



中国科学院大学
University of Chinese Academy of Sciences

博士学位论文

三类应用优化问题的高效算法

作者姓名: 杨沐明

指导教师: 戴或虹 研究员

中国科学院数学与系统科学研究院

学位类别: 理学博士

学科专业: 计算数学

培养单位: 中国科学院数学与系统科学研究院

2021 年 6 月

Efficient Algorithms for Three Optimization Problems with
Application Background

A dissertation submitted to the
University of Chinese Academy of Sciences
in partial fulfillment of the requirement
for the degree of
Doctor of Philosophy
in Computational Mathematics

By

Yang Muming

Supervisor: Professor Dai Yu-Hong

Academy of Mathematics and Systems Science
Chinese Academy of Sciences

June, 2021

中国科学院大学 学位论文原创性声明

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明或致谢。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日 期：

中国科学院大学 学位论文授权使用声明

本人完全了解并同意遵守中国科学院大学有关保存和使用学位论文的规定，即中国科学院大学有权保留送交学位论文的副本，允许该论文被查阅，可以按照学术研究公开原则和保护知识产权的原则公布该论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存、汇编本学位论文。

涉密及延迟公开的学位论文在解密或延迟期后适用本声明。

作者签名：

日 期：

导师签名：

日 期：

摘要

最优化问题是在给定约束限制下寻求最优决策的问题, 常见于生产、生活中的各个环节. 为解决最优化问题而发展出的最优化理论与方法被广泛应用于通信、金融、石油化工、交通等领域. 因此最优化是一门具有极高应用价值的学科. 本文主要研究了石油运输、航天制导和物流过程优化领域中的难点问题, 并为这些问题设计了新的高效算法.

对于热油管道问题, 我们给出了一个混合整数非线性规划模型, 并证明了它是 \mathcal{NP} 难的. 通过分析这一模型的两个松弛问题在一定条件下的等价性, 本文设计了基于分支定界的全局优化算法 **HOPBB**, 并证明了算法的有限终止性和全局最优性. 此外, 我们还设计了预处理方法保证算法可行以及引入外逼近方法加速松弛问题的求解. 将 **HOPBB** 算法应用于实际数据所得的最优方案节省了当前方案 6.83% 的运行费用. **HOPBB** 算法的高效性也在与通用混合整数非线性规划求解器的对比中得到了充分的验证.

对于动力下降制导问题, 我们考虑了带有气动阻力和推力与速度方向夹角约束的动力下降制导问题的最优控制模型. 这一模型经无损凸化、时间离散化和线性化步骤后转化为二阶锥规划. 我们分析了已有的序列凸化方法在求解此模型时出现的数值不稳定和迭代效率较低的问题. 基于部分非凸约束计算困难和时间变量收敛更快的特点, 本文提出了一个两阶段的序列凸化方法. 数值实验的结果表明两阶段方法比单阶段方法更加稳定且高效. 为了研究一般动力下降制导问题有收敛性保证的方法, 我们将它的最优控制模型转化为非线性二阶锥规划进行分析. 通过引入 ℓ_1 势函数和非光滑线搜索, 本文将求解仅含非线性锥约束的可行方向内点算法推广至求解更一般的非线性二阶锥规划问题, 并分析了方法的可行性和全局收敛性. 对随机产生的模型问题和动力下降制导问题的数值测试展示了所提方法相比于已有非线性规划和非线性二阶锥规划算法在求解速度、解的质量和稳定性几个方面的优势.

对于 **AGV** 道口分布问题, 我们将其转化为多商品设施选址模型, 并利用整数规划求解器进行求解. 为了更快得到高质量可行解, 本文提出了紧问题启发式方法和结合邻域搜索的改进方法, 并说明了两种启发式方法可应用于任何可行的实

例. 数值实验的结果表明紧问题启发式可以大大提升整数规划求解器求解部分问题的效率. 对于紧问题启发式效果不佳的情形, 结合邻域搜索的改进启发式方法也可以快速获得问题的高质量可行解.

关键词: 热油管道问题, 动力下降制导问题, AGV 道口分布问题, 设施选址问题, 分支定界方法, 序列凸化, 非线性二阶锥规划, 内点法, 启发式方法

Abstract

Optimization problem is a problem of seeking optimal decision under given constraints, which is common in the whole procedure of production and daily life. The optimization theory and method developed for solving optimization problems are widely implemented in communication, finance, petrochemical industry, transportation and other fields. Therefore optimization is a subject with the high application value. This thesis mainly studied the difficult problems in oil transport, aerospace guidance and logistics procedure optimization fields, designed new efficient algorithms for them.

For the heated oil pipeline problem, we introduced a mixed integer nonlinear programming model, and proved its \mathcal{NP} -hardness. Through the analysis on the equivalence between two relaxation problems of the model under some conditions, we designed a global optimization algorithm called HOPBB based on the branch-and-bound framework, and proved the finite termination and global optimality properties of the algorithm. Besides, we designed a preprocessing algorithm to guarantee the feasibility of the algorithm, and introduced the outer approximation method to accelerate the solution procedure of the relaxation problems. The optimal scheme obtained by implementing HOPBB algorithm to the real data saves 6.83% operation cost of the current scheme. The high efficiency of the HOPBB algorithm is fully verified under the comparison among some general mixed integer nonlinear programming solvers.

For the powered descent guidance problem, we considered the optimal control model with aerodynamic drag and the angle constraints of thrust and velocity. By lossless convexification, time discretization and linearization procedures, the optimal control model is transformed to a second order cone programming. We analyzed the numerical instability and low efficiency of the existing successive convexification method when solving this model. Based on the phenomena that some nonconvex constraints are hard to solve and the time variable converges faster than other variables, this thesis proposed a two-stage successive convexification method. The numerical results showed that the two-stage method is more stable and efficient than the single stage method. For

analyzing general methods for powered descent guidance problems with convergence guarantee, we abstracted the optimal control model into a nonlinear second order cone programming. By introducing the ℓ_1 potential function and a nonsmooth line search technique, this thesis generalized the feasible direction interior point method for solving the problem with nonlinear cone constraints to the method for solving more general nonlinear second order cone programming problems, and analyzed the feasibility and global convergence properties of the method. The numerical tests on random problems and the powered descent guidance problems showed that the proposed method has advantages on the solution speed, the quality of the solution and the stability comparing with some existing nonlinear programming and nonlinear second order cone programming methods.

For the AGV dock distribution problem, we transformed it into a multi-product facility location model, and solved it with the integer programming solver. For getting high quality feasible solutions faster, this thesis proposed the compact problem heuristic and the improving method combining with the local search, and proved that two heuristic methods can be implemented on any feasible instance. The numerical results showed that the compact problem heuristic improves the efficiency of the integer programming solver on part of the problems. For the cases that the compact problem heuristic works not well, the improving method combining with the local search can achieve high quality feasible solutions quickly.

Keywords: heated oil pipeline problem, powered descent guidance problem, AGV dock distribution problem, facility location problem, branch-and-bound, successive convexification, nonlinear second order cone programming, interior point method, heuristic method

目 录

| | |
|------------------------------|----|
| 第 1 章 引言 | 1 |
| 1.1 问题背景及研究现状 | 1 |
| 1.1.1 热油管道问题 | 1 |
| 1.1.2 动力下降制导问题 | 3 |
| 1.1.3 AGV 道口分布问题 | 5 |
| 1.2 预备知识 | 6 |
| 1.2.1 整数规划与分支定界算法 | 6 |
| 1.2.2 二阶锥及其 Jordan 代数 | 8 |
| 1.2.3 非线性二阶锥规划 | 9 |
| 1.3 本文主要贡献 | 10 |
| 第 2 章 热油管道问题的确定性全局优化算法 | 13 |
| 2.1 混合整数非线性规划模型 | 13 |
| 2.2 全局优化算法 | 18 |
| 2.2.1 非凸松弛问题 | 18 |
| 2.2.2 凸松弛问题 | 19 |
| 2.2.3 松弛问题的等价性 | 21 |
| 2.2.4 HOPBB 算法 | 27 |
| 2.3 预处理算法和外逼近方法 | 28 |
| 2.3.1 预处理算法 | 28 |
| 2.3.2 外逼近方法和热启动策略 | 32 |
| 2.4 数值实验 | 33 |
| 2.5 小结 | 42 |
| 第 3 章 动力下降制导问题的两类模型和算法 | 43 |
| 3.1 最优控制模型 | 43 |
| 3.2 序列凸化方法 | 46 |
| 3.2.1 模型转化 | 46 |
| 3.2.2 SOCP 模型 | 47 |
| 3.2.3 模型求解 | 48 |
| 3.2.4 两阶段法 | 49 |
| 3.3 非线性二阶锥规划模型与可行方向法 | 51 |
| 3.3.1 下降方向与可行方向 | 52 |

| | |
|--------------------------------------|-----------|
| 3.3.2 可行方向内点算法 | 55 |
| 3.3.3 收敛性分析 | 57 |
| 3.3.4 FDIPA 的一些解释 | 59 |
| 3.4 数值实验 | 60 |
| 3.4.1 两阶段序列凸化方法 | 60 |
| 3.4.2 可行方向内点法 | 62 |
| 3.5 小结 | 68 |
| 第 4 章 AGV 道口分布问题的启发式方法 | 69 |
| 4.1 整数规划模型 | 69 |
| 4.2 基于松弛解的启发式算法 | 71 |
| 4.2.1 紧问题启发式 | 72 |
| 4.2.2 结合邻域搜索的改进启发式 | 74 |
| 4.3 数值实验 | 82 |
| 4.4 小结 | 84 |
| 第 5 章 总结与展望 | 85 |
| 参考文献 | 87 |
| 作者简历及攻读学位期间发表的学术论文与研究成果 | 95 |
| 致谢 | 97 |

图形列表

| | |
|--|----|
| 1.1 热油管道及热泵站示意图. | 3 |
| 2.1 MINLP 模型中的部分常量 (蓝色正体文字) 和变量 (红色斜体文字) 符号含义. | 14 |
| 2.2 例 2.1 中的压头曲线 (左), 温度曲线 (中) 和方案费用 (右). | 20 |
| 2.3 (HOP)、(HOPnr1) 及 (HOPnr2) 的关系示意图. | 24 |
| 2.4 例 2.2 中的压头曲线. 前三个子图中的实线和虚线分别代表第 1、2、3 步和前一步的压头曲线. 第四个子图将最终曲线 (绿色圆形标记实线) 与之前迭代中步骤 2 的两条不可行曲线 (虚线) 比较展示了不可行性是如何消除的. | 25 |
| 2.5 Q-T 管线布局示意图. | 34 |
| 2.6 Q-T 管线的里程和高程图. | 34 |
| 2.7 原油粘温曲线. | 36 |
| 2.8 实际方案与最优方案的压力 (左)、温度 (右) 曲线. | 37 |
| 2.9 不同出站温度上界下的最优运行费用. | 38 |
| 2.10 实际方案及出站温度上界分别取 48°C 和 54°C 的两个最优方案的运行费用. | 38 |
| 2.11 不同方法求解 10 个 (HOP) 问题实例的表现. 同一时刻求解实例个数越多, 性能越好. | 40 |
| 3.1 火箭受力分析及飞行场景示意图. | 45 |
| 3.2 各变量与最终解的距离. | 50 |
| 3.3 3D 位置轨迹 (左) 和 2D 位置速度轨迹 (右). 左图中蓝色曲线表示位置轨迹, 红色箭头表示加速度, 绿色圆形表示着陆点. | 61 |
| 3.4 质量 (左)、推力大小 (中) 和推力与速度反方向夹角曲线 (右). | 61 |
| 3.5 最终解处实际推力大小 $\ T\ $ 与推力上界 Γ 的关系. | 61 |
| 3.6 不同终端时刻猜测值 τ_f 下的 t_f 收敛结果. | 62 |
| 3.7 各阶段最终解中的速度 (第一行)、推力 (第二行)、速度大小 (第三行左)、推力大小 (第三行中) 和推力与速度反方向夹角曲线 (第三行右). | 63 |
| 4.1 AGV 道口分布问题实例的上、下界随求解时间的变化曲线. | 70 |
| 4.2 移位: 将一个设施转移到一个无设施的候选位置. | 76 |
| 4.3 增加: 在一个无设施的候选位置建造一个设施. | 76 |
| 4.4 互换: 交换两个设施所生产的商品种类. | 77 |
| 4.5 移除: 将一个设施去除. | 77 |

表格列表

| | |
|--|----|
| 2.1 Q-T 热油管道问题中的部分常量和参数设置..... | 35 |
| 2.2 Q-T 管线所输原油的动力粘度..... | 35 |
| 2.3 Q-T 管线各热泵站的数据..... | 36 |
| 2.4 最优方案与实际方案的加压、加热方案及费用对比 (费用单位: 人民币/天)..... | 37 |
| 2.5 求解 Q-T 热油管道问题的不同方法效果对比..... | 39 |
| 2.6 不同方法求解 10 个 (HOP) 实例的求解时间, 迭代次数和 gap 对比. BARON 由于遇到数值问题而在未满足 gap 的要求情况下提前终止... | 41 |
| 3.1 测试问题数据及参数取值..... | 60 |
| 3.2 单阶段法和两阶段法的迭代次数对比..... | 62 |
| 3.3 测试问题规模..... | 64 |
| 3.4 测试问题 (3.25) 的求解时间和迭代次数对比..... | 64 |
| 3.5 测试问题 (3.25) 的可行性对比..... | 65 |
| 3.6 动力下降制导测试问题的迭代次数和可行性对比..... | 67 |
| 3.7 NSOCP 求解方法性质对比..... | 68 |
| 4.1 某 (FLP) 问题的松弛解 y^{LP} | 73 |
| 4.2 第一组测试算例..... | 82 |
| 4.3 第一组算例上界和 gap 对比..... | 82 |
| 4.4 第二组测试算例..... | 83 |
| 4.5 第二组算例上界、gap 和时间对比 (时间单位: 秒)..... | 84 |

符号列表

通用符号

| 符号 | 含义 |
|----------------|------------|
| \mathbb{R} | 实数集合 |
| \mathbb{R}_+ | 非负实数集合 |
| \mathbb{Z}_+ | 非负整数集合 |
| S_+ | 实对称半正定矩阵集合 |
| S_{++} | 实对称正定矩阵集合 |

集合的右上角标表示维数, 如 \mathbb{R}^n 表示 n 维实数空间, S_+^n 表示 n 阶实对称半正定矩阵集合. 如不做特别说明, 本文对向量的符号做以下规定:

1. 所有向量均为列向量;
2. 若向量正上方 (或正下方) 有标记, 则自动将其分量的记号也视为有同样的标记. 如 $x = (x_1, x_2)$, 则 $\hat{x} = (\hat{x}_1, \hat{x}_2)$, \underline{x}_2 可视为 \underline{x} 的第二个 (或块) 分量;
3. 将 (x, y) 视为向量 x 和 y 组合后的列向量, 即 $(x^\top, y^\top)^\top$.

此外, 范数符号 $\|\cdot\|$ 如不加脚标则默认表示欧氏范数, 即 $\|x\| := \sqrt{\sum_{i=1}^n x_i^2}$.

第 2 章符号

如不做特别说明, 仅含下标 j 的符号表示第 j 个热泵站内的物理量, 如 N_j^{CP} 、 $H_{in}^{S_j}$ 等. 含下标 j 和 r 的符号表示第 j 站和第 $j+1$ 站间第 r 个管道片段内的物理量, 如 $H_{out}^{P_{jr}}$ 等. 所有关于压头 H 和温度 T (除 T_g 和 T_f) 的物理量和变量均是非负的.

| 符号 | 含义 | 单位 |
|------------|-------------------------|-----------------|
| N^S | 热泵站个数 | - |
| N_j^{CP} | 定速泵个数 | - |
| N_j^{SP} | 调速泵个数 | - |
| N_j^P | 第 j 站和 $j+1$ 站间的管道片段数 | - |
| ρ | 原油密度 | kg/m^3 |

| | | |
|---|-----------------|------------------------|
| π | 圆周率 | - |
| c | 原油比热容 | J/(kg·°C) |
| g | 重力加速度 | m/s ² |
| C_p | 电力费用单价 | RMB/J |
| C_f | 燃气费用单价 | RMB/m ³ |
| V_c | 燃气热值 | J/m ³ |
| ξ_j^{CP} | 定速泵效率 | - |
| ξ_j^{SP} | 调速泵效率 | - |
| η_j | 加热炉效率 | - |
| K_{jr} | 热传导系数 | W/(m ² ·°C) |
| Q_{jr} | 体积流量 | m ³ /h |
| L_{jr} | 管道片段长度 | m |
| D_{jr} | 管道内径 | m |
| d_{jr} | 管道外径 | m |
| ΔZ_{jr} | 高程差 | m |
| H_j^{CP} | 定速泵压头 | m |
| $\underline{H}_j^{SP}(\overline{H}_j^{SP})$ | 调速泵最小 (最大) 压头 | m |
| $T_g^{P_{jr}}$ | 土壤温度 | °C |
| $T_f^{P_{jr}}$ | 因摩擦热导致的环境温度变化量 | °C |
| $\underline{x}_j(\overline{x}_j)$ | 最小 (最大) 定速泵开启数目 | - |
| $\underline{y}_j(\overline{y}_j)$ | 最小 (最大) 调速泵开启数目 | - |
| $\Delta \overline{T}_j$ | 最大温升 | °C |
| $\underline{H}_{in}^{S_j}(\underline{H}_{out}^{S_j})$ | 入 (出) 站压头下界 | m |
| $\overline{H}_{in}^{S_j}(\overline{H}_{out}^{S_j})$ | 入 (出) 站压头上界 | m |
| $\underline{T}_{in}^{S_j}(\underline{T}_{out}^{S_j})$ | 入 (出) 站温度下界 | °C |
| $\overline{T}_{in}^{S_j}(\overline{T}_{out}^{S_j})$ | 入 (出) 站温度上界 | °C |
| $\underline{H}_{out}^{P_{jr}}(\overline{H}_{out}^{P_{jr}})$ | 压头下 (上) 界 | m |
| x_j | 开启的定速泵数目 | - |

| | | |
|-------------------------------|----------|----|
| y_j | 开启的定速泵数目 | - |
| ΔH_j^{SP} | 调速泵输出总压头 | m |
| ΔT_j | 温度升高量 | °C |
| $H_{in}^{S_j}(H_{out}^{S_j})$ | 入(出)站压头 | m |
| $T_{in}^{S_j}(T_{out}^{S_j})$ | 入(出)站温度 | °C |
| $H_{out}^{P_{jr}}$ | 出口压头 | m |
| $T_{out}^{P_{jr}}$ | 出口温度 | °C |
| $T_{ave}^{P_{jr}}$ | 平均温度 | °C |
| F_{jr} | 水力摩擦 | m |

第 1 章 引言

最优化问题 (optimization problem)[1, 2] 是指在给定若干约束条件下, 如何找到一组决策方案使其在某个指标下达到最优的问题. 为解决最优化问题而发展出的最优化理论与方法被广泛应用于通信、金融、石油化工、交通等领域 [3–6]. 自 1947 年 Dantzig 提出线性规划的单纯形方法以来, 线性规划、非线性规划、整数规划等优化问题以及内点法、梯度法、分支定界方法、启发式方法等优化方法迅速发展起来 [7–11]. 这些理论和方法的出现大大促进了各种工业应用领域的发展, 对人类社会产生了深远的影响.

然而在一些传统应用行业中, 优化问题的解决方法没有得到很好地研究. 此外, 随着人类科技的进步以及社会的发展, 新应用场景下的新优化问题也在不断地产生. 这些问题往往带有规模大、求解困难, 或是可靠性和实时性要求极高的特点, 对最优化学科和工作者们提出了一个又一个挑战. 本论文从石油运输、航天制导和物流过程优化三个领域中的应用问题入手, 分别建立混合整数非线性规划模型、非线性二阶锥规划模型和 0-1 整数规划模型, 并设计确定性全局优化算法、两阶段序列凸化和可行方向内点方法以及基于线性规划松弛解的启发式方法对问题进行高效地求解. 本章首先介绍三个应用问题的背景以及研究动机和现状, 然后介绍与本文相关的基本优化知识, 最后列举本论文的主要贡献.

1.1 问题背景及研究现状

1.1.1 热油管道问题

原油作为众多石油产品的原材料, 对工业生产及日常生活都十分重要. 在原油提炼前, 原油需要从油田输运到提炼厂. 据不完全统计, 全世界超过 50% 的原油是通过管道进行运输. 在使用管道运输时, 原油需要进行加压处理以使其安全通过管线. 同时, 部分原油还需进行加热处理以避免凝管或因粘度过高而难以正常通过管道. 比如我国大庆油田出产的原油就有高达 32 度的凝固点. 对于这样的原油, 人们利用装配有加压泵和加热炉的热油管道 (heated oil pipeline, HOP) 进行输运. 对于长输热油管道, 对输运方案进行优化来降低输运成本是十分必要的.

热油管道的运行方案主要包括加压泵组合方案和泵、炉运行方案. 为了达到

安全运输的要求, 一个可行的方案需要满足进、出站和翻越点处的压力和温度界约束. 只要管线中的热泵站输出充足的压力和热量, 这样的可行方案总是存在的. 但是不同可行方案间的运行费用可能差距很大. 通常来说较高的输油温度需要较高的加热费用, 但同时由于油温越高粘度越低, 输运所需的压力就可以相应降低, 节省了加压费用. 热油管道问题就是在满足安全运输要求的前提下, 找到加压和加热总成本最低的运行方案.

为了获得最经济的方案, 我们可以将加压泵的启、停状态建模为整数变量, 将加热炉的提供的温升建模为连续变量, 以此得到热油管道问题的混合整数非线性规划 (mixed integer nonlinear programming, MINLP) 问题进行最优化求解. 热油管道问题相关的研究已有 30 年的历史. 据我们所知, 吴长春等 [12] 是最早研究稳态热油管道问题的学者. 他们在考虑加压泵、加热炉、温度变化以及站—管匹配的基础上给出了一个不精确的 MINLP 模型, 并设计了一个两级递阶方法. 该方法将 MINLP 模型分解成泵组合与管路最优匹配两个子模型, 并分别利用动态规划和非线性规划方法进行求解. 之后, 这一模型和方法还被应用于热油管道的能耗分析和预测 [13, 14]. 此外, 更多热油管道问题相关的工作聚焦于元启发式方法. Liu 等 [15] 同样考虑了稳态热油管道问题, 并在考虑多种加热炉型号的基础上提出一个新的 MINLP 模型. 他们利用改进基因算法求解这一模型. Zhou 等 [16] 针对非稳态的情形提出了一个动态热油管道模型. 这一模型首先计算土壤温度, 然后计算管道中的压力和温度数值. 他们结合粒子群和差分进化算法设计了 PSO-DE 方法求解这一模型. Zhang 等 [17] 结合 PSO-DE 方法和神经网络设计了在线优化方法. 此外, 还有学者利用线性逼近 [18], 非线性优化 [19], 模拟退火 [20], 遗传算法 [21] 等方法求解热油管道问题. 更多输油管道相关工作可参考综述文献 [22].

事实上, 为了更准确地反映管线的实际状况, 考虑更精细的管线配置建模以及水力计算是十分重要的. 如在热泵站中通常有两类加压泵, 定速泵和调速泵 (如图 1.1 所示). 根据两种泵的叶轮转速是否可调的性质, 它们分别可以给原油增加常量压力和变量压力. 我们在建模时应根据它们特性的不同进行分别考虑. 为了适应更多的实际生产情形, 管线的水力计算也应在更一般的模型上进行考虑. 此外, 绝大多数已有的工作或是聚焦于元启发式方法, 或是利用工程上有效但没有收敛性保证的技巧进行求解. 为了获得更优的运行方案, 我们尝试使用诸如分支定界的确定性全局优化方法对热油管道问题进行求解. 热油管道问题本质上是非

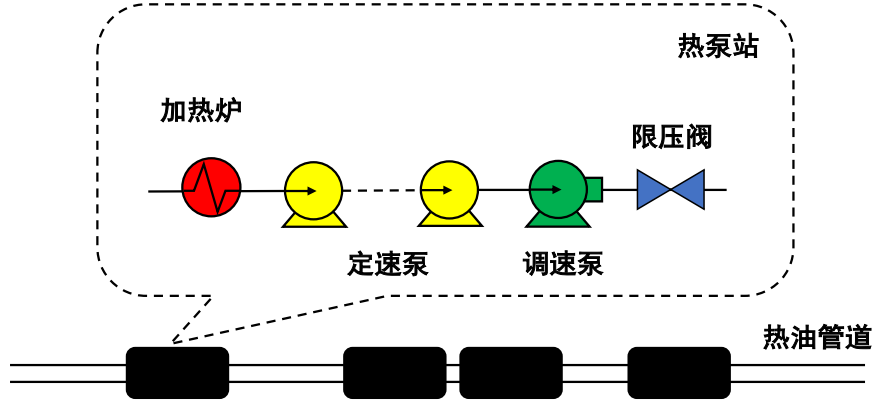


图 1.1 热油管道及热泵站示意图.

凸的 MINLP 问题. 这是由于在水力摩擦计算时我们通常使用 Darcy-Weisbach 公式 [23], 即

$$\text{HF}(T, Q, D, L) = \beta(T) \frac{Q^{2-m(T)}}{D^{5-m(T)}} [\nu(T)]^{m(T)} L, \quad (1.1)$$

其中 T 是原油温度, Q 是原油体积流量, D 是管道内径, L 是管道长度, $\nu(\cdot)$ 表示原油运动粘度, $\beta(\cdot)$ 和 $m(\cdot)$ 是分段常数函数. 公式 (1.1) 给热油管道问题引入了非凸性, 使得我们即使将所有整数变量松弛成连续变量后得到的也仍然是非凸可行域. 尽管目前已有一些商业或开源求解器, 如 BARON [24], ANTIGONE [25], LINDOGlobal [26], Couenne [27], SCIP [28] 等, 可以得到非凸 MINLP 的全局最优解, 但这些软件的求解效率未必能达到热油管道问题实际应用的要求. 因此为热油管道问题设计专门的全局优化方法是十分必要的.

1.1.2 动力下降制导问题

近年来, “可重复使用火箭”是航空航天领域十分热门的话题. 传统火箭所采用的一次性助推器在完成助推任务后当即报废, 无法回收利用. 通过对助推器下落过程进行制导控制, 可重复利用火箭技术可实现对火箭助推器的回收再利用. 据估计, 一套成熟的火箭回收方案可使火箭发射节约高达 50% 的成本. 可见这一技术对国家航天实力的增强以及太空旅行的普及都具有重要的意义. 2015 年 11 月 24 日, 美国 Blue Origin 公司成功进行了火箭回收试验, 成为全世界首家实现这一技术的航天公司. 此后, 美国 SpaceX 公司推出其研制的 Falcon 9 型可重复使用火箭, 并仅在 2017 年就成功完成了 13 次陆上和海上回收任务.

作为火箭回收技术的难点之一, 着陆段的精确制导控制是通过求解动力下降

制导 (powered descent guidance) 问题来实现的. 其可表示为带有目标函数 (如极小化燃料消耗) 和一系列动力学约束, 状态和控制约束 (如推力大小、方向、变化速率) 的最优控制问题. 通过时间离散化, 这一问题可转化为非线性规划 (nonlinear programming, NLP) 问题进行求解. 由于包含了非凸的动力学约束和控制约束, 求解此 NLP 问题的时间复杂度通常是 \mathcal{NP} 难的. 尽管已有许多针对一般 NLP 问题求解方法的研究, 但因为问题是非凸的, 这些方法往往难以满足可靠性和高效性的工程需求. 特别的, 着陆段制导采用的闭环制导控制方式通常需要在线且实时地求解轨迹优化问题. 极其有限的箭上计算资源和极高的实时性要求给问题的求解带来了极大挑战.

不同于非凸 NLP 问题, 凸规划问题因其多项式可解 [29–32] 的特点而备受航天应用者的青睐. 在过去近 20 年间, 许多学者提出将轨迹优化问题通过序列凸化 (successive convexification) 和无损凸化 (lossless convexification) 转化为一系列凸规划问题进行求解. 2007 年, Acikmese 和 Ploen [33] 将火星着陆的三自由度 (three-degree-of-freedom, 3-DOF) 燃料最优动力下降制导问题转化为凸优化问题进行求解, 并首次提出应用于推力大小约束的无损凸化技巧. Blackmore 等人 [34] 在此工作的基础上进一步考虑在没有初始可行轨迹的前提下利用凸优化求解极小化着陆误差的动力下降问题. 此后, 凸优化方法被广泛应用于各类航空航天领域问题, 如动力下降制导 [35–40], 极大横程轨迹优化 [41], 再入轨迹优化 [42, 43], 交会对接轨迹优化 [44], 无人飞行器路径规划 [45, 46], 导弹飞行路径规划 [47] 等. 根据这类最优控制问题的状态和控制约束特点, 以上研究中的凸规划问题可具体表示为二阶锥规划 (second order cone programming, SOCP) 问题. 在给定解的精度下, 基于内点法 (interior point method, IPM) 的软件如 MOSEK, SeDuMi, SDPT3, ECOS 等可以确定 SOCP 问题求解的迭代次数上限. 因此, SOCP 求解算法十分适用于着陆段制导所需的在线实时计算. 虽然序列凸化方法在许多场景十分高效, 但对于一些特殊的动力下降制导问题模型的求解可能会出现迭代效率低或数值不稳定的问题. 此外, 对序列凸化方法应用于航天应用问题的收敛性分析大多聚焦于一些特殊的情形 [48]. 因此, 为通用动力下降制导问题模型设计高效且有收敛性保证的算法以满足高可靠性的要求是十分重要且迫切的.

1.1.3 AGV 道口分布问题

在电商行业高速发展的今天, 物流过程优化是这一行业发展中的重要课题. 除了缩短长途运输时间, 提高货物分拣效率也是加速物流的重要一环. 随着快递包裹数量的急剧增加, 传统的人工分拣已经不能满足高效、低成本的需求. 取而代之的是一种新型自动化分拣方案——自动导引运输车 (automated guided vehicle, AGV) 自动分拣. 它依靠上百台 AGV 自主将包裹由工作站运输至对应的道口, 从而实现包裹的分拣. AGV 道口分布问题是决定系统中每个道口对应的包裹目的地的问题, 即在满足任何去向的包裹都有对应的道口的前提下, 综合考虑工作站与道口的位置关系, 以及不同目的地包裹量的差异, 为每个道口指定一个目的地, 使得分拣效率最高. 据初步测算, 一个好的分布方案能带来约 20% 的效率提升.

设施选址问题 (facility location problem, FLP) 是运筹学领域中的一类经典问题. 这一问题是在满足顾客对商品的需求前提下, 为一些设施选定建造地点, 使得总费用尽可能少. 这里的费用通常包含商品的运输费用和设施建造费用. 许多实际问题如运输、供应链设计、工厂选址、物流管理等问题均可建模成 FLP 模型. 由于 FLP 的应用范围十分广泛, 研究 FLP 的模型和求解算法显得尤为重要. 在过去 40 年里, 许多学者对设施选址及相关问题进行了研究. 选址问题依据对设施容量, 商品种类, 问题周期的要求可以分为: (1) 有容量和无容量限制问题; (2) 单商品和多商品问题; (3) 静态和动态问题等. 关于问题分类可参考文献 [49, 50]. 针对这些问题, 目前的求解方法有直接建模成整数规划 (integer programming, IP) 或混合整数规划 (mixed integer programming, MIP)[9], 然后调用求解器进行求解; 也有针对特定问题结构设计的诸如邻域搜索方法 [51–53], 禁忌搜索方法 [54], 近似算法 [55, 56], Lagrangian 松弛方法 [57, 58], 分解方法 [59, 60] 和启发式方法 [61] 等. 在众多设施选址问题中, 多商品设施选址问题 (multi-product facility location problem, MPFLP) 是其中重要的一类. 这一问题拓展了经典的单商品选址问题, 考虑顾客的需求不只限于一种商品, 应用范围更加广泛. Karkazis[62] 和 Klincewicz[63] 针对无容量限制的 MPFLP 提出了基于对偶的方法. 对于有容量限制的 MPFLP, Geoffrion[59] 在 1974 年给出了基于 Benders 分解的求解算法. 文献 [57] 考虑带有多种设施类型的 MPFLP, 并给出了基于 Lagrangian 松弛的方法. Klincewicz[61] 和 Montoya[64] 为 MPFLP 设计了启发式方法进行求解.

事实上, 如果将 AGV 道口问题中的工作站视为顾客, 道口视为生产各种商

品的工厂,则可将其转化为一类无容量限制的单货源多商品设施选址问题 (uncapacitated single-source multi-product facility location problem, USSMPFLP), 即每个设施可以满足顾客任意多的需求; 每个顾客对每种商品的需求仅可从一处设施得到满足; 每个顾客对多类型商品均有需求的设施选址问题. 文献 [56, 58] 考虑了此问题, 并分别给出了近似比为 $2I + 1$ 的近似算法和基于 Lagrangian 松弛的启发式方法. 这里 I 表示商品种类数. 虽然上述算法的效率较高, 但找到的可行解往往较差, 不能满足实际需求. 对于大规模的 USSMPFLP, 直接求解方法又难以避免求解时间过长的问题. 对于 AGV 道口问题这类规模比较大的问题, 设计高效、实用的算法是十分必要的.

1.2 预备知识

1.2.1 整数规划与分支定界算法

整数规划问题是对含有整数变量的最优化问题的统称, 可根据变量类型细分为纯整数规划 (变量全为整数变量)、混合整数规划 (同时含整数变量和连续变量)、0-1 整数规划 (变量全为二元整数变量) 等, 也可根据约束类型分为混合整数线性规划、混合整数非线性规划、非凸混合整数非线性规划等. 整数规划问题通常是 \mathcal{NP} 难的. 分支定界方法是求解整数规划的一种确定性全局优化方法. 通过求解整数规划问题的线性松弛问题, 分支定界方法将一个整数规划问题转化为许多个可行域更小且不相交的整数规划子问题, 其本质上是一种隐枚举法. 一个简单的分支定界算法框架如算法 1 所述. 下面给出本文涉及的一些整数规划相关名词定义.

定义 1.1.

