6 和 Gurobi Optimizer 7.5.1 实现，时间限制为一小时。计算结果列于表2中。我们将候选位置的数量 n设为 20 至 50。我们使用 100 组亮度、温度和湿度读数来评估熵。对于所有 B_q，基数上界 $n/10$ 均设为 $q \in \{1, \ldots, k\}$。我们报告 DCG 的计算统计数据，包括运行时间、添加的 k-次模不等式数量、访问的分支定界节点数量以及最终间隙（即 (UB–LB)/UB）。对于 ES，我们报告平均运行时间。

DCG 在半小时内即可将所有测试实例求解至最优。相比之下，对于 $k = 2$ 和 $k = 3$，ES 无法求解规模超过 $n = 20$ 的任何实例。以 $k = 2$ 和 $n = 50$ 的测试案例为例，ES 需要枚举 50!/(5!5!40!) $\approx 2.59 \times 10^{12}$ 个可行解才能找到最优解。每次熵评估平均耗时 1.6×10^{-4} 秒，因此 ES 大约需要 13.13 年才能达到最优。相比之下，我们的算法在八分钟内即可找到最优解。

我们进一步考虑耦合传感器布置问题的鲁棒变体。此时，我们的目标是确定一个最坏情况熵最高的耦合传感器布置方案。具体而言，我们希望布设方案能够抵御两类潜在故障。首先，对于任意布设方案，最多会安装 $W \in \mathbb{Z}_+$个类型错误的传感器。其次，部分传感器会因软件或硬件故障而失灵，但至少有 B_1' 个 1 型传感器和 B_2' 个 2 型传感器能够正常工作。问题（28）是鲁棒

表 2：DCG 和穷举搜索在 k 型传感器布设中的计算性能。统计数据取三次试验的均值。上标 $^{\ell}$ 表示有 ℓ 个实例达到一小时的时间限制。

k	n	时间（秒）	# 割	# 节点	最终间隙	穷举搜索时间（秒）
2	20	1.09	38.67	42.33	–	17.68
	30	13.04	250.33	253.67	–	–[3]
	40	161.80	1783.33	1791.00	–	–[3]
	50	478.09	3559.33	3566.33	–	–[3]
3	20	1.58	32.67	35.33	–	2953.18
	30	20.27	222.33	226.67	–	–[3]
	40	164.97	1043.00	1049.33	–	–[3]
	50	1779.49	6907.67	6920.00	–	–[3]

耦合传感器布置问题。

$$\max_{(S_1,S_2)\in 3^N} \quad \min_{(T_1,T_2)\in 3^{S_1\cup S_2}} \quad f\ (T_1,\ T_2) \tag{28a}$$

$$\text{s.t.} \quad |S_1| = B_1, |S_2| = B_2, \tag{28b}$$

$$|T_1| \geq B_1', |T_2| \geq B_2', \tag{28c}$$

$$|T_1 \cap S_2| + |T_2 \cap S_1| \leq W. \tag{28d}$$

给定一个布设方案 $(S_1, S_2) \in 3^N$，在外层最大化问题中选定后，内层最小化问题中的决策 (T_1, T_2) 对应于最坏情况下两类故障发生后正常工作的传感器。该内层最小化问题是约束双子模最小化。由于目标函数是非线性的，且存在基数约束，我们采用 DCG 算法处理内层问题的混合整数重构（29）。在这一形式中，$\mathcal{C}$ 在约束（29b）中由所有极值多双拟阵不等式定义。可以采用经典的鲁棒优化技术（例如 L 形方法）[45] 来处理外层最大化问题。

$$\min \quad w \tag{29a}$$

$$\text{s.t. } (\mathbf{x}\ w) \in \mathcal{C}, \tag{29b}$$

$$x_i = y_i^1 - y_i^2, \qquad \text{for all } i \in N, \tag{29c}$$

$$y_i^1 + y_i^2 \leq 1, \qquad \text{for all } i \in N, \tag{29d}$$

$$\sum_{i\in N} y_i^1 \geq B_1', \tag{29e}$$

$$\sum_{i\in N} y_i^2 \geq B_2', \tag{29f}$$

$$y_i^1, y_i^2 = 0, \qquad \text{for all } i \in N\backslash(S_1 \cup S_2), \tag{29g}$$

$$\sum_{i\in S_1} y_i^2 + \sum_{i\in S_2} y_i^1 \leq W, \qquad \text{for all } i \in N, \tag{29h}$$

$$y_i^1, y_i^2 \in \{0, 1\}, \qquad \text{for all } i \in N. \tag{29i}$$

我们再次使用 Intel Berkeley 研究实验室数据集中的温度和湿度读数生成随机实例。我们随机选取 n 个候选位置，并改变 t 用于熵评估的温度和湿度读数数量。我们设置 $B_1 = \lfloor 2n/5 \rfloor$，$B_2 = \lfloor n/2 \rfloor$，$B_1' = \lfloor 4B_1/5 \rfloor$，$B_2' = \lfloor 3B_2/5 \rfloor$，以及 $W = \lfloor 3\ (B_1 + B_2)\ /5 \rfloor$。在表 3中，我们报告 DCG 的平均运行时间（秒）、新增极值多双拟阵不等式的平均数量，以及访问的分支定界节点平均数量。结果表明，DCG 平均能在五分钟内求解所有测试实例。对于规模较小且满足 $n \leq 10$的测试用例，DCG 可在五秒内达到最优解。总体而言，DCG 是一种高效的方法，能够精确优化内部的约束双子模最小化问题。

表 3：DCG 在鲁棒耦合传感器布设中的双子模最小化子问题上的计算结果。统计数据为十个实例的平均值。

n	t	时间（秒）	割数	节点数
5	10	0.003	2.1	2.0
	20	0.012	10.2	25.9
	50	0.011	4.9	6.4
	100	0.035	11.4	28.9
	500	0.096	8.0	20.7
10	10	0.104	62.4	493.1
	20	0.238	97.1	1113.1
	50	0.339	77.9	750.0
	100	0.896	114.9	1024.1
	500	4.261	142.6	1162.3
20	10	0.405	103.8	2015.4
	20	1.121	167.5	2906.7
	50	88.371	1568.9	127030.7
	100	154.587	2257.2	122514.8
	500	277.127	1800.5	111079.2

至此，我们已展示了子模性的多集合扩展——k-子模性——在现实应用中的建模能力。这些决策问题会产生定义在 k-集合上的非线性优化问题，而这些问题无法直接输入优化求解器。我们讨论了将这些具有挑战性的问题重新表述为混合整数线性规划的方法。尽管约束数量庞大，DCG 算法仍能利用混合整数线性规划求解器的能力，高效地将重新表述的问题求至最优。接下来，我们将简要介绍另一种广义子模性的概念。

5 子模性的混合整数扩展

回顾第1节，我们曾阐述子模性的混合整数扩展：在这种扩展中，我们可以将多个甚至分数个给定的同质物品放入每个“箱子”中（见图2）。与次模集合函数优化中的二元决策变量不同，现在我们需要纳入混合整数决策变量。具体而言，本教程重点讨论递减收益（*DR*）子模性。

令 $\mathbf{e}^i \in \mathbb{R}^n$ 为一个向量，其第 i 个分量为 1，其余分量均为 0。

定义 5.1 函数 $f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$ 称为 DR-子模 函数，当且仅当

$$f(\mathbf{x} + \alpha\mathbf{e}^i) - f(\mathbf{x}) \geq f(\mathbf{y} + \alpha\mathbf{e}^i) - f(\mathbf{y})$$

对于每个 $i \in \{1, 2, \ldots, n\}$，对于所有满足 $\mathbf{x} \leq \mathbf{y}$（逐分量成立）的 $\mathbf{x}$、$\mathbf{y} \in \mathcal{X}$，以及所有满足 $\mathbf{x} + \alpha\mathbf{e}^i$、$\mathbf{y} + \alpha\mathbf{e}^i \in \mathcal{X}$ 的 $\alpha \in \mathbb{R}_+$。

与次模集合函数的定义 2.2类似，定义 5.1 体现了边际收益递减的直觉。向逐分量较小的向量 α 的第 i 个坐标增加一个正步长，其$\mathbf{x}$ 边际贡献大于向较大的向量 $\mathbf{y}$增加同样步长所产生的边际贡献。连续 DR-子模函数可以是凸函数、凹函数，也可以既非凸也非凹。图 5 （左）给出了一个非凸且非凹的 DR-子模函数示例。一个二阶可微函数是 DR-子模函数，当且仅当其海森矩阵元素在定义域内每一点都非正 [12]。

还存在另一种相关的子模性的混合整数扩展。

定义 5.2 考虑函数 $f : \mathcal{X} \subseteq \mathbb{R}^n \to \mathbb{R}$。假设

$$f(\mathbf{x}) + f(\mathbf{y}) \geq f(\mathbf{x} \wedge \mathbf{y}) + f(\mathbf{x} \vee \mathbf{y}),$$

对于所有$\mathbf{x}$、$\mathbf{y} \in \mathcal{X}$，其中 $\wedge$和 $\vee$分别是逐分量最小值和最大值算子。此时，f称为次模（参见 *[10, 12, 29, 74, 78]*）；当 $\mathcal{X} \subseteq \mathbb{Z}^n$ 时，则称为格次模（参见 *[22, 71, 73]*）。

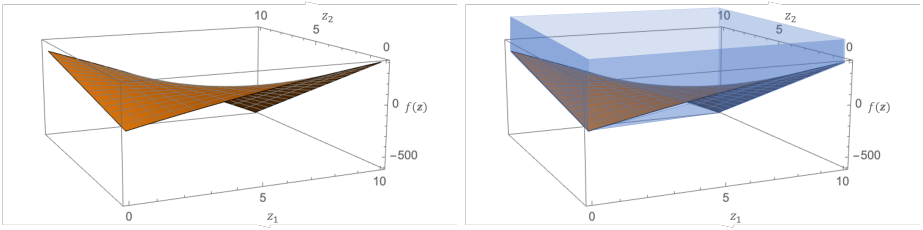

图 5：盒约束下的连续非凸且非凹的 DR-子模函数 $f(\mathbf{x}) = -x_1^2 - 13x_1x_2 + 50x_1 + 30x_2$及其上图的凸包。该凸包是多面体的。