- 分数变量: 整数变量在线性松弛问题的解中取分数值的变量称为分数变量
- 上界、下界和相对间隙: 上界 (upper bound, UB) 可以是 MIP 问题的任一可行解对应的目标函数值. 下界 (lower bound, LB) 可通过 MIP 的松弛问题最优目标值得到. MIP 的相对间隙 (relative gap, gap), 定义为

$$\text{gap} = \frac{UB - LB}{|UB|} \times 100\%$$

算法 1 分支定界算法

```
1: 初始化根结点  $q_0$ ,  $q_0^{LB} := -\infty$ ,  $Q := \{q_0\}$ ,  $GLB := -\infty$ ,  $GUB := \infty$ ;  
2: while  $Q \neq \emptyset$  且  $GLB < GUB$  do  
3:   选取  $q \in Q$ , 更新  $Q := Q \setminus q$ , 更新  $GLB := \min\{q^{LB} \mid q \in Q\}$ ;  
4:   if  $q$  的线性松弛问题可行 then  
5:     记线性松弛问题的解为  $x$ , 最优值为  $c$ ;  
6:     if  $c < GUB$  then  
7:       if  $x$  是原问题可行解 then  
8:         令  $x^* := x$ ,  $GUB := c$ ;  
9:       else  
10:        将  $q$  分成  $k$  个子问题  $q_1, \dots, q_k$ , 更新  $Q := Q \cup \{q_1, \dots, q_k\}$ ;  
11:      end if  
12:    end if  
13:  end if  
14: end while  
15: if  $GUB < \infty$  then  
16:   return 最优解  $x^*$  和最优值  $GUB$ ;  
17: else  
18:   return 问题不可行.  
19: end if
```

1.2.2 二阶锥及其 Jordan 代数

当 $m \geq 2$ 时, m 维二阶锥 (second order cone, 又称 Lorentz cone), 可表示为

$$\mathcal{K}^m = \{(x_1, \bar{x}) \in \mathbb{R} \times \mathbb{R}^{m-1} \mid x_1 \geq \|\bar{x}\|\}$$

当 $m = 1$ 时, 二阶锥退化为非负卦限锥, 即 $\mathcal{K}^1 = \mathbb{R}_+$. 对于任意 $y = (y_1, \bar{y}) \in \mathbb{R}^m$, $z = (z_1, \bar{z}) \in \mathbb{R}^m$, 定义 y 与 z 的 *Jordan* 乘积为

$$y \circ z = \begin{pmatrix} y^\top z \\ y_1 \bar{z} + z_1 \bar{y} \end{pmatrix} \in \mathbb{R}^m.$$

定义单位元 $e = (1, \mathbf{0})$. 因此有 $y \circ e = y$. 定义 y 的箭矩阵 (arrow matrix)

$$\text{Arw}(y) = \begin{pmatrix} y_1 & \bar{y}^\top \\ \bar{y} & y_1 I_{m-1} \end{pmatrix} \in S^m.$$

因此有 $\text{Arw}(y)z = y \circ z$. 定义 y 的谱分解 (spectral decomposition) 为 $y = \lambda_1 u_1 + \lambda_2 u_2$, 其中

$$\lambda_i = y_1 + (-1)^i \|\bar{y}\|, \quad u_i = \begin{cases} \frac{1}{2} \begin{pmatrix} 1, (-1)^i \frac{\bar{y}}{\|\bar{y}\|} \end{pmatrix}, & \text{若 } \bar{y} \neq 0, \\ \frac{1}{2} (1, (-1)^i \bar{v}), & \text{若 } \bar{y} = 0, \end{cases} \quad i = 1, 2.$$

$\bar{v} \in \mathbb{R}^{m-1}$ 是任意单位元. λ_i 是谱值 (spectral value), u_i 是谱向量 (spectral vector). $\{u_1, u_2\}$ 是一组 *Jordan* 基底 (Jordan frame). y 与 z 算子可交换 (operator commute) 当且仅当存在一组 *Jordan* 基底 $\{u_1, u_2\}$ 和标量 $\alpha_i, \beta_i, i = 1, 2$ 使得 $y = \alpha_1 u_1 + \alpha_2 u_2$ 和 $z = \beta_1 u_1 + \beta_2 u_2$. 命题 1.1 给出了本文所需的与箭矩阵和算子可交换相关性质.

命题 1.1.

- $y \in \mathcal{K}^m \Leftrightarrow \text{Arw}(y) \in S_+^m, y \in \text{int}(\mathcal{K}^m) \Leftrightarrow \text{Arw}(y) \in S_{++}^m$;
- $\text{Arw}(y)\text{Arw}(z) = \text{Arw}(z)\text{Arw}(y)$ 当且仅当 y 与 z 算子可交换;
- 若 $y \in \text{bd}(\mathcal{K}^m)$, $z \in \mathbb{R}^m$ 且 $y \circ z = 0$, 则 y 与 z 算子可交换.

1.2.3 非线性二阶锥规划

一般非线性二阶锥规划 (nonlinear second order cone programming, NSOCP) 问题可表示为以下的最优化问题

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s. t.} \quad & h(x) = 0, \\ & g(x) \in \mathcal{K}. \end{aligned} \quad (\text{NSOCP})$$

其中 \mathcal{K} 表示若干非负卦限锥和二阶锥的笛卡尔乘积. 事实上, 将 NSOCP 作为特殊的非线性优化问题进行研究是有必要的. 如果将二阶锥约束建模为一般的非线性约束并利用通用的非线性优化求解算法进行求解可能会遇到问题. 二阶锥约束的非线性约束表示有如下几种: (1) 按定义: $y_1 - \|\bar{y}\| \geq 0$, 约束是凸的但在 $\bar{y} = 0$ 时不可导. 可能会对传统求解器造成困难 [65]. (2) 光滑化: $y_1^2 - \|\bar{y}\|^2 \geq 0, y_1 \geq 0$, 此时约束处处可导但非凸, 且当 $\bar{y} = 0$ 时线性独立约束品性不满足. (3) 变量替换: $\|t\| \leq 1, \bar{y} = y_1 t, y_1 \geq 0, t \in \mathbb{R}^{m-1}$, 此时约束处处可导且满足线性独立约束品性, 但仍然非凸, 而且当 $y = 0$ 时 t 不唯一. 引入了额外变量 t 使得问题规模变大.

当 f, h, g 均为线性函数时, (NSOCP) 退化为线性二阶锥规划 (linear second order cone programming, LSOCP) 问题, 或简称二阶锥规划 (second order cone programming, SOCP) 问题. 此问题是凸规划问题, 有若干内点法 [66] [67] 可以高效地进行求解, 也有成熟的内点法求解软件 (SDPT3[68], SeDuMi[69], ECOS[70]). LSOCP 是线性半定规划的一个特殊情形, 也可由多种经典凸优化问题 (线性规划, 凸二次规划, 凸二次约束二次规划) 转化而来. LSOCP 的应用也十分广泛 [71], 如鲁棒凸规划, 天线角度设计, 投资组合优化等.

相比于 LSOCP, NSOCP 也有很重要的应用 [33] [72], 但相关的工作较少. 序列二次规划 (sequential quadratic programming, SQP) 型方法 [73] [74] [75] 对 (NSOCP) 的约束线性化, Lagrangian 函数进行二次近似, 在每次迭代求解 LSOCP 问题. 此外还有增广 Lagrangian 方法 [76] [77] [78], 原始对偶内点法 [79], 半光滑牛顿法 [80], 平方松弛变量方法 [81], 可行方向法 [82], 无罚方法 [83] 等.

记 $Jh(x) \in \mathbb{R}^{p \times n}$, $Jg(x) \in \mathbb{R}^{m \times n}$ 为 $h(x)$ 和 $g(x)$ 的 Jacobian 矩阵. (NSOCP) 的 Lagrangian 函数为

$$L(x, y, z) = f(x) - y^\top g(x) - z^\top h(x).$$

其中 $y \in \mathbb{R}^m$, $z \in \mathbb{R}^p$ 为乘子. (NSOCP) 的 Karush-Kuhn-Tucker (KKT) 条件为

$$\nabla_x L(x, y, z) = \nabla f(x) - Jg(x)^\top y - Jh(x)^\top z = 0, \quad (1.2)$$

$$g_j(x) \circ y_j = 0, \quad j = 1, \dots, J, \quad (1.3)$$

$$h(x) = 0, g_j(x) \in \mathcal{K}^{m_j}, y_j \in \mathcal{K}^{m_j}, \quad j = 1, \dots, J. \quad (1.4)$$

1.3 本文主要贡献

对于热油管道问题, 我们主要有以下三方面的贡献. 首先, 我们为稳态热油管道问题提出了一个精确具体的 MINLP 模型. 并且在模型中我们考虑了不同种类的加压泵建模以及通用的水力计算建模. 其次, 我们为这一 MINLP 模型设计了高效的分支定界算法 HOPBB. 这一算法利用了原模型的两个松弛问题间的等价性, 在有限步内可以求得问题的全局最优解. 为了实际应用 HOPBB 算法, 我们还设计了预处理方法保证上述等价性的成立, 以及引入外逼近方法和热启动技术对算法进行加速. 数值实验的结果也表明我们的算法可以求得全局最优解, 且比通用求解器更加高效. 最后, 我们利用 HOPBB 算法得到的最优方案分析了节省热油管道运行费用的方式及潜力. 分析指出采用算法得到的最优方案可以比现有方案节省 6.83% 的运行费用. 进一步, 如果在某些站增加输油温度上限, 可节省超过 8% 的费用.

对于动力下降制导问题, 我们研究了含有非凸动力学约束 (带气动阻力项), 非凸推力大小、方向控制约束等的三自由度燃料最优模型. 为了消除非凸性给问题求解带来的困难, 我们通过无损凸化、时间离散化和线性化技术将非凸最优控制问题转化为 SOCP 问题进行求解. 传统的序列凸化策略在求解本文的模型时会出现数值不稳定的现象. 为了应对这一问题并提高算法迭代效率, 我们提出一个新的两阶段序列凸化方法. 数值结果表明所提方法仅需求解少量的凸优化问题, 且两阶段方法拥有比单阶段方法更稳定、高效的数值表现. 针对序列凸化方法可能存在的子问题不可行、算法不收敛的问题, 我们进一步将动力下降制导问题抽象为一般的非线性二阶锥规划问题. 我们将已有的可行方向内点法推广, 使其可以求解一般的 NSOCP 问题, 并分析了算法的可行性和全局收敛性. 基于随机算例和动力下降制导算例的数值实验展示了可行方向法在一些情形下的高效性.

对于 AGV 道口分布问题, 我们将之转化为无容量限制的单货源多商品设施

选址问题, 并分别提出了基于线性规划松弛解的紧问题启发式和结合邻域搜索的改进启发式方法. 分析表明两种方法均可实施于任意可行的 **USSMPFLP**. 数值实验的结果说明了两种启发式方法可以找到比传统的 **MIP** 求解器和已有的 **Lagrangian** 松弛方法更好的可行解. 对部分问题, 紧问题启发式方法更是可以显著提升求解器的求解效率. 这些结果充分展示了基于 **LP** 解的启发式方法对 **USSMPFLP** 的重大意义.

第2章 热油管道问题的确定性全局优化算法

由于热油管道中油流相比天然气管线或管网中的气体流是较为稳定的,且稳态的情形也是较为常见的研究对象 [12, 14, 15, 18], 因此我们考虑稳态下的热油管道问题, 即原油流量是固定不变的. 在加热原油输送过程中, 热散失会导致原油温度下降. 同时原油会因水力摩擦的存在而逐渐降低压力. 为了使站内和管道中油压和油温处于安全阈值内, 管线中的热泵站需要提供合适的压力和温升. 因此热油管道问题的主要决策变量是热泵站内加压泵的启、停状态和加热炉的温度控制. 给定管线起点处输入油压和油温, 热油管道问题的目标是在所有满足安全运输要求的可行决策方案中找到最经济的方案.

为达到这一目标, 我们为热油管道问题设计一个精确的 MINLP 模型. 利用模型的两个松弛问题, 非凸松弛和凸松弛之间的等价性, 我们提出基于分支定界框架的全局优化算法, 并证明了算法得到解的全局最优性. 为保证算法的可行性和提高算法效率, 我们又分别设计了预处理算法以及引入带热启动的外逼近方法. 实际数据的计算结果说明了算法得到的最优方案比现有方案更好. 与通用求解器的对比结果说明了算法的高效性. 本章的安排如下: 2.1 节详细介绍热油管道问题的 MINLP 模型及其难度分析; 2.2 节给出 MINLP 模型的两个松弛问题和它们的等价性分析, 以及分支定界算法和收敛性分析; 预处理算法和外逼近方法在 2.3 节进行介绍; 2.4 节展示数值实验的结果; 最后我们在 2.5 节进行本章总结.

2.1 混合整数非线性规划模型

本节将给出热油管道问题的 MINLP 模型. 所涉及的部分变量我们通过图 2.1 展示. 更多更详细的符号请参见符号列表部分.

我们考虑的热油管道结构为串行结构. 除作为终点的最后一站外, 管线中的所有工作站均装配有加压泵和加热炉. 为了突出热油管道问题的主要特点, 方便描述解决问题的思想, 我们假设每个中间站中每类设施有相同的运行效率, 如同一热泵站中的多个定速泵有相同的压头和效率. 在管线中存在有多处原油注入和分输点, 用以从油田注入管线或从管线分输给附近的提炼厂. 这些点位既可能在站内也可能在两站之间的管道某处. 假定管线中有 N^S 个热泵站. 为了更准确地

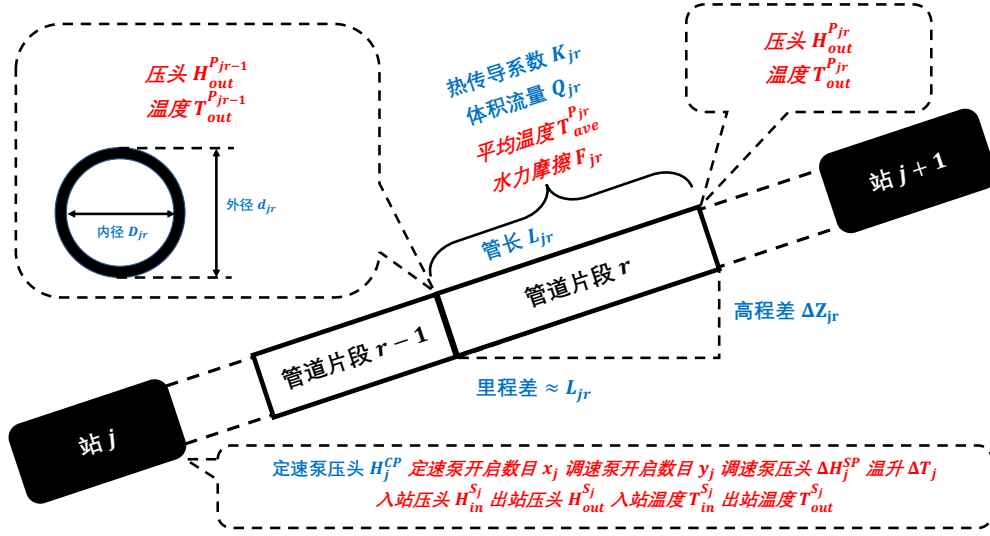


图 2.1 MINLP 模型中的部分常量 (蓝色正体文字) 和变量 (红色斜体文字) 符号含义.

计算管道中的油压和油温状况, 我们将第 j 站和第 $j+1$ 站之间的管道分为 N_j^P 个管道片段, $j = 1, \dots, N^S - 1$. 则每个管道片段两端的压头有如下的关系:

$$H_{out}^{Pjr} = H_{out}^{Pj,r-1} - F_{jr} - \Delta Z_{jr}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P. \quad (2.1)$$

其中 $H_{out}^{Pj,r-1}$ 和 H_{out}^{Pjr} 分别表示第 r 个管道片段入口和出口处的压头. (2.1) 式表明管道中的压力变化来自两方面: 水力摩擦 F_{jr} 和高程差 ΔZ_{jr} . 在本文的模型中我们使用非线性函数 f 来表示水力摩擦系数, 即:

$$F_{jr} = f(T_{ave}^{Pjr}, Q_{jr}, D_{jr}) L_{jr}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P. \quad (2.2)$$

其中 T_{ave}^{Pjr} 是第 r 个管道片段的平均油温. (2.2) 式中的非线性函数 f 可通过套用不同的摩擦或粘度公式得到不同场景下的水力计算模型. 对于平均温度 T_{ave}^{Pjr} 的计算, 我们首先利用如下的轴向温降公式计算管道片段的出口温度 T_{out}^{Pjr} :

$$T_{out}^{Pjr} = T_g^{Pjr} + T_f^{Pjr} + \left[T_{out}^{Pj,r-1} - (T_g^{Pjr} + T_f^{Pjr}) \right] e^{-\alpha_{jr} L_{jr}}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P. \quad (2.3)$$

其中 T_g^{Pjr} 是地面温度, T_f^{Pjr} 表示输运产生的摩擦热导致的环境温度变化量, 参数 α_{jr} 定义为:

$$\alpha_{jr} = \frac{K_{jr} \pi d_{jr}}{\rho Q_{jr} c}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P.$$

其中 K_{jr} 是热传导系数, ρ 为原油密度, c 是原油比热容. 事实上 $T_f^{P_{jr}}$ 是随水力摩擦 F_{jr} 变化的. 我们将其近似为常量以简化模型. 简化的合理性将 2.5 节讨论. 根据管道片段出、入口的油温, 平均油温 $T_{ave}^{P_{jr}}$ 可通过以下经验公式得到:

$$T_{ave}^{P_{jr}} = \frac{1}{3}T_{out}^{P_{j,r-1}} + \frac{2}{3}T_{out}^{P_{jr}}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P. \quad (2.4)$$

在管线中的各个热泵站中, 加压泵和加热炉输出压力和温升以保证原油的安全输运. 当流量固定时, 定速泵可以提供固定的压头, 而调速泵可以提供连续变化的压头. 对于第 j 个热泵站中的加压泵, 我们引入整数变量 x_j 和 y_j 分别表示定速泵和调速泵的开启数量, 引入连续变量 ΔH_j^{SP} 表示调速泵输出的总压头. 热泵站的出口压力上界由入站压力和站内压力变化决定, 即:

$$H_{in}^{S_j} + x_j H_j^{CP} + \Delta H_j^{SP} \geq H_{out}^{S_j}, \quad j = 1, \dots, N^S - 1. \quad (2.5)$$

需要注意的是每个热泵站配置了限压阀. 当出站压力过高时, 限压阀可以在原油出站前将压力降低. 因此 (2.5) 式是不等式. 事实上这样的不等式约束体现了热油管道问题的特殊结构. 调速泵的压头通常有上、下界限制, 即

$$y_j \underline{H}_j^{SP} \leq \Delta H_j^{SP} \leq y_j \overline{H}_j^{SP}, \quad j = 1, \dots, N^S - 1. \quad (2.6)$$

对于加热炉, 我们引入连续变量 ΔT_j 表示原油在第 j 站的温升. 则出站温度可以通过下式得到.

$$T_{in}^{S_j} + \Delta T_j = T_{out}^{S_j}, \quad j = 1, \dots, N^S - 1. \quad (2.7)$$

为了建立站—管之间的联系, 我们定义 $T_{out}^{P_{j0}}$ 和 $H_{out}^{P_{j0}}$ 分别表示第 j 站的出站压头和温度. 同时, 当 $r = N_j^P$ 时, $T_{out}^{P_{jr}}$ 和 $H_{out}^{P_{jr}}$ 即是第 $j+1$ 站的入站压头和温度. 则有:

$$T_{out}^{P_{j0}} = T_{out}^{S_j}, \quad T_{out}^{P_{jN_j^P}} = T_{in}^{S_{j+1}}, \quad j = 1, \dots, N^S - 1, \quad (2.8)$$

$$H_{out}^{P_{j0}} = H_{out}^{S_j}, \quad H_{out}^{P_{jN_j^P}} = H_{in}^{S_{j+1}}, \quad j = 1, \dots, N^S - 1. \quad (2.9)$$

此外, MINLP 模型中还有若干界约束. 在每个热泵站中, 处于开启状态的定速泵和调速泵数量是有界的 ((2.10), (2.11) 式); 温升是有界的 ((2.12) 式); 出、入站的压头和温度是有界的 ((2.13)–(2.16) 式). 各个管道片段中的压头也是有界的 ((2.17)

式). 由于我们可以在管线翻越点处将管道分段, 因此翻越点处的压力限制也包含在 (2.17) 式中.

$$x_j \in \{\underline{x}_j, \underline{x}_j + 1, \dots, \bar{x}_j\}, \quad j = 1, \dots, N^S - 1, \quad (2.10)$$

$$y_j \in \{\underline{y}_j, \underline{y}_j + 1, \dots, \bar{y}_j\}, \quad j = 1, \dots, N^S - 1, \quad (2.11)$$

$$0 \leq \Delta T_j \leq \Delta \bar{T}_j, \quad j = 1, \dots, N^S - 1, \quad (2.12)$$

$$\underline{H}_{in}^{S_j} \leq H_{in}^{S_j} \leq \bar{H}_{in}^{S_j}, \quad j = 2, \dots, N^S, \quad (2.13)$$

$$\underline{H}_{out}^{S_j} \leq H_{out}^{S_j} \leq \bar{H}_{out}^{S_j}, \quad j = 1, \dots, N^S - 1, \quad (2.14)$$

$$\underline{T}_{in}^{S_j} \leq T_{in}^{S_j} \leq \bar{T}_{in}^{S_j}, \quad j = 2, \dots, N^S, \quad (2.15)$$

$$\underline{T}_{out}^{S_j} \leq T_{out}^{S_j} \leq \bar{T}_{out}^{S_j}, \quad j = 1, \dots, N^S - 1, \quad (2.16)$$

$$\underline{H}_{out}^{P_{jr}} \leq H_{out}^{P_{jr}} \leq \bar{H}_{out}^{P_{jr}}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P - 1. \quad (2.17)$$

最终, 热油管道问题的目标是极小化加压泵和加热炉的能耗, 即加压泵的电能耗和加热炉的燃气消耗. 总费用函数可以表示为:

$$C(x, \Delta H^{SP}, \Delta T) = \sum_{j=1}^{N^S-1} \left[C_p \rho Q_{j0} g \left(\frac{x_j H_j^{CP}}{\xi_j^{CP}} + \frac{\Delta H_j^{SP}}{\xi_j^{SP}} \right) + C_f c \rho Q_{j0} \frac{\Delta T_j}{\eta_j V_c} \right].$$

其中 C_p 和 C_f 分别表示电费和燃气费单价. ξ^{CP} 和 ξ^{SP} 分别表示定速泵和调速泵的效率. η 是加热炉的效率. V_c 为燃气热值. 定义 $z := (x, y)$, $\underline{z} := (\underline{x}, \underline{y})$, $\bar{z} := (\bar{x}, \bar{y})$. 令 Ψ 表示模型中所有变量组成的向量, 即:

$$\Psi := (z, \Delta H^{SP}, \Delta T, H_{in}^S, H_{out}^S, T_{in}^S, T_{out}^S, H_{out}^P, T_{out}^P, T_{ave}^P, F).$$

现在我们给出热油管道问题的 MINLP 模型如下:

$$\begin{aligned} & \min_{\Psi} C(x, \Delta H^{SP}, \Delta T) \\ & \text{(HOP}(\underline{z}, \bar{z})) \\ & \text{s. t. } (2.1) - (2.17). \end{aligned}$$

对于初始问题, 我们有:

$$\underline{z} = (\mathbf{0}, \mathbf{0}), \quad \bar{z} = (N^{CP}, N^{SP}).$$

对于变量 x 和 y 没有特别指定的情形, 我们简称上述模型为 (HOP). 类似简称也将应用于后文中的松弛问题. 尽管除约束 (2.2) 外, (HOP) 问题的目标函数和约束

均为线性约束, 但 (HOP) 问题属于非凸 MINLP. 对于一般的非凸 MINLP, 即使是求解连续松弛问题也难以找到全局最优解. 下面的命题表明, 在 $\mathcal{P} \neq \mathcal{NP}$ 的情况下不存在多项式时间的算法可以求解到 (HOP) 问题的全局最优解.

命题 2.1. (HOP) 问题是 \mathcal{NP} 难的.

证明. 我们说明 (HOP) 问题可以归约为属于 \mathcal{NP} 难问题的 cutting stock 问题 [84, 85]. 在以下的参数设置情形下:

- $\underline{z} = (\mathbf{0}, \mathbf{0}), \bar{z} = (N^{CP}, \mathbf{0}), N_j^{CP} \in \mathbb{Z}_+, H_j^{CP} \in \mathbb{Z}_+, j = 1, \dots, N^S - 1;$
- $L_{jr} = 0, j = 1, \dots, N^S - 1, r = 1, \dots, N_j^P;$
- $\Delta Z_{jr} = 0, j = 1, \dots, N^S - 1, r = 1, \dots, N_j^P;$
- $\underline{H}_{in}^{S_{N^S}} - H_{in}^{S_1} \in (0, \sum_{j=1}^{N^S-1} N_j^{CP} H_j^{CP}) \cap \mathbb{Z}_+;$
- $\underline{H}_{in}^{S_j} = H_{in}^{S_1}, j = 2, \dots, N^S - 1;$
- $\bar{H}_{in}^{S_j} = H_{in}^{S_1} + \sum_{j=1}^{N^S-1} N_j^{CP} H_j^{CP}, j = 2, \dots, N^S;$
- $\underline{H}_{out}^{S_j} = \underline{H}_{in}^{S_{j+1}}, \bar{H}_{out}^{S_j} = \bar{H}_{in}^{S_{j+1}}, j = 1, \dots, N^S - 1;$
- $\underline{H}_{out}^{P_{jr}} = \underline{H}_{in}^{S_{j+1}}, \bar{H}_{out}^{P_{jr}} = \bar{H}_{in}^{S_{j+1}}, j = 1, \dots, N^S - 1, r = 1, \dots, N_j^P;$
- $\underline{T}_{in}^{S_j} = \bar{T}_{in}^{S_j} = T_{in}^{S_1}, j = 2, \dots, N^S, \underline{T}_{out}^{S_j} = \bar{T}_{out}^{S_j} = T_{in}^{S_1}, j = 1, \dots, N^S - 1,$

原 (HOP(\underline{z}, \bar{z})) 归约为

$$\begin{aligned} \min_x \quad & \sum_{j=1}^{N^S-1} C_p \rho Q_{j0} g \frac{H_j^{CP}}{\xi_j^{CP}} x_j \\ \text{s. t.} \quad & \sum_{j=1}^{N^S-1} x_j H_j^{CP} \geq \underline{H}_{in}^{S_{N^S}} - H_{in}^{S_1}, \\ & x_j \in \{0, \dots, N_j^{CP}\}, \quad j = 1, \dots, N^S - 1. \end{aligned}$$

令 $V := \underline{H}_{in}^{S_{N^S}} - H_{in}^{S_1}$ 表示钢管需求量, x_j 为第 j 个切割方式的应用次数, $U_j := H_j^{CP}$ 表示每使用一次第 j 种切割方式时生产的钢管个数, $W_j := C_p \rho Q_{j0} g H_j^{CP} / \xi_j^{CP}$ 表示第 j 种切割方式消耗的原料费用. 根据 [85], 此约减后的问题可以视为 cutting stock 问题. 证毕. \square

命题 2.1 表明只考虑 (HOP) 问题中的加压泵组合也是 \mathcal{NP} 难的. 然而实际的需求又促使我们寻找 (HOP) 问题的全局最优. 基于此目标, 我们考虑通过连续松弛和凸化松弛分别处理 (HOP) 问题的两个难点: 整数变量 z 和非凸约束 (2.2).

通过找到可求解到全局最优的子问题并结合分支定界框架来设计 (HOP) 的全局优化算法. 我们将在接下来的 2.2 节中详细说明这一全局优化算法的设计思路及算法步骤.

2.2 全局优化算法

为应对整数变量和非凸约束给 (HOP) 引入的难度, 我们分别考虑非凸松弛和凸松弛问题. 我们可以证明在一定条件下, 这两个松弛问题是等价的. 进而我们可以利用这一等价性设计分支定界算法求得问题的全局最优解.

2.2.1 非凸松弛问题

通常来说, 连续问题比含有整数变量的离散问题更容易处理. 因此我们将 (HOP) 问题中的整数变量 x 和 y 松弛成连续变量, 即

$$x_j \in [\underline{x}_j, \bar{x}_j], \quad j = 1, \dots, N^S - 1, \quad (2.18)$$

$$y_j \in [\underline{y}_j, \bar{y}_j], \quad j = 1, \dots, N^S - 1. \quad (2.19)$$

于是得到如下的 (HOP) 非凸松弛问题

$$\begin{aligned} (\text{HOPnr1}(\underline{z}, \bar{z})) \quad & \min_{\Psi} C(x, \Delta H^{SP}, \Delta T) \\ & \text{s. t. } (2.1) - (2.9), (2.12) - (2.19). \end{aligned}$$

(HOPnr1) 是连续非线性规划 (nonlinear programming, NLP) 问题, 因此可以应用多种求解 NLP 问题的算法 [86]. 然而 (HOPnr1) 仍含有非凸约束 (2.2), 这使得算法难以求得全局最优解, 进而无法得到 (HOP) 问题的下界.

幸运的是, (HOPnr1) 的每个可行解可以转化成 (HOP) 问题的可行解, 即可以通过求解 (HOPnr1) 得到原问题的上界. 我们将这一性质总结为如下的命题 2.2.

命题 2.2. $(\text{HOP}(\underline{z}, \bar{z}))$ 问题是可行的当且仅当 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 是可行的. 特别地, 假设 $\check{\Psi}$ 是 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 的可行解, 则存在 $\hat{\Psi}$, 其中

$$\begin{aligned} \hat{x} &= [\check{x}], \quad \hat{y} = [\check{y}], \\ \Delta \hat{H}_j^{SP} &= \begin{cases} \Delta \check{H}_j^{SP}, & \text{若 } \hat{y}_j \underline{H}_j^{SP} \leq \Delta \check{H}_j^{SP} \leq \hat{y}_j \bar{H}_j^{SP}; \\ \hat{y}_j \underline{H}_j^{SP}, & \text{否则,} \end{cases} \quad j = 1, \dots, N^S - 1, \end{aligned}$$

使得 $\hat{\Psi}$ 可行于 $(\text{HOP}(\underline{z}, \bar{z}))$.

证明. (HOPnr1) 是 (HOP) 问题的松弛, 因此仅需证明 (HOPnr1) 可行时 (HOP) 也可行. 注意到两个问题的不同点仅在整数变量 x 和 y 的界约束, 且包含 x 和 y 的其他约束仅有 (2.5) 和 (2.6). 取 $\hat{\Psi}$ 满足

$$\hat{\Psi} = (\hat{z}, \Delta \hat{H}^{SP}, \Delta \hat{T}, \check{H}_{in}^S, \check{H}_{out}^S, \check{T}_{in}^S, \check{T}_{out}^S, \check{H}_{out}^P, \check{T}_{out}^P, \check{T}_{ave}^P, \check{F}).$$

下面证明 $\hat{\Psi}$ 是 (HOP) 的可行解. 对于 $j = 1, \dots, N^S - 1$, 由于

$$\begin{aligned} \hat{x}_j &= [\check{x}_j] \in \{\underline{x}_j, \dots, \bar{x}_j\}, \\ \hat{y}_j &= [\check{y}_j] \in \{\underline{y}_j, \dots, \bar{y}_j\}, \\ \check{H}_{in}^{S_j} + \hat{x}_j H_j^{CP} + \Delta \hat{H}_j^{SP} &\geq \check{H}_{in}^{S_j} + \check{x}_j H_j^{CP} + \Delta \check{H}_j^{SP} \geq \check{H}_{out}^{S_j} = \hat{H}_{out}^{S_j}. \end{aligned}$$

得证. □

命题 2.2 是设计分支定界算法求解 (HOP) 问题的关键之一. 然而为了得到 (HOP) 的下界, 我们还需要处理 (HOPnr1) 的非凸性. 若能够通过凸化将非凸约束 (2.2) 转化为凸约束, 则可得到可求得全局最优解的子问题.

2.2.2 凸松弛问题

在进一步分析之前, 我们首先给出针对非线性函数 f 的一个假设.

假设 2.3. 给定 Q_{jr}, D_{jr} , 函数 f 是关于变量 $T_{ave}^{P_{jr}}$ 的单调非增凸函数.

事实上假设 2.3 是基于实际生产经验的合理假设. 通常来说, 运动粘度 ν 关于油温是凸的且单调递减的. 此外, 实际管线中的油流通常是水力光滑的. 这意味着在水力摩擦公式 (1.1) 中的参数 β 和 m 是处于某个常数片内 ($\beta \equiv 0.0246$, $m \equiv 0.25$). 因此水力摩擦 F 与粘度 ν 有相同的凸性和单调性. 基于假设 2.3, 我们可以将非凸约束 (2.2) 松弛为如下的凸不等式约束:

$$F_{jr} \geq f\left(T_{ave}^{P_{jr}}, Q_{jr}, D_{jr}\right) L_{jr}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P. \quad (2.20)$$

由此得到 (HOP) 的另一松弛问题:

$$\begin{aligned} & \min_{\Psi} C(x, \Delta H^{SP}, \Delta T) \\ & \text{(HOPnr2}(\underline{z}, \bar{z})) \\ & \text{s. t. } (2.1), (2.3) - (2.9), (2.12) - (2.20). \end{aligned}$$

(HOPnr2) 是凸规划问题, 它的局部最优解同时也是全局最优解. 因此我们可以通过求解 (HOPnr2) 得到 (HOP) 问题的下界.

然而由于 (HOPnr2) 松弛了约束 (2.2), 它的解可能使约束 (2.20) 中的不等式严格成立. 根据这样的解得到的热油管道方案的温度曲线是正确的, 但压头曲线不正确. 因为此时的 F_{jr} 不再反映真实的水力摩擦值. 此时, 即使这个 (HOPnr2) 的解满足整数约束 (2.10) 和 (2.11), 这样的解对于原 (HOP) 问题仍然是不可行的. 上述情形可能导致在分支定界算法执行时仅能不断改进下界, 但无法得到可行上界的问题. 有趣的是, 这样的情形本不该发生在 (HOPnr2) 的求解中. 注意到目标函数是极小化加压泵压力和加热炉温升. 在这样的目标函数下, 约束 (2.20) 的左端项 F_{jr} 作为压头损失的一部分应尽可能地小, 而右端项中真实的水力摩擦系数 f 在假设 2.3 下关于油温是单调的, 应尽可能地大. 因此目标函数应促使约束 (2.20) 成为积极约束. 究竟约束 (2.20) 能否在 (HOPnr2) 最优解处成为不积极约束, 以及为什么会出现这样的情形. 我们用如下的一个简单且有趣的例子来解释这两个问题.

例 2.1. 考虑包含三个热泵站 A、B、C 的热油管道问题. 图 2.2 展示了此热油管道的两个运行方案, 分别来自 (HOPnr1) 的最优解和 (HOPnr2) 的一个可行解. (HOPnr2) 的解输油温度更高, 但总费用更低. (HOPnr2) 的红色压头曲线 (原始) 直接来自它的可行解, 满足了压头的上、下限约束. 然而注意到 (HOPnr1) 的压头曲线在热泵站 B 恰好达到下界, 同时在 C 站恰好达到上界. 则真实的 (HOPnr2) 压头曲线 (修正) 在 C 站的入站压头应比 (HOPnr1) 的曲线数值更高.

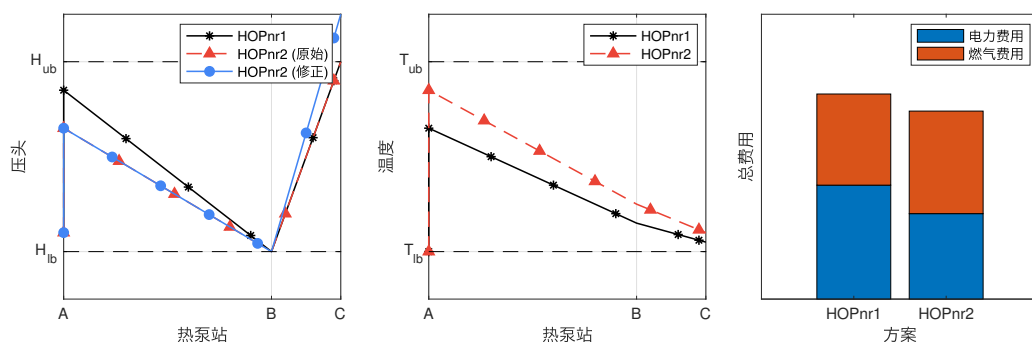


图 2.2 例 2.1 中的压头曲线 (左), 温度曲线 (中) 和方案费用 (右).

在例 2.1 中, 容易验证的是约束 (2.20) 在 (HOPnr2) 的全局最优解处是不积极的. 这是由于在热泵站 B 和 C 之间, 最大压头是油温的隐含约束. 当压力和温度

之间仅有的联系约束 (2.2) 被松弛后, 这样的隐含约束也同时被松弛了. 针对例 2.1 的情形, 是否存在将这样的隐含约束保留在 (HOPnr2) 中的方法. 比如限制热泵站 B 的最大出站温度等于 (HOPnr1) 最优方案的温度, 使得 (HOPnr2) 的最优方案不会违背约束 (2.2). 沿着这一思路, 我们发现 (HOPnr1) 与 (HOPnr2) 在一定条件下是等价的, 即它们有相同的最优目标值.

2.2.3 松弛问题的等价性

在给出重要的等价性定理前, 我们先给出如下两个引理. 其中引理 2.5 揭示了等价性定理成立的条件.