请注意，DR-子模性是（格）次模性的一个子类 [12]。由于具有边际收益递减性质，DR-子模性广泛应用于许多领域，包括但不限于最优预算分配、收益管理、传感器能量管理、图的稳定数和平均场推断 [12, 52, 62,72, 73]。因此，本节重点讨论DR-子模性。

5.1 DR-子模性的应用

下面我们详细介绍 DR-子模函数的两个应用。

5.1.1 广告规划。

我们关注的是设计一个具有最优影响力的广告计划。设 $N = \{1, 2, \ldots, n\}$ 为媒体平台的集合（例如插播广告、电视广告时段、社交媒体网红）， T 为我们的目标受众集合。每个媒体选项 $i \in N$ 对受众 $t \in T$产生影响的概率为 p_{it} 。对于所有 $i \in N$，我们的决策是通过媒体选项 i投放的广告投放量 x_i 。该变量可以是连续的（例如电视播出时间），也可以是离散的（例如社交媒体网红的数量）。给定计划 $\mathbf{x} \in \mathbb{R}^n$，该计划的总影响力为$f(\mathbf{x}) = \sum_{t\in T}\left[1 - \prod_{i\in N}(1 - p_{it})^{x_i}\right]$，其为 DR-子模函数 [12]。

5.1.2 传感器能量管理。

全球能源危机使能源效率成为许多应用中的重要考量。我们描述了一个问题：如何布设能量水平可调的传感器，以检测污染事件 [71]。给定$N = \{1, 2, \ldots, n\}$ 个用于布设传感器的候选位置，我们的任务是决定向 $i \in N$提供的能量水平 x_i。能量可取连续范围内的数值，也可取离散水平，因此我们的决策变量 $\mathbf{x}$ 分别为连续变量和离散变量。若未在 i布设传感器，则令 $x_i = 0$ 。当发生污染事件 $e \in E$ 时， $z(i, e)$ 表示在该事件中，污染物到达传感器位置 i 所需的时间。令 $0 < p < 1$ 为一个常数，表示任一传感器位置处单位能量导致检测失败的概率。当污染物到达传感器位置 i时，成功检测到它的概率为 $1 - p^{x_i}$。我们注意到，能量越高，成功检测的概率也越大。给定 $e \in E$，令 I_e 为一个随机变量，表示在事件 e中首个检测到污染物的传感器。令 $z_\infty = \max_{i\in N, e\in E} z(i, e)$ 表示事件 e 中的污染物到达任一传感器位置 $i \in N$所需的最长时间。如果事件 e中没有传感器检测到污染物，则令 $z(i, e) = z_\infty$。污染物检测的预期时间缩短由函数 $f(\mathbf{x}) = \mathbb{E}_{e\in E}\mathbb{E}_{I_e}[z_\infty - z(I_e, e)]$建模。已有研究表明，函数 f 是 DR-子模的 [71]。

5.2 DR-次模优化

受众多应用的推动，DR-子模最大化受到了广泛关注。对于连续DR-子模最大化，研究人员开发了近似算法和全局优化算法 [11,12, 30, 51, 62]。其他研究探讨了具有整数决策变量的 DR-子模最大化，并提出了具有理论保证的近似算法 [72, 73]。

很少有研究关注 DR-子模最小化。这些研究通常假设决策变量受盒约束限制（即 $0 \le x_i \le u_i$ 对于 $u_i \in \mathbb{R}_+$, $i \in N$）。在 [22], 中，作者将含整数决策变量 的 DR-次模优化问题化为次模集合函数优化问题，这意味着盒约束下的整数 DR-子模最小化问题具有多项式可解性。其复杂度取决于决策空间的大小以及盒约束参数值的对数（另见 [74]）。

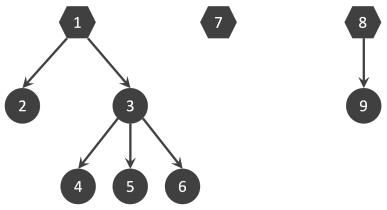

图 6：有向根森林示例。

回顾一下，（格）次模性涵盖 DR-子模性，因此关于（格）次模最小化的研究也适用于 DR-次模最小化。Topkis [78] 研究了一类参数化（格）次模最小化问题的极小化解的性质。Bach [10]将 Choquet 积分（等价于 Lov`asz 延拓）推广到（格）次模函数，并提出了一种在盒约束下最小化（格）次模函数的算法。该算法的时间复杂度关于决策空间大小和盒约束参数为多项式，因此是伪多项式的。Han 等人 [29] 研究了一类含有混合二元变量的约束子模极小化问题，并证明了其具有多项式可解性。我们采用多面体方法来处理含有混合整数变量的约束 DR-子模极小化问题，见 [90]。

5.2.1 DR-次模最小化

接下来，我们讨论如何将混合整数规划方法应用于带约束的混合整数 DR-子模最小化问题 [90]。

考虑以下问题:

$$\min_{\mathbf{x}\in\mathcal{X}(\mathcal{G},\mathbf{u})} f(\mathbf{x}), \tag{30}$$

其中 $\mathcal{X}(\mathcal{G},\ \mathbf{u})$ 由盒约束以及可能存在的单调性约束定义:

$$\mathcal{X}(\mathcal{G},\mathbf{u}) := \{\mathbf{x}\in\mathbb{Z}^n\times\mathbb{R}^m : \mathbf{0}\le\mathbf{x}\le\mathbf{u}, x_i\le x_j,\ \forall\,(i,j)\in\mathcal{A}\}. \tag{31}$$

我们的决策变量 $\mathbf{x}$ 包含 $n\in\mathbb{Z}_+$ 一般整数变量（$n\ge 1$）以及 $m\in\mathbb{Z}_+$ 连续变量。令$\mathcal{V}=\{1,\dots,n+m\}$ 表示所有变量的指标集，并令 N 表示对应于整数变量的指标集合。记号 $\mathcal{G}$ 表示一个顶点集为 $\mathcal{V}$、弧集为 $\mathcal{A}$的有向图。弧集 $\mathcal{A}\subset\mathcal{V}\times\mathcal{V}$ 决定可行集的单调性约束。这些单调性约束源自单调线性不等式组，即每个不等式都含有两个变量，且其系数符号相反 [4, 17, 32, 69]。这类约束也自然出现在各种应用中。例如，在传感器能量管理中（见第 5.1.2 节），某些传感器位置比其他位置能提供更有用的信息，因此我们可能希望对能量分配施加偏序。

我们允许 $\mathcal{G}$ 为任意有向根森林，即由若干有向树不相交并组成的结构；每棵树都有一个指定的根，且所有弧均从根向外指向。图6 展示了一个有向根森林的示例。这类有向图不含环，因为环会导致冗余等式。下面给出与有向根森林相关的定义。对于 $i\in\mathcal{V}$，我们用 $R^+(i)\subseteq\mathcal{V}$ 表示 i 沿着 $\mathcal{A}$中的弧可以到达的顶点。我们称这些顶点为 的后代 （descendants） i。

令人惊讶的是，约束混合整数非线性优化问题（30）可以改写为连续线性规划。为此，我们首先将问题（30）改写为

$$\min\left\{w : (\mathbf{x},w)\in\operatorname{conv}\left(\mathcal{P}_f^{\mathcal{X}(\mathcal{G},\mathbf{u})}\right)\right\}, \tag{32}$$

其中

$$\mathcal{P}_f^{\mathcal{X}(\mathcal{G},\mathbf{u})} := \{(\mathbf{x},w)\in\mathcal{X}(\mathcal{G},\mathbf{u})\times\mathbb{R} : w\ge f(\mathbf{x})\}$$

是 f在 $\mathcal{X}(\mathcal{G},\ \mathbf{u})$ 下的上图。定理5.3 给出了 $\operatorname{conv}\left(\mathcal{P}_f^{\mathcal{X}(\mathcal{G},\mathbf{u})}\right)$ 的完整刻画，而该凸包恰好是多面体。因此，（32）变为一个包含连续变量和指数数量线性约束的线性规划。图5 （右）以示意图展示了一个示例中的这一结果。

定理 5。3 *[90] 在某些技术条件下，与有效排列相关的所有 DR 不等式，以及混合整数舍入（MIR）不等式、盒约束和单调性约束，完整描述了 conv* $(\mathcal{X}(\mathcal{G},\mathbf{u}))$*，从而描述 conv* $\left(\mathcal{P}_f^{\mathcal{X}(\mathcal{G},\mathbf{u})}\right)$。

接下来，我们定义定理 5.3 中提到的 MIR 不等式和 DR 不等式。MIR 不等式是将混合整数可行集 $\mathcal{X}(\mathcal{G}, \mathbf{u})$ 凸化所必需的。令

$$\Psi := \{\psi \in \mathcal{V} : u_\psi \notin \mathbb{Z}, R^+(\psi) \cap N \neq \emptyset\}$$

是对应于非整数上界变量的顶点集合，其中每个变量至少有一个离散后代。对于任意 $\psi \in \Psi$, $u_\psi \notin \mathbb{Z}$。这类变量的存在往往会使 $\mathcal{X}(\mathcal{G}, \mathbf{u})$ 的连续松弛与其凸包不同。因此，我们需要额外的有效不等式来强化连续松弛。与 ψ 相关的 MIR 不等式如下所示：

$$-x_{\mathrm{ch}(\psi)} - \frac{u_\psi - x_\psi}{u_\psi - \lfloor u_\psi \rfloor} \leq -\lceil u_\psi \rceil,$$

其中 ch (ψ) 是一个 具有 $(\psi,\mathrm{ch}\,(\psi))$ 的顶点 $\in \mathcal{A}$。我们在例5中给出了 MIR 不等式的示例 .4.