引理 2.4. 假设 $\check{\Psi}$ 和 $\hat{\Psi}$ 是 (HOPnr2) 的可行解. 对任意 $j = 1, \dots, N^S - 1$, 若 $\check{T}_{out}^{S_j} \geq \hat{T}_{out}^{S_j}$, 则有

$$\check{T}_{ave}^{P_{jr}} \geq \hat{T}_{ave}^{P_{jr}}, \quad r = 0, \dots, N_j^P.$$

证明. 首先证明 $\check{T}_{out}^{P_{jr}} \geq \hat{T}_{out}^{P_{jr}}, r = 0, \dots, N_j^P$. 当 $r = 0$ 时, 由约束 (2.9) 可知结论成立. 假设 $\check{T}_{out}^{P_{jr_0}} \geq \hat{T}_{out}^{P_{jr_0}}, r_0 < N_j^P$. 我们说明结论对 $r_0 + 1$ 成立. 根据约束 (2.3), 有

$$\begin{aligned} \check{T}_{out}^{P_{j,r_0+1}} &= T_g^{P_{j,r_0+1}} + T_f^{P_{j,r_0+1}} + \left[\check{T}_{out}^{P_{jr_0}} - \left(T_g^{P_{j,r_0+1}} + T_f^{P_{j,r_0+1}} \right) \right] e^{-\alpha_{j,r_0+1} L_{j,r_0+1}} \\ &\geq T_g^{P_{j,r_0+1}} + T_f^{P_{j,r_0+1}} + \left[\hat{T}_{out}^{P_{jr_0}} - \left(T_g^{P_{j,r_0+1}} + T_f^{P_{j,r_0+1}} \right) \right] e^{-\alpha_{j,r_0+1} L_{j,r_0+1}} \\ &= \hat{T}_{out}^{P_{j,r_0+1}}. \end{aligned}$$

结论仍成立. 结合 (2.4) 可知引理成立. \square

引理 2.5. 假定假设 2.3 成立. 对于 $j = 1, \dots, N^S - 1$, 若存在 (HOPnr1) 的可行解 $\hat{\Psi}$ 使得 $\bar{T}_{out}^{S_j}$ 满足

$$\bar{T}_{out}^{S_j} = \bar{T}_{out}^{S_j},$$

则 (HOPnr2) 可行. 此外, 对于每个 (HOPnr2) 的可行解 $\check{\Psi}$, 存在 (HOPnr1) 的可行解 $\hat{\Psi}$ 使得

$$C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}) = C(\hat{x}, \Delta \hat{H}^{SP}, \Delta \hat{T}).$$

证明. (HOPnr2) 是 (HOPnr1) 的松弛, 因此 (HOPnr2) 显然可行. 下面我们证明存

在 (HOPnr1) 的可行解 $\hat{\Psi}$ 满足以下条件

$$\begin{aligned} \hat{z} &= \check{z}, \\ \Delta \hat{H}^{SP} &= \Delta \check{H}^{SP}, \\ \Delta \hat{T} &= \Delta \check{T}, \\ \hat{H}_{out}^{S_j} &= \begin{cases} \check{H}_{out}^{S_j}, & \text{若对所有 } r = 1, \dots, N_j^P, \\ & \check{F}_{jr} \text{ 和 } \check{T}_{ave}^{P_{jr}} \text{ 满足约束 (2.2);} \\ \check{H}_{out}^{S_j} - M_j, & \text{若存在 } r_0 \in \{1, \dots, N_j^P\}, \\ & \check{F}_{jr_0} \text{ 和 } \check{T}_{ave}^{P_{jr_0}} \text{ 不满足约束 (2.2),} \end{cases} \quad j = 1, \dots, N^S - 1. \end{aligned}$$

其中对于 $j = 1, \dots, N^S - 1$, 若 $\check{H}_{out}^{S_j} - \sum_{t=1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - U_{jr_1} > 0$, 则 M_j 定义为

$$M_j = \check{H}_{out}^{S_j} - \sum_{t=1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - \max \{ \check{H}_{out}^{P_{jr_1}}, \check{H}_{out}^{P_{jr_1}} \}.$$

否则 $M_j = 0$. r_1 和 U_{jr} 定义为

$$\begin{aligned} r_1 &= \arg \max_{r=1, \dots, N_j^P} \left\{ \check{H}_{out}^{S_j} - \sum_{t=1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - U_{jr} \right\}, \\ U_{jr} &= \begin{cases} \overline{H}_{out}^{P_{jr}}, & r = 1, \dots, N_j^P - 1; \\ \overline{H}_{in}^{S_{j+1}}, & r = N_j^P. \end{cases} \end{aligned}$$

$\check{H}_{out}^{P_{jr_1}}$ 是满足 $\check{T}_{out}^{S_j} = \overline{T}_{out}^{S_j}$ 的解 $\check{\Psi}$ 的分量.

注意到 (HOPnr1) 与 (HOPnr2) 的区别仅在非线性约束 2.2. 对任一 j , 若对所有的 $r = 1, \dots, N_j^P$, \check{F}_{jr} 和 $\check{T}_{ave}^{P_{jr}}$ 均满足约束 (2.2), 则有 $\hat{F}_{jr} = \check{F}_{jr}$, $r = 1, \dots, N_j^P$. 因此 $\hat{\Psi}$ 的所有下标等于此 j 的分量的可行性与 $\check{\Psi}$ 的相同. 对于存在 r_0 使得 \check{F}_{jr_0} 与 $\check{T}_{ave}^{P_{jr_0}}$ 不满足约束 (2.2) 的情形, 仅需证明 $\hat{H}_{out}^{S_j}$, $\hat{H}_{in}^{S_{j+1}}$ 以及 $\hat{H}_{out}^{P_{jr}}$ 满足约束 (2.5), (2.13), (2.14) 和 (2.17), $r = 1, \dots, N_j^P - 1$.

若 $M_j = 0$, 则有 $\hat{H}_{out}^{S_j} = \check{H}_{out}^{S_j}$. 此时约束 (2.14) 满足. 根据约束 (2.20), 我们可

以得出对任意的 $r = 1, \dots, N_j^P$, 有

$$\begin{aligned}
 \hat{H}_{out}^{P_{jr}} &= \check{H}_{out}^{S_j} - \sum_{t=1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] \\
 &\geq \check{H}_{out}^{S_j} - \sum_{t=1}^r [\check{F}_{jr} + \Delta Z_{jt}] = \check{H}_{out}^{P_{jr}}, \\
 \hat{H}_{out}^{P_{jr}} &= \check{H}_{out}^{S_j} - \sum_{t=1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] \\
 &\leq \check{H}_{out}^{S_j} - \sum_{t=1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - U_{jr_1} + U_{jr} \leq U_{jr}.
 \end{aligned}$$

因此界约束 (2.13) 和 (2.17) 同样满足.

对于 $\check{H}_{out}^{S_j} - \sum_{t=1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - U_{jr_1} > 0$ 的情形, 由 $\check{\Psi}$ 和 $\check{\Psi}$ 在约束 (2.17) 的可行性有 $M_j > 0$. 于是有 $\hat{H}_{out}^{S_j} \leq \check{H}_{out}^{S_j}$. 再由引理 2.4 以及 f 的单调性, 得到

$$\begin{aligned}
 \hat{H}_{out}^{P_{jr}} &= \check{H}_{out}^{S_j} - M_j - \sum_{t=1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] \\
 &\leq \check{H}_{out}^{S_j} - M_j - \sum_{t=1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] - U_{jr_1} + U_{jr} \\
 &= U_{jr} - U_{jr_1} + \max \{ \check{H}_{out}^{P_{jr_1}}, \check{H}_{out}^{P_{jr_1}} \} \leq U_{jr}, \quad r = 1, \dots, N_j^P, \\
 \hat{H}_{out}^{P_{jr}} &= \sum_{t=r+1}^{r_1} \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] + \max \{ \check{H}_{out}^{P_{jr_1}}, \check{H}_{out}^{P_{jr_1}} \} \\
 &\geq \check{H}_{out}^{S_j} - \sum_{t=1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] = \check{H}_{out}^{P_{jr}}, \quad r = 0, \dots, r_1 - 1, \\
 \hat{H}_{out}^{P_{jr}} &= - \sum_{t=r_1+1}^r \left[f \left(\check{T}_{ave}^{P_{jr}}, Q_{jt}, D_{jt} \right) L_{jt} + \Delta Z_{jt} \right] + \max \{ \check{H}_{out}^{P_{jr_1}}, \check{H}_{out}^{P_{jr_1}} \} \\
 &\geq \check{H}_{out}^{S_j} - \sum_{t=1}^r [\check{F}_{jr} + \Delta Z_{jt}] = \check{H}_{out}^{P_{jr}}, \quad r = r_1 + 1, \dots, N_j^P.
 \end{aligned}$$

由 $\check{\Psi}$ 和 $\check{\Psi}$, 约束 (2.13), (2.14) 和 (2.17) 成立. 最后, 对于所有情形均有

$$\check{H}_{in}^{S_{j+1}} \geq \hat{H}_{in}^{S_{j+1}}, \quad \hat{H}_{out}^{S_j} \leq \check{H}_{out}^{S_j}.$$

因为

$$\hat{H}_{in}^{S_j} + \hat{x}_j H_j^{CP} + \Delta \hat{H}_j^{SP} \geq \check{H}_{in}^{S_j} + \check{x}_j H_j^{CP} + \Delta \check{H}_j^{SP} \geq \check{H}_{out}^{S_j} \geq \hat{H}_{out}^{S_j}.$$

约束 (2.5) 对所有的 $j = 1, \dots, N^S - 1$ 均满足. 因此 $\hat{\Psi}$ 是 (HOPnr1) 的可行解且

$$C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}) = C(\hat{x}, \Delta \hat{H}^{SP}, \Delta \hat{T}).$$

证毕. □

定理 2.6. 在引理 2.5 的条件下, (HOPnr1) 与 (HOPnr2) 有相同的最优目标值.

证明. 假设 $\check{\Psi}^*$ 是 (HOPnr2) 的全局最优解. 记 C^* 是 (HOPnr1) 的最优目标值. 一方面, 因为 (HOPnr2) 是 (HOPnr1) 的松弛, 我们有

$$C(\check{x}^*, \Delta \check{H}^{SP*}, \Delta \check{T}^*) \leq C^*.$$

另一方面, 根据引理 2.5, 存在 (HOPnr1) 的可行解 $\hat{\Psi}^*$ 使得

$$C(\check{x}^*, \Delta \check{H}^{SP*}, \Delta \check{T}^*) = C(\hat{x}^*, \Delta \hat{H}^{SP*}, \Delta \hat{T}^*).$$

因此有

$$C^* \leq C(\hat{x}^*, \Delta \hat{H}^{SP*}, \Delta \hat{T}^*) = C(\check{x}^*, \Delta \check{H}^{SP*}, \Delta \check{T}^*) \leq C^*.$$

证毕. □

引理 2.5 与定理 2.6 是设计分支定界算法的另一关键要素. 结合 2.2.1 节的命题 2.2, 我们将 (HOP) 问题和它的两个松弛问题的关系整理成如图 2.3 所示. 当引理 2.5 的条件成立时, 根据定理 2.6, 我们可以通过求解 (HOPnr2) 得到 (HOPnr1) 的全局最优解. 同时, 通过 (HOPnr1) 的解又可以得到 (HOP) 的上界. 因此求解 (HOPnr2) 至最优使得我们同时得到了 (HOP) 问题的下界和上界. 最终利用这一特点我们可以设计分支定界算法求解 (HOP) 问题.

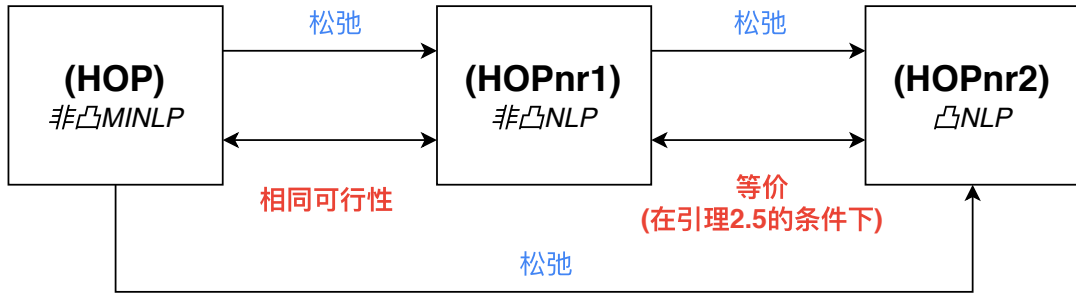


图 2.3 (HOP)、(HOPnr1) 及 (HOPnr2) 的关系示意图.

在给出分支定界算法前, 我们仍需仔细考虑违背引理 2.5 条件的情形. 这样的情形可能是: (1) (HOPnr1) 不可行; (2) 出站温度上界 $\bar{T}_{out}^{S_j}, j = 1, \dots, N^S - 1$ 是松的, 即不存在可行解达到这一上界. 情形 (1) 意味着 (HOP) 不可行, 因此无需进一步考虑. 对于情形 (2), 温度上界可在不改变 (HOPnr1) 可行域的前提下进行缩紧. 下面的例 2.2 对缩紧的基本想法进行了说明.

例 2.2. 考虑包含两个热泵站 A 和 B 的热油管道问题. H_A 和 T_A 分别表示站 A 的出站压头和温度. 假设站 A 的最大温升和站 B 的最大入站温度的数值都足够大. 通过下面的操作可以得到 T_A 的精确上界 (如图 2.4 所示):

1. 初始化 $H_A := H_{lb}, T_A := 60$ (60 是初始上界), 计算压头曲线;
2. 找到对压头下界 H_{lb} 的最大违背点, 增加出站压头 H_A 直到这个最大违背点满足约束;
3. 找到对压头上界 H_{ub} 的最大违背点, 降低出站温度 T_A 使得最大违背点恰好满足约束, 重置 $H_A := H_{lb}$ 并更新压头曲线;
4. 重复步骤 2 和 3 直至压头曲线上的所有点满足上、下界约束.

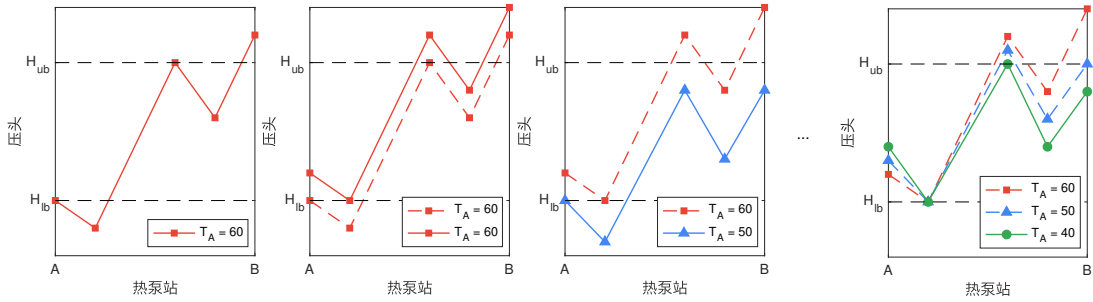


图 2.4 例 2.2 中的压头曲线. 前三个子图中的实线和虚线分别代表第 1、2、3 步和前一步的压头曲线. 第四个子图将最终曲线 (绿色圆形标记实线) 与之前迭代中步骤 2 的两条不可行曲线 (虚线) 比较展示了不可行性是如何消除的.

例 2.2 仅展示了两个站的情形. 在接下来的章节中我们将给出一个可应对多个站情形的预处理算法. 这一算法首先检查 (HOPnr1) 的可行性. 在可行的情况下缩紧 $T_{out}^{S_j}$ 来满足引理 2.5 的条件. 经过预处理, 分支定界树中的每个子问题要么因为不可行被剪枝, 要么松弛得到 (HOPnr2) 进行求解. 即预处理算法保证了分支定界树中的每个可行子问题的两个松弛问题 (HOPnr1) 和 (HOPnr2) 都是等价的.

算法 2 HOPBB: 求解 (HOP) 的分支定界算法

```

1: 初始化  $\underline{z} := 0, \bar{z} := (N^{CP}, N^{SP}), LB := -\infty, GLB := -\infty, GUB := +\infty, Q := \{(\underline{z}, \bar{z}, LB)\}$ ;
2: while  $Q \neq \emptyset$  且  $GLB < GUB$  do
3:     选取  $q \in Q, Q := Q \setminus q$ , 更新  $GLB, \underline{z}, \bar{z}, LB$ ;
4:     对  $(\text{HOPnr1}(\underline{z}, \bar{z}))$  进行预处理检查可行性并达成引理 2.5 的条件;
5:     if  $LB < GUB$  且  $(\text{HOPnr1}(\underline{z}, \bar{z}))$  可行 then
6:         求解  $(\text{HOPnr2}(\underline{z}, \bar{z}))$  并得到最优解  $\check{\Psi}$ , 更新  $LB := C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T})$ , 利用  $\check{\Psi}$  和引理
            2.5 得到  $(\text{HOPnr1}(\underline{z}, \bar{z}))$  的最优解  $\hat{\Psi}$ ;
7:         利用  $\hat{\Psi}$  和命题 2.2 得到 (HOP) 的可行解  $\hat{\Psi}$ ;
8:         if  $C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}) < GUB$  then
9:             更新  $GUB := C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}), \Psi^* := \hat{\Psi}$ ;
10:        end if
11:        if 存在  $j_0 \in \{1, \dots, N^S - 1\}$  使得  $\check{x}_{j_0} \notin \mathbb{Z} (\check{y}_{j_0} \notin \mathbb{Z})$  then
12:            令
                
$$z^L := \{z \in \mathbb{Z}_+^{2N^S-2} \mid \text{对任意 } j \neq j_0, x_j = \bar{x}_j, y_j = \bar{y}_j, x_{j_0} = \lfloor \check{x}_{j_0} \rfloor (y_{j_0} = \lfloor \check{y}_{j_0} \rfloor)\},$$


$$z^R := \{z \in \mathbb{Z}_+^{2N^S-2} \mid \text{对任意 } j \neq j_0, x_j = \underline{x}_j, y_j = \underline{y}_j, x_{j_0} = \lceil \check{x}_{j_0} \rceil (y_{j_0} = \lceil \check{y}_{j_0} \rceil)\},$$

            更新  $Q := Q \cup \{(\underline{z}, z^L, LB), (z^R, \bar{z}, LB)\}$ ;
13:        end if
14:    end if
15:    按如下方式更新  $GLB$ ;
        
$$GLB := \min_{q \in Q} \{LB \mid LB \text{ 是 } q \text{ 的下界}\}.$$

16: end while
17: if  $GUB = +\infty$  then
18:     return (HOP) 问题不可行;
19: else
20:     return 当前候选解  $\Psi^*$  及  $GUB$ .
21: end if
    
```

2.2.4 HOPBB 算法

结合第 1 章的分支定界算法框架 (算法 1), 我们给出求解 (HOP) 问题的分支定界算法 HOPBB (算法 2). 分支定界树中的所有子节点是通过对整数变量 x_j 和 $y_j, j = 1, \dots, N^S - 1$ 进行分支得到的. 对每个可行的 (HOP) 子问题, 根据 2.2.3 节的讨论, 我们通过求解 (HOPnr2) 得到子问题的上、下界. 若得到的解 x 和 y 均为仅包含整数值的向量, 则子问题的全局最优解已经找到, 可以进行剪枝. 否则我们需要对任一分数变量进行分支, 得到两个新的子问题. 接着选取分支定界树中任一未求解叶子节点继续求解. 可以证明, 算法 2 可以在有限步内得到 (HOP) 的全局最优解.

定理 2.7. 对于任意的 (HOP) 问题实例, 算法 2 是有限终止的. 特别地, 当 (HOP) 问题实例可行时, 算法 2 将在终止时返回这一实例的全局最优解.

证明. 分支定界树中的每个子问题的可行性由预处理方法保证. 通过对分数变量进行分支, 我们得到两个包含不相交可行域的子问题. 因此 HOPBB 算法产生的分支定界树中不存在两个子问题有相同的整数变量界. 又因为以下集合仅包含有限个元素

$$\left\{ (\underline{z}, \bar{z}) \mid 0 \leq \underline{x}_j \leq \bar{x}_j \leq N_j^{CP}, 0 \leq \underline{y}_j \leq \bar{y}_j \leq N_j^{SP}, j = 1, \dots, N^S - 1 \right\}.$$

因此分支定界树有有限个节点, 子问题的求解个数也是有限个. HOPBB 的有限终止性得证. 下面我们利用反证法证明算法 2 的最优性. 显然如果 (HOP) 问题可行, 算法 2 总可以返回它的候选解 $\check{\Psi}$ 和全局上界 GUB . 假设 Ψ^* 是 (HOP) 问题的全局最优解, 且 $C(x^*, \Delta H^{SP*}, \Delta T^*) < C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T})$ 成立. 由于分支仅会对可行域进行分割, 不会使其缩减, 则存在节点 $(\underline{z}, \bar{z}, LB)$ 满足 $\underline{z} \leq z^* \leq \bar{z}$ 被算法 2 剪枝. 注意到包含 Ψ^* 的节点不可能因为不可行或定界被剪枝. 因此有 (HOPnr1(\underline{z}, \bar{z})) 的最优解 $\check{\Psi}$ 满足所有整数约束. 令 $\hat{\Psi}$ 是由 $\check{\Psi}$ 和命题 2.2 得到的 (HOP) 可行解 (即算法 2 的步骤 7). 因为 $\hat{\Psi}$ 满足

$$\hat{z} = [\check{z}] = \check{z}, \quad \Delta \hat{H}^{SP} = \Delta \check{H}^{SP}, \quad \Delta \hat{T} = \Delta \check{T}.$$

由 $\check{\Psi}$ 的最优性又有

$$\begin{aligned} C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}) &= GUB \\ &\leq C(\hat{x}, \Delta \hat{H}^{SP}, \Delta \hat{T}) \\ &= C(\check{x}, \Delta \check{H}^{SP}, \Delta \check{T}) \\ &\leq C(x^*, \Delta H^{SP*}, \Delta T^*). \end{aligned}$$

这与前述假设矛盾. 证毕. \square

2.3 预处理算法和外逼近方法

根据 2.2.3 节的讨论, 我们需要设计预处理算法检查 (HOPnr1) 的可行性和缩紧出站温度上界以达到引理 2.5 的条件. 此外, 对于子问题松弛 (HOPnr2) 的求解, 我们可以结合外逼近方法和热启动策略对算法进行加速.

2.3.1 预处理算法

预处理算法的详细步骤如算法 3 所述. 其中, $\phi_{jr}^{r_0}$ 和 $\psi_{jr}^{r_0}$ 定义为

$$\begin{aligned} \phi_{jr}^{r_0} &= \frac{1}{3} [1 + 2 \exp(-\alpha_{j,r_0+r} L_{j,r_0+r})] \exp\left(-\sum_{t=1}^{r-1} \alpha_{j,r_0+t} L_{j,r_0+t}\right), \\ \psi_{jr}^{r_0} &= \frac{1}{3} [1 + 2 \exp(-\alpha_{j,r_0+r} L_{j,r_0+r})] \sum_{t=1}^{r-1} (T_g^{P_{j,r_0+t}} + T_f^{P_{j,r_0+t}}) [1 - \exp(-\alpha_{j,r_0+t} L_{j,r_0+t})] + \\ &\quad \frac{2}{3} (T_g^{P_{j,r_0+r}} + T_f^{P_{j,r_0+r}}) [1 - \exp(-\alpha_{j,r_0+r} L_{j,r_0+r})]. \end{aligned}$$

规定 $\underline{H}_{in}^{S_1} = \overline{H}_{in}^{S_1} = H_{in}^{S_1}$, $\underline{T}_{in}^{S_1} = \overline{T}_{in}^{S_1} = T_{in}^{S_1}$. 为了推广例 2.2 中的操作以应对更多热泵站的情形, 算法 3 的设计思路是将一个多站系统分解成多个双站系统. 由于约束 (2.5) 是不等式, 只要左端项充分大, 其右端项总可以取到 $\underline{H}_{out}^{S_j}$. 对于温度, 如果假设温升 ΔT_j 是上无界的, 那么在 $T_{in}^{S_j}$ 充分小时出站温度总可以取到 $\overline{T}_{out}^{S_j}$. 利用这两个特点, 我们可以从第 $j = N^S - 1$ 站开始向前遍历所有的站, 将第 j 站和第 $j + 1$ 站视为如例 2.2 中的双站系统. 在我们找到了达到实际最大出站温度的方案后, 我们将隐含在最新的出站温度上界中的可行性要求通过缩紧 $\underline{H}_{in}^{S_j}$ 和 $\overline{T}_{in}^{S_j}$ 以及线性域传播的方式传递到前面的站. 在继续讨论前, 我们给出如下算法 3 所需的假设.

假设 2.8.

- 对所有的 $j = 1, \dots, N^S - 1$, 有 $\Delta \overline{T}_j = +\infty$;

算法 3 (HOPnr1) 的预处理算法

```

1: for  $j = N^S - 1, N^S - 2, \dots, 1$  do
2:   对下标  $j$  的约束 (2.3), (2.8), (2.9), (2.16) 和下标  $j + 1$  的约束 (2.13), (2.15) 进行线性域
      传播;
3:   令  $T_{out}^{S_j} := \overline{T}_{out}^{S_j}$ , 利用约束 (2.2), (2.3), (2.4), (2.8) 计算  $T_{out}^{P_{jr}}, T_{ave}^{P_{jr}}, F_{jr}, r = 1, \dots, N_j^P, T_{in}^{S_{j+1}}$ ;
4:   令  $H_{out}^{S_j} := \underline{H}_{out}^{S_j}$ , 利用约束 (2.1), (2.9) 计算  $H_{out}^{P_{jr}}, r = 0, \dots, N_j^P$ ;
5:   令  $r_0 := \arg \max_{r=0, \dots, N_j^P} \{ \underline{H}_{out}^{P_{jr}} - H_{out}^{P_{jr}} \}$ ,  $\delta H := \underline{H}_{out}^{P_{r_0}} - H_{out}^{P_{r_0}}$ , 更新  $H_{out}^{P_{jr}} := H_{out}^{P_{jr}} + \delta H$ ;
6:   if 存在  $r_1 \in \{0, \dots, N_j^P\}$  使得  $H_{out}^{P_{r_1}} > \overline{H}_{out}^{P_{r_1}}$  then
7:     if  $r_1 < r_0$  then
8:       return (HOPnr1( $\underline{z}, \overline{z}$ )) 不可行;
9:     end if
10:    求解下面的非线性方程
        
$$\overline{H}_{out}^{P_{r_1}} = \underline{H}_{out}^{P_{r_0}} - \sum_{t=1}^{r_1-r_0} [f(\phi_{jt}^{r_0} u + \psi_{jt}^{r_0}, Q_{j,r_0+t}, D_{j,r_0+t}) L_{j,r_0+t} + \Delta Z_{j,r_0+t}]. \quad (\text{NLEQN})$$

        其中  $u \in \mathbb{R}$  是变量;
11:    if 非线性方程 (NLEQN) 无解 then
12:      return (HOPnr1( $\underline{z}, \overline{z}$ )) 不可行;
13:    end if
14:    令
        
$$\overline{T}_{out}^{S_j} := \sum_{t=1}^{r_0} (T_g^{P_{jt}} + T_f^{P_{jt}}) [1 - \exp(\alpha_{jt} L_{jt})] \prod_{k=1}^{t-1} \exp(\alpha_{jk} L_{jk}) + \prod_{t=1}^{r_0} \exp(\alpha_{jt} L_{jt}) u.$$

        其中  $u$  是 (NLEQN) 的解, 转步骤 3;
15:  end if
16:  按以下方式缩紧变量界  $\underline{H}_{out}^{S_j}, \underline{H}_{in}^{S_j}$  和  $\overline{T}_{in}^{S_j}$ 
        
$$\underline{H}_{out}^{S_j} := H_{out}^{S_j}, \underline{H}_{in}^{S_j} := \max\{\underline{H}_{in}^{S_j}, \underline{H}_{out}^{S_j} - \overline{x}_j H_j^{CP} - \overline{y}_j \overline{H}_j^{SP}\}, \overline{T}_{in}^{S_j} := \min\{\overline{T}_{in}^{S_j}, \overline{T}_{out}^{S_j}\};$$

17:  if 界约束 (2.13)–(2.17) 表示的集合为空 then
18:    return (HOPnr1( $\underline{z}, \overline{z}$ )) 不可行;
19:  end if
20: end for
21: return (HOPnr1( $\underline{z}, \overline{z}$ )) 可行.
    
```

- 给定 Q_{jr}, D_{jr} , 函数 f 是关于变量 $T_{ave}^{P_{jr}}$ 的严格单调递减凸函数.

假设 2.8 的第一条保证了分解多站系统的可行性. 尽管每个热泵站的热负荷在实际中并不是无限的, 但在此假设下研究管线的节能潜力也是十分重要的. 第二条中的严格单调性保证了算法 3 中非线性方程 (NLEQN) 解的唯一性. 通常来说此对单调性的附加假设并不违背 2.2.2 节中讨论的假设合理性. 可以证明, 算法 3 或是说明 (HOPnr1) 的不可行性, 或是在问题可行时缩紧变量界以达到引理 2.5 的条件. 在给出这一结论前, 我们先引入如下三个基本的引理.

引理 2.9. 假定假设 2.3 成立且非线性方程 (NLEQN) 有可行解 \tilde{u} . 则 \tilde{u} 是变量 $T_{out}^{P_{jr0}}$ 的一个上界. 此外, 变量 $T_{out}^{S_j}$ 有如下上界:

$$\sum_{t=1}^{r_0} \left(T_g^{P_{jt}} + T_f^{P_{jt}} \right) [1 - \exp(\alpha_{jt} L_{jt})] \prod_{k=1}^{t-1} \exp(\alpha_{jk} L_{jk}) + \prod_{t=1}^{r_0} \exp(\alpha_{jt} L_{jt}) \tilde{u}.$$

证明. 令

$$h(u) = - \sum_{t=1}^{r_1-r_0} \left[f(\phi_{jt}^{r_0} u + \psi_{jt}^{r_0}, Q_{j,r_0+t}, D_{j,r_0+t}) L_{j,r_0+t} + \Delta Z_{j,r_0+t} \right].$$

注意到 f 单调非增, 且 $L_{j,r_0+r} \geq 0, r = 1, \dots, r_1 - r_0$. 容易验证 h 是单调非减的. 根据约束 (2.1), (2.2), (2.3), (2.4) 以及 \tilde{u} 的可行性, 有

$$h(T_{out}^{P_{jr0}}) = H_{out}^{P_{jr1}} - H_{out}^{P_{jr0}} \leq \overline{H}_{out}^{P_{jr1}} - \underline{H}_{out}^{P_{jr0}} = h(\tilde{u}).$$

因此有 $T_{out}^{P_{jr0}} \leq \tilde{u}$. 由约束 (2.3), 可以证明 $T_{out}^{S_j}$ 存在陈述中的上界. 证毕. \square

引理 2.10. 假定假设 2.8 成立且 (HOPnr1(\underline{z}, \bar{z})) 可行. 对所有的 $0 \leq r_0 < r_1 \leq N_j^P$, 若存在 \hat{u} 使得

$$\overline{H}_{out}^{P_{jr1}} < \underline{H}_{out}^{P_{jr0}} - \sum_{t=1}^{r_1-r_0} \left[f(\phi_{jt}^{r_0} \hat{u} + \psi_{jt}^{r_0}, Q_{j,r_0+t}, D_{j,r_0+t}) L_{j,r_0+t} + \Delta Z_{j,r_0+t} \right].$$

则非线性方程 (NLEQN) 有唯一解. 进一步, 若 \tilde{u} 是 (NLEQN) 的可行解, 则以下性质满足:

$$\overline{T}_{out}^{S_j} > \sum_{t=1}^{r_0} \left(T_g^{P_{jt}} + T_f^{P_{jt}} \right) [1 - \exp(\alpha_{jt} L_{jt})] \prod_{k=1}^{t-1} \exp(\alpha_{jk} L_{jk}) + \prod_{t=1}^{r_0} \exp(\alpha_{jt} L_{jt}) \tilde{u}.$$

证明. 在假设 2.8 下, 引理 2.9 的证明中定义的 h 函数是严格单调递增的. 令 $\tilde{\Psi}$ 是 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 的可行解. 则有

$$h(\tilde{T}_{out}^{P_{jr0}}) = \tilde{H}_{out}^{P_{jr1}} - \tilde{H}_{out}^{P_{jr0}} \leq \overline{H}_{out}^{P_{jr1}} - \underline{H}_{out}^{P_{jr0}} < h(\hat{u}) \Rightarrow \tilde{T}_{out}^{P_{jr0}} < \hat{u}.$$

此外, 由于 f 是凸函数, 因此 h 在 $[\tilde{T}_{out}^{P_{jr0}}, \hat{u}]$ 上连续. 存在唯一的 \tilde{u} 使得

$$\tilde{T}_{out}^{P_{jr0}} < \tilde{u} < \hat{u}, \quad h(\tilde{u}) = \overline{H}_{out}^{P_{jr1}} - \underline{H}_{out}^{P_{jr0}}.$$

进而有 \tilde{u} 是 (NLEQN) 的唯一解. 根据 $\tilde{u} < \hat{u}$, 引理陈述的性质是满足的. 得证. \square

引理 2.11. 假定假设 2.3 成立. 对所有的 $0 \leq r_1 < r_0 \leq N_j^P$, 若存在 \hat{u} 使得

$$\begin{aligned} \underline{H}_{out}^{P_{jr0}} &> \overline{H}_{out}^{P_{jr1}} - \sum_{t=1}^{r_0-r_1} \left[f(\phi_{jt}^{r_1} \hat{u} + \psi_{jt}^{r_1}, Q_{j,r_1+t}, D_{j,r_1+t}) L_{j,r_1+t} + \Delta Z_{j,r_1+t} \right], \\ \overline{T}_{out}^{S_j} &= \sum_{t=1}^{r_1} \left(T_g^{P_{jt}} + T_f^{P_{jt}} \right) [1 - \exp(\alpha_{jt} L_{jt})] \prod_{k=1}^{t-1} \exp(\alpha_{jk} L_{jk}) + \prod_{t=1}^{r_1} \exp(\alpha_{jt} L_{jt}) \hat{u}. \end{aligned}$$

则 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 不可行.

证明. 我们利用反证法证明结论. 假设 $\tilde{\Psi}$ 是 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 的可行解. 令

$$h'(u) = - \sum_{t=1}^{r_0-r_1} \left[f(\phi_{jt}^{r_1} u + \psi_{jt}^{r_1}, Q_{j,r_1+t}, D_{j,r_1+t}) L_{j,r_1+t} + \Delta Z_{j,r_1+t} \right].$$

类似引理 2.9 的证明, 可知 h' 是单调的. 同时有 \hat{u} 是变量 $T_{out}^{P_{jr1}}$ 的上界. 因此有

$$h'(\hat{u}) < \underline{H}_{out}^{P_{jr0}} - \overline{H}_{out}^{P_{jr1}} \leq \tilde{H}_{out}^{P_{jr0}} - \tilde{H}_{out}^{P_{jr1}} = h'(\tilde{T}_{out}^{P_{jr1}}) \Rightarrow \hat{u} < \tilde{T}_{out}^{P_{jr1}}.$$

这与 $\tilde{\Psi}$ 的可行性矛盾. 证毕. \square

定理 2.12. 算法 3 不改变 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 的可行域. 当算法 3 终止时, 要么存在 $\tilde{\Psi}$ 使得

$$\tilde{T}_{out}^{S_j} = \overline{T}_{out}^{S_j}, \quad j = 1, \dots, N^S - 1,$$

要么返回 $(\text{HOPnr1}(\underline{z}, \bar{z}))$ 不可行.

证明. 首先证明算法 3 中的变量缩紧步 16 是正确的. 然后验证步骤 8, 12 和 18 的正确性. 最后给出可行解 $\tilde{\Psi}$ 满足陈述条件.