例*5.4* 考虑可行集 $\mathcal{X}(\mathcal{G}, u) = \{x_1 \in \mathbb{R}, x_2 \in \mathbb{Z}: 0 \leq x_1 \leq 2.4, 0 \leq x_2 \leq 3, x_1 \leq x_2\}$。我们注意到 $\Psi = \{1\}$。在图 7中，黑色实心圆点和线段表示混合整数可行点。虚线区域是 $\mathcal{X}(\mathcal{G}, u)$ 的连续松弛，紫色区域表示 *conv* $(\mathcal{X}(\mathcal{G}, u))$。我们观察到， $\mathcal{X}(\mathcal{G}, u)$ 的连续松弛与凸包并不一致。紫色虚线对应于与 $\psi = 1$ 相关的 *MIR* 不等式 $-x_2 -(2.4 - x_1)/0.4 \leq -3$ 。该 *MIR* 不等式以及盒约束和单调性约束完整刻画了 *conv* $(\mathcal{X}(\mathcal{G}, u))$。

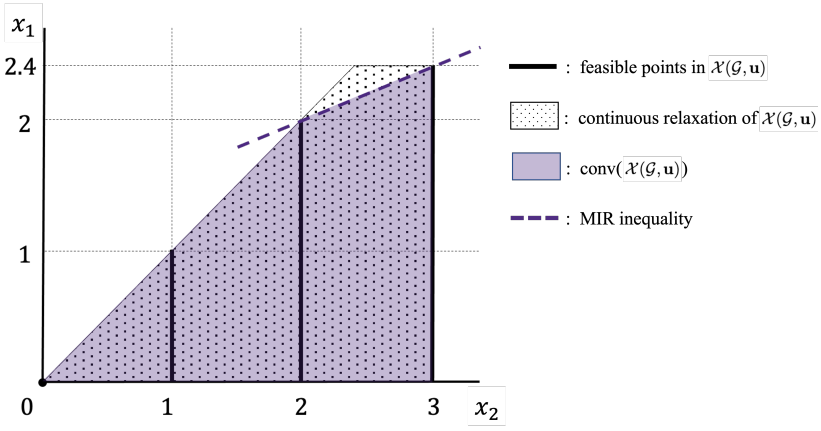

图7：MIR不等式的一个示例。

正如下一个定理所述，与所有 $\psi \in \Psi$ 相关联的MIR不等式对于混合整数可行集的凸包描述十分重要。

定理 5。**5** *[90]* 与所有 $\psi \in \Psi$相关联的*MIR*不等式，以及盒约束和单调性约束，在某些技术条件下，足以描述可行集 $\mathcal{X}(\mathcal{G}, \mathbf{u})$ 的凸包。

DR 不等式还涉及上图变量 w。令 $\boldsymbol{\delta}$ 为 $\mathcal{V}$的一个排列。我们定义与 $\boldsymbol{\delta}$ 相关联的*DR* 不等式 如下：

$$w \geq \sum_{k=1}^{|\mathcal{V}|} t(\boldsymbol{\delta}, \mathbf{x})_k [f(P(\boldsymbol{\delta}, k)) - f(P(\boldsymbol{\delta}, k-1))], \tag{33}$$

对于每个 $k \in \{0, 1, \ldots, |\mathcal{V}|\}$， $t(\boldsymbol{\delta}, \mathbf{x})_k$是关于 x 的线性函数，而 $P(\boldsymbol{\delta}, k)$ 是 conv $(\mathcal{X}(\mathcal{G}, \mathbf{u}))$ 的一个极点。 $t(\cdot, \cdot)$ 和 $P(\cdot)$ 的闭式形式见 [90]。需要指出的是，所有 DR 不等式都是线性且齐次的。此外，当 f 是经典子模函数时，它们化为著名的 EPI (7)。鉴于每个 DR 不等式都与 $\mathcal{V}$ 的一个排列相关联（该排列满足某些性质），因此 DR 不等式的数量呈指数级增长。每个 DR 不等式都可以通过一个 $\mathcal{O}(|\mathcal{V}|^2 \log |\mathcal{V}|)$ 算法 [90] 精确分离。基于分离与优化之间的等价性，我们得出以下结论：

推论 5。**6** *[90]* 在混合整数可行集 $\mathcal{X}(\mathcal{G}, \mathbf{u})$ 上的 *DR*-次模最小化，在满足某些技术条件时可在多项式时间内求解。

6 结论

在本教程中，我们概述子模性及其两种扩展，并介绍广义子模性。我们通过详细描述其应用实例，展示广义子模性的建模能力。这些应用会产生具有广义子模目标函数的混合整数非线性优化问题。由于决策空间是离散的，且问题具有非线性，这些优化问题难以精确求解。我们采用混合整数规划技术克服这些挑战。具体而言，我们将原问题重新表述为混合 0-1 规划或连续线性规划，并通过应用延迟约束生成框架、发挥混合整数线性规划求解器的优势，设计有效的精确求解方法。