由引理 2.9, 2.10 和约束 (2.3) 可知算法 3 中步 14 对变量 $T_{out}^{S_j}$ 上界的缩紧是正确的, 且 $\bar{T}_{out}^{S_j}$ 在算法过程中是单调递减的. 我们利用反证法说明变量 $H_{out}^{S_j}$ 的缩紧是正确的. 假设在预处理前存在 (HOPnr1(\underline{z}, \bar{z})) 的可行解 $\tilde{\Psi}$, 且 $\tilde{H}_{out}^{S_j} < \underline{H}_{out}^{S_j}$ 成立. 其中 $\underline{H}_{out}^{S_j}$ 是在步 16 后得到的. 根据步 4 和 5, 有

$$\tilde{H}_{out}^{S_j} < \underline{H}_{out}^{S_j} = \max_{r=0, \dots, N_j^P} \left\{ \underline{H}_{out}^{P_{jr}} + \sum_{t=1}^r \left[f(\phi_{jt}^0 \bar{T}_{out}^{S_j} + \psi_{jt}^0, Q_{jt}, D_{jt}) L_{jt} + \Delta Z_{jt} \right] \right\}.$$

则存在 r_2 使得

$$\begin{aligned} \tilde{H}_{out}^{S_j} &= \tilde{H}_{out}^{P_{jr_2}} + \sum_{t=1}^{r_2} \left[f(\phi_{jt}^0 \tilde{T}_{out}^{S_j} + \psi_{jt}^0, Q_{jt}, D_{jt}) L_{jt} + \Delta Z_{jt} \right] \\ &< \underline{H}_{out}^{P_{jr_2}} + \sum_{t=1}^{r_2} \left[f(\phi_{jt}^0 \bar{T}_{out}^{S_j} + \psi_{jt}^0, Q_{jt}, D_{jt}) L_{jt} + \Delta Z_{jt} \right]. \end{aligned}$$

因为 f 是单调的, 有 $\tilde{H}_{out}^{P_{jr_2}} < \underline{H}_{out}^{P_{jr_2}}$ 或 $\tilde{T}_{out}^{S_j} > \bar{T}_{out}^{S_j}$ 成立. 任何一条的成立都与 $\tilde{\Psi}$ 的可行性矛盾. 约束 (2.5), (2.7) 和 (2.12) 保证了步 16 对 $H_{in}^{S_j}$ 与 $T_{in}^{S_j}$ 的缩紧是正确的.

引理 2.11 和 2.10 保证了步 8 和 12 对不可行性的判断是正确的. 显然步 18 也是正确的.

当算法 3 终止时, 我们得到的解 $\tilde{\Psi}$ 满足

$$\tilde{z} = \bar{z}, \Delta \tilde{H}_j^{SP} = \bar{y}_j H_j^{SP}, \tilde{H}_{out}^{S_j} = \underline{H}_{out}^{S_j}, j = 1, \dots, N^S - 1,$$

且 ΔT_j 的值由 $T_{out}^{S_j} = \bar{T}_{out}^{S_j}$ 以及约束 (2.3), (2.4) 确定. 因此 $\tilde{\Psi}$ 满足定理陈述的条件. 注意到下标 j 每迭代一次, 我们都得到一个部分解满足 $H_{out}^{S_j} = \underline{H}_{out}^{S_j}$, $T_{out}^{S_j} = \bar{T}_{out}^{S_j}$. 这样的部分解组成了 $\tilde{\Psi}$. 因此 $\tilde{\Psi}$ 的可行性可被算法 3 保证. 得证. \square

2.3.2 外逼近方法和热启动策略

由于 (HOPnr2) 是凸规划问题, 因此可以使用内点法和外逼近 [87] 等方法进行求解. 由于 HOPBB 算法需要多次求解仅有部分界约束不同的凸规划问题, 若能选取合适的算法降低每个凸问题的求解时间则可大大提升 HOPBB 算法的效率. 这里我们提出结合外逼近方法和热启动策略求解 (HOPnr2) 的算法. 首先将 (HOPnr2) 转化为线性外逼近问题, 即对约束 (2.20) 进行线性外逼近, 得到如下的

线性规划问题.

$$\begin{aligned}
 & \min_{\Psi} \quad C(x, \Delta H^{SP}, \Delta T) \\
 & \text{s. t.} \quad (2.1), (2.3) - (2.9), (2.12) - (2.19), \\
 & \text{(HOPlr}(\underline{z}, \bar{z}, \Omega)) \quad F_{jr} \geq \left[\zeta_{j,r,\omega} T_{ave}^{P_{jr}} + f(\omega, Q_{jr}, D_{jr}) - \zeta_{j,r,\omega} \right] L_{jr}, \\
 & \quad (j, r, \omega) \in \Omega.
 \end{aligned}$$

其中 $\zeta_{j,r,\omega} \in \partial f(\omega, Q_{jr}, D_{jr})$. ∂f 是函数 f 的次梯度集合. 算法 4 展示了我们利用外逼近方法求解 (HOPnr2) 的算法. 其中 Ω_0 在将算法 4 应用于 HOPBB 算法中时可选取为前次调用算法 4 时 Ω 的最终值. 这是由于不同子问题可以共用 f 的外逼近模型. 2.4 节的数值结果表明利用算法 4 和热启动求解 (HOPnr2) 的 HOPBB 算法比调用内点法的版本更加高效.

算法 4 (HOPnr2(\underline{z}, \bar{z})) 的外逼近算法

- 1: 初始化 $\Omega := \Omega_0$, 容限参数 $\epsilon > 0$;
- 2: 求解 (HOPlr($\underline{z}, \bar{z}, \Omega$)) 并得到最优解 $\check{\Psi}$, 利用 $\check{T}_{ave}^{P_{jr}}$ 和约束 (2.2) 计算 \hat{F} ;
- 3: 通过

$$(j_0, r_0) := \arg \max_{\substack{j=1, \dots, N^S-1 \\ r=1, \dots, N_j^P}} \{ \hat{F}_{jr} - \check{F}_{jr} \}, \quad vio_{max} := \hat{F}_{j_0 r_0} - \check{F}_{j_0 r_0};$$

找到最大违反度 vio_{max} 及其对应的下标 j_0, r_0

- 4: **if** $vio_{max} > \epsilon$ **then**
 - 5: 更新 $\Omega := \Omega \cup \{(j_0, r_0, \check{T}_{ave}^{P_{j_0 r_0}})\}$, 转步 2;
 - 6: **else**
 - 7: **return** $\check{\Psi}$ 是 (HOPnr2(\underline{z}, \bar{z})) 的近似最优解.
 - 8: **end if**
-

2.4 数值实验

本节中我们通过将 HOPBB 算法应用于一个热油管道的实例来说明方法的可行性和效率. 我们在 MATLAB R2018a 平台实现 HOPBB 算法. 如不做特别说明, 本节中的实验在一个装有酷睿 i7-7567U 型号的 CPU 和 16GB 的内存的计算机上进行.

我们的测试算例基于中国的 Q-T 热油管线. 整个管线包含 9 个热泵站以及若干原油注入点和分输点. 管线示意图如图 2.5 所示. 图 2.6 给出了 Q-T 管线的

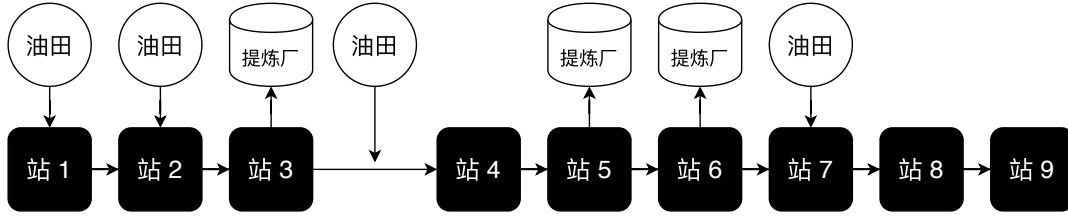


图 2.5 Q-T 管线布局示意图.

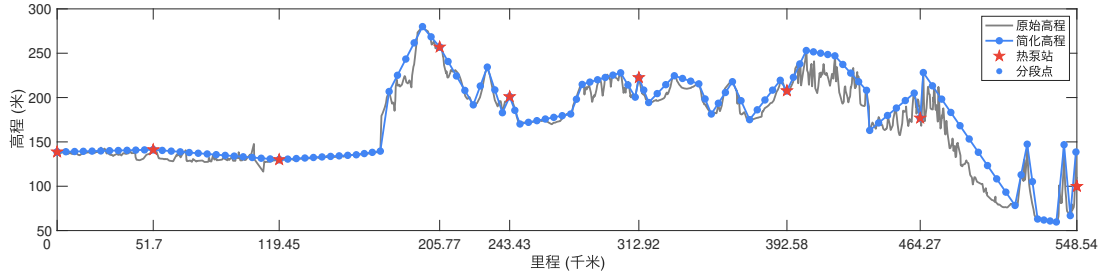


图 2.6 Q-T 管线的里程和高程图.

里程和高程图. 为了方便计算, 我们在测试中将图 2.6 中的“简化高程”作为高程数据进行建模. 进一步, 我们将管线中的管道分成总共 131 个管道片段. 每个片段的长度不超过 5 公里. 表 2.1–2.3 给出了加压泵、管道和原油数据. 通过曲线拟合的方式 (如图2.7所示), 我们得到如下的动力粘度 ν 表达式:

$$\nu(T) = [8.166 \times 10^6 \exp(-0.3302T) + 77.04 \exp(-0.02882T)] / 1000/\rho.$$

进而水力摩擦 F_{jr} 可由公式 (1.1) 计算得到. 容易验证的是当 $T > 0$ 时, 对应的水力摩擦系数 f 是光滑的, 且满足假设 2.8.

利用以上数据, 我们建模得到 (HOP) 问题并利用 HOPBB 算法进行求解, 将得到的解对应的运行方案 (下称“最优方案”) 与目前石油公司实际应用的方案 (下称“实际方案”) 进行对比. 表 2.4 和图 2.8 展示了加压、加热方案, 方案费用以及管线的压力、温度曲线. 表 2.4 中的 C_{power} 、 C_{fuel} 和 C_{total} 分别表示每站的加压总费用、加热总费用以及全线合计费用. 两方案的比较结果表明, 最优方案可节省 Q-T 管线 6.83% 的运行费用, 意味着当前的实际方案还有一定的节能空间. 具体的改进主要有两方面: (1) 最优方案对每站的加压、加热负载分配更加合理. 如两方案中前三站的温升差别较大. 由于第二站的体积流量比前三站中的另两站要大得多, 最优方案通过减少这一站的温升以降低费用. 此外, 选取更优的泵组合方案也很重要. 对于没有调速泵的站, 可能会通过限压阀降低压力以达到安全的

表 2.1 Q-T 热油管道问题中的部分常量和参数设置.

| | |
|---|----------------------------------|
| V_c | $3.36 \times 10^7 \text{ J/m}^3$ |
| η_j | 80% |
| C_p | 0.65 RMB/(kW·h) |
| C_f | 1.15 RMB/m ³ |
| 管壁厚度 | 8 mm |
| ρ | 859 kg/m ³ |
| c | 2400 J/(kg·°C) |
| $\underline{H}_{in}^{S_1}, \overline{H}_{in}^{S_1}$ | 57 m |
| $\underline{T}_{in}^{S_1}, \overline{T}_{in}^{S_1}$ | 40.7 °C |
| $\underline{H}_{in}^{S_j}, \underline{H}_{out}^{S_j}, \underline{H}_{out}^{P_{jr}}$ | 35.6 m |
| $\overline{H}_{in}^{S_j}, \overline{H}_{out}^{S_j}$ | 748.4 m |
| $\underline{T}_{in}^{S_j}, \underline{T}_{out}^{S_j}$ | 35 °C |
| $\overline{T}_{in}^{S_j}, \overline{T}_{out}^{S_j}$ | 48 °C |

表 2.2 Q-T 管线所输原油的动力粘度.

| 温度 (°C) | 动力粘度 (mPa·s) |
|---------|--------------|
| 35 | 107.4 |
| 36 | 82.6 |
| 37 | 64.1 |
| 38 | 55.7 |
| 39 | 48.3 |
| 40 | 41.9 |
| 42 | 27.6 |
| 46 | 21.9 |
| 50 | 19.2 |
| 54 | 16.9 |
| 60 | 13.7 |
| 68 | 10.7 |

表 2.3 Q-T 管线各热泵站的数据.

| 热泵站 | Q_{j0} | D_{j0} | $T_g^{P_{j0}}$ | K_{j0} | H_j^{CP} | N_j^{CP} | ξ_j^{CP} | \underline{H}_j^{SP} | \overline{H}_j^{SP} | N_j^{SP} | ξ_j^{SP} |
|-----|----------|----------|----------------|----------|------------|------------|--------------|------------------------|-----------------------|------------|--------------|
| 1 | 2212 | 0.740 | 4.50 | 1.87 | 235.76 | 4 | 78.7% | - | - | 0 | - |
| 2 | 2810 | 0.740 | 1.69 | 1.09 | - | 0 | - | - | - | 0 | - |
| 3 | 2215 | 0.772 | 8.80 | 0.79 | 245.70 | 3 | 78.0% | 94.01 | 244.24 | 1 | 80.3% |
| 4 | 2941 | 0.622 | 0.10 | 1.63 | 222.36 | 4 | 83.5% | - | - | 0 | - |
| 5 | 2751 | 0.685 | 2.70 | 1.17 | 229.96 | 3 | 83.3% | 102.84 | 231.35 | 1 | 83.8% |
| 6 | 2022 | 0.695 | 6.53 | 1.68 | 239.56 | 3 | 75.7% | 95.34 | 239.56 | 1 | 79.5% |
| 7 | 2102 | 0.715 | 2.60 | 1.89 | 237.96 | 3 | 77.0% | 92.94 | 237.96 | 1 | 79.6% |
| 8 | 2047 | 0.705 | 1.60 | 1.48 | - | 0 | - | - | - | 0 | - |

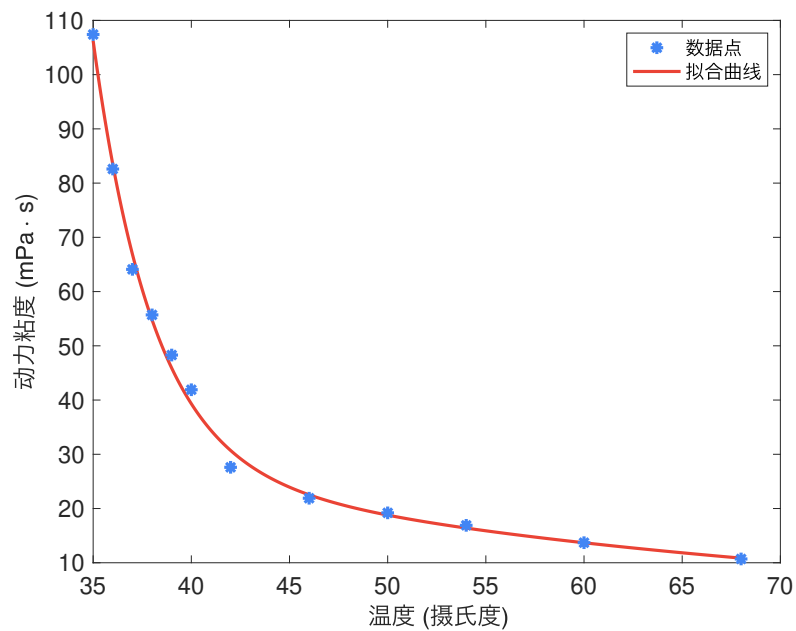


图 2.7 原油粘温曲线.

出站压力要求. 对这样的情况可通过调整压力负载到其他的站以减少不必要的能源浪费. 实际方案中浪费掉的压力有 11.84 MPa, 而最优方案仅有 0.13 MPa. (2) 最优方案有更优的整体运行策略. 图 2.8 表明最优方案的平均出站温度更高. 且注意到第一和第三站的出站温度达到设置的上限. 若提高每站的出站温度上限 $\bar{T}_{out}^{S_j}$ 可能使总体运行费用进一步降低. 图 2.9 展示了 $\bar{T}_{out}^{S_j}$ 选取为 45–54°C 下的最优费用. 这一结果表明在 $\bar{T}_{out}^{S_j}$ 超过 53.6°C 前, 总运行费用会随着出站温度上界的升高而降低. 由此, 若能通过改变管道和加热炉的配置提升热泵站的出站温度, Q-T 线的总运行费用的降幅可超过 8% (如图 2.10 所示).

表 2.4 最优方案与实际方案的加压、加热方案及费用对比 (费用单位: 人民币/天).

| 热泵站 | 实际方案 | | | | | 最优方案 | | | | |
|-------------|-----------|-----------------|-------------|------------|------------|-----------|-----------------|-------------|------------|------------|
| | x | ΔH^{SP} | C_{power} | ΔT | C_{fuel} | x | ΔH^{SP} | C_{power} | ΔT | C_{fuel} |
| 1 | 3 | 0 | 72510.05 | 3.00 | 14047.23 | 3 | 0 | 72510.05 | 7.30 | 34181.59 |
| 2 | 0 | 0 | 0 | 3.50 | 20818.94 | 0 | 0 | 0.00 | 1.24 | 7391.16 |
| 3 | 1 | 219.96 | 47578.26 | 2.90 | 13597.40 | 1 | 244.24 | 50020.79 | 10.13 | 47496.64 |
| 4 | 3 | 0 | 85751.64 | 2.80 | 17431.60 | 1 | 0 | 28583.88 | 3.83 | 23825.27 |
| 5 | 2 | 104.33 | 67879.53 | 4.10 | 23875.84 | 1 | 231.35 | 55390.45 | 0.00 | 0 |
| 6 | 1 | 170.27 | 39140.92 | 6.48 | 27735.78 | 2 | 224.68 | 67530.40 | 6.84 | 29293.14 |
| 7 | 1 | 203.94 | 43330.03 | 8.80 | 39156.12 | 0 | 175.05 | 16858.35 | 7.46 | 33178.22 |
| 8 | 0 | 0 | 0 | 6.90 | 29898.62 | 0 | 0 | 0.00 | 9.10 | 39416.83 |
| C_{total} | 542751.95 | | | | | 505676.76 | | | | |

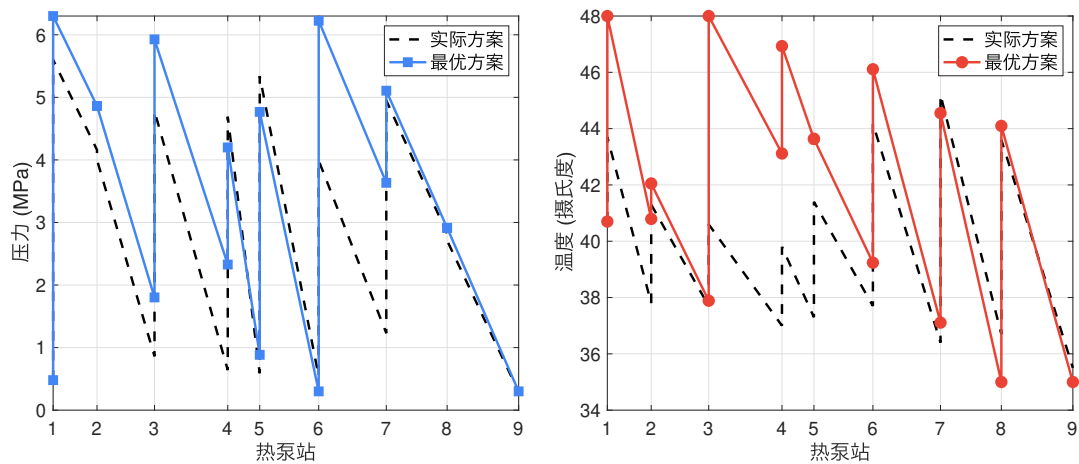


图 2.8 实际方案与最优方案的压力 (左)、温度 (右) 曲线.

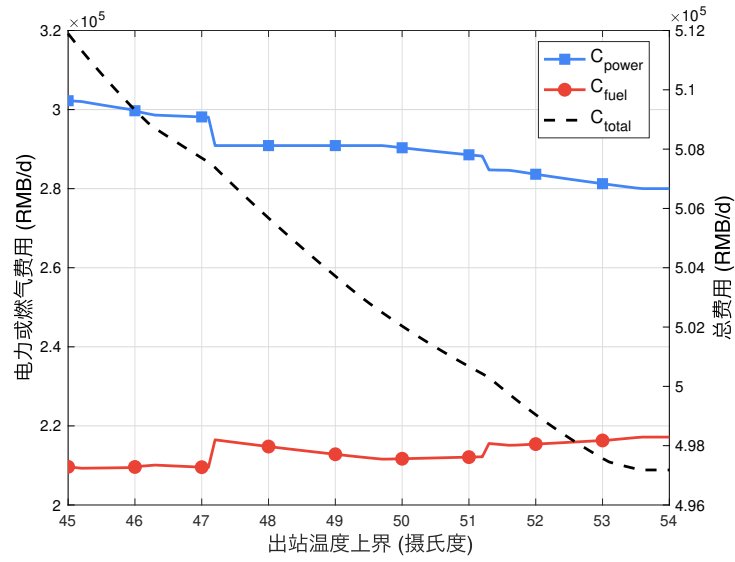


图 2.9 不同出站温度上界下的最优运行费用。

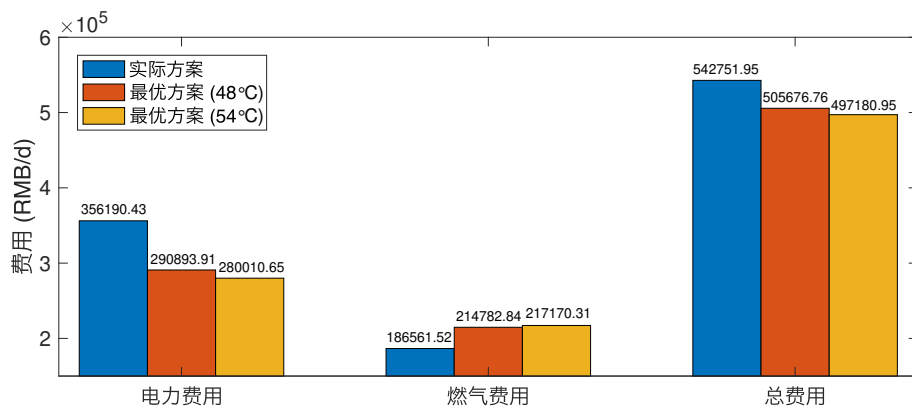


图 2.10 实际方案及出站温度上界分别取 48°C 和 54°C 的两个最优方案的运行费用。

此外, 我们关心 HOPBB 算法在求解速度和解质量的表现. 由于 Q-T 热油管道问题中的非线性函数 f 是光滑的, 因此这个实例的 (HOPnr2) 松弛问题可通过两种方法进行求解: 内点法 (记为 HOPBB-IPM) 与结合热启动的外逼近算法 4 (记为 HOPBB-OAP). 我们通过将这两个版本的 HOPBB 算法与目前最先进的 MINLP 求解器进行比较来说明 HOPBB 的效率. 包括商业求解器 BARON, ANTIGONE, LINDOGlobal 和开源求解器 Couenne, SCIP 在内的 MINLP 求解器均可以求解到非凸 MINLP 问题的全局最优解. 我们利用 GAMS 建模语言对 (HOP) 问题进行建模, 在 NEOS 服务器 [88] 对这些 MINLP 求解器进行测试. HOPBB-IPM 使用开源 NLP 求解器 IPOPT [89] 求解每个子问题的 (HOPnr2) 松弛. HOPBB-OAP 调用 MATLAB 内建的 LINPROG 函数求解 (HOPlr). 算法 4 中的容限 ϵ 设置为 10^{-6} . MINLP 求解器的设置保持为 NEOS 服务器的默认设置. 所有的方法求解时间限制为 3600 秒, gap 取 10^{-5} .

表 2.5 求解 Q-T 热油管道问题的不同方法效果对比.

| 方法 | 最优目标值 | gap | 可行性 | 迭代次数 | 时间 (秒) |
|-------------|----------|-----------------------|------------------------|---------|--------|
| Couenne | 21073.49 | 5.33×10^{-3} | 9.59×10^{-8} | 196239 | 3600 |
| SCIP | 21069.87 | 2.62×10^{-3} | 2.71×10^{-9} | 1548507 | 3600 |
| LINDOGlobal | 21069.87 | 1.11×10^{-8} | 8.38×10^{-13} | 845 | 1047 |
| ANTIGONE | 21069.87 | 1.00×10^{-5} | 5.84×10^{-8} | 34439 | 651 |
| BARON | 21069.87 | 9.85×10^{-3} | 2.55×10^{-10} | 28007 | 541 |
| HOPBB-IPM | 21069.86 | 0 | 7.48×10^{-6} | 25 | 41 |
| HOPBB-OAP | 21069.85 | 0 | 8.59×10^{-7} | 25 | 9 |

表 2.5 展示了 7 种方法的求解效果对比. 其中“可行性”表示最优解的最大约束违反量; “迭代次数”对 ANTIGONE 表示使用的割平面总个数, 对其他方法表示总节点数. 表 2.5 的结果表明, 尽管所有的方法得到的最优解均能达到可接受的可行性水平, 但它们的表现差距是较大的. 开源求解器 Couenne 和 SCIP 没有在给定时间限制内求解完毕. 其中 Couenne 未求得全局最优解, 而 SCIP 在求解了 150 多万节点 (消耗内存 1233 MB) 后也未能完成解的最优性证明. 对于三款商业求解器, LINDOGlobal 与 ANTIGONE 成功在时间限制内完成优化求解. BARON 由于遇到了数值问题而提前终止了算法 (gap 未达到 10^{-5} 以下). 相比于这五个通用的 MINLP 求解器, 两个版本的 HOPBB 算法均有更高效表现. HOPBB-OAP 的求解速度比内点法的版本更快, 且在这一算例的计算上比 BARON 快 50 多倍.

尽管在不同的计算平台进行算法效率对比是不够公平的,但从迭代次数这一项来看也足以说明 HOPBB 算法与其他方法的巨大差距. 需要额外说明的是,商业 MINLP 求解器的数值表现离不开它们依赖的强大的求解子问题的商业软件,如 CPLEX [90], MINOS [91], CONOPT [92] 等. 然而 HOPBB-IPM 借助的是开源求解器 IPOPT. 两个 HOPBB 算法能取得更好的求解效率主要是因为所设计的分支定界算法降低了计算复杂度. 为了避免结论的偶然性,我们通过变化 Q-T 热油管道问题建模时采用的管道片段的长度,从 2000 米到 20000 米,得到了 10 个不同规模的算例. 完整的测试结果以及性能分析结果分别如表 2.6 和图 2.11 所示. 总之,对于求解 (HOP) 问题, HOPBB 算法是十分高效的.

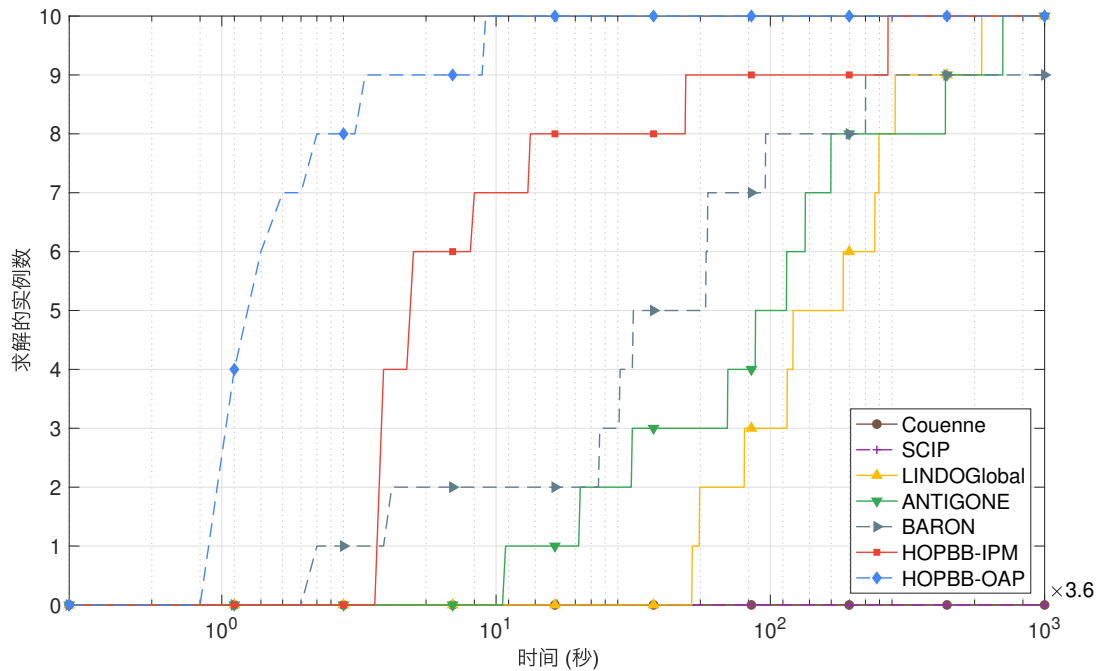


图 2.11 不同方法求解 10 个 (HOP) 问题实例的表现. 同一时刻求解实例个数越多, 性能越好.

表 2.6 不同方法求解 10 个 (HOP) 实例的求解时间, 迭代次数和 gap 对比. BARON 由于遇到数值问题而在未满足 gap 的要求情况下提前终止.

| 算例 | 最大管道片段长度 (米) | 标准 | Couenne | SCIP | LINDOGlobal | ANTIGONE | BARON | HOPBB-IPM | HOPBB-OAP |
|----|--------------|--------|----------|----------|-------------|----------|----------|-----------|-----------|
| 1 | 20000 | 时间 (秒) | 3600 | 3600 | 187 | 251 | 212 | 14 | 3 |
| | | 迭代次数 | 423987 | 3002817 | 435 | 128874 | 18104 | 25 | 25 |
| 2 | 18000 | gap | 4.47E-03 | 8.79E-04 | 1.11E-08 | 9.95E-06 | 9.96E-03 | 0 | 0 |
| | | 时间 (秒) | 3600 | 3600 | 289 | 38 | 101 | 14 | 3 |
| 3 | 16000 | 迭代次数 | 394213 | 2964184 | 563 | 7271 | 11089 | 25 | 25 |
| | | gap | 4.67E-03 | 8.03E-04 | 9.76E-06 | 9.95E-06 | 1.09E-02 | 0 | 0 |
| 4 | 14000 | 时间 (秒) | 3600 | 3600 | 198 | 72 | 86 | 14 | 3 |
| | | 迭代次数 | 507918 | 2395276 | 523 | 10450 | 8194 | 25 | 25 |
| 5 | 12000 | gap | 4.93E-03 | 9.86E-04 | 9.49E-06 | 9.95E-06 | 1.11E-02 | 0 | 0 |
| | | 时间 (秒) | 3600 | 3600 | 436 | 112 | 114 | 13 | 4 |
| 6 | 10000 | 迭代次数 | 382188 | 2848651 | 761 | 17357 | 10930 | 25 | 25 |
| | | gap | 4.96E-03 | 9.82E-04 | 9.48E-06 | 9.95E-06 | 1.05E-02 | 0 | 0 |
| 7 | 8000 | 时间 (秒) | 3600 | 3600 | 664 | 482 | 15 | 18 | 4 |
| | | 迭代次数 | 337350 | 2446325 | 829 | 120119 | 1104 | 25 | 25 |
| 8 | 6000 | gap | 4.82E-03 | 1.05E-03 | 9.35E-06 | 9.95E-06 | 1.63E-02 | 0 | 0 |
| | | 时间 (秒) | 3600 | 3600 | 414 | 317 | 7 | 18 | 5 |
| 9 | 4000 | 迭代次数 | 353961 | 2086513 | 805 | 39680 | 380 | 25 | 25 |
| | | gap | 4.87E-03 | 1.20E-03 | 9.25E-06 | 9.96E-06 | 1.66E-02 | 0 | 0 |
| 10 | 2000 | 时间 (秒) | 3600 | 3600 | 896 | 598 | 210 | 30 | 5 |
| | | 迭代次数 | 261048 | 1870753 | 853 | 219813 | 21569 | 25 | 25 |
| 11 | 4000 | gap | 5.04E-03 | 1.68E-03 | 1.11E-08 | 9.96E-06 | 9.44E-03 | 0 | 0 |
| | | 时间 (秒) | 3600 | 3600 | 865 | 412 | 345 | 48 | 7 |
| 12 | 6000 | 迭代次数 | 201114 | 1615645 | 1105 | 30449 | 20333 | 25 | 25 |
| | | gap | 4.66E-03 | 2.25E-03 | 9.00E-06 | 9.96E-06 | 9.44E-03 | 0 | 0 |
| 13 | 4000 | 时间 (秒) | 3600 | 3600 | 1027 | 1565 | 800 | 176 | 11 |
| | | 迭代次数 | 177110 | 818881 | 819 | 71602 | 34903 | 25 | 25 |
| 14 | 2000 | gap | 4.74E-03 | 3.24E-03 | 8.82E-06 | 1.00E-05 | 8.92E-03 | 0 | 0 |
| | | 时间 (秒) | 3600 | 3600 | 2123 | 2534 | 3600 | 966 | 33 |
| 15 | 2000 | 迭代次数 | 48574 | 586670 | 941 | 78380 | 51565 | 25 | 25 |
| | | gap | 5.71E-03 | 5.33E-03 | 8.67E-06 | 9.97E-06 | 8.82E-03 | 0 | 0 |

2.5 小结

在本章节中, 我们研究了热油管道问题的全局算法. 通过考虑两种加压泵以及通用的水力摩擦非线性约束建模, 我们给出了热油管道问题的一个精确的非凸 MINLP 数学模型. 不同于传统的基于元启发式的方法, 我们在分析了两种松弛问题的等价性后, 为热油管道问题设计了基于分支定界框架的 HOPBB 全局优化算法. 此外为了实际应用这一算法, 我们还设计了预处理算法保证等价性的成立, 即实现了算法的可行性, 以及提出结合外逼近和热启动的凸松弛问题求解方法. 为了说明算法的可行性和计算效率, 我们将 HOPBB 算法应用于我国的 Q-T 热油管线实例. 数值实验结果表明我们提出的算法能够比现有方案节省 6.83% 的运行费用, 并且给出了 Q-T 热油管道进一步改进的方向. 此外, 与最先进的 MINLP 求解器的对比结果也体现出 HOPBB 算法在热油管道问题上的求解效率具有极大的优势.

事实上, 本工作中的 MINLP 模型和 HOPBB 算法还可以进一步改进提升. 约束 (2.3) 简化了压力和温度间的相互作用. 精确的轴向温降公式应考虑 $T_f^{P_{jr}}$ 随水力摩擦的变化而变化, 即

$$T_f^{P_{jr}} = \frac{g F_{jr} \rho Q_{jr}}{K_{jr} \pi d_{jr} L_{jr}}, \quad j = 1, \dots, N^S - 1, \quad r = 1, \dots, N_j^P.$$

精确的温降公式虽然建模更加准确, 但大大提升了 MINLP 问题的难度. 由于约束 (2.2) 在松弛之后可能导致压头曲线和温度曲线同时不可行, 因此 (HOPnr1) 和 (HOPnr2) 间的等价性难以得到保证. 在本工作中考虑的 Q-T 热油管线, $T_f^{P_{jr}}$ 在最大和最小原油粘度间的变化在 1°C 左右, 且由于 $e^{-\alpha_{jr} L_{jr}}$ 接近 1, $T_{out}^{P_{jr}}$ 的变化更小. 因此本工作中将 $T_f^{P_{jr}}$ 近似为常量在一定精度下是合理的. 针对近似不适用的工程场景, 如何设计 MINLP 模型的不确定性全局优化算法将是未来的工作方向. 此外, 如何将本工作中的建模和算法设计思想拓展到其他类似的管道优化问题, 如自来水管或天然气管道运行优化问题, 也是一个值得思考的方向.

第3章 动力下降制导问题的两类模型和算法

本章考虑动力下降制导问题的模型和求解算法. 对这一问题, 我们分别转化为 **SOCP** 和 **NSOCP** 问题, 并利用序列凸化和可行方向内点算法进行求解. 本章安排如下: 3.1 节介绍本论文考虑的带有气动阻力和推力限制约束的动力下降制导最优控制模型; 3.2 节给出转化成 **SOCP** 问题的具体步骤和序列凸化算法的具体描述; 3.3 节给出第二种模型——**NSOCP** 模型和相应的可行方向内点算法; 3.4 节展示两类模型和算法的数值结果; 最后在 3.5 节进行本章总结. 由于两种模型的记号较多, 为保持记号简洁性, 我们将 3.1, 3.2 和 3.4.1 节分为第一部分, 将 3.3 和 3.4.2 节分为第二部分. 如无特别说明, 两部分的数学记号不通用.