致谢

本研究部分由ONR资助项目N00014-22-1-2602资助。

参考文献

[1] Abujubbeh, M., Al-Turjman, F. 和 Fahrioglu, M.（2019）。智能电网中的软件定义无线传感器网络：综述。《可持续城市与社会》，51:101754。[2] Ahmed, S. 和 Atamt¨urk, A.（2011）。最大化一类子模效用函数。《数学规划》，128(1):149–169。[3] Ando, K. 和 Fujishige, S.（1996）。关于双次模多面体的结构。《数学规划》，74(3):293–317。[4] Aspvall, B. 和 Shiloach, Y.（1980）。求解每个不等式含两个变量的线性不等式组的多项式时间算法。SIAM 计算杂志，9(4):827–845。[5] Atamt¨urk, A. 和 G´omez, A.（2020）。锥二次混合0-1优化中的子模性。《运筹学》，68(2):609–630。[6] Atamt¨urk, A. 和 G´omez, A.（2022）。含指示变量的二次优化中的超模性与有效不等式。《数学规划》，提前发表：1–44。[7] Atamt¨urk, A. 和 Jeon, H.（2019）。含指示变量的均值-风险优化中的提升多拟阵不等式。《全球优化杂志》，73(4):677–699。[8] Atamt¨urk, A. 和 Narayanan, V.（2008）。离散优化中的多拟阵与均值-风险最小化。《运筹学快报》，36(5):618–622。[9] Atamt¨urk, A. 和 Narayanan, V.（2022）。子模函数最小化与对偶性。《数学规划》，196(1–2):57–67。[10] Bach, F.（2019）。子模函数：从离散域到连续域。《数学规划》，175(1):419–459。[11] Bian, A., Levy, K., Krause, A. 和 Buhmann, J. M.（2017a）。连续 DR-子模最大化：结构与算法。《神经信息处理系统进展》，30。[12] Bian, A. A., Mirzasoleiman, B., Buhmann, J. 和 Krause, A.（2017b）。有保证的非凸优化：连续域上的次模最大化。载于《人工智能与统计》，第111–120页。机器学习研究论文集。[13] Birge, J. R. 和 Louveaux, F.（2011）。随机规划导论。施普林格科学与商业媒体。[14] Bodik, P., Hong, W., Guestrin, C., Madden, S., Paskin, M. 和 Thibaux, R.（2004）。英特尔实验室数据。在线数据集。http://db.csail.mit.edu/labdata/labdata.html。

[15] Bouchet, A.（1987）。贪心算法与对称拟阵。《数学规划》，38(2):147–159。[16] Calinescu, G., Chekuri, C., P´al, M. 和 Vondr´ak, J.（2007）。拟阵约束下次模集合函数的最大化。载于《整数规划与组合优化国际会议》，第182–196页。施普林格。[17] Cohen, E. 和 Megiddo, N.（1991）。每个不等式含两个变量的线性不等式的改进算法。载于《第二十三届 ACM 计算理论年度研讨会论文集》，第145–155页。[18] Coniglio, S., Furini, F. 和 Ljubi´c, I.（2022）。对由集合并运算符复合的凹效用函数进行次模最大化及其在最大覆盖选址问题中的应用。《数学规划》，196(1-2):9–56。[19] Cornuejols, G., Fisher, M. 和 Nemhauser, G. L.（1977）。无容量限制选址问题。载于《离散数学年刊》，第1卷，第163–177页。爱思唯尔。[20] Cunningham, W. H.（1985）。关于子模函数最小化。Combinatorica，5(3):185–192。[21] Edmonds, J.（2003）。子模函数、拟阵与某些多面体。载于《组合优化——尤里卡，你变小了！》，第11–26页。施普林格。[22] Ene, A. 和 Nguyen, H. L.（2016）。一种优化具有边际收益递减性质的格次模函数的归约方法。arXiv 预印本 arXiv:1606.08362。[23] Feldman, E. R., Lehrer, F. 和 Ray, T.（1966）。连续规模经济下的仓库选址。Management Science，12(9):670–684。[24] Fujishige, S. 和 Iwata, S.（2005）。双子模函数最小化。SIAM 离散数学杂志，19(4):1065–1073。[25] Ghayvat, H., Liu, J., Mukhopadhyay, S. C. 和 Gui, X.（2015）。健康监测传感器网络：面向辅助生活的智能家居方案与实现。IEEE Sensors Journal，15(12):7341–7348。[26] G´omez, A.（2018）。含指示变量的非线性优化中的子模性与有效不等式。http://www.optimization-online.org/DB_FILE/2018/11/6925.pdf。[27] Gr¨otschel, M., Lov´asz, L. 和 Schrijver, A.（1981）。椭球法及其在组合优化中的应用。Combinatorica，1(2):169–197。[28] Hajiaghayi, M. T., Mahdian, M. 和 Mirrokni, V. S.（2003）。采用一般成本函数的设施选址问题。Networks: An International Journal，42(1):42–47。[29] Han, S., G´omez, A. 和 Pang, J.-S.（2022）。关于组合马尔可夫随机场的多项式时间可解性。arXiv 预印本 arXiv:2209.13161。[30] Hassani, H., Soltanolkotabi, M. 和 Karbasi, A.（2017）。次模最大化的梯度法。《神经信息处理系统进展》，30。[31] Hassin, R. 和 Tamir, A.（1989）。拟阵上的双参数目标函数类最大化。Mathematics of Operations Research，14(2):362–375。[32] Hochbaum, D. S. 和 Naor, J.（1994）。每个不等式含两个变量的线性规划与整数规划的简单快速算法。SIAM 计算杂志，23(6):1179–1192。[33] Hu, Y., Wang, R. 和 Chen, F.（2019）。用于从在线健康论坛检测药物相互作用（DDIs）的双次模优化（BSMO）。Journal of Healthcare Informatics Research，3(1):19–42。[34] Huber, A. 和 Kolmogorov, V.（2012）。迈向 DR-子模函数最小化。载于《组合优化国际研讨会》，第451–462页。施普林格。[35] Iwata

, S. (2008) . 子模函数最小化 . 数学规划 , 112(1):45 –64.