3.1 最优控制模型

本节我们给出垂直起降方案下连续时间的燃料最优动力下降制导模型. 我们约定: 标量用普通字体表示, 如质量 m 等; 矢量用粗体表示, 如位置 \mathbf{r} , 速度 \mathbf{v} , 推力 \mathbf{T} 等; 字母记号上方的点表示物理量对时间的导数, 如 $\dot{\mathbf{r}} = d\mathbf{r}/dt$. 本节考虑的火箭运动模型为 3-DOF 模型, 即将火箭主体视为一个质点且在除推力方向约束外仅考虑火箭的平行移动. 三维坐标的三个维度正方向依次为北 (north, N)、上 (up, U)、东 (east, E). 令 $t \in \mathbb{R}_+$ 表示飞行时刻, t_f 表示终端时刻. 在燃料最优的最优控制模型中, 我们希望通过确定各个时刻的推力指令 \mathbf{T} 和飞行总时长 (也即终端时刻) t_f 使得火箭着陆时的剩余燃料质量极大化, 即极大化火箭终端质量 $m(t_f)$. 火箭的飞行需要满足若干动力学约束. 结合牛顿第二定律可以得到如下的动力学方程:

$$\dot{m}(t) = -\alpha \|\mathbf{T}(t)\|, \quad (3.1)$$

$$\dot{\mathbf{r}}(t) = \mathbf{v}(t), \quad (3.2)$$

$$\dot{\mathbf{v}}(t) = \frac{1}{m(t)} [\mathbf{T}(t) + \mathbf{D}(t)] + \mathbf{g}. \quad (3.3)$$

其中 $m \in \mathbb{R}_+$ 、 $\mathbf{r} \in \mathbb{R}^3$ 、 $\mathbf{v} \in \mathbb{R}^3$ 、 $\mathbf{T} \in \mathbb{R}^3$ 分别表示火箭的质量、位置、速度和所受推力. 在飞行过程中, 火箭的质量随着燃料的消耗而逐渐减小. 因此 \dot{m} 反映了燃料消耗速度, 它与发动机输出推力大小有关. 常参数 α 和 \mathbf{g} 分别为发动机比冲的

倒数和重力加速度. $\mathbf{D}(t) \in \mathbb{R}^3$ 表示 t 时刻火箭所受气动阻力, 其具体计算公式为:

$$\mathbf{D}(t) = -\frac{1}{2}\rho S_D C_D \|\mathbf{v}(t)\| \mathbf{v}(t).$$

其中 ρ 为大气密度, S_D 为火箭有效横截面积, C_D 表示气动力轴向系数.

此外, 火箭的动力下降制导中还需满足一定的状态和控制约束. 首先, 终端时刻的燃料剩余量必须是非负的, 即:

$$m(t_f) \geq m_{\text{dry}}. \quad (3.4)$$

m_{dry} 表示火箭的干质量. 其次, 火箭在飞行过程中既不能与地面相撞, 也不能在除着陆点外的其他位置与地面距离过近以防被地面凸起物干涉飞行. 因此将火箭的飞行区域限制在一个锥形区域内 (见图 3.1), 即:

$$\|\mathbf{r}(t)\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}(t). \quad (3.5)$$

其中 $\mathbf{e}_U = [0 \ 1 \ 0]^\top$ 表示 U 方向的单位向量. 最后, 发动机推力的大小、方向 and 变化速率也需要一定的限制: 发动机不仅有推力上限, 也不能在飞行途中熄火 (有推力下限); 推力方向需与箭体呈小角度锐角; 推力大小在单位时间内的变化量是有限的. 以上限制可分别表示为如下的约束:

$$T_{\min} \leq \|\mathbf{T}(t)\| \leq T_{\max}, \quad (3.6)$$

$$\frac{(-\mathbf{v}(t))^\top \mathbf{T}(t)}{\|-\mathbf{v}(t)\| \|\mathbf{T}(t)\|} \geq \cos \theta_2, \quad (3.7)$$

$$\dot{T}_{\min} \leq \frac{d \|\mathbf{T}(t)\|}{dt} \leq \dot{T}_{\max}. \quad (3.8)$$

注意到在 3-DOF 模型中, 由于将箭体抽象为质点, 我们无法精确计算攻角. 事实上, 着陆段开始前的调姿操作会将箭体主轴与速度方向调节成接近平行 (见图 3.1). 因此在约束 (3.7) 中, 我们把箭体方向近似为速度反方向 $-\mathbf{v}$.

除以上约束外, 模型中还必须给定部分状态和控制变量的边界条件, 即初始质量、位置、速度、推力方向以及终端位置、速度和推力方向.

$$m(0) = m_{\text{init}}, \mathbf{r}(0) = \mathbf{r}_{\text{init}}, \mathbf{v}(0) = \mathbf{v}_{\text{init}}, \mathbf{T}(0) = \|\mathbf{T}(0)\| \frac{-\mathbf{v}_{\text{init}}}{\|-\mathbf{v}_{\text{init}}\|}, \quad (3.9)$$

$$\mathbf{r}(t_f) = \mathbf{0}, \mathbf{v}(t_f) = \mathbf{0}, \mathbf{T}(t_f) = \|\mathbf{T}(t_f)\| \mathbf{e}_U. \quad (3.10)$$

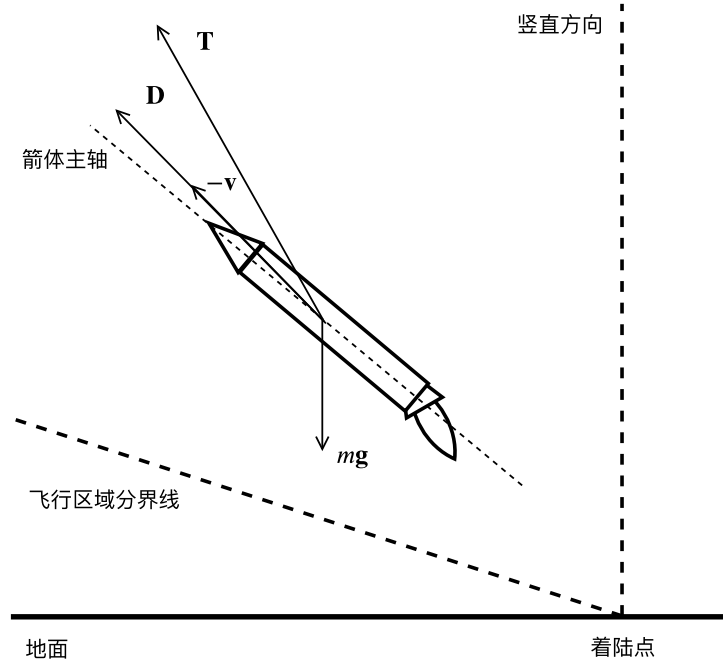


图 3.1 火箭受力分析及飞行场景示意图.

在约束 (3.9) 中, 初始的推力方向选取为速度反方向. 终端条件 (3.10) 表示火箭不仅要降落在指定位置, 而且落地速度和横向加速度必须为 0.

令 $\mathbf{X} \in \mathbb{R}^7$ 为包含所有状态变量的矢量, 即:

$$\mathbf{X}(t) = [m(t) \ \mathbf{r}(t) \ \mathbf{v}(t)].$$

综上, 燃料最优动力下降制导的最优控制模型可归结为:

$$\begin{aligned} (\text{ROC}) \quad & \min_{\mathbf{X}, \mathbf{T}, t_f} \quad -m(t_f) \\ & \text{s. t.} \quad (3.1) - (3.10). \end{aligned}$$

注意到 (ROC) 问题是连续时间的最优控制问题, 可通过时间离散化的方式将此问题转化为 NLP 问题进行求解. 此外, (ROC) 问题还包含非凸约束, 如约束 (3.1)、(3.3)、(3.6)、(3.8) 等. 在下一节中我们通过无损凸化、时间离散化和线性化的方式将 (ROC) 问题转化为凸 SOCP 问题.

3.2 序列凸化方法

3.2.1 模型转化

3.2.1.1 无损凸化

参考文献 [33] 等的无损凸化方法, 我们引入新的控制变量 Γ 表示推力大小的上界, 即

$$\|\mathbf{T}(t)\| \leq \Gamma(t). \quad (3.11)$$

同时将约束 (3.1)、(3.6)–(3.10) 中的 $\|\mathbf{T}\|$ 替换为 Γ . 文献 [33] 证明了在最优解处这种凸化方式是无损的, 即最终得到的解使 (3.11) 式取等号. 虽然目前还不能证明包含气动阻力动力学约束的模型同样可以保持无损性, 但从数值结果 (见图 3.5) 来看, 这种凸化方式在我们的模型中也是无损的.

3.2.1.2 时间离散化

将 0 时刻到 t_f 时刻的飞行时间均匀分成 N 等份, 每份时间长度记为 Δt , 同时以下标 k 表示第 k 个时间点, 即

$$t_f = N\Delta t, \quad \mathbf{X}_k = \mathbf{X}(kt_f/N), \quad k = 0, 1, \dots, N.$$

当 N 确定时, 由于 t_f 是变量, 因此 Δt 也是变量. 微分方程约束 (3.1)–(3.3) 和 (3.8) 可通过差分方法转化为如下有限个线性或非线性约束:

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k(\mathbf{Y}_k), \quad k = 0, 1, \dots, N-1, \quad (3.12)$$

$$\dot{T}_{\min}\Delta t \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max}\Delta t, \quad k = 0, 1, \dots, N-1.$$

其中矢量值函数 $\mathbf{F}_k(\cdot) : \mathbb{R}^{23} \rightarrow \mathbb{R}^7$ 和矢量 $\mathbf{Y}_k \in \mathbb{R}^{23}$ 定义为:

$$\begin{aligned} \mathbf{F}_k(\mathbf{Y}_k) &= \frac{\Delta t}{2} \begin{pmatrix} -\alpha(\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[\frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}} \right] + 2\mathbf{g} \end{pmatrix}, \\ \mathbf{Y}_k &= [\Delta t \quad \mathbf{X}_k \quad \mathbf{U}_k \quad \mathbf{X}_{k+1} \quad \mathbf{U}_{k+1}], \\ \mathbf{U}_k &= [\mathbf{T}_k \quad \Gamma_k]. \end{aligned}$$

3.2.1.3 线性化

最后, 将非凸动力学约束 (3.12) 和推力方向约束 (3.7) 通过线性化的方式进行凸化. 令

$$G_k(\mathbf{Y}_k) = \mathbf{v}_k^\top \mathbf{T}_k + \cos \theta_2 \|\mathbf{v}_k\| \Gamma_k,$$

可以得到如下线性约束:

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1,$$

$$G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1.$$

这里 $\bar{\mathbf{Y}}_k$ 是一组已知轨迹, $\mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k) \in \mathbb{R}^{7 \times 23}$ 、 $\mathcal{J}G_k(\bar{\mathbf{Y}}_k) \in \mathbb{R}^{1 \times 23}$ 分别是 \mathbf{F}_k 、 G_k 在 $\bar{\mathbf{Y}}_k$ 处的 Jacobian 矩阵. 在序列凸化方法中, $\bar{\mathbf{Y}}_k$ 可选取为上一次迭代的轨迹, 因此需要在算法开始时给定一条初始轨迹. 合理地选取初始轨迹将有助于减少迭代次数. 我们将在 3.2.3.1 节给出针对此模型的构造初始解方法.

3.2.2 SOCP 模型

综上, 我们给出 (ROC) 问题经离散化和凸化后的凸规划模型:

$$(\text{RSOCP}(\bar{\mathbf{Y}})) \quad \min_{\mathbf{Y}} \quad -m_N$$

$$\text{s. t.} \quad \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1, \quad (3.13)$$

$$T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \quad (3.14)$$

$$\|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \quad (3.15)$$

$$G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1, \quad (3.16)$$

$$\dot{T}_{\min} \Delta t \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta t, \quad k = 0, 1, \dots, N-1, \quad (3.17)$$

$$m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|\mathbf{v}_{\text{init}}\|}, \quad (3.18)$$

$$m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U. \quad (3.19)$$

上述问题仅含线性约束和二阶锥约束, 因此是凸 SOCP 问题. 当 (RSOCP) 问题的解 \mathbf{Y} 满足以下条件时

$$\mathbf{Y}_k = \bar{\mathbf{Y}}_k, \quad k = 0, 1, \dots, N-1,$$

$$\|\mathbf{T}_k\| = \Gamma_k, \quad k = 1, 2, \dots, N-1,$$

其可视为 (ROC) 问题的一个离散解.

3.2.3 模型求解

3.2.3.1 初始化

我们采用类似文献 [37] 中的方式计算初始轨迹 $\mathbf{Y}^{(0)}$. 给定终端时刻猜测值 τ_f , 可得变量 Δt 的猜测值 $\Delta\tau = \tau_f/N$. 假定终端质量为干质量 m_{dry} . 利用质量 m 和速度 \mathbf{v} 的初始和终端值, 通过线性插值的方式得到全部 $N+1$ 个时刻的质量和速度轨迹, 分别记为 μ 和 \mathbf{w} , 即

$$\begin{aligned}\mu_k &= m_{\text{init}} - \frac{k}{N} (m_{\text{init}} - m_{\text{dry}}), \quad k = 0, 1, \dots, N, \\ \mathbf{w}_k &= \mathbf{v}_{\text{init}} - \frac{k}{N} \mathbf{v}_{\text{init}}, \quad k = 0, 1, \dots, N.\end{aligned}$$

通过将变量 Δt 近似为 $\Delta\tau$ 和部分变量 m 和 \mathbf{v} 近似为 μ 和 \mathbf{w} , 可得到如下的初始化问题:

$$\begin{aligned}(\text{RSOCP-INIT}) \quad & \min_{\mathbf{Z}} \quad -m_N + \beta \|\mathbf{a}^R\| \\ \text{s. t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{\text{init}}(\mathbf{Z}_k), \quad k = 0, 1, \dots, N-1, \\ & T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \\ & \|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^\top \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\ & \mathbf{w}_k^\top \mathbf{T}_k + \cos \theta_2 \|\mathbf{w}_k\| \Gamma_k \leq 0, \quad k = 1, 2, \dots, N-1, \\ & \dot{T}_{\min} \Delta\tau \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta\tau, \quad k = 0, 1, \dots, N-1, \\ & m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|\mathbf{v}_{\text{init}}\|}, \\ & m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U.\end{aligned}$$

其中 \mathbf{a}^R 、 \mathbf{Z} 、 \mathbf{Z}_k 、 $\mathbf{F}_k^{\text{init}}$ 定义为:

$$\begin{aligned}\mathbf{a}^R &= [\mathbf{a}_0^R \quad \mathbf{a}_1^R \quad \dots \quad \mathbf{a}_N^R], \\ \mathbf{Z} &= [\mathbf{Y} \quad \mathbf{a}^R], \quad \mathbf{Z}_k = [\mathbf{Y}_k \quad \mathbf{a}_k^R \quad \mathbf{a}_{k+1}^R], \\ \mathbf{F}_k^{\text{init}}(\mathbf{Z}_k) &= \frac{\Delta\tau}{2} \begin{pmatrix} -\alpha (\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[\frac{\mathbf{T}_k + \tilde{\mathbf{D}}_k}{\mu_k} + \frac{\mathbf{T}_{k+1} + \tilde{\mathbf{D}}_{k+1}}{\mu_{k+1}} \right] + \mathbf{a}_k^R + \mathbf{a}_{k+1}^R + 2\mathbf{g} \end{pmatrix}, \\ \tilde{\mathbf{D}}_k &= -\frac{1}{2} \rho S_D C_D \|\mathbf{w}_k\| \mathbf{v}_k.\end{aligned}$$

虽然 (RSOCP-INIT) 问题的目标函数是非线性函数, 但可通过引入新的变量和二阶锥约束进而将问题转化成 SOCP 问题进行求解. 参考文献 [37] 的方式, 我们在加速度中引入辅助加速度 \mathbf{a}^R 来避免问题不可行. 注意到当问题给定时, 初始轨迹仅依赖于终端时刻猜测值 τ_f . 事实上, 我们并不要求 τ_f 与最优轨迹终端时刻很接近, 即算法的收敛性对估计值 τ_f 的准确度依赖程度较低. 不同 τ_f 下的算法迭代次数和最终解的比较将在 3.4.1 节给出.

3.2.3.2 序列凸化

利用前一小节得到的初始解, 我们可以通过迭代的方式求解 (RSOCP), 直至相邻两次的轨迹比较接近时终止算法. 简要的算法步骤描述如下:

1. 通过求解初始化问题 (RSOCP-INIT) 得到可行解 $\mathbf{Y}^{(0)}$, $i := 0$;
2. 求解 (RSOCP($\mathbf{Y}^{(i)}$)) 问题, 得到解 $\mathbf{Y}^{(i+1)}$;
3. 若 $\|\mathbf{Y}^{(i+1)} - \mathbf{Y}^{(i)}\| < \epsilon$, 终止; 否则令 $i := i + 1$, 继续执行步 2.

在求解 (RSOCP) 问题时, 我们仍采用上一小节给出的引入辅助加速度 \mathbf{a}^R 的方式来避免问题不可行. 在实际应用此序列凸化算法进行计算时, 我们发现算法会出现数值不稳定或迭代效率较低的问题, 其原因有: (1) 推力与速度反方向夹角约束给问题求解带来困难, 将约束 (3.16) 从 (RSOCP) 问题中去掉可改善数值不稳定的问题; (2) 变量 Δt 相较其他变量与真解更接近 (见图 3.2), 仅需少量迭代就已经满足 Δt 的最优性条件. 参考以上两点, 我们给出改进的两阶段序列凸化方法.

3.2.4 两阶段法

针对序列凸化算法存在的问题, 我们设计如下的两阶段序列凸化方法:

- **第一阶段:** 去掉约束 (3.16), 仅在其余约束的基础上进行优化, 直至 $\Delta t^{(i)}$ 满足如下条件

$$|\Delta t^{(i)} - \Delta t^{(i-1)}| < \eta$$

时切换至第二阶段;

- **第二阶段:** 将变量 Δt 的取值固定为 $\Delta \tau := \Delta t^{(i)}$ 并加入约束 (3.16), 继续优化直至满足序列凸化终止条件.

两个阶段的优化问题 (RSOCP-S1) 和 (RSOCP-S2) 分别定义如下:

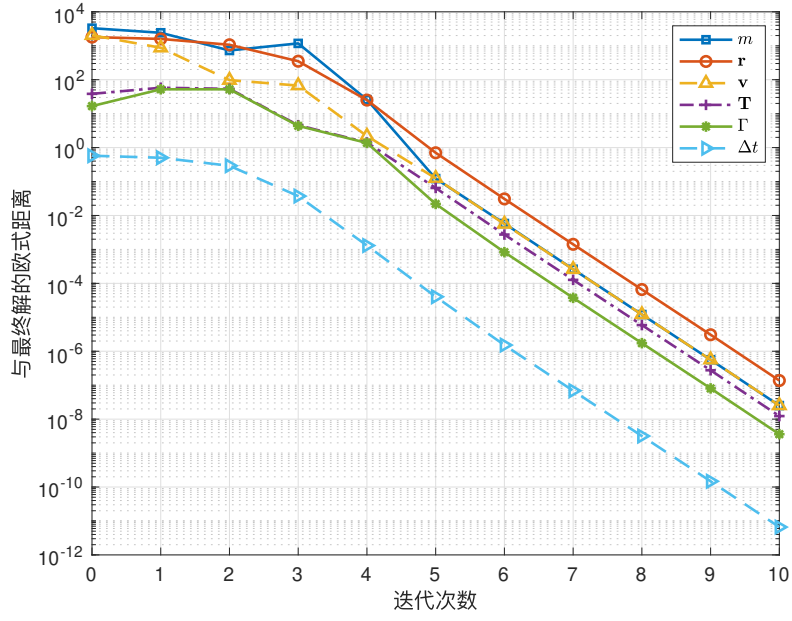


图 3.2 各变量与最终解的距离。

$$\begin{aligned}
 (\text{RSOCP-S1}(\bar{\mathbf{Z}})) \quad & \min_{\mathbf{Z}} \quad -m_N + \beta \|\mathbf{a}^R\| \\
 \text{s. t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{s1}(\bar{\mathbf{Z}}_k) + \mathcal{J}\mathbf{F}_k^{s1}(\bar{\mathbf{Z}}_k)(\mathbf{Z}_k - \bar{\mathbf{Z}}_k), \quad k = 0, 1, \dots, N-1, \\
 & T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \\
 & \|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^T \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\
 & \dot{T}_{\min} \Delta t \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta t, \quad k = 0, 1, \dots, N-1, \\
 & m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|\mathbf{v}_{\text{init}}\|}, \\
 & m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U.
 \end{aligned}$$

$$\begin{aligned}
 (\text{RSOCP-S2}(\bar{\mathbf{Y}})) \quad & \min_{\mathbf{Y}} \quad -m_N \\
 \text{s. t.} \quad & \mathbf{X}_{k+1} = \mathbf{X}_k + \mathbf{F}_k^{s2}(\bar{\mathbf{Y}}_k) + \mathcal{J}\mathbf{F}_k^{s2}(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k), \quad k = 0, 1, \dots, N-1, \\
 & T_{\min} \leq \Gamma_k \leq T_{\max}, \quad k = 0, 1, \dots, N, \\
 & \|\mathbf{r}_k\| \cos \theta_1 \leq \mathbf{e}_U^T \mathbf{r}_k, \quad \|\mathbf{T}_k\| \leq \Gamma_k, \quad k = 1, 2, \dots, N-1, \\
 & G_k(\bar{\mathbf{Y}}_k) + \mathcal{J}G_k(\bar{\mathbf{Y}}_k)(\mathbf{Y}_k - \bar{\mathbf{Y}}_k) \leq 0, \quad k = 1, 2, \dots, N-1, \\
 & \dot{T}_{\min} \Delta \tau \leq \Gamma_{k+1} - \Gamma_k \leq \dot{T}_{\max} \Delta \tau, \quad k = 0, 1, \dots, N-1, \\
 & m_0 = m_{\text{init}}, \quad \mathbf{r}_0 = \mathbf{r}_{\text{init}}, \quad \mathbf{v}_0 = \mathbf{v}_{\text{init}}, \quad \mathbf{T}_0 = \Gamma_0 \frac{-\mathbf{v}_{\text{init}}}{\|\mathbf{v}_{\text{init}}\|}, \\
 & m_N \geq m_{\text{dry}}, \quad \mathbf{r}_N = \mathbf{0}, \quad \mathbf{v}_N = \mathbf{0}, \quad \mathbf{T}_N = \Gamma_N \mathbf{e}_U.
 \end{aligned}$$

其中 $\mathbf{F}_k^{s1}, \mathbf{F}_k^{s2}$ 的定义为:

$$\mathbf{F}_k^{s1}(\mathbf{Z}_k) = \frac{\Delta t}{2} \begin{pmatrix} -\alpha (\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[\frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}} \right] + \mathbf{a}_k^R + \mathbf{a}_{k+1}^R + 2\mathbf{g} \end{pmatrix},$$

$$\mathbf{F}_k^{s2}(\mathbf{Y}_k) = \frac{\Delta \tau}{2} \begin{pmatrix} -\alpha (\Gamma_k + \Gamma_{k+1}) \\ \mathbf{v}_k + \mathbf{v}_{k+1} \\ \left[\frac{\mathbf{T}_k + \mathbf{D}_k}{m_k} + \frac{\mathbf{T}_{k+1} + \mathbf{D}_{k+1}}{m_{k+1}} \right] + 2\mathbf{g} \end{pmatrix}.$$

3.3 非线性二阶锥规划模型与可行方向法

3.2 节给出的序列凸化方法在工程上是一个十分高效的算法, 但已有的工作鲜有对一般动力下降制导问题的序列凸化算法可行性、全局收敛性和最终解的性质的分析 [48]. 注意到 (ROC) 问题在经过 3.2.1.1 节的无损凸化和 3.2.1.2 节的时间离散化后转化为了含二阶锥约束和一般非线性约束的非线性规划问题. 其可抽象为如下的非线性二阶锥规划模型

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & f(x) \\ \text{s. t.} \quad & h(x) = 0, \\ & g(x) \in \mathcal{K}. \end{aligned} \tag{NSOCP}$$

其中

- $f : \mathbb{R}^n \rightarrow \mathbb{R}$;
- $h = (h_1, h_2, \dots, h_p)^\top, h : \mathbb{R}^n \rightarrow \mathbb{R}^p$;
- $g = (g_1, g_2, \dots, g_J)^\top, g_j : \mathbb{R}^n \rightarrow \mathbb{R}^{m_j}, m = \sum_{j=1}^J m_j$;
- f, h, g_j 连续可微;
- $\mathcal{K} = \mathcal{K}^{m_1} \times \mathcal{K}^{m_2} \times \dots \times \mathcal{K}^{m_J}$
- $m_j = 1, \mathcal{K}^1 = \mathbb{R}_+$;
- $m_j \geq 2, \mathcal{K}^{m_j} = \{(x_1, \bar{x}) \in \mathbb{R} \times \mathbb{R}^{m_j-1} \mid x_1 \geq \|\bar{x}\|\}$.

若 Ω 是优化问题的可行域. 向量 $d \in \mathbb{R}^n \setminus \{0\}$ 称为 $x \in \Omega$ 处的可行方向, 当且仅当存在 $\theta > 0$ 使得任意 $t \in [0, \theta]$ 都有 $x + td \in \Omega$. Herskovits [93] 在 1998 年提出应用可行方向法求解一般的非线性规划问题. 可行方向法就是从问题可行域的一个内点出发, 通过牛顿法求解 KKT 系统找到目标函数的下降方向和可行

方向, 在这个方向上计算步长使得迭代点保持在可行域内部. 此方法的优势在于每次迭代仅需求解两个有相同系数矩阵的线性系统, 效率较高, 且当问题仅含不等式约束时, 每个迭代点都是优化问题的可行点, 可根据应用需求提前终止迭代. Canelas 等人 [82] 在 2019 年将可行方向法推广到仅含非线性锥约束的 NSOCP 问题. 本论文在 [82, 93] 的基础上研究利用可行方向法求解更一般的 (NSOCP) 问题.

3.3.1 下降方向与可行方向

记 (NSOCP) 的可行域 Ω_0 为

$$\Omega_0 := \{x \in \mathbb{R}^n \mid h(x) = 0, g(x) \in \mathcal{K}\}.$$

定义扩充可行域 Ω 为

$$\Omega := \{x \in \mathbb{R}^n \mid g(x) \in \mathcal{K}\}.$$

在继续讨论前给出以下必要假设.

假设 3.1.

1. 存在 $a \in \mathbb{R}$ 使得 $\Omega_a = \{x \in \Omega \mid f(x) \leq a\}$ 是紧的且 $\text{int}(\Omega_a) \neq \emptyset$
2. 每个 $x \in \text{int}(\Omega_a)$ 满足 $g(x) \in \text{int}(\mathcal{K})$;
3. 对 $j = 1, \dots, J$, 映射 $\nabla f : \mathbb{R}^n \rightarrow \mathbb{R}^n, Jh : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times p}, Jg_j : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m_j}$ 是 *Lipschitz* 连续的;
4. 每个 $x \in \Omega_a$ 是非退化的 [94], 即

$$Jg(x)(\mathbb{R}^n) + \text{Lin}(\mathcal{T}_{\mathcal{K}}(g(x))) = \mathbb{R}^{m_1} \times \dots \times \mathbb{R}^{m_J},$$

且 $Jh_i(x), i = 1, \dots, p$ 线性独立. 其中 $\mathcal{T}_{\mathcal{K}}$ 表示 \mathcal{K} 的切锥, $\text{Lin}(C)$ 表示 C 中的最大线性空间.

引入 ℓ_1 势函数

$$\phi_c(x) := f(x) + \sum_{i=1}^p c_i |h_i(x)|,$$

其中 $c_i > 0, i = 1, \dots, p$. 令

$$\text{Arw}(g(x)) := \text{diag}(\text{Arw}(g_j(x))), \quad \text{Arw}(y) := \text{diag}(\text{Arw}(y_j))$$

定义 $g(x)$ 与 y 算子可交换当且仅当 $g_j(x) \in \mathcal{K}^{m_j}$ 与 $y_j \in \mathcal{K}^{m_j}$ 算子可交换, $j = 1, \dots, J$. 利用牛顿法求解 KKT 系统 (1.2)–(1.4)

$$\begin{pmatrix} B & -Jg(x)^\top & -Jh(x)^\top \\ \text{Arw}(y)Jg(x) & \text{Arw}(g(x)) & 0 \\ Jh(x) & 0 & 0 \end{pmatrix} \begin{pmatrix} x_a - x \\ y_a - y \\ z_a - z \end{pmatrix} = - \begin{pmatrix} \nabla_x L(x, y, z) \\ \text{Arw}(g(x))y \\ h(x) \end{pmatrix}.$$

令 $d_a := x_a - x$,

$$\begin{pmatrix} B & -Jg(x)^\top & -Jh(x)^\top \\ \text{Arw}(y)Jg(x) & \text{Arw}(g(x)) & 0 \\ Jh(x) & 0 & 0 \end{pmatrix} \begin{pmatrix} d_a \\ y_a \\ z_a \end{pmatrix} = \begin{pmatrix} -\nabla f(x) \\ 0 \\ -h(x) \end{pmatrix}. \quad (\text{LEQN1})$$

引理 3.2. 令 $x \in \Omega$, $y \in \text{int}(\mathcal{K})$ 使得 $g(x)$ 与 y 算子可交换. 若对任意 $B \in S_{++}^n$, 满足 $c_i > |z_{ai}|$, $i = 1, \dots, p$, 则 (LEQN1) 的解 d_a 是 ϕ_c 的下降方向.

证明. 由于 d_a 是 (LEQN1) 的解, 因此 ϕ_c 的方向导数满足

$$\begin{aligned} D(\phi_c(x); d_a) &= \lim_{\alpha \rightarrow 0_+} \frac{\phi_c(x + \alpha d_a) - \phi_c(x)}{\alpha} \\ &= \nabla f(x)^\top d_a + \sum_{i=1}^p c_i \lim_{\alpha \rightarrow 0_+} \frac{|h_i(x + \alpha d_a)| - |h_i(x)|}{\alpha} \\ &= \nabla f(x)^\top d_a + \sum_{i=1}^p c_i \lim_{\alpha \rightarrow 0_+} \frac{|h_i(x) + \alpha Jh_i(x)d_a| - |h_i(x)|}{\alpha} \\ &= \nabla f(x)^\top d_a + \sum_{i=1}^p c_i \lim_{\alpha \rightarrow 0_+} \frac{|(1 - \alpha)h_i(x)| - |h_i(x)|}{\alpha} \\ &= \nabla f(x)^\top d_a - \sum_{i=1}^p c_i |h_i(x)|. \end{aligned}$$

对 (LEQN1) 第一式左乘 d_a^\top , 得到

$$d_a^\top B d_a - d_a^\top Jg(x)^\top y_a - d_a^\top Jh(x)^\top z_a = -d_a^\top \nabla f(x).$$

根据命题 1.1, 有

$$\text{Arw}(y) \in S_{++}^n, \quad \text{Arw}(g(x))\text{Arw}(y) = \text{Arw}(y)\text{Arw}(g(x)).$$

又因为

$$\begin{aligned} [(\text{Arw}(y))^{-1} \text{Arw}(g(x))]^\top &= \text{Arw}(g(x))(\text{Arw}(y))^{-1} \\ &= (\text{Arw}(y))^{-1} \text{Arw}(y) \text{Arw}(g(x))(\text{Arw}(y))^{-1} \\ &= (\text{Arw}(y))^{-1} \text{Arw}(g(x)). \end{aligned}$$

所以 $(\text{Arw}(y))^{-1} \text{Arw}(g(x)) \in S_+^n$ [95]. 因此有

$$-y_a^\top Jg(x)d_a = y_a^\top (\text{Arw}(y))^{-1} \text{Arw}(g(x))y_a \geq 0.$$

于是

$$\begin{aligned} D(\phi_c(x); d_a) &= -d_a^\top B d_a + d_a^\top Jg(x)^\top y_a + d_a^\top Jh(x)^\top z_a - \sum_{i=1}^p c_i |h_i(x)| \\ &\leq \sum_{i=1}^p (|z_{ai}| - c_i) |h_i(x)| \\ &< 0. \end{aligned}$$

证毕. □

当 $d_a = 0$ 时, 由 (LEQN1) 可知 x 是稳定点; 若同时有 $y_a \in \mathcal{K}$, 则 x 是 KKT 点. 当 $d_a \neq 0$ 时, 存在 $\tau > 0$ 使得对任意 $t \in (0, \tau)$ 有

$$\phi_c(x + t d_a) < \phi_c(x).$$

然而下面的命题指出, d_a 未必总是 x 处的可行方向.

命题 3.3 ([82]). 令 $x \in \Omega$, $d \in \mathbb{R}^n$. 假定 $h(x) \leq 0$, $g(x) \in \text{bd}(\mathcal{K})$. 令 $\lambda_1(g(x))u_1 + \lambda_2(g(x))u_2$ 为 $g(x)$ 的谱分解. 若

- 当 $g(x) = 0$ 时, $\nabla g(x)^\top d \in \text{int}(\mathcal{K})$, 或者;
- 当 $g(x) \in \text{bd}(\mathcal{K}) \setminus \{0\}$ 时, $u_1^\top \nabla g(x)^\top d > 0$,

则 d 是 x 处的扩充可行方向.

在 (LEQN1) 的基础上考虑如下的线性系统

$$\begin{pmatrix} B & -Jg(x)^\top & -Jh(x)^\top \\ \text{Arw}(y)Jg(x) & \text{Arw}(g(x)) & 0 \\ Jh(x) & 0 & 0 \end{pmatrix} \begin{pmatrix} d \\ \hat{y} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} -\nabla f(x) \\ \rho y \\ -h(x) \end{pmatrix} \quad (3.20)$$

其中 $\rho, \xi_j, \eta_i > 0, i = 1, \dots, p, j = 1, \dots, m$.

引理 3.4 ([82]). 令 $x \in \Omega$, $y \in \text{int}(\mathcal{K})$ 使得 $g(x)$ 和 y 算子可交换. 那么线性系统 (3.20) 的解 d 是 x 处的可行方向.

为使 d 同时是 x 处势函数的下降方向, 考虑如下的线性系统

$$\begin{pmatrix} B & -Jg(x)^\top & -Jh(x)^\top \\ \text{Arw}(y)Jg(x) & \text{Arw}(g(x)) & 0 \\ Jh(x) & 0 & 0 \end{pmatrix} \begin{pmatrix} d_b \\ y_b \\ z_b \end{pmatrix} = \begin{pmatrix} 0 \\ y \\ 0 \end{pmatrix} \quad (\text{LEQN2})$$

则有 $d = d_a + \rho d_b$. 虽然 d_b 未必是 ϕ_c 的下降方向, 但下面的引理表明合适地选取 ρ 可使 d 成为 ϕ_c 的下降方向.

引理 3.5. 令 $x \in \Omega$, $\eta \in \mathbb{R}_{++}^p$, d_a 和 d_b 分别是 (LEQN1) 和 (LEQN2) 的解, d_a 是 ϕ_c 的下降方向. 若 $\rho > 0$ 满足

$$\rho \leq (\xi - 1) \frac{\nabla f(x)^\top d_a - \sum_{i=1}^p c_i |h_i(x)|}{\nabla f(x)^\top d_b}, \text{ 当 } \nabla f(x)^\top d_b > 0 \text{ 时.}$$

其中 $\xi \in (0, 1)$. 则 $d = d_a + \rho d_b$ 是 ϕ_c 在 x 处的下降方向.

证明. 由于

$$\begin{aligned} D(\phi_c(x); d) &= \nabla f(x)^\top d + \sum_{i=1}^p c_i \lim_{\alpha \rightarrow 0_+} \frac{|h_i(x + \alpha d)| - |h_i(x)|}{\alpha} \\ &= \nabla f(x)^\top d + \sum_{i=1}^p c_i \lim_{\alpha \rightarrow 0_+} \frac{|(1 - \alpha)h_i(x) - \alpha \rho \eta_i| - |h_i(x)|}{\alpha} \\ &\leq D(\phi_c(x); d_a) + \rho \nabla f(x)^\top d_b \\ &\leq \xi D(\phi_c(x); d_a) \\ &< 0. \end{aligned}$$

引理得证. □

3.3.2 可行方向内点算法

通过线搜索方法沿着 d 寻找下一个迭代点, 使得新的迭代点仍保持在可行域内, 且使势函数下降. 将这个过程具体描述为以下算法 5.