[36] Iwata，S.、Fleischer，L. 和 Fujishige，S.（2001）。用于最小化子模函数的组合强多项式算法。《ACM学报》（JACM），48(4):761–777。[37] Iwata，S. 和 Orlin，J. B.（2009）。一种简单的子模函数最小化组合算法。载于第二十届ACM-SIAM离散算法研讨会论文集，第1230–1237页。SIAM。[38] Iwata，S.、Tanigawa，S.-i. 和 Yoshida，Y.（2013）。双子模函数最大化及其扩展。技术报告，METR 2013-16，东京大学。[39] Iwata，S.、Tanigawa，S.-i. 和 Yoshida，Y.（2016）。改进的 k-子模函数最大化近似算法。载于第二十七届ACM-SIAM离散算法研讨会论文集，第404–413页。SIAM。[40] Jegelka，S. 和 Bilmes，J.（2011）。超越子模能量的子模性：图割中的耦合边。载于CVPR 2011，第1897–1904页。IEEE。[41] Kempe，D.、Kleinberg，J. 和 Tardos，´E.（2015）。通过社交网络最大化影响力传播。《计算理论》，11(4):105–147。[42] Kılınc-Karzan，F.、K¨uc¨ukyavuz，S. 和 Lee，D.（2022）。联合机会约束规划与混合集合的交集：子模性视角。《数学规划》，195(1–2):283–326。[43] Kılınc-Karzan，F.、K¨uc¨ukyavuz，S.、Lee，D. 和 Shafieezadeh-Abadeh，S.（2020）。锥混合二元集合：凸包刻画及应用。arXiv预印本，arXiv:2012.14698。[44] Krause，A.、Singh，A. 和 Guestrin，C.（2008）。高斯过程中的近最优传感器布置：理论、高效算法与实证研究。《机器学习研究杂志》，9(2)。[45] Laporte，G. 和 Louveaux，F. V.（1993）。具有完全补救的随机整数规划的整数L形方法。《运筹学快报》，13(3):133–142。[46] Lee，J.、Mirrokni, V. S.、Nagarajan，V. 和 Sviridenko，M.（2010）。在拟阵或背包约束下最大化非单调子模函数。SIAM 离散数学杂志，23(4):2053–2078。[47] Lee，Y. T.、Sidford，A. 和 Wong，S. C.-w.（2015）。更快的切平面方法及其对组合优化和凸优化的影响。载于 2015 年 IEEE 第 56 届计算机科学基础年度研讨会论文集，第1049–1065页。IEEE。[48] Lov´asz，L.（1983）。子模函数与凸性。载于《数学规划：现状》，第235–257页。施普林格。[49] McCormick，S. T.（2005）。子模函数最小化。《运筹学与管理科学手册》，12:321–391。[50] McCormick，S. T. 和 Fujishige，S.（2010）。双子模函数最小化的强多项式与完全组合算法。《数学规划》，122(1):87。[51] Medal，H. 和 Ahanor，I.（2022）。最大化不可因式分解的单调连续子模函数的空间分支定界方法。arXiv预印本，arXiv:2208.13886。[52] Motzkin，T. S. 和 Straus，E. G.（1965）。图的最大值及Tur´án定理的新证明。《加拿大数学杂志》，17:533–540。[53] Nemhauser，G. L. 和 Wolsey，L. A.（1978）。用于近似子模集合函数最大值的最佳算法。Mathematics of Operations Research，3(3):177–188。[54] Nemhauser，G. L.、Wolsey，L. A. 和 Fisher，M. L.（1978）。最大化子模集合函数的近似分析——I。《数学规划》，14:265–294。[55] Ohsaka，N. 和 Yoshida，Y.（2015）。带规模约束的单调 k-子模函数最大化。载于Cortes，C.、Lawrence，N. D.、Lee，D. D.、Sugiyama，M. 和 Garnett，R.（编），《神经信息处理系统进展28》，第694–702页。Curran Associates，Inc.