下面的引理表明线搜索中的条件 (3.23) 总可以被满足.

引理 3.6 ([82]). 令 $x \in \Omega$, $y \in \text{int}(\mathcal{K})$ 使得 $g(x)$ 和 y 算子可交换. 令 $\{u_1^j, u_2^j\}$ 是 $g_j(x)$ 的 Jordan 基底. 若 (d, \hat{y}, \hat{z}) 是 (3.20) 的解且有 $\hat{y}_j^\top u_i^j \leq 0$, 那么 d 是 $-\lambda_i(g_j(x))$ 的下降方向.

算法 5 (NSOCP) 的可行方向内点算法 (FDIPA)

- 1: 选取参数 $c \in \mathbb{R}_{++}^p$, $\psi > 0$, $\epsilon > 0$, $\xi \in (0, 1)$, $\mu > 0$, $\lambda_m > 0$, $v \in (0, 1)$. 令 $x^{(0)} \in \text{int}(\Omega)$, $y^{(0)} \in \text{int}(\mathcal{K})$ 使得 $g(x^{(0)})$ 和 $y^{(0)}$ 算子可交换. 令 $B^{(0)} \in S_{++}^n$, $k := 0$;
- 2: 求解 (LEQN1), 其中 $(x, y) := (x^{(k)}, y^{(k)})$, 得到 $(d_a^{(k)}, y_a^{(k)}, z_a^{(k)}) := (d_a, y_a, z_a)$; 若 $\|d_a^{(k)}\| < \epsilon$, 算法终止;
- 3: 求解 (LEQN2), 其中 $(x, y) := (x^{(k)}, y^{(k)})$, 得到 $(d_b^{(k)}, y_b^{(k)}, z_b^{(k)}) := (d_b, y_b, z_b)$;
- 4: 若 $c_i < 1.2 |z_{ai}^{(k)}|$, 更新 $c_i := 1.5 |z_{ai}^{(k)}|$, $i = 1, \dots, p$;
- 5: 令

$$\rho = \begin{cases} \min \left\{ \|d_a^{(k)}\|, \psi \frac{\|d_a^{(k)}\|}{\|d_b^{(k)}\|} \right\}, & \text{若 } \nabla f(x^{(k)})^\top d_b^{(k)} \leq 0, \\ \min \left\{ \|d_a^{(k)}\|, \psi \frac{\|d_a^{(k)}\|}{\|d_b^{(k)}\|}, (\xi - 1) \frac{\nabla f(x^{(k)})^\top d_a^{(k)} - \sum_{i=1}^p c_i |h_i(x^{(k)})|}{\nabla f(x^{(k)})^\top d_b^{(k)}} \right\}, & \text{否则.} \end{cases}$$

- 6: 更新 $d^{(k)} := d_a^{(k)} + \rho d_b^{(k)}$, $\hat{y}^{(k)} := y_a^{(k)} + \rho y_b^{(k)}$;
- 7: 寻找步长 $t^{(k)} \in 1, v, v^2, \dots$ 满足:

$$\phi_c(x^{(k)} + t^{(k)} d^{(k)}) \leq \phi_c(x^{(k)}) + t^{(k)} \mu D(\phi_c(x^{(k)}); d^{(k)}), \quad (3.21)$$

$$g(x^{(k)} + t^{(k)} d^{(k)}) \in \text{int}(\mathcal{K}), \quad (3.22)$$

$$\lambda_i^j(g_j(x^{(k)} + t^{(k)} d^{(k)})) \geq \lambda_i^j(g_j(x^{(k)})), \text{ 若 } \hat{y}_j^{(k)\top} u_i^j(g_j(x^{(k)})) < 0 \text{ 且 } \lambda_i^j(g_j(x^{(k)})) < \lambda_m. \quad (3.23)$$

- 8: 更新 $x^{(k+1)} := x^{(k)} + t^{(k)} d^{(k)}$. 令 $y^{(k+1)} \in \text{int}(\mathcal{K})$ 使得 $g(x^{(k+1)})$ 与 $y^{(k+1)}$ 算子可交换. 更新 $B^{(k+1)} \in S_{++}^n$, $k := k + 1$, 转第 2 步.
-

3.3.3 收敛性分析

为方便分析, 本节中的锥 $\mathcal{K} = \mathcal{K}^m$, 即锥约束仅含一个二阶锥. 关于非负卦限锥 (即非负约束) 的分析已在文献 [93] 中给出. 在分析算法 5 的收敛性之前我们先给出以下额外假设条件.

假设 3.7.

1. 存在 $\sigma_1, \sigma_2 > 0$ 使得对任意 $d \in \mathbb{R}^n$, 都有

$$\sigma_1 \|d\|^2 \leq d^\top B d \leq \sigma_2 \|d\|^2;$$

2. 存在 $c^I, c^S > 0$ 使得对任意 $j = 1, \dots, J$, 都有

$$y_j - c^I e \in \mathcal{K}^{m_j}, \quad c^S e - y_j \in \mathcal{K}^{m_j}.$$

引理 3.8. 令 $B \in S_{++}^n$, $x \in \Omega$, $y \in \text{int}(\mathcal{K})$ 使得 $g(x)$ 和 y 算子可交换. 则 (LEQN1) 与 (LEQN2) 均有唯一解.

证明. 仅需证明如下齐次方程组

$$\begin{pmatrix} B & -Jg(x)^\top & -Jh(x)^\top \\ \text{Arw}(y)Jg(x) & \text{Arw}(g(x)) & 0 \\ Jh(x) & 0 & 0 \end{pmatrix} \begin{pmatrix} d \\ \hat{y} \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (\text{LEQN2})$$

有唯一解 $(d, \hat{y}, z) = (0, 0, 0)$. 由于

$$d^\top B d - d^\top Jg(x)^\top \hat{y} - d^\top Jh(x)^\top z = d^\top B d - d^\top Jg(x)^\top \hat{y} = 0,$$

$$\text{Arw}(y)Jg(x)d + \text{Arw}(g(x))\hat{y} = 0,$$

根据假设 3.1 和 [82] 的引理 3.4 的证明有 $(d, \hat{y}) = (0, 0)$. 又因为 $Jh(x)^\top$ 列满秩且 $Jh(x)^\top z = 0$, 有 $z = 0$. 证毕. \square

引理 3.9. 存在 $\tau > 0$ 使得 FDIPA 中的线搜索条件(3.21)–(3.23)可被任意 $t \in [0, \tau]$ 满足.

证明. 由引理 3.6 及 [82] 的引理 3.7 可知存在 $\tau_1 > 0$ 使得任意 $t \in [0, \tau_1]$ 满足条件(3.22)–(3.23). 对于条件(3.21), 记 $\Phi := \phi_c(x + td) - \phi_c(x) - t\mu D(\phi_c(x); d)$. 显然

$t = 0$ 时 $\Phi \leq 0$. 当 $t > 0$ 时, 由泰勒定理, 存在 $k_f > 0$ 使得

$$\begin{aligned}\Phi &= f(x + td) - f(x) + \sum_{i=1}^p c_i (|h_i(x + td)| - |h_i(x)|) - t\mu D(\phi_c(x); d) \\ &\leq t^2 \|d\|^2 k_f + t \left(\nabla f(x)^\top d + \sum_{i=1}^p c_i \frac{|h_i(x + td)| - |h_i(x)|}{t} - \mu D(\phi_c(x); d) \right) \\ &:= t^2 \|d\|^2 k_f + t\Psi\end{aligned}$$

再由引理 3.2 和 3.5 的证明, 有

$$D(\phi_c(x); d) \leq \xi D(\phi_c(x); d_a) \leq -\xi d_a^\top B d_a \leq -\xi \sigma_1 \|d_a\|^2.$$

因此当 $d_a \neq 0$ 时,

$$\lim_{t \rightarrow 0_+} \Psi = (1 - \mu) D(\phi_c(x); d) \leq -(1 - \mu) \xi \sigma_1 \|d_a\|^2 < 0.$$

即存在 $\tau_2 > 0$ 使得任意 $t \in (0, \tau_2]$ 都有 $\Psi < 0$. 令 $\tau_3 = \min\{\tau_2, -\Psi/\|d\|^2 k_f\}$. 当 $t \in (0, \tau_3]$ 时,

$$\Phi \leq t(\Psi + t\|d\|^2 k_f) \leq 0.$$

令 $\tau = \min\{\tau_1, \tau_3\}$. 得证. □

定理 3.10. 令 $x^{(0)} \in \text{int}(\Omega)$. 假设 $FDIPA$ 产生的 $\{x^{(k)}\}_{k \in \mathbb{N}}$ 有界且对任意的 k 有 $c_i \geq |z_{ai}^{(k)}|$, $i = 1, \dots, p$. 那么 $\{x^{(k)}\}_{k \in \mathbb{N}}$ 的任意聚点都是 (NSOCP) 的 KKT 点.

证明. 由假设 3.7 有 $B^{(k)}$ 和 $y^{(k)}$ 有界. 由引理 3.8 有 $d_a^{(k)}, d_b^{(k)}, y_a^{(k)}, y_b^{(k)}, z_a^{(k)}, z_b^{(k)}$ 有界. 又因为 $t^{(k)} \in [0, 1]$, 我们有 $X^{(k)} := (x^{(k)}, B^{(k)}, y^{(k)}, d_a^{(k)}, d_b^{(k)}, y_a^{(k)}, y_b^{(k)}, z_a^{(k)}, z_b^{(k)}, t^{(k)})$ 有界. 取其收敛子列 $\{X^{(k_i)}\}_{i \in \mathbb{N}}$ 并记其聚点为 $X^* := (x^*, B^*, y^*, d_a^*, d_b^*, y_a^*, y_b^*, z_a^*, z_b^*, t^*)$. 假设 x^* 不是 (NSOCP) 的稳定点. 若 $d_a^* = 0$, 由 (LEQN1) 的第一式可知与假设矛盾. 因此存在 $\zeta > 0$ 使得 $\|d_a^*\| > \zeta > 0$. 又由引理 3.9 可知 $t^* > 0$. 因此有

$$\begin{aligned}\phi_c(x^*) &= \lim_{k \rightarrow \infty} \phi_c(x^{(k+1)}) \leq \lim_{k \rightarrow \infty} \phi_c(x^{(k)}) + t^{(k)} \eta D(\phi_c(x^{(k)}); d) \\ &\leq \phi_c(x^*) - t^* \eta \xi \sigma_1 \zeta^2.\end{aligned}$$

依然矛盾. 所以 x^* 是 (NSOCP) 的稳定点.

由于 (LEQN1) 的第三式说明 $h(x^*) = 0$, 为说明 x^* 是 (NSOCP) 的 KKT 点仅需证明 $y_a^* \in \mathcal{K}$. 假设 $g(x^*)$ 的谱分解为 $\lambda_1(g(x^*))u_1^* + \lambda_2(g(x^*))u_2^*$. 由 $g(x^*) \in \mathcal{K}$

有 $\lambda_l(g(x^*)) \geq 0, l = 1, 2$. 对于 $g(x^*) \in \text{int}(\mathcal{K})$ 的情形, 容易通过 $\text{Arw}(g(x^*))y_a^* = -\text{Arw}(y^*)\mathcal{J}(g(x^*))d_a^* = 0$ 得到 $y_a^* = 0 \in \mathcal{K}$. 下面讨论 $g(x^*) = 0$ 和 $g(x^*) \in \text{bd}(\mathcal{K}) \setminus \{0\}$ 两种情形.

若 $\lambda_l(g(x^*)) = 0, l = 1, 2$, 由于 $\lambda_l(g(x^{(k_i)})) > 0$ 且 $\lim_{i \rightarrow \infty} \lambda_l(g(x^{(k_i)})) = 0$, 可不失一般性假设 $\lambda_l(g(x^{(k_i)})) < \lambda_l(g(x^{(k_{i-1}})))$ 且 $\lambda_l(g(x^{(k_i)})) < \lambda_m$. 记 $g(x^{(k_i)})$ 的谱分解为

$$g(x^{(k_i)}) = \lambda_1(g(x^{(k_i)}))u_1^{(k_i)} + \lambda_2(g(x^{(k_i)}))u_2^{(k_i)}.$$

由于 $u_l^{(k_i)}, l = 1, 2$ 的范数是常数, 可不失一般性假设收敛到 u_l^* . 由算法 5 的线搜索条件 3.23, 有 $\hat{y}^{(k_i)\top} u_l^{(k_i)} \geq 0$. 根据 $\hat{y}^{(k_i)} = y_a^{(k_i)} + \rho^{(k_i)} y_b^{(k_i)}, \rho^{(k_i)} \rightarrow 0$ (因为 $d_a^{(k_i)} \rightarrow 0$) 且 $y_b^{(k_i)}$ 有界, 有 $\hat{y}^{(k_i)} \rightarrow y_a^*$. 对 (LEQN1) 取极限得到 $\text{Arw}(g(x^*))y_a^* = 0$. 由命题 1.1 可知 $g(x^*)$ 与 y_a^* 算子可交换, 即存在 $\alpha_l, l = 1, 2$ 使得 $y_a^* = \alpha_1 u_1^* + \alpha_2 u_2^*$. 对 $\hat{y}^{(k_i)\top} u_l^{(k_i)} \geq 0$ 取极限可知 $\alpha_l \geq 0$. 因此 $y_a^* \in \mathcal{K}$.

若 $\lambda_1(g(x^*)) = 0, \lambda_2(g(x^*)) > 0$, 与之前的证明类似可得到 $\alpha_1 \geq 0$. 再根据 $\text{Arw}(g(x^*))y_a^* = 0$ 得到 $\alpha_2 \lambda_2(g(x^*)) = 0$, 即 $\alpha_2 = 0$. 因此 $y_a^* \in \mathcal{K}$. 证毕. \square

3.3.4 FDIPA 的一些解释

对于初始内点的获取以及线性等式约束的处理, 如果问题不含特殊结构, 我们可以通过求解以下辅助问题得到 FDIPA 的初始内点

$$\begin{aligned} \min_{x, \theta} \quad & \theta \\ \text{s. t.} \quad & h_i(x) = 0, \quad i \in I, \\ & g(x) + \theta E \in \mathcal{K}, \end{aligned} \tag{3.24}$$

其中 $I \subseteq \{1, \dots, p\}$ 是所有线性约束的指标集, $E = (e^{m_1}, \dots, e^{m_J})^\top$, e^{m_j} 是 \mathcal{K}^{m_j} 的单位元. 利用 FDIPA 求解问题 (3.24) 直到 $\theta < 0$. 问题 (3.24) 的初始点很容易得到. 得到 (3.24) 的解后, FDIPA 将一直保持 $h_i(x), i \in I$ 的积极性.

对于乘子 y 的更新, 我们可以在 y_a 的基础上做尽可能小的变动来产生 $y^{(k+1)}$ 使其与 $g(x^{(k+1)})$ 算子可交换: (1) 记 $g(x^{(k+1)})$ 的谱分解为 $\lambda_1 u_1 + \lambda_2 u_2$; (2) 将 y_a 投影到 u_1, u_2 生成的空间, 即 $\tilde{y} = P_{\text{span}(u_1, u_2)}(y_a) = \tilde{y}_1 u_1 + \tilde{y}_2 u_2$; (3) 将 \tilde{y} 投影到 $\{y \mid y - c^I e \in \mathcal{K}, c^S e - y \in \mathcal{K}\}$, 即

$$y^{(k+1)} = y_1^{(k+1)} u_1 + y_2^{(k+1)} u_2, \quad y_i^{(k+1)} = \max\{c^I, \min\{c^S, \tilde{y}_i\}\}, \quad i = 1, 2.$$

3.4 数值实验

我们分别用 3.4.1 和 3.4.2 小节展示两阶段序列凸化方法和可行方向内点法的数值实验及其结果. 本节的所有测试均在装有 i7-7567U 的 CPU 和 16GB 内存的笔记本电脑上完成, 代码实现和测试平台为 MATLAB R2019b.

3.4.1 两阶段序列凸化方法

本小节将以一个问题测试算例展示两阶段序列凸化方法求解 (ROC) 问题的数值表现. 我们分别实现 3.2.3.2 和 3.2.4 小节基于序列凸化方法的单阶段算法和两阶段算法, 调用建模工具 CVX [96, 97] 和求解器 ECOS [70] 实现 SOCP 问题的建立和求解. 测试数据 (见表 3.1) 是在文献 [37] 中测试数据的基础上根据本文研究模型 (如推力方向约束) 修改得到.

表 3.1 测试问题数据及参数取值.

| 参数 | 取值 | 单位 | 参数 | 取值 | 单位 |
|------------------|------------------------|-------------------|----------------------------|-------------------------|------------------|
| N | 30 | - | T_{\min} | 100000 | 牛顿 |
| α | 3.399×10^{-4} | 秒/米 | T_{\max} | 250000 | 牛顿 |
| ρ | 1 | 千克/米 ³ | \dot{T}_{\min} | -100000 | 牛顿/秒 |
| S_D | 10 | 米 ² | \dot{T}_{\max} | 100000 | 牛顿/秒 |
| C_D | 1 | - | \mathbf{g} | $[0 \ -9.807 \ 0]^T$ | 米/秒 ² |
| m_{dry} | 10000 | 千克 | m_{init} | 15000 | 千克 |
| θ_1 | 80 | 度 | \mathbf{r}_{init} | $[-600 \ 2000 \ 600]^T$ | 米 |
| θ_2 | 15 | 度 | \mathbf{v}_{init} | $[80 \ -150 \ -80]^T$ | 米/秒 |

我们以表 3.1 中的数据建立 (ROC) 问题, 并利用 3.2.4 节的两阶段算法进行求解. 辅助加速度的罚参数 β 取 5×10^5 . 控制第一、第二阶段终止条件的参数 η 和 ϵ 分别取 10^{-3} 和 10^{-6} . 在没有对飞行轨迹进行分析预测的情况下, 我们粗略地选取 $\tau_f = 5$. 两阶段算法在 11 次迭代后终止. 终端质量和终端时刻分别为 13557.8 和 22.25. 所得飞行轨迹, 质量、推力曲线和推力大小与变量 Γ 的关系如图 3.3–图 3.5 所示. 可以看到, 两阶段算法求解非凸最优控制问题 (ROC) 仅需求解 12 个 (包括初始化 1 个和两阶段共 11 个) SOCP 问题, 算法效率较高; 算法得到的结果满足推力上下限和推力方向等约束, 飞行轨迹较为平滑; $\|\mathbf{T}_k\|$ 与 Γ_k 在最终解处的距离不超过 7×10^{-8} , 所用无损凸化技巧确实达到“无损”的效果; 最终解的终端时刻与预测值相差 17.25 秒, 算法在如此粗糙的预测下仍能正常迭代并收敛.

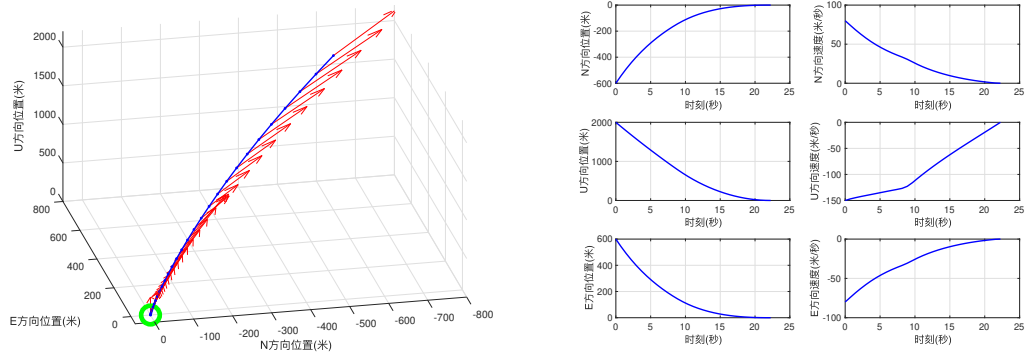


图 3.3 3D 位置轨迹 (左) 和 2D 位置速度轨迹 (右). 左图中蓝色曲线表示位置轨迹, 红色箭头表示加速度, 绿色圆形表示着陆点.

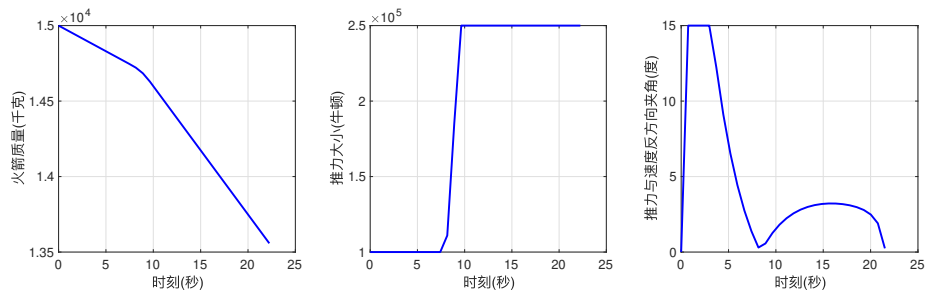


图 3.4 质量 (左)、推力大小 (中) 和推力与速度反方向夹角曲线 (右).

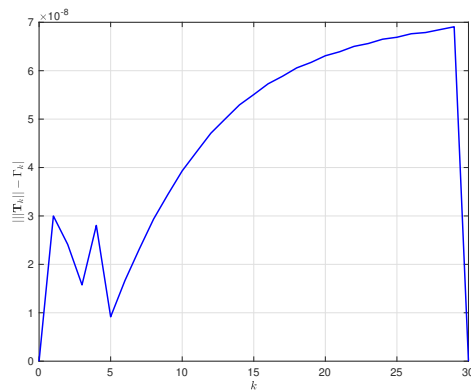


图 3.5 最终解处实际推力大小 $\|T\|$ 与推力上界 Γ 的关系.

为了进一步验证算法对 τ_f 的弱依赖性, 同时展示两阶段方法相较于 3.2.3.2 节的单阶段序列凸化方法更加稳定和高效的特点, 我们选取不同的 τ_f 进行多次测试. 图 3.6 展示了在 10 个不同的猜测值 τ_f 下, 算法总可以使终端时刻 t_f 快速收敛 (6 次迭代之内), 且收敛到同一个值. 表 3.2 比较了单阶段方法和两阶段方法的迭代次数. 单阶段法在 τ_f 取 10、15 和 25 时出现了数值不稳定性, 导致算法迭代次数异常增加. 相比之下, 两阶段法在 10 次测试的迭代次数却可以控制在 11–13 次. 不仅在数值表现上更加稳定, 而且在单阶段法表现正常的测试中两阶段法也以更少的迭代次数胜出. 图 3.7 中给出了与约束 (3.16) 相关的物理量曲线. 可以看到, 第一阶段所得的轨迹在经过第二阶段的迭代后满足了约束 (3.16), 但此轨迹与最终解差别很小. 由此可以猜测, 约束 (3.16) 比其他线性化约束对初始轨迹的质量更敏感. 两阶段方法中的第一阶段提供了很好的轨迹, 使得非凸约束 (3.7) 的序列凸化更加稳定. 因此两阶段方法具有比单阶段方法更好的稳定性.

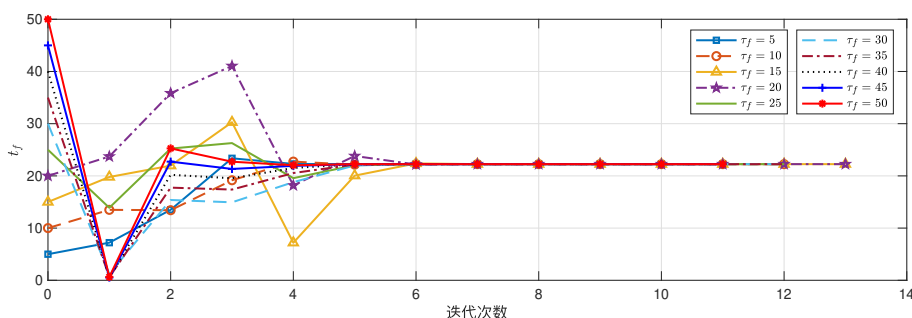


图 3.6 不同终端时刻猜测值 τ_f 下的 t_f 收敛结果.

表 3.2 单阶段法和两阶段法的迭代次数对比.

| τ_f | 5 | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50 |
|----------|----|----|----|----|-----|----|----|----|----|----|
| 单阶段法 | 11 | 30 | 88 | 13 | 225 | 17 | 19 | 19 | 15 | 14 |
| 两阶段法 | 11 | 12 | 13 | 13 | 12 | 12 | 12 | 11 | 11 | 11 |

3.4.2 可行方向内点法

本小节我们通过两个数值实验展示算法 5 的表现. 算法 5 的参数选取为

$$c = 1, \xi = 0.7, \psi = 0.5, \mu = 0.3, \nu = 0.7, B^{(0)} = I.$$

我们分别在对比了如下几种 NLP 和 NSOCP 求解算法: (1) NLP 的内点法软件包 IPOPT [98]; (2) NSOCP 的 SQP 型方法 [73] (记为 SQP-type); (3) NSOCP 的无罚方

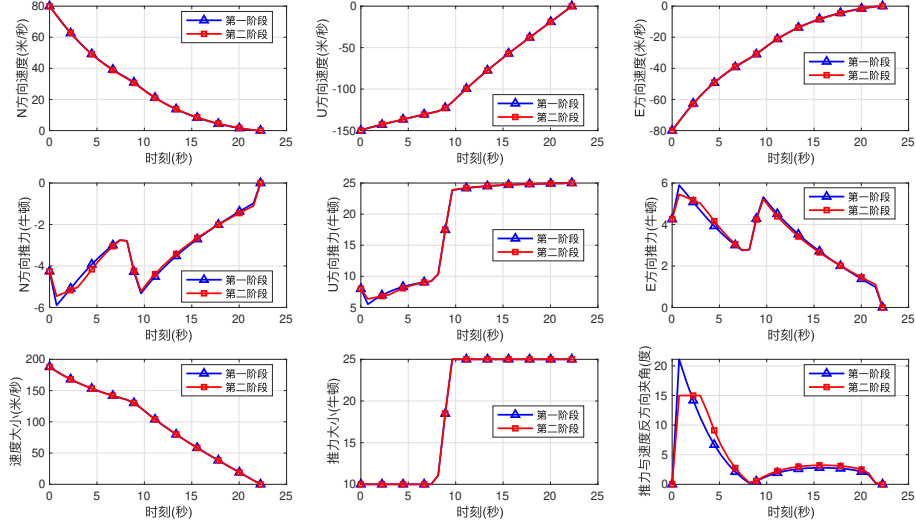


图 3.7 各阶段最终解中的速度 (第一行)、推力 (第二行)、速度大小 (第三行左)、推力大小 (第三行中) 和推力与速度反方向夹角曲线 (第三行右).

法 [83] (记为 PF); (4) 动力下降制导问题的定制化序列凸化方法 (记为 SC) (5) 算法 5 (记为 FDIPA). 其中 SQP-type 和 PF 方法中 LSOCP 的求解是调用 ECOS [70] 软件包. ECOS 是开源的 LSOCP 求解器, 在求解小规模问题 (变量少于 1000) 时与商业求解器 MOSEK 有相近的效率. 在本节的所有数值实验中, IPOPT 及文献中的方法均采用默认参数设置. SQP-type 和 SC 的终止条件设置为 $\|\Delta x\| < 10^{-6}$. FDIPA 的终止条件为 $\|d_a\| < 10^{-6}$. PF 的终止条件与 [83] 相同, 且保留最大迭代次数上限为 200 这一条件. SQP-type、PF 和 FDIPA 中的拟牛顿矩阵采用带 Powell 修正的 BFGS 方法 [99]. 特别地在 FDIPA 中每 n 次迭代重置拟牛顿矩阵为 $B^{(k)} = I$. 为对比各个方法得到的解的可行性, 定义 feasH 为 $\|h(x^*)\|_\infty$, feasG 为 $|\min\{0, \min_{1 \leq j \leq J} \{\lambda_1(g_j(x^*))\}\}|$, 其中当 $m_j = 1$ 时定义 $\lambda_1(g_j(x^*)) = g_j(x^*)$.

首先我们利用随机产生的模型问题进行测试. 将文献 [73] 中的部分测试问题拓展为

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & x^\top C x + \sum_{i=1}^n p_i x_i^4 + q_i x_i \\ \text{s. t.} \quad & x^\top Q_i Q_i^\top x + q_i^\top x = 0, i = 1, \dots, n/2, \\ & Ax + b \in \mathcal{K}, \end{aligned} \quad (3.25)$$

其中 $Z \in [0, 1]^{n \times n}$, $A \in [0, 2]^{n \times n}$, $Q \in [0, 1]^{n \times (n/2)}$, $q \in [0, 1]^{n \times (n/2)}$, $d_i \in [0, 1]^n$, $f_i \in [-1, 1]^n$ 为随机生成的矩阵和向量. $C = Z^\top Z \in \mathcal{S}_{++}^n$, $\mathcal{K} = \mathcal{K}^{m_1} \times \dots \times \mathcal{K}^{m_J}$,

$b = (b_1, \dots, b_J)$, $b_j = (1, \mathbf{0})$. 通过选取不同的 n 和 \mathcal{K} , 我们生成了如表 3.3 所示的 5 种测试问题. A–E 每种测试问题各随机生成 3 个算例. 我们在 (3.25) 上测试了除 SC 外的 4 种方法. SQP-type、PF 和 FDIPA 的初始点设置为 $\mathbf{0} \in \mathbb{R}^n$. IPOPT 的初始点选取为 $10^{-6} \in \mathbb{R}^n$ 以避免数值问题.

表 3.3 测试问题规模

| 算例类型 | n | m_j |
|------|-----|-------------------------|
| A | 50 | 5,5,20,20 |
| B | 100 | 10,10,20,20,20,20 |
| C | 100 | 4,4,...,4 |
| D | 200 | 4,4,...,4 |
| E | 340 | 40,40,50,50,50,40,30,40 |

表 3.4 测试问题 (3.25) 的求解时间和迭代次数对比

| 算例 | IPOPT | | SQP-type | | PF | | FDIPA | |
|----|--------|-----|----------|-----------|--------|-----------|--------|-----|
| | 时间 (秒) | 迭代 | 时间 (秒) | 迭代 | 时间 (秒) | 迭代 | 时间 (秒) | 迭代 |
| 1 | 0.25 | 49 | 0.42 | 320(31) | 25.95 | 1099(43) | 0.21 | 37 |
| 2 | 0.17 | 38 | 0.31 | 230(22) | 18.43 | 837(31) | 0.16 | 27 |
| 3 | 0.20 | 57 | 0.44 | 297(32) | 25.37 | 982(43) | 0.23 | 34 |
| 4 | 0.72 | 60 | 2.77 | 477(50) | 53.22 | 1608(67) | 0.76 | 57 |
| 5 | 0.69 | 61 | 2.35 | 434(47) | 67.16 | 2078(85) | 0.70 | 65 |
| 6 | 0.71 | 66 | 2.14 | 401(42) | 49.15 | 1506(62) | 0.61 | 55 |
| 7 | 1.23 | 110 | 3.23 | 591(56) | 206.14 | 4787(187) | 0.97 | 64 |
| 8 | 1.16 | 92 | 3.48 | 660(60) | 135.42 | 3262(123) | 0.96 | 67 |
| 9 | 1.30 | 106 | 3.52 | 666(62) | 168.55 | 3951(154) | 0.85 | 66 |
| 10 | 12.52 | 179 | 50.09 | 1730(165) | 446.49 | 5280(200) | 4.46 | 125 |
| 11 | 11.56 | 162 | 44.92 | 1341(117) | 471.41 | 5368(200) | 4.56 | 125 |
| 12 | 9.55 | 129 | 40.45 | 1369(121) | 497.55 | 5238(200) | 4.57 | 124 |
| 13 | 27.06 | 86 | 220.31 | 1349(112) | 815.79 | 4347(188) | 10.06 | 153 |
| 14 | 27.01 | 91 | 173.57 | 1150(110) | 696.26 | 3963(173) | 10.38 | 151 |
| 15 | 25.86 | 93 | 172.08 | 1172(103) | 654.99 | 3853(175) | 9.35 | 154 |

表 3.4 和 3.5 分别展示了求解时间、迭代次数和解的可行性对比. 其中表 3.4 同时展示了 SQP-type 和 PF 方法的主迭代次数 (括号内) 和 LSOCp 子问题总迭代次数 (括号外). 可以看到, 在求解效率方面, FDIPA 方法在 12 个算例上的求解

表 3.5 测试问题 (3.25) 的可行性对比

| 算例 | IPOPT | | SQP-type | | PF | | FDIPA | |
|----|----------|----------|----------|-------|----------|-------|----------|-------|
| | feasH | feasG | feasH | feasG | feasH | feasG | feasH | feasG |
| 1 | 1.99E-16 | 9.96E-09 | 5.84E-10 | 0 | 2.74E-11 | 0 | 9.29E-13 | 0 |
| 2 | 1.67E-16 | 9.98E-09 | 1.15E-09 | 0 | 6.52E-11 | 0 | 1.39E-12 | 0 |
| 3 | 2.08E-16 | 9.96E-09 | 1.19E-09 | 0 | 1.90E-11 | 0 | 7.02E-13 | 0 |
| 4 | 1.72E-16 | 9.91E-09 | 3.73E-09 | 0 | 2.50E-11 | 0 | 5.63E-13 | 0 |
| 5 | 2.24E-15 | 9.91E-09 | 1.98E-08 | 0 | 2.45E-11 | 0 | 9.91E-13 | 0 |
| 6 | 6.44E-16 | 9.94E-09 | 3.63E-09 | 0 | 4.06E-11 | 0 | 7.49E-13 | 0 |
| 7 | 1.68E-15 | 9.94E-09 | 1.08E-07 | 0 | 3.08E-10 | 0 | 8.83E-12 | 0 |
| 8 | 9.89E-17 | 9.89E-09 | 7.04E-08 | 0 | 2.58E-11 | 0 | 9.95E-12 | 0 |
| 9 | 1.39E-16 | 9.85E-09 | 1.37E-10 | 0 | 8.25E-11 | 0 | 4.51E-13 | 0 |
| 10 | 1.67E-16 | 9.77E-09 | 4.77E-12 | 0 | 3.52E-04 | 0 | 7.37E-12 | 0 |
| 11 | 2.64E-16 | 9.82E-09 | 3.13E-09 | 0 | 2.62E-04 | 0 | 1.48E-12 | 0 |
| 12 | 3.68E-16 | 9.11E-09 | 1.08E-09 | 0 | 1.09E-05 | 0 | 2.58E-11 | 0 |
| 13 | 9.32E-17 | 9.90E-09 | 2.68E-08 | 0 | 7.59E-11 | 0 | 8.69E-13 | 0 |
| 14 | 1.11E-16 | 9.91E-09 | 1.81E-08 | 0 | 3.95E-11 | 0 | 4.35E-13 | 0 |
| 15 | 4.13E-16 | 9.90E-09 | 2.71E-08 | 0 | 3.30E-11 | 0 | 7.89E-13 | 0 |

时间均优于其它 3 种方法, 且有 11 个算例的迭代次数最少. SQP-type 和 PF 与 FDIPA 的求解效率差距较大. 在最终解的可行性方面, FDIPA 方法表现同样优秀. 算法 5 保证了迭代点始终满足锥约束可行. IPOPT 将锥约束建模为一般非线性约束, 解的锥约束可行性劣于其它 3 种方法. FDIPA 的非线性等式约束的可行性虽不及 IPOPT, 但仍达到较好的水平, 在大部分算例上优于 SQP-type 和 PF.

然后我们考察这些方法对动力下降制导问题的求解效果. 为了尽量满足 FDIPA 等方法的收敛性要求, 我们考虑在 3.1 节 (ROC) 问题的基础上去掉非光滑项 $\mathbf{D}(t)$ 和非光滑约束 (3.7), 加入 [37] 中的倾角约束

$$\|\mathbf{T}(t)\| \cos \theta_3 \leq \mathbf{e}_U^T \mathbf{T}(t).$$

然后通过无损凸化和时间离散化得到形如 (NSOCP) 的问题. 初始位置 $\mathbf{r}_{\text{init}} = [-600 \ 3000 \ 600]^T$, 初始速度 $\mathbf{v}_{\text{init}} = [50 \ -150 \ -50]^T$, 角度 $\theta_1 = 85$, $\theta_3 = 30$. 分段数 N 取 10, 15, 20, 25 产生 4 个算例. 对每个算例, 结合类似 3.2.3.1 节 (RSOCP-INIT) 初始化问题, 终端时刻猜测值 τ_f 取 25, 27, 29, 31, 33, 35 产生 6 个初始点. 问题中的其它参数取值同表 3.1. 由于 SQP-type 方法在求解动力下降制导问题时会遇到

LSOCP 子问题不可行而无法继续求解的问题, 因此这部分的测试仅针对 IPOPT、SC、PF 和 FDIPA. 这里的 SC 方法是类似 3.2 节的单阶段序列凸化方法. 为了避免步长过小, FDIPA 方法线搜索中的充分下降条件 (3.21) 在本实验中替换为如下的非单调充分下降条件:

$$\phi_c(x^{(k)} + t^{(k)}d^{(k)}) \leq \phi_c^{max}(x^{(k)}) + t^{(k)}\mu D(\phi_c(x^{(k)}); d^{(k)}).$$

其中 ϕ_c^{max} 表示 $\max\{\phi_c(x^{(i)}) : \max\{0, k - M + 1\} \leq i \leq k\}$. M 取 30.

表 3.6 展示了 4 种方法的对比结果. k_{ave} 、 k_{min} 和 k_{max} 分别表示某方法从 6 个初始点出发求解单个算例的平均、最少和最多迭代次数. 在求解效率方面, SC 方法在 4 种方法中最优. FDIPA 方法略优于 IPOPT. PF 方法与其它 3 种方法的差距较大. 在稳定性方面, SC 方法在规模较小的两个算例上稳定性较好. FDIPA 在 $N = 20$ 和 $N = 25$ 两个算例上的稳定性优于 IPOPT 和 SC. 在解的可行性方面, PF 和 FDIPA 明显优于 IPOPT 和 SC. 对比第一部分的测试结果 (表 3.5), IPOPT 变化较大, 特别是等式约束的可行性变差了许多. 相比之下, FDIPA 的可行性变化较小.

综合算法的理论性质和以上的数值结果, 我们从迭代次数、子问题可行性、收敛性和解的质量角度将算法的对比总结为表 3.7. 尽管 FDIPA 方法在求解动力下降制导问题时的效率不及定制化的 SC 方法, 但在算法可行性、收敛性和解的可行性方面均有优势. 对于可靠性要求高的航天应用场景, FDIPA 的这些优势无疑是十分重要的.

表 3.6 动力下降制导测试问题的迭代次数和可行性对比

| N | IPOPT | | | SC | | | PF | | | FDIPA | | |
|-----|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| | k_{ave} | k_{min} | k_{max} | k_{ave} | k_{min} | k_{max} | k_{ave} | k_{min} | k_{max} | k_{ave} | k_{min} | k_{max} |
| 10 | 162 | 80 | 262 | 132(7) | 119(6) | 139(7) | 3964(121) | 2400(74) | 6613(200) | 147 | 109 | 247 |
| 15 | 217 | 122 | 361 | 110(6) | 106(6) | 125(7) | 4740(132) | 3109(89) | 6019(166) | 192 | 156 | 246 |
| 20 | 254 | 159 | 545 | 170(10) | 108(6) | 307(17) | 7079(175) | 4599(123) | 8352(200) | 235 | 193 | 325 |
| 25 | 291 | 132 | 420 | 231(11) | 102(5) | 436(21) | 7962(185) | 6184(139) | 8861(200) | 262 | 204 | 323 |

| N | IPOPT | | | SC | | | PF | | | FDIPA | | |
|-----|----------|----------|--|----------|----------|--|----------|----------|--|----------|-------|--|
| | feasH | feasG | | feasH | feasG | | feasH | feasG | | feasH | feasG | |
| 10 | 2.62E-06 | 2.62E-07 | | 4.35E-07 | 7.45E-12 | | 3.19E-11 | 1.01E-12 | | 5.09E-11 | 0 | |
| 15 | 1.68E-06 | 2.63E-07 | | 3.39E-07 | 2.49E-11 | | 3.40E-11 | 4.09E-12 | | 5.58E-11 | 0 | |
| 20 | 1.23E-06 | 2.61E-07 | | 4.06E-07 | 1.30E-11 | | 5.58E-11 | 1.84E-12 | | 6.31E-11 | 0 | |
| 25 | 9.76E-07 | 2.61E-07 | | 2.01E-07 | 4.11E-11 | | 4.61E-11 | 6.43E-12 | | 5.82E-11 | 0 | |

表 3.7 NSOCP 求解方法性质对比

| 方法 | 迭代次数 | LSOCP 子问题 (如果有) 可行性 | 全局收敛性 | 解的可行性 |
|----------|------|---------------------|-------|-------|
| SC | 少 | 不确定 | 未知 | 一般 |
| SQP-type | 多 | 不确定 | 有 | 一般 |
| PF | 很多 | 始终可行 | 有 | 较好 |
| FDIPA | 中等 | — | 有 | 较好 |

3.5 小结

本章节考虑了包含非凸气动阻力项和非凸推力方向控制约束的三自由度燃料最优动力下降制导最优控制问题, 以及从中抽象出的非线性二阶锥规划问题.

对于动力下降制导问题, 我们通过无损凸化、时间离散化和线性化技术, 非凸最优控制问题被转化为 SOCP 问题进行求解. 为了避免数值不稳定的发生同时提高算法迭代效率, 我们在单阶段序列凸化方法的基础上提出一个新的两阶段序列凸化方法. 数值结果表明所提方法可以通过求解少量的凸优化问题得到非凸最优控制问题的解, 并且算法的收敛性不依赖于初始猜测值的选取. 此外, 多个测试问题的结果均说明两阶段方法拥有比单阶段方法更稳定、高效的数值表现.

对于 NSOCP 问题, 我们利用 ℓ_1 势函数和非光滑线搜索, 将文献 [93][82] 所提的针对 NLP 和仅含锥约束的 NSOCP 问题的可行方向内点法推广至可以求解含非线性等式约束的问题 (NSOCP), 并分析了算法的全局收敛性. 通过基于随机产生的问题和动力下降制导问题的数值实验, 我们说明了 FDIPA 方法相比于若干 NLP 和 NSOCP 求解算法在求解速度、算法可行性、稳定性、收敛性和解的可行性方面的优势, 展示了 FDIPA 在高可靠航天应用场景的应用潜力.

第4章 AGV 道口分布问题的启发式方法

本章针对 AGV 道口分布问题的 USSMPFLP 模型, 介绍从加速直接求解方法的角度出发而设计的基于线性规划松弛解的紧问题启发式和邻域搜索启发式方法. 通过已知的线性松弛解, 这两种方法利用启发式思想快速得到原问题的可行解. 作为问题上界, 此可行解能够对直接求解方法进行加速. 本章安排如下: 4.1 节介绍 USSMPFLP 的整数规划模型及算法设计动机, 4.2 节介绍两种启发式方法, 4.3 节展示启发式方法的数值结果, 最后的小结在 4.4 节给出.

4.1 整数规划模型

无容量限制的单货源多商品设施选址问题可以描述为以下的最优化问题:

$$\begin{aligned}
 (\text{FLP}) \quad & \min_{x,y} \quad \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J c_{kij} x_{kij} + \sum_{i=1}^I \sum_{j=1}^J f_{ij} y_{ij} \\
 \text{s. t.} \quad & \sum_{j=1}^J x_{kij} = 1, \forall k \in \mathcal{K}, i \in \mathcal{I}, \quad (4.1)
 \end{aligned}$$

$$x_{kij} \leq y_{ij}, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}, \quad (4.2)$$

$$\sum_{i=1}^I y_{ij} \leq 1, \forall j \in \mathcal{J}, \quad (4.3)$$

$$x_{kij}, y_{ij} \in \{0, 1\}, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}. \quad (4.4)$$

其中 $\mathcal{K} = \{1, \dots, K\}$ 是顾客的指标集, $\mathcal{I} = \{1, \dots, I\}$ 是商品类型的指标集, $\mathcal{J} = \{1, \dots, J\}$ 是设施候选位置的指标集. c_{kij} 为顾客 k 从 j 处设施获得第 i 类商品的费用, 通常是商品的运输费用. f_{ij} 表示在 j 处建造生产第 i 种商品的设施的费用. (FLP) 的变量 x_{kij} 和 y_{ij} 都是 0-1 整数变量, 其中 $x_{kij} = 1 (x_{kij} = 0)$ 代表顾客 k 要 (不要) 从 j 处设施获得第 i 类商品, $y_{ij} = 1 (y_{ij} = 0)$ 代表在 j 处要 (不要) 建造生产第 i 种商品的设施. (FLP) 的目标是极小化商品运输总费用和设施建造总费用. 约束 (4.1) 和 (4.4) 表示顾客对各种商品的需求必须得到满足, 且仅从一处设施得到满足. 约束 (4.2) 是适定性约束, 表示只有当对应的设施存在时才能满足顾客的需求. 约束 (4.3) 限定了每个设施只能生产一种商品. 对于一个可行的 USSMPFLP, 显然有设施候选位置个数大于等于商品种类数, 即 $J \geq I$. 本文假定这一条件总成立.

(FLP) 属于 0-1 整数规划问题. 这类问题的求解通常是 \mathcal{NP} 难的. Huang[56] 证明了仅含两种商品的 USSMPFLP 是 \mathcal{NP} 完全的. 对于较大规模的 (FLP), 即 K, I, J 较大时, 即使调用目前国际上最高效的商用整数规划求解器也难以在合理的时间内求得质量较好的解. 目前应用于整数规划求解器 (如 Gurobi [100], CPLEX [90], SCIP [101] 等) 中较为流行的求解方法有分支定界方法和分支割平面 (branch-and-cut) 方法. 在这类方法的求解过程中, (FLP) 的上、下界分别由当前的候选解和分支定界树中未求解子问题的线性规划松弛解给出. 图 4.1 给出了调用商业整数规划求解器 Gurobi 求解一个 AGV 道口分布问题实例的计算结果.

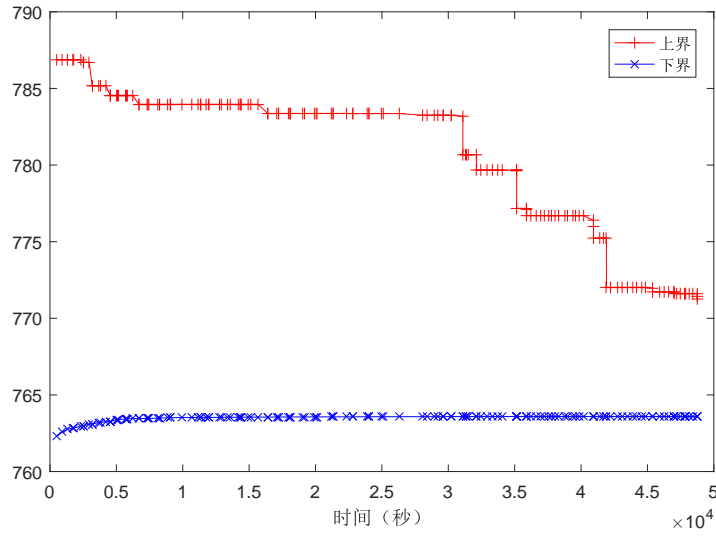


图 4.1 AGV 道口分布问题实例的上、下界随求解时间的变化曲线.

这一实例在近 12.5 小时的求解之后才最终达到求解器的终止条件 (gap 低于 1%)! 由上、下界的变化情况可以得出: (1) 求解器难以在短时间内找到高质量的候选解, 导致上界降低缓慢; (2) 与初始上界相比, 初始下界更接近最终下界, 初始线性规划松弛解可能比较接近最优解. 针对此实例, 若能通过启发式方法在较短时间内给出较高质量的解, 其求解效率将大大改善. 4.3 节将进一步对比和分析不同方法求解此实例的结果.

此外, (FLP) 中的变量 x_{kij} 可在设施位置固定后通过最优性来显式地确定. 当

$y_{ij} = y_{ij}^h \in \{0, 1\}$ 且 $\sum_{i=1}^I y_{ij}^h \leq 1$ 时, (FLP) 约简为如下关于 x_{kij} 的优化问题:

$$\begin{aligned}
 (\text{FLP}(y^h)) \quad & \min_x \quad \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J c_{kij} x_{kij} \\
 \text{s. t.} \quad & \sum_{j=1}^J x_{kij} = 1, \forall k \in \mathcal{K}, i \in \mathcal{I}, \\
 & x_{kij} \leq y_{ij}^h, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}, \\
 & x_{kij} \in \{0, 1\}, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}.
 \end{aligned}$$

(FLP(y^h)) 有显式解 x^h , 其中:

$$x_{kij}^h = \begin{cases} 1, & j = \arg \min_{t \in \mathcal{J}_i} \{c_{kit}\}; \\ 0, & \text{其它;} \end{cases}, \quad k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}.$$

这里 $\mathcal{J}_i = \{j \mid y_{ij}^h = 1, j \in \mathcal{J}\}$. 即使是仅有部分变量 y_{ij} 被固定, 也可类似地通过最优性确定部分变量 x_{kij} 的取值.

本项工作根据以上的观察, 利用 (FLP) 的线性松弛解设计启发式方法确定或部分确定设施位置 y^h 进而得到 (FLP) 的可行解 (x^h, y^h) . 随后将 (x^h, y^h) 对应的目标函数值作为上界来加速求解器的求解效率.

4.2 基于松弛解的启发式算法

本节介绍应用于 (FLP) 的两种启发式算法, 分别是紧问题启发式和结合邻域搜索的改进启发式方法.

4.2.1 紧问题启发式

记 (x^{LP}, y^{LP}) 为 (FLP) 的线性规划松弛解, $\mathcal{F} \subseteq \mathcal{I} \times \mathcal{J}$ 为一指标集. 首先给出如下紧问题的形式:

$$\begin{aligned}
 (\text{C}(y^{LP}, \mathcal{F})) \quad & \min_{x,y} \quad \sum_{k=1}^K \sum_{i=1}^I \sum_{j=1}^J c_{kij} x_{kij} + \sum_{i=1}^I \sum_{j=1}^J f_{ij} y_{ij} \\
 \text{s. t.} \quad & \sum_{j=1}^J x_{kij} = 1, \forall k \in \mathcal{K}, i \in \mathcal{I}, \\
 & x_{kij} \leq y_{ij}, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J}, \\
 & \sum_{i=1}^I y_{ij} \leq 1, \forall j \in \mathcal{J}, \\
 & y_{ij} = y_{ij}^{LP}, \forall (i, j) \in \mathcal{F}, \\
 & x_{kij}, y_{ij} \in \{0, 1\}, \forall k \in \mathcal{K}, i \in \mathcal{I}, j \in \mathcal{J},
 \end{aligned} \tag{4.5}$$

问题 $(\text{C}(y^{LP}, \mathcal{F}))$ 仅比原问题 (FLP) 多出了约束 (4.5), 即部分变量 y_{ij} 被固定为线性松弛解的值. 由于固定了部分变量, $(\text{C}(y^{LP}, \mathcal{F}))$ 的可行域比 (FLP) 的可行域更小, 因此前者是后者的紧问题. 紧问题启发式的思想是: 通过线性规划松弛解 y^{LP} 构造指标集 \mathcal{F} , 从而固定一部分 y_{ij} 变量; 再由 4.1 节的观察, 部分 x_{kij} 也可被固定, 从而问题规模可进一步降低; 通过快速求解小规模紧问题, 得到同时可行于原问题 (FLP) 的解 (x^h, y^h) ; 此可行解可作为 (FLP) 的一个上界, 从而提高 (FLP) 的求解效率. 关于这一启发式方法的实施, 有两个问题需要回答: (1) 如何设置规则来生成指标集 \mathcal{F} ; (2) 能否总是可以从紧问题 $(\text{C}(y^{LP}, \mathcal{F}))$ 中获得最优解或者可行解, 即紧问题是否可行.

关于第一个问题, 我们引入在解 (x^{LP}, y^{LP}) 下的“确信”变量的概念.

定义 4.1. 对于 (FLP) 的线性规划松弛解 (x^{LP}, y^{LP}) , 变量 y_{ij} , $(i, j) \in \mathcal{I} \times \mathcal{J}$ 的“确信”取值 (若存在) 为:

- 对每个 $i \in \mathcal{I}$, 若对任意的 $j \in \mathcal{J}$ 均有 $y_{ij}^{LP} = 0$ 或 $y_{ij}^{LP} = 1$, 则变量 y_{ij} , $j \in \mathcal{J}$ 的“确信”取值是 y_{ij}^{LP} ;
- 对每个 $j \in \mathcal{J}$, 若存在某个 $i_0 \in \mathcal{I}$ 使得 $y_{i_0 j}^{LP} = 1$, 则变量 y_{ij} , $i \in \mathcal{I}$ 的“确信”取值是 y_{ij}^{LP} .

称所有存在“确信”取值的变量为“确信”变量.

所有“确信”变量的指标构成了指标集 \mathcal{F} . “确信”变量的定义思想来自 4.1 节中 (FLP) 初始下界的观察. 由于 (FLP) 的线性规划松弛解的质量很高, 若出现某种商品对于生产候选地的选取较为明确 (y_{ij} 取 0 或 1), 或是某个候选地对所生产产品种类的选取较为明确 (y_{ij} 取 0 或 1), 则可认为这样的选取是较优的, 可将这部分变量固定下来. 其他在线性松弛解中取分数的变量在紧问题中仍是变量. 这类变量的取值倾向在线性松弛解中并不明确, 需要在紧问题中进一步优化. 下面的例 4.1 具体展示了指标集 \mathcal{F} 是如何生成的.

例 4.1. 某 (FLP) 问题实例满足 $I = 4, J = 5$. 其线性松弛解中变量 y 的部分如表 4.1 所示. 表中粗体数值代表的是“确信”取值, 即对应的指标集为:

$$\mathcal{F} = (I \times 2) \cup (I \times 3) \cup (I \times 5) \cup (4 \times J).$$

表 4.1 某 (FLP) 问题的松弛解 y^{LP} .

| $j \backslash i$ | 1 | 2 | 3 | 4 |
|------------------|----------|----------|----------|----------|
| 1 | 0.6 | 0.1 | 0.2 | 0 |
| 2 | 1 | 0 | 0 | 0 |
| 3 | 0 | 1 | 0 | 0 |
| 4 | 0 | 0 | 0.8 | 0 |
| 5 | 0 | 0 | 0 | 1 |

对于紧问题可行性的判断, 以下定理指出在原问题 (FLP) 可行的前提下, 紧问题 $(C(y^{LP}, \mathcal{F}))$ 总存在可行解.

定理 4.1. 若 $I \leq J$, (x^{LP}, y^{LP}) 是 (FLP) 的线性松弛解且指标集 \mathcal{F} 由定义 4.1 产生, 则问题 $(C(y^{LP}, \mathcal{F}))$ 是可行的.

证明. 记

$$U_1 = \{i \mid \forall j \in J, y_{ij}^{LP} \neq 1\}, u_1 = |U_1|,$$

$$U_2 = \{j \mid \exists i \in I, (i, j) \notin \mathcal{F}\}, u_2 = |U_2|,$$

$$S = \sum_{(i,j) \in (I \times J) \setminus \mathcal{F}} y_{ij}^{LP},$$

其中 U_1 和 U_2 分别表示紧问题 $(C(y^{LP}, \mathcal{F}))$ 中还未被分配设施的商品种类指标集和还未被指定生产商品的种类的设施候选位置指标集. 根据“确信”变量的定义可知以下结论成立:

$$U_1 \times U_2 \subseteq (\mathcal{I} \times \mathcal{J}) \setminus \mathcal{F} \subseteq \mathcal{I} \times U_2.$$

要证紧问题的可行性, 只需证明 $u_1 \leq u_2$.

一方面, 对于每个 $i \in U_1$, 若存在 $j_0 \in \mathcal{J}$ 使得 $(i, j_0) \in \mathcal{F}$, 则必存在 $i_0 \in \mathcal{I}$ 使得 $y_{i_0 j_0}^{LP} = 1$. 此时由 (FLP) 的约束可知 $y_{ij_0}^{LP} = 0$. 于是有

$$S \geq \sum_{i \in U_1} \sum_{j \in U_2} y_{ij}^{LP} = \sum_{i \in U_1} \sum_{j \in \mathcal{J}} y_{ij}^{LP} \geq \sum_{i \in U_1} 1 = u_1.$$

另一方面, 对于每个 $j \in U_2$, 若存在 $i_0 \in \mathcal{I}$ 使得 $(i_0, j) \in \mathcal{F}$, 则必有 $y_{i_0 j}^{LP} = 0$. 于是有

$$S = \sum_{j \in U_2} \sum_{i \in \mathcal{I}} y_{ij}^{LP} \leq \sum_{j \in U_2} 1 = u_2.$$

综上, 有 $u_1 \leq u_2$. 证毕. \square

由定理 4.1 可知, 紧问题启发式总可以为原问题 (FLP) 找到一个上界. 数值实验的效果也表明这一上界的质量往往是很高的. 事实上, 紧问题启发式中的指标集 \mathcal{F} 可通过迭代的方式进一步扩充, 我们将完整的紧问题启发式算法总结为算法 6. 当首个集合 \mathcal{F} 生成后, 可利用紧问题 $(C(y^{LP}, \mathcal{F}))$ 的线性松弛问题的解更新 y^{LP} 和 \mathcal{F} , 直到集合 \mathcal{F} 的元素不再增加为止. 由于此时的松弛解 y^{LP} 是包含之前“确信”变量取值的信息的, 因此指标集 \mathcal{F} 中的元素数量是单调增加的. 又因为 $\mathcal{F} \subseteq \mathcal{I} \times \mathcal{J}$, 此迭代总是有限终止的. 通过此方法最终可获得规模更小的紧问题, 有利于提高紧问题启发式的执行效率.

4.2.2 结合邻域搜索的改进启发式

对于 4.2.1 节给出的紧问题启发式方法, 在实际应用时若生成的指标集 \mathcal{F} 所含元素较少, 则这一方法的效果是不理想的. 因为此时的紧问题与原问题的规模相当, 难以发挥此启发式方法的优势. 针对这种现象, 可相应提出线性松弛解 (x^{LP}, y^{LP}) 下的“伪确信”变量的定义.

定义 4.2. 对于 (FLP) 的线性规划松弛解 (x^{LP}, y^{LP}) , 变量 y_{ij} , $(i, j) \in \mathcal{I} \times \mathcal{J}$ 均为“伪确信”变量, 且它们的“伪确信”取值可通过以下步骤得到:

算法 6 紧问题启发式算法

输入: (FLP) 的线性松弛解 y^{LP} , 紧问题求解时间上限 T_{max} , 参数 $\epsilon > 0$.

输出: (FLP) 的可行解 (x^h, y^h) .

- 1: 初始化 $\mathcal{F}^{(0)} := \emptyset, t := 0$;
- 2: **repeat**
- 3: 令 $t := t + 1, \mathcal{F}^{(t)} := \mathcal{F}^{(t-1)}$;
- 4: **for** $i \in \mathcal{I}$ **do**
- 5: **if** 对任意 $j \in \mathcal{J}$ 有 $y_{ij}^{LP} = 0$ 或 $y_{ij}^{LP} = 1$ **then**
- 6: 更新 $\mathcal{F}^{(t)} := \mathcal{F}^{(t)} \cup \{(i, j) \mid \forall j \in \mathcal{J}\}$;
- 7: **end if**
- 8: **end for**
- 9: **for** $j \in \mathcal{J}$ **do**
- 10: **if** 存在 $i_1 \in \mathcal{I}$ 使得 $y_{i_1 j}^{LP} = 1$ **then**
- 11: 更新 $\mathcal{F}^{(t)} := \mathcal{F}^{(t)} \cup \{(i, j) \mid \forall i \in \mathcal{I}\}$;
- 12: **end if**
- 13: **end for**
- 14: 求解 $(C(y^{LP}, \mathcal{F}^{(t)}))$ 的线性松弛问题并更新 y^{LP} 为其最优解 (y 的部分);
- 15: **until** $\mathcal{F}^{(t)} = \mathcal{F}^{(t-1)}$
- 16: 求解 $(C(y^{LP}, \mathcal{F}^{(t)}))$, 并设置终止条件为 gap 小于 ϵ 和最大求解时间 T_{max} , 返回得到的最佳候选解 (x^h, y^h) .

- 对每个 $i \in \mathcal{I}$ 和 $j \in \mathcal{J}$, 初始化 $y_{ij}^h = 0$;
- 依次取 $i = 1, \dots, I$, 记 $j_0 \in \arg \max_{j \in \mathcal{E}} \{y_{ij}^{LP}\}$, 令 $y_{ij_0}^h = 1, y_{ij}^h = 0, j \in \mathcal{J} \setminus j_0$, 其中 $\mathcal{E} = \{j \in \mathcal{J} \mid y_{ij}^h = 0, \forall i \in \mathcal{I}\}$;
- y_{ij}^h 即为各变量 y_{ij} , $(i, j) \in \mathcal{I} \times \mathcal{J}$ 的“伪确信”取值.

由以上定义不难验证: (1) y^h 满足原问题 (FLP) 中约束, 可产生可行解 (x^h, y^h) ; (2) 每种商品类型仅指定了一个设施来生产. 虽然“伪确信”变量的定义可以为 (FLP) 提供一个上界, 但实际应用时发现此上界的质量往往很差. 这是因为“伪确信”变量的定义对线性松弛解中信息的提取比较粗糙. 此方法仅考虑了 y^{LP} 中最大分数的信息, 且在候选设施位置数量 J 相比于商品种类 I 较大时, 所获得的启发式解 y^h 中设施建造数量偏少, 使得商品运输费用较高, 解的质量自然就比较差.

为了获得更高质量的启发式解, 我们引入如文献 [58] 中的邻域搜索的方法. 这些方法包括: 移位, 增加, 移除, 互换.

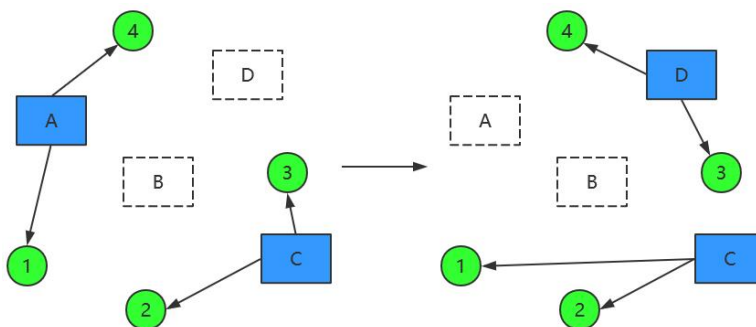


图 4.2 移位: 将一个设施转移到一个无设施的候选位置.

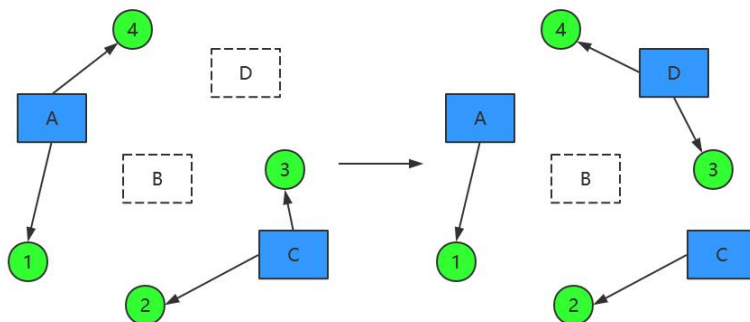


图 4.3 增加: 在一个无设施的候选位置建造一个设施.

图 4.2–4.5 中的圆形表示顾客, 矩形表示候选设施位置. 四种邻域搜索的具体步骤如算法 7–10 所述. 通过以上的邻域搜索方法, 我们将由“伪确信”定义得到的可行解 (x^h, y^h) 作为初始解, 利用邻域搜索的方法不断改善此解的质量, 即单调

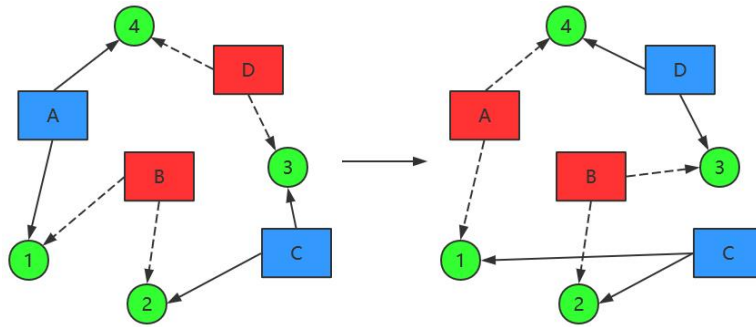


图 4.4 互换：交换两个设施所生产的商品种类。

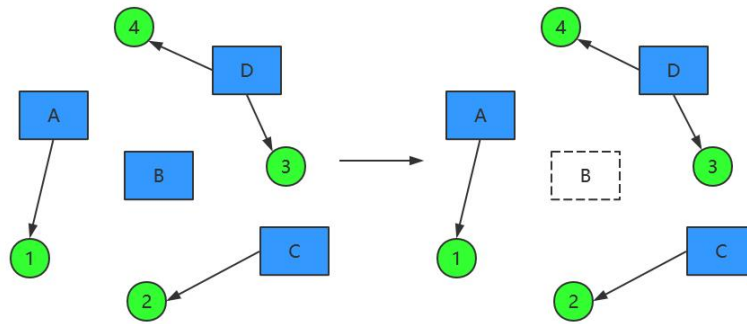


图 4.5 移除：将一个设施去除。

地降低此可行解的目标函数值, 直到找到局部最优解为止. 完整的结合邻域搜索的改进启发式算法总结为算法 11.

算法 7 邻域搜索之移位

输入: (FLP) 的可行解 y^h , 指标集 $\mathcal{E} := \{j \in \mathcal{J} \mid y_{ij}^h = 0, \forall i \in \mathcal{I}\}$, 参数 $\epsilon > 0$.

输出: 改进后的 (FLP) 可行解 y^h 及对应的 \mathcal{E} .

```

1: for  $i \in \mathcal{I}$  do
2:   令  $\mathcal{J}_1 := \{j \in \mathcal{J} \mid y_{ij}^h = 1\}$ ;
3:   for  $j_1 \in \mathcal{J}_1$  do
4:     令  $r_{max} := 0$ ;
5:     for  $j_2 \in \mathcal{E}$  do
6:       令  $\mathcal{J}'_1 := \{j \in \mathcal{J} \mid y_{ij}^h = 1\}$ ,  $\mathcal{J}_2 := \mathcal{J}'_1 \setminus j_1 \cup j_2$ ,
7:        $r := \sum_{k \in \mathcal{K}} \min\{c_{kij} \mid j \in \mathcal{J}'_1\} - \min\{c_{kij} \mid j \in \mathcal{J}_2\} + f_{ij_1} - f_{ij_2}$ ;
8:       if  $r > r_{max}$  then
9:         令  $r_{max} := r$ ,  $j_{max} := j_2$ ;
10:      end if
11:    end for
12:    if  $r_{max} > \epsilon$  then
13:      令  $y_{ij_1}^h := 0$ ,  $y_{ij_{max}}^h := 1$ , 更新  $\mathcal{E} := \mathcal{E} \setminus j_{max} \cup j_1$ .
14:    end if
15:  end for
16: end for
    
```

算法 8 邻域搜索之增加

输入: (FLP) 的可行解 y^h , 指标集 $\mathcal{E} := \{j \in \mathcal{J} \mid y_{ij}^h = 0, \forall i \in \mathcal{I}\}$, 参数 $\epsilon > 0$.

输出: 改进后的 (FLP) 可行解 y^h 及对应的 \mathcal{E} .

```

1: 初始化  $C := \emptyset$ ;
2: for  $i \in \mathcal{I}$  do
3:   for  $j_1 \in \mathcal{E}$  do
4:     令  $\mathcal{J}_1 := \{j \in \mathcal{J} \mid y_{ij}^h = 1\}$ ,  $\mathcal{J}_2 := \mathcal{J}_1 \cup j_1$ ,
5:      $r := \sum_{k \in \mathcal{K}} \min\{c_{kij} \mid j \in \mathcal{J}_1\} - \min\{c_{kij} \mid j \in \mathcal{J}_2\} - f_{ij_1}$ ;
6:     if  $r > \epsilon$  then
7:       令  $C := C \cup (i, j_1, r)$ ;
8:     end if
9:   end for
10: end for
11: while  $C \neq \emptyset$  do
12:   令  $(i^*, j^*, r^*) := \arg \max\{r \mid (i, k, r) \in C\}$ ;
13:   令  $y_{i^*j^*}^h := 1$ , 更新  $C := C \setminus \{(i, j, r) \in C \mid i = i^* \text{ 或 } j = j^*\}$ .
14: end while

```

算法 9 邻域搜索之互换

输入: (FLP) 可行解 y^h , 参数 $\epsilon > 0$.

输出: 改进解 y^h .

```

1: for  $i_1 \in \mathcal{I}$  do
2:   令  $\mathcal{J}_1 := \{j \in \mathcal{J} \mid y_{i_1 j}^h = 1\}$ ;
3:   for  $j_1 \in \mathcal{J}_1$  do
4:     令  $r_{max} := 0$ ;
5:     for  $(i_2, j_2) \in \{(i, j) \in \mathcal{I} \times \mathcal{J} \mid y_{ij}^h = 1, i \neq i_1, j \neq j_1\}$  do
6:       令  $\mathcal{J}'_1 := \{j \in \mathcal{J} \mid y_{i_1 j}^h = 1\}$ ,  $\mathcal{J}''_1 := \mathcal{J}'_1 \setminus j_1 \cup j_2$ ;
7:       令  $\mathcal{J}_2 := \{j \in \mathcal{J} \mid y_{i_2 j}^h = 1\}$ ,  $\mathcal{J}'_2 := \mathcal{J}_2 \setminus j_2 \cup j_1$ ,
8:        $r := \sum_{k \in \mathcal{K}} \min\{c_{kij} \mid j \in \mathcal{J}'_1\} + \min\{c_{kij} \mid j \in \mathcal{J}_2\} - \min\{c_{kij} \mid j \in \mathcal{J}''_1\} - \min\{c_{kij} \mid$ 
           $j \in \mathcal{J}'_2\} + f_{i_1 j_1} + f_{i_2 j_2} - f_{i_1 j_2} - f_{i_2 j_1}$ ;
9:       if  $r > r_{max}$  then
10:        令  $r_{max} := r, i_{max} = i_2, j_{max} = j_2$ ;
11:       end if
12:     end for
13:     if  $r_{max} > \epsilon$  then
14:       令  $y_{i_1 j_1}^h := 0, y_{i_{max} j_{max}}^h := 0, y_{i_1 j_{max}}^h := 1, y_{i_{max} j_1}^h := 1$ .
15:     end if
16:   end for
17: end for
    
```

算法 10 邻域搜索之移除

输入: (FLP) 的可行解 y^h , 指标集 $\mathcal{E} := \{j \in \mathcal{J} \mid y_{ij}^h = 0, \forall i \in \mathcal{I}\}$, 参数 $\epsilon > 0$.

输出: 改进后的 (FLP) 可行解 y^h 及对应的 \mathcal{E} .

```

1: 令  $x_{kij}^h$  为问题  $(P(y^h))$  的最优解;
2: for  $(i_1, j_1) \in \{(i, j) \in \mathcal{I} \times \mathcal{J} \mid y_{ij}^h = 1\}$  do
3:   if  $x_{k i_1 j_1}^h = 0, \forall k \in \mathcal{K}$  then
4:     令  $y_{i_1 j_1}^h := 0$ , 更新  $\mathcal{E} := \mathcal{E} \cup j_1$ .
5:   end if
6: end for
    
```

算法 11 结合邻域搜索的改进启发式算法

输入: (FLP) 的线性松弛解 y^{LP} , 紧问题求解时间上限 T_{max} , 参数 $\epsilon > 0$.

输出: (FLP) 的可行解 (x^h, y^h) .