[56] Onn, S.（2003）。凸拟阵优化。《SIAM 离散数学杂志》，17(2)：249–253。[57] Orlin, J. B.（2009）。一种更快的子模函数最小化强多项式时间算法。数学规划，118(2)：237–251。[58] Orlin, J. B., Schulz, A. S., and Udwani, R.（2018）。鲁棒单调子模函数最大化。数学规划，172(1)：505–537。[59] Pirmohamed, M. and Orme, M.（1998）。具有临床重要性的药物相互作用。《药物不良反应》Davies 教科书，第888–912页。[60] Qi, L.（1988）。有向子模性、有向拟阵与有向子模流。数学规划，42(1-3)：579–599。[61] Rokach, L. and Maimon, O. Z.（2008）。决策树数据挖掘：理论与应用，第69卷。世界科学出版社。[62] Sadeghi, O. and Fazel, M.（2021）。用于单调强DR-子模最大化的更快一阶算法。arXiv预印本 arXiv:2111.07990。[63] Saeys, Y., Inza, I., and Larra˜naga, P.（2007）。生物信息学中特征选择技术综述。《生物信息学》，23(19)：2507–2517。[64] Sakaue, S.（2017）。关于在拟阵约束下最大化单调 k子模函数。《离散优化》，23：105–113。[65] Schrijver, A.（2000）。一种在强多项式时间内最小化子模函数的组合算法。《组合理论杂志B辑》，80(2)：346–355。[66] Shalev-Shwartz, S. and Ben-David, S.（2014）。理解机器学习：从理论到算法。剑桥大学出版社。[67] Shen, H. and Jiang, R.（2022）。具有Wasserstein模糊集的机会约束集合覆盖。数学规划，预先在线发表：1–54。[68] Shi, X., Prokopyev, O. A., and Zeng, B.（2022）。一组次模最大化问题的序列无关提升。数学规划，196(1-2)：69–114。[69] Shostak, R.（1981）。通过计算环路余数判定线性不等式。《ACM学报》（JACM），28(4)：769–779。[70] Singh, A., Guillory, A., and Bilmes, J.（2012）。关于双子模最大化。载于《人工智能与统计》，第1055–1063页。[71] Soma, T. and Yoshida, Y.（2015）。通过整数格上的边际收益递减性质推广子模覆盖。《神经信息处理系统进展》，28：847–855。[72] Soma, T. and Yoshida, Y.（2017）。非单调DR-子模函数最大化。载于《AAAI人工智能会议论文集》，第31卷(1)。[73] Soma, T. and Yoshida, Y.（2018）。在整数格上最大化单调子模函数。数学规划，172(1)：539–563。[74] Staib, M. and Jegelka, S.（2017）。通过连续子模函数实现鲁棒预算分配。载于《国际机器学习会议》，第3230–3240页。机器学习研究论文集。[75] Sviridenko, M.（2004）。关于在背包约束下最大化次模集合函数的一则注记。《运筹学快报》，32(1)：41–43。[76] Svitkina, Z. and Fleischer, L.（2011）。子模近似：基于采样的算法与下界。SIAM 计算杂志，40(6)：1715–1737。

[77] Tatonetti, N. P., Fernald, G. H., and Altman, R. B.（2012）。一种用于在不良事件报告中识别隐藏药物相互作用的新型信号检测算法。美国医学信息学会杂志，19(1)：79–85。[78] Topkis, D. M.（1978）。在格上最小化子模函数。《运筹学》，26(2)：305–321。[79] Ward, J. and ˇZivn´y, S.（2014）。最大化双次模函数和 k子模函数。载于第二十五届ACM-SIAM离散算法研讨会论文集，第1468–1481页。SIAM。[80] Ward, J. and ˇZivn´y, S.（2016）。最大化 k子模函数及其他问题。《ACM算法汇刊》（TALG），12(4)：1–26。[81] Wolsey, L. A. and Nemhauser, G. L.（1999）。整数与组合优化，第55卷。John Wiley & Sons。[82] Wu, H.-H. and K¨u¸c¨ukyavuz, S.（2018）。一种用于社交网络影响力最大化的两阶段随机规划方法。《计算优化与应用》，69(3)：563–595。[83] Wu, H.-H. and K¨u¸c¨ukyavuz, S.（2019）。带有机会约束预言机的概率部分集合覆盖。SIAM优化杂志，29(1)：690–718。[84] Wu, H.-H. and K¨u¸c¨ukyavuz, S.（2020）。一种针对一类随机次模函数条件风险价值约束最大化的精确方法。《运筹学快报》，48(3)：356–361。[85] Xie, W.（2021）。关于具有Wasserstein距离的分布鲁棒机会约束规划。数学规划，186(1–2)：115–155。[86] Yu, J. and Ahmed, S.（2017a）。带约束的一类子模效用函数最大化。数学规划，162(1-2)：145–164。[87] Yu, J. and Ahmed, S.（2017b）。一类基数约束子模最小化问题的多面体结果。《离散优化》，24：87–102。[88] Yu, Q. and K¨u¸c¨ukyavuz, S.（2020）。双子模函数最小化的多面体方法。《运筹学快报》，49(1)：5–10。[89] Yu, Q. and K¨u¸c¨ukyavuz, S.（2021）。一种用于 k子模函数最大化的精确切平面方法。《离散优化》，42：100670。[90] Yu, Q. and K¨u¸c¨ukyavuz, S.（2022）。受约束的混合整数DR次模最小化。arXiv预印本 arXiv:2211.07726。[91] Yu, Q. and K¨u¸c¨ukyavuz, S.（2023）。基数约束下某类凹次模最小化问题的强有效不等式。即将在数学规划上发表，第1–59页。[92] Zanella, A., Bui, N., Castellani, A., Vangelista, L., and Zorzi, M.（2014）。面向智慧城市的物联网。IEEE物联网杂志，1(1)：22–32。[93] Zhang, Y., Jiang, R., and Shen, S.（2018）。基于均值—协方差信息的含糊机会约束二进制规划。SIAM 优化杂志，28(4)：2922–2944。