- 1: 初始化 $y^h := 0, \mathcal{E} := \mathcal{J}$;
 - 2: **for** $i \in \mathcal{I}$ **do**
 - 3: 令 $j_1 := \arg \max \{y_{ij}^{LP} \mid j \in \mathcal{J}\}, y_{ij_1}^{LP} := 1$, 更新 $\mathcal{E} := \mathcal{E} \setminus j_1$;
 - 4: **end for**
 - 5: **repeat**
 - 6: 令 $\tilde{y}^h := y^h$;
 - 7: 调用算法 7–10 更新 y^h 和 \mathcal{E} ;
 - 8: 令 x^h 为 (FLP(y^h)) 的最优解;
 - 9: **until** $y^h = \tilde{y}^h$
 - 10: **return** (x^h, y^h) .
-

4.3 数值实验

本节将本文给出的两种启发式方法与商业求解器及文献中的算法进行对比测试. 测试对象分别是某物流公司的实际问题算例及文献 [58] 中的测试算例. 本节所有测试均在装有 i7-6700 处理器和 16GB 内存的台式机上完成.

首先测试紧问题启发式方法对求解器求解效率的提升效果. 我们将紧问题启发式方法嵌入商业求解器 Gurobi (版本 7.5.2) 中来求解 (FLP), 并将这一方案 (记为 CPH) 与直接调用 Gurobi 求解问题 (记为 GRB) 以及文献 [58] 中给出的 Lagrangian 松弛方法 (记为 LR) 进行对比. 这一部分的测试算例是来自某物流公司的实际问题. 算例的规模如表 4.2 所示. 此处的算例 3 即为 4.1 节中提及的问题实例. Gurobi 求解器的终止条件设置为 gap 低于 1%. Lagrangian 松弛方法的最大迭代次数设置为 1000 次. 三种方案的求解时间、上界和 gap 对比如表 4.3 所示.

表 4.2 第一组测试算例.

| 算例 | K | I | J | 变量 | 约束 | 非零元素 |
|----|-----|-----|-----|-----------|-----------|-----------|
| 1 | 32 | 20 | 301 | 198,660 | 193,581 | 583,940 |
| 2 | 32 | 54 | 301 | 536,382 | 522,157 | 1,576,638 |
| 3 | 32 | 260 | 301 | 2,582,580 | 2,512,941 | 7,591,220 |
| 4 | 32 | 100 | 301 | 993,300 | 966,701 | 2,919,700 |
| 5 | 32 | 150 | 301 | 1,489,950 | 1,449,901 | 4,739,550 |
| 6 | 32 | 200 | 301 | 1,986,600 | 1,933,101 | 5,839,400 |

表 4.3 第一组算例上界和 gap 对比.

| 算例 | CPH | | | GRB | | | LR | | |
|----|--------|-------|--------|--------|-------|---------|--------|--------|-------|
| | 上界 | gap | 时间 | 上界 | gap | 时间 | 上界 | gap | 时间 |
| 1 | 230.22 | 0.00% | 8.4 | 230.22 | 0.00% | 11.5 | 268.20 | 15.09% | 52.6 |
| 2 | 324.21 | 0.12% | 126.9 | 324.41 | 0.18% | 113.0 | 411.45 | 21.92% | 142.2 |
| 3 | 770.61 | 0.99% | 5504.0 | 771.28 | 1.00% | 48338.8 | 866.22 | 27.08% | 804.6 |
| 4 | 443.20 | 0.66% | 2953.4 | 443.99 | 0.84% | 6138.5 | 591.55 | 26.07% | 289.2 |
| 5 | 512.70 | 0.68% | 3573.0 | 517.25 | 1.54% | 83564.9 | 667.72 | 25.01% | 464.0 |
| 6 | 600.76 | 0.92% | 2555.2 | 601.30 | 1.00% | 26322.1 | 745.61 | 33.58% | 559.5 |

从以上测试结果可以看出: (1) LR 方法进行 1000 次迭代后的 gap 依然很大, 且在此迭代数下进展已经十分缓慢, 因此并不适用于高质量求解此类问题; (2) 直

接调用求解器确实可以最终得到高质量的解,但大规模问题的求解时间极长,算例3和算例5的求解时间均超过12小时;(3)紧问题启发式方法对Gurobi求解器的提升效果是极为明显的,不但可以找到更好的可行解,而且对大规模算例的求解效率提升极大,最高达到了22倍(算例5).这一结果充分说明了本文所提的紧问题启发式方法对USSMPFLP问题的求解是有巨大提升的.

第二组测试算例来自文献[58],其规模如表4.4所示.对于这组算例,虽然其规模相较第一组中的一些算例要小得多,但其特殊的结构导致问题的难度较大.由于紧问题启发式方法对第二组算例的效果不佳(指标集 \mathcal{F} 的元素个数较少),这一部分我们对比结合邻域搜索的改进启发式算法(记为LSH),直接调用Gurobi求解器和Lagrangian松弛三种方案.其中LSH方案是先调用Gurobi求解(FLP)的线性松弛问题,然后利用得到的松弛解经过4.2.2节所述的方法得到上界后即终止.在这一部分我们限定Gurobi求解器的求解时间上限为2000秒.三种方案的求解结果对比如表4.5所示.

表4.4 第二组测试算例.

| 算例 | J | K | I | 变量 | 约束 |
|----|-----|-----|-----|---------|---------|
| 1 | 50 | 100 | 30 | 151,500 | 153,050 |
| 2 | 50 | 100 | 30 | 151,500 | 153,050 |
| 3 | 50 | 100 | 30 | 151,500 | 153,050 |
| 4 | 50 | 100 | 30 | 151,500 | 153,050 |
| 5 | 50 | 100 | 30 | 151,500 | 153,050 |
| 6 | 50 | 100 | 30 | 151,500 | 153,050 |
| 7 | 30 | 80 | 25 | 60,750 | 62,030 |
| 8 | 30 | 80 | 25 | 60,750 | 62,030 |
| 9 | 30 | 80 | 25 | 60,750 | 62,030 |
| 10 | 50 | 100 | 25 | 126,250 | 127,550 |
| 11 | 50 | 100 | 25 | 126,250 | 127,550 |
| 12 | 50 | 100 | 25 | 126,250 | 127,550 |
| 13 | 100 | 100 | 20 | 202,000 | 202,100 |
| 14 | 100 | 100 | 20 | 202,000 | 202,100 |
| 15 | 100 | 100 | 20 | 202,000 | 202,100 |

上述结果表明LSH方案在较短的时间内提供的上界在多数情形下要优于GRB和LR得到的上界.这一结果充分说明结合邻域搜索的改进启发式方法也可以为(FLP)提供较好的可行解.

表 4.5 第二组算例上界、gap 和时间对比 (时间单位: 秒).

| 算例 | LSH | | | GRB | | | LR | | |
|----|-----------|-------|------|-----------|--------|--------|-----------|--------|------|
| | 上界 | gap | 时间 | 上界 | gap | 时间 | 上界 | gap | 时间 |
| 1 | 77967.46 | 5.77% | 25.5 | 79439.03 | 7.51% | 2000.0 | 80895.30 | 9.45% | 59.2 |
| 2 | 69929.94 | 6.44% | 21.4 | 71407.62 | 8.37% | 2000.0 | 73275.63 | 10.91% | 58.9 |
| 3 | 53854.91 | 8.36% | 22.2 | 55326.47 | 10.79% | 2000.0 | 57044.77 | 13.72% | 59.0 |
| 4 | 450936.70 | 0.97% | 31.3 | 452392.17 | 1.29% | 2000.0 | 452325.74 | 1.32% | 57.8 |
| 5 | 91849.47 | 4.78% | 30.0 | 93304.93 | 6.27% | 2000.0 | 93147.15 | 6.27% | 58.3 |
| 6 | 55940.74 | 7.85% | 23.5 | 57369.20 | 10.14% | 2000.0 | 57368.86 | 10.74% | 57.9 |
| 7 | 516942.20 | 0.36% | 5.3 | 517077.54 | 0.38% | 2000.0 | 517367.60 | 0.48% | 29.2 |
| 8 | 46800.05 | 3.97% | 4.8 | 46982.26 | 4.33% | 2000.0 | 47196.45 | 4.99% | 28.7 |
| 9 | 23292.94 | 7.98% | 5.0 | 23434.84 | 8.49% | 2000.0 | 23795.88 | 10.37% | 30.1 |
| 10 | 210951.47 | 0.37% | 18.0 | 211212.47 | 0.49% | 2000.0 | 211048.89 | 0.58% | 48.1 |
| 11 | 181440.63 | 2.09% | 20.7 | 183021.32 | 2.93% | 2000.0 | 184369.31 | 3.73% | 48.6 |
| 12 | 177363.94 | 2.22% | 21.4 | 178938.71 | 3.08% | 2000.0 | 180158.08 | 3.85% | 48.5 |
| 13 | 396809.87 | 0.00% | 15.8 | 396809.87 | 0.00% | 17.6 | 396945.15 | 0.12% | 70.0 |
| 14 | 124619.36 | 0.34% | 46.4 | 124551.42 | 0.27% | 2000.0 | 143216.56 | 13.41% | 66.4 |
| 15 | 75065.58 | 2.33% | 53.6 | 74663.51 | 1.77% | 2000.0 | 90250.33 | 19.19% | 66.4 |

4.4 小结

本章针对 AGV 道口分布问题的无容量限制单货源多商品设施选址问题模型, 分别提出了基于线性规划松弛解的紧问题启发式和结合邻域搜索的改进启发式方法. 分析表明两种方法均可实施于任意可行的 (FLP) 问题. 数值实验的结果说明了两种启发式方法可以找到比传统的整数规划求解器和 Lagrangian 松弛方法更好的可行解. 对部分问题, 紧问题启发式方法更是可以显著提升求解器的求解效率. 这些结果充分展示了基于线性规划松弛解的启发式方法对 USSMPFLP 的重大意义.

事实上, 本章给出的两种启发式方法还存在很大的改进和提升空间. 对于紧问题启发式方法, 我们并没有给出其得到可行解的质量的理论分析. 若能得到这一方法给出的解对 (FLP) 问题最优解的近似比则可更好地解释其数值表现好的原因. 此外, 对于紧问题启发式难以发挥作用的情形, 本论文提出的邻域搜索方法是一类贪婪算法. 如何通过结合随机性、重启动、路径重连 [11] 等技巧改进纯贪婪的邻域搜索方法以得到更好的可行解也是一个值得探索的方向.

第5章 总结与展望

本文主要研究了热油管道问题、动力下降制导问题和 AGV 道口分布问题三个应用优化问题的模型和算法理论.

首先, 我们给出了热油管道问题的一个确定性全局优化算法. 通过考虑两种加压泵以及通用的水力摩擦非线性约束建模, 我们给出了热油管道问题的一个精确的非凸 MINLP 数学模型. 不同于传统的基于元启发式的方法, 我们在分析了两种松弛问题的等价性后, 为热油管道问题设计了基于分支定界框架的 HOPBB 全局优化算法. 此外为了实际应用这一算法, 我们还设计了预处理算法保证等价性的成立, 即实现了算法的可行性, 以及提出结合外逼近和热启动的凸松弛问题求解方法. 为了说明算法的可行性和计算效率, 我们将 HOPBB 算法应用于我国的 Q-T 热油管线实例. 数值实验结果表明我们提出的算法能够比现有方案节省 6.83% 的运行费用, 并且给出了 Q-T 热油管道进一步改进的方向. 此外, 与最先进的 MINLP 求解器的对比结果也体现出 HOPBB 算法在热油管道问题上的求解效率具有极大的优势.

其次, 我们研究了动力下降制导问题的两类模型和算法. 我们通过无损凸化、时间离散化和线性化技术, 将非凸最优控制问题转化为二阶锥规划问题进行求解. 为了避免数值不稳定的发生同时提高算法迭代效率, 我们在单阶段序列凸化方法的基础上提出一个新的两阶段序列凸化方法. 数值结果表明所提方法可以通过求解少量的凸优化问题得到非凸最优控制问题的解, 并且算法的收敛性不依赖于初始猜测值的选取. 此外, 多个测试问题的结果均说明两阶段方法拥有比单阶段方法更稳定、高效的数值表现. 为进一步考虑一般动力下降制导问题有收敛性保证的算法, 我们将其抽象为非线性二阶锥规划问题. 我们利用 ℓ_1 势函数和非光滑线搜索, 将 [93][82] 所提的针对 NLP 和仅含非线性锥约束的 NSOCP 问题的可行方向内点法推广至可以求解含非线性等式约束的 (NSOCP) 问题, 并分析了算法的全局收敛性. 通过基于随机产生的问题和动力下降制导问题的数值实验, 我们说明了 FDIPA 方法相比于若干 NLP 和 NSOCP 求解算法在求解速度、算法可行性、稳定性、收敛性和解的可行性方面的优势, 展示了 FDIPA 在高可靠航天应用场景的应用潜力.

最后,我们为 AGV 道口分布问题设计了两种启发式算法. 针对 AGV 道口分布问题的无容量限制单货源多商品设施选址问题模型, 分别提出了基于线性规划松弛解的紧问题启发式和邻域搜索启发式方法. 分析表明两种方法均可实施于任意可行的 (FLP) 问题. 数值实验的结果说明了两种启发式方法可以找到比传统的整数规划求解器和 Lagrangian 松弛方法更好的可行解. 对部分问题, 紧问题启发式方法更是可以显著提升求解器的求解效率. 这些结果充分展示了基于线性规划松弛解的启发式方法对 USSMPFLP 的重大意义.

本文所考虑的三个应用优化问题还存在一些未攻克的难点, 这里列举其中一部分并说明我们的一些初步想法.

对于热油管道问题, 本文中的 MINLP 模型和 HOPBB 算法还可以进一步改进提升. 约束 (2.3) 简化了压力和温度间的相互作用. 精确的轴向温降公式应考虑 $T_f^{P_{jr}}$ 随水力摩擦的变化而变化, 但大大提升了 MINLP 问题的难度. 对这样的情形, 可考虑通过分别优化温度和压力变量的方式, 交替求解 MINLP 问题.

对于动力下降制导问题, 主要有两方面亟待解决的难题. 一是带有气动阻力项的问题的无损凸化是否可以在理论上证明是无损的; 二是带有气动阻力项的问题实际是非光滑 NSOCP 问题, 如何针对这类问题设计有收敛性保证的算法, 设计出的算法能否在效率上接近或是超过已有的序列凸化方法.

对于 AGV 道口分布问题的设施选址模型, 在实际计算时我们发现紧问题启发式得到的可行解与最优解十分接近, 如何通过分析近似比或是其它方式说明这一现象的原理. 此外结合邻域搜索的改进方法中, 邻域搜索采取的策略是纯贪婪策略. 通过结合随机性、重启动、路径重连 [11] 等技巧改进纯贪婪的策略以得到更好的可行解也是一个值得探索的方向.

参考文献

- [1] 袁亚湘, 孙文瑜. 最优化理论与方法 [M]. 科学出版社, 1997.
- [2] Nocedal J, Wright S. Numerical optimization [M]. Springer Science and Business Media, 2006.
- [3] 付云姗. 几类应用优化问题的新算法 [D]. 中国科学院研究生院, 2010.
- [4] 刘亚锋. 无线通信中的最优资源分配——复杂性分析与算法设计 [D]. 中国科学院研究生院, 2012.
- [5] 董志龙. 稀疏优化中的若干问题研究 [D]. 中国科学院大学, 2017.
- [6] 傅凯. 三线性约束石油混流问题研究 [D]. 中国科学院大学, 2020.
- [7] 戴彧虹, 刘新为. 线性与非线性规划算法与理论 [J]. 运筹学学报, 2014, 18(1): 69-92.
- [8] 袁亚湘. 非线性优化计算方法 [M]. 科学出版社, 2008.
- [9] Conforti M, Cornuéjols G, Zambelli G, et al. Integer programming: volume 271 [M]. Springer, 2014.
- [10] Dai Y H, Yuan Y. A nonlinear conjugate gradient method with a strong global convergence property [J]. SIAM Journal on Optimization, 1999, 10(1): 177-182.
- [11] Resende M G, Ribeiro C C. Optimization by GRASP [M]. Springer, 2016.
- [12] 吴长春, 严大凡. 热油管道稳态运行的两级递阶优化模型 [J]. 石油学报, 1989, 10(3): 109-117.
- [13] Wang Y, Liu Y, Zhao J, et al. Energy consumption analysis and comprehensive optimization in oil pipeline system [C]//Advanced Materials Research: volume 648. Trans Tech Publ, 2013: 251-254.
- [14] Zuo L, Wu C, Liu S, et al. Predicting monthly energy consumption of crude oil pipelines using process simulation and optimization [C]//2018 12th International Pipeline Conference. American Society of Mechanical Engineers Digital Collection, 2018.
- [15] Liu E, Li C, Yang L, et al. Research on the optimal energy consumption of oil pipeline [J]. Journal of Environmental Biology, 2015, 36(4): 703.
- [16] Zhou M, Zhang Y, Jin S. Dynamic optimization of heated oil pipeline operation using PSO-DE algorithm [J]. Measurement, 2015, 59: 344-351.
- [17] Zhang Y, Wang B, Huang X. Online optimization of heated-oil pipeline operation based on neural network system identification [J]. Journal of Pipeline Systems Engineering and Practice, 2020, 11(1): 04019040.
- [18] Li C, Wang J, Wu X, et al. Operation optimization of heated oil transportation pipeline [M]//

- ICPTT 2011: Sustainable Solutions For Water, Sewer, Gas, And Oil Pipelines. 2011: 733-743.
- [19] 孟振虎, 陈毅忠, 马平. 输油管道运行优化实用分析 [J]. 油气储运, 2002, 21(4): 9-13.
- [20] 宋红波, 杨毅. 原油管道运行优化技术研究 [J]. 西南石油大学学报, 2007(S1): 70-72.
- [21] 高松竹, 汪玉春, 许萍. 混合遗传算法在输油管道优化运行中的应用 [J]. 油气储运, 2004, 7.
- [22] Wang Y, Tian C H, Yan J c, et al. A survey on oil/gas pipeline optimization: problems, methods and challenges [C]//Proceedings of 2012 IEEE International Conference on Service Operations and Logistics, and Informatics. IEEE, 2012: 150-155.
- [23] Darcy H. Recherches expérimentales relatives au mouvement de l'eau dans les tuyaux: volume 1 [M]. Impr. Impériale, 1857.
- [24] Tawarmalani M, Sahinidis N V. A polyhedral branch-and-cut approach to global optimization [J]. Mathematical Programming, 2005, 103: 225-249.
- [25] Misener R, Floudas C A. ANTIGONE: Algorithms for continuous/integer global optimization of nonlinear equations [J]. Journal of Global Optimization, 2014, 59(2-3): 503-526.
- [26] Lin Y, Schrage L. The global solver in the LINDO API [J]. Optimization Methods and Software, 2009, 24(4-5): 657-668.
- [27] Couenne [EB/OL]. <https://www.coin-or.org/Couenne/>.
- [28] Vigerske S, Gleixner A. SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework [J]. Optim. Methods Softw., 2018, 33(3): 563-593.
- [29] Nesterov Y, Nemirovskii A. Interior-point polynomial algorithms in convex programming [M]. SIAM, 1994.
- [30] Nesterov Y E, Todd M J. Self-scaled barriers and interior-point methods for convex programming [J]. Mathematics of Operations research, 1997, 22(1): 1-42.
- [31] Sturm J F. Implementation of interior point methods for mixed semidefinite and second order cone optimization problems [J]. Optimization Methods and Software, 2002, 17(6): 1105-1154.
- [32] Boyd S, Boyd S P, Vandenberghe L. Convex optimization [M]. Cambridge University Press, 2004.
- [33] Acikmese B, Ploen S R. Convex programming approach to powered descent guidance for mars landing [J]. Journal of Guidance, Control, and Dynamics, 2007, 30(5): 1353-1366.
- [34] Blackmore L, Açikmeşe B, Scharf D P. Minimum-landing-error powered-descent guidance for mars landing using convex optimization [J]. Journal of Guidance, Control, and Dynamics, 2010, 33(4): 1161-1171.

- [35] Carson J M, Açikmeşe B, Blackmore L. Lossless convexification of powered-descent guidance with non-convex thrust bound and pointing constraints [C]//Proceedings of the 2011 American Control Conference. IEEE, 2011: 2651-2656.
- [36] Acikmese B, Carson J M, Blackmore L. Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem [J]. IEEE Transactions on Control Systems Technology, 2013, 21(6): 2104-2113.
- [37] Szmuk M, Acikmese B, Berning A W. Successive convexification for fuel-optimal powered landing with aerodynamic drag and non-convex constraints [C]//2016 AIAA Guidance, Navigation, and Control Conference. 2016: 0378.
- [38] Szmuk M, Eren U, Acikmese B. Successive convexification for mars 6-dof powered descent landing guidance [C]//2017 AIAA Guidance, Navigation, and Control Conference. 2017: 1500.
- [39] Szmuk M, Acikmese B. Successive convexification for 6-dof mars rocket powered landing with free-final-time [C]//2018 AIAA Guidance, Navigation, and Control Conference. 2018: 0617.
- [40] Szmuk M, Reynolds T, Acikmese B, et al. Successive convexification for 6-dof powered descent guidance with compound state-triggered constraints [C]//AIAA Scitech 2019 Forum. 2019: 0926.
- [41] Liu X, Shen Z, Lu P. Solving the maximum-crossrange problem via successive second-order cone programming with a line search [J]. Aerospace Science and Technology, 2015, 47: 10-20.
- [42] Liu X, Shen Z, Lu P. Entry trajectory optimization by second-order cone programming [J]. Journal of Guidance, Control, and Dynamics, 2016, 39(2): 227-241.
- [43] Wang Z, Grant M J. Autonomous entry guidance for hypersonic vehicles by convex optimization [J]. Journal of Spacecraft and Rockets, 2018, 55(4): 993-1006.
- [44] Lu P, Liu X. Autonomous trajectory planning for rendezvous and proximity operations by conic optimization [J]. Journal of Guidance, Control, and Dynamics, 2013, 36(2): 375-389.
- [45] Szmuk M, Pascucci C A, Dueri D, et al. Convexification and real-time on-board optimization for agile quad-rotor maneuvering and obstacle avoidance [C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017: 4862-4868.
- [46] Zhang Z, Li J, Wang J. Sequential convex programming for nonlinear optimal control problems in uav path planning [J]. Aerospace Science and Technology, 2018, 76: 280-290.
- [47] Liu X, Shen Z, Lu P. Exact convex relaxation for optimal flight of aerodynamically controlled missiles [J]. IEEE Transactions on Aerospace and Electronic Systems, 2016, 52(4): 1881-1892.

- [48] Liu X, Lu P, Pan B. Survey of convex optimization for aerospace applications [J]. *Astrodynamics*, 2017, 1(1): 23-40.
- [49] Daskin M S. What you should know about location modeling [J]. *Naval Research Logistics (NRL)*, 2008, 55(4): 283-294.
- [50] Klose A, Drexl A. Facility location models for distribution system design [J]. *European Journal of Operational Research*, 2005, 162(1): 4-29.
- [51] Arya V, Garg N, Khandekar R, et al. Local search heuristics for k-median and facility location problems [J]. *SIAM Journal on Computing*, 2004, 33(3): 544-562.
- [52] Fisher M L. The Lagrangian relaxation method for solving integer programming problems [J]. *Management Science*, 1981, 27(1): 1-18.
- [53] Zhang J, Chen B, Ye Y. A multiexchange local search algorithm for the capacitated facility location problem [J]. *Mathematics of Operations Research*, 2005, 30(2): 389-403.
- [54] Sun M. Solving the uncapacitated facility location problem using tabu search [J]. *Computers and Operations Research*, 2006, 33(9): 2563-2589.
- [55] Chudak F A, Shmoys D B. Improved approximation algorithms for the uncapacitated facility location problem [J]. *SIAM Journal on Computing*, 2003, 33(1): 1-25.
- [56] Huang H C, Li R. A k-product uncapacitated facility location problem [J]. *European Journal of Operational Research*, 2008, 185(2): 552-562.
- [57] Mazzola J B, Neebe A W. Lagrangian-relaxation-based solution procedures for a multiproduct capacitated facility location problem with choice of facility type [J]. *European Journal of Operational Research*, 1999, 115(2): 285-299.
- [58] Nezhad A M, Manzour H, Salhi S. Lagrangian relaxation heuristics for the uncapacitated single-source multi-product facility location problem [J]. *International Journal of Production Economics*, 2013, 145(2): 713-723.
- [59] Geoffrion A M, Graves G W. Multicommodity distribution system design by benders decomposition [J]. *Management Science*, 1974, 20(5): 822-844.
- [60] Lee C Y. A cross decomposition algorithm for a multiproduct-multitype facility location problem [J]. *Computers and Operations Research*, 1993, 20(5): 527-540.
- [61] Klincewicz J G, Luss H, Rosenberg E. Optimal and heuristic algorithms for multiproduct uncapacitated facility location [J]. *European Journal of Operational Research*, 1986, 26(2): 251-258.
- [62] Karkazis J, Boffey T. The multi-commodity facilities location problem [J]. *Journal of the Operational Research Society*, 1981, 32(9): 803-814.
- [63] Klincewicz J G, Luss H. A dual-based algorithm for multiproduct uncapacitated facility location [J]. *Transportation Science*, 1987, 21(3): 198-206.

- [64] Montoya A, Vélez-Gallego M C, Villegas J G. Multi-product capacitated facility location problem with general production and building costs [J]. NETNOMICS: Economic Research and Electronic Networking, 2016, 17(1): 47-70.
- [65] Goldberg N, Leyffer S. An active-set method for second-order conic-constrained quadratic programming [J]. SIAM Journal on Optimization, 2015, 25(3): 1455-1477.
- [66] Monteiro R D, Tsuchiya T. Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions [J]. Mathematical Programming, 2000, 88(1): 61-83.
- [67] Tsuchiya T. A convergence analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming [J]. Optimization Methods and Software, 1999, 11(1-4): 141-182.
- [68] Toh K C, Todd M J, Tütüncü R H. SDPT3-a MATLAB software package for semidefinite programming, version 1.3 [J]. Optimization Methods and Software, 1999, 11(1-4): 545-581.
- [69] Sturm J F. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones [J]. Optimization Methods and Software, 1999, 11(1-4): 625-653.
- [70] Domahidi A, Chu E, Boyd S. ECOS: An SOCP solver for embedded systems [C]//2013 European Control Conference (ECC). IEEE, 2013: 3071-3076.
- [71] Lobo M S, Vandenberghe L, Boyd S, et al. Applications of second-order cone programming [J]. Linear Algebra and Its Applications, 1998, 284(1-3): 193-228.
- [72] Liu X, Shen Z, Lu P. Entry trajectory optimization by second-order cone programming [J]. Journal of Guidance, Control, and Dynamics, 2015, 39(2): 227-241.
- [73] Kato H, Fukushima M. An SQP-type algorithm for nonlinear second-order cone programs [J]. Optimization Letters, 2007, 1(2): 129-144.
- [74] Zhang X, Liu Z, Liu S. A trust region SQP-filter method for nonlinear second-order cone programming [J]. Computers and Mathematics with Applications, 2012, 63(12): 1569-1576.
- [75] Okuno T, Yasuda K, Hayashi S. SIIQP based algorithm with trust region technique for solving nonlinear second-order cone programming problems [J]. Interdisciplinary Information Sciences, 2015, 21(2): 97-107.
- [76] Liu Y J, Zhang L W. Convergence analysis of the augmented Lagrangian method for nonlinear second-order cone optimization problems [J]. Nonlinear Analysis: Theory, Methods and Applications, 2007, 67(5): 1359-1373.
- [77] Liu Y J, Zhang L W. Convergence of the augmented Lagrangian method for nonlinear optimization problems over second-order cones [J]. Journal of Optimization Theory and Applications, 2008, 139(3): 557-575.

- [78] Fukuda E H, Silva P J, Fukushima M. Differentiable exact penalty functions for nonlinear second-order cone programs [J]. *SIAM Journal on Optimization*, 2012, 22(4): 1607-1633.
- [79] Yamashita H, Yabe H. A primal–dual interior point method for nonlinear optimization over second-order cones [J]. *Optimization Methods and Software*, 2009, 24(3): 407-426.
- [80] Kanzow C, Ferenczi I, Fukushima M. On the local convergence of semismooth Newton methods for linear and nonlinear second-order cone programs without strict complementarity [J]. *SIAM Journal on Optimization*, 2009, 20(1): 297-320.
- [81] Fukuda E H, Fukushima M. The use of squared slack variables in nonlinear second-order cone programming [J]. *Journal of Optimization Theory and Applications*, 2016, 170(2): 394-418.
- [82] Canelas A, Carrasco M, López J. A feasible direction algorithm for nonlinear second-order cone programs [J]. *Optimization Methods and Software*, 2019, 34(6): 1322-1341.
- [83] Zhao Q, Chen Z. A line search penalty-free method for nonlinear second-order cone programming [J]. *Acta Applicandae Mathematicae*, 2020, 170: 291-317.
- [84] Gilmore P C, Gomory R E. A linear programming approach to the cutting-stock problem [J]. *Operations Research*, 1961, 9(6): 849-859.
- [85] Gilmore P C, Gomory R E. A linear programming approach to the cutting stock problem—Part II [J]. *Operations research*, 1963, 11(6): 863-888.
- [86] Bertsekas D P. Nonlinear programming [J]. *Journal of the Operational Research Society*, 1997, 48(3): 334-334.
- [87] Kelley J E. The cutting-plane method for solving convex programs [J]. *Journal of the Society for Industrial and Applied Mathematics*, 1960, 8(4): 703-712.
- [88] Gropp W, Moré J J. Optimization environments and the NEOS server [M]//Buhman M D, Iserles A. *Approximation Theory and Optimization*. Cambridge University Press, 1997: 167-182.
- [89] Wächter A, Biegler L T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming [J]. *Mathematical Programming*, 2006, 106(1): 25-57.
- [90] CPLEX optimizer [EB/OL]. <https://www.ibm.com/analytics/cplex-optimizer>.
- [91] MINOS [EB/OL]. http://www.sbsi-sol-optimize.com/asp/sol_product_minos.htm.
- [92] CONOPT [EB/OL]. <http://www.conopt.com/>.
- [93] Herskovits J. Feasible direction interior-point technique for nonlinear optimization [J]. *Journal of Optimization Theory and Applications*, 1998, 99(1): 121-146.
- [94] Bonnans J F, Shapiro A. *Perturbation analysis of optimization problems* [M]. Springer Science and Business Media, 2013.
- [95] Horn R, Johnson C. *Matrix analysis* [M]. Cambridge University Press, 1990.

-
- [96] Grant M, Boyd S. CVX: MATLAB software for disciplined convex programming, version 2.1 [Z]. 2014.
- [97] Grant M C, Boyd S P. Graph implementations for nonsmooth convex programs [M]//Recent advances in learning and control. Springer, 2008: 95-110.
- [98] Wächter A, Biegler L T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming [J]. Mathematical Programming, 2006, 106(1): 25-57.
- [99] Powell M J D. A fast algorithm for nonlinearly constrained optimization calculations [C]//Watson G A. Numerical Analysis. Berlin, Heidelberg: Springer Berlin Heidelberg, 1978: 144-157.
- [100] Gurobi optimizer [EB/OL]. <http://www.gurobi.com/>.
- [101] Achterberg T. SCIP: solving constraint integer programs [J]. Mathematical Programming Computation, 2009, 1(1): 1-41.

作者简历及攻读学位期间发表的学术论文与研究成果

作者简历

杨沐明, 男, 天津市人, 1993 年 9 月生.

2012 年 9 月至 2016 年 7 月, 南开大学数学科学学院, 获学士学位.

2016 年 9 月至 2021 年 7 月, 中国科学院数学与系统科学研究院计算数学所, 攻读博士学位.

已发表的学术论文:

1. Chen W K, Chen L, Yang M M, Dai Y H. Generalized coefficient strengthening cuts for mixed integer programming [J]. Journal of Global Optimization, 2018, 70(1): 289-306.
2. 杨沐明, 黄亚魁, 戴或虹. 一类多商品设施选址问题的基于线性松弛解的启发式方法 [J]. 运筹学学报, 2019, 23(3): 15.
3. Yang M, Huang Y, Dai Y H, Li B. An efficient global optimization algorithm for heated oil pipeline problems [J]. Industrial and Engineering Chemistry Research, 2020, 59(14): 6638-6649.
4. 杨沐明, 攸国攸. 动力下降制导问题的两阶段序列凸化方法 [J]. 中国科学: 数学, 2020, 50(9): 234-247.
5. Chen L, Chen W K, Yang M M, Dai Y H. An exact separation algorithm for unsplittable flow capacitated network design arc-set polyhedron [J]. Journal of Global Optimization, 2021: 1-31.

学术报告:

1. Solving Heated Oil Pipeline Problems Via Mixed Integer Nonlinear Programming Approach, 第二届北京大学计算与应用数学拔尖博士生研讨会, 邀请报告, 2019.

致 谢

时光荏苒, 岁月如梭, 五年的硕博生涯即将结束. 正值论文完成之际, 谨在此向一路陪我走来的师长、朋友和家人们表示衷心的感谢.

首先, 我要感谢我的导师戴或虹研究员. 感谢戴老师五年来对我的指导和关心. 初见时, 戴老师的和蔼可亲和平易近人给我留下很深刻的印象. 在入学面试时, 戴老师并没有因为我的学习成绩不理想而把我拒之门外, 这令我非常感动, 也让我下定决心在研究生期间学好知识, 做好科研. 戴老师为我提供了广阔的学习和展示自我的舞台, 引导我了解最新最前沿的整数规划相关的成果和技术, 创造接触各种应用优化问题的机会, 并资助我到国外知名大学向知名教授学习知识、锻炼能力. 在学习和科研之外, 戴老师也非常关心我的心理成长. 我也常常在自省时想起戴老师对我的教导. 此外, 戴老师对慈善事业的贡献和无私的精神永远值得我去学习. 正是您这五年来对我的教育和影响, 使我进步使我成长. 衷心感谢您为我做的一切.

感谢课题组的袁亚湘院士. 袁老师的学识和口才令我十分钦佩. 在讨论班上, 袁老师常常教导我们要看清问题的本质, 思考科研工作的意义, 这让我十分受益. 袁老师组织的爬山、打牌活动丰富了我们科研之外的生活, 让我感受到课题组大家庭的温暖.

感谢课题组的刘歆研究员. 刘老师对知识的一丝不苟永远是我学习的榜样. 在怀柔校区学习时我曾有幸聆听刘老师的两节优化课程. 刘老师授课时的风趣幽默令我印象深刻.

感谢课题组的刘亚锋副研究员. 您勤奋的工作态度和乐观的生活态度对我影响很大. 感谢您在这五年里对我如兄长般的关心和帮助.

感谢新加坡国立大学的 Christine A. Shoemaker 教授. 感谢您在我到新加坡交流期间在课题研究上的指导和生活上提供的便利. 感谢 Songhao Wang 博士等对我在科研方面的帮助. 感谢新加坡国立大学的 Wei Lu 博士和数学院的李林秋同学和董晓宇师妹. 在新冠疫情期间, 是你们的陪伴和帮助让我在异国他乡度过了那段艰难的日子.

感谢课题组的崔春风师姐、王树雄师兄、董乾师兄、董志龙师兄、康冬师姐、

顾然师兄、曾燎原师姐、金哲师兄、陈诚师兄、陈伟坤师兄、高斌师兄、赵亮师兄、周睿智师姐、陈亮师兄、傅凯师兄、王小玉师姐、张瑞师兄、肖纳川、吴宇宸、陈雅丹、吉振远师弟、张瑞进师弟、赵浩天师弟、刘为师弟、陈圣杰师弟、张吾帅君师弟、姜博鸥师妹、黄磊师弟、汪思维师弟、裴骞师弟、王圣超师弟、张跃师弟、赵成师弟、谢鹏程师弟、陈硕师弟、王磊师弟、胡雨宽师弟、武哲宇师妹, 以及博士后李志保师兄、黄亚魁师兄、张国涵师兄、黄娜师姐、张婷师姐、刘泽显师兄、戴金雨师兄.

感谢同我一起在计算数学所学习的曲晓昆、贾云飞、于文洋、邝超智、杨斌、付姚姚、张晓迪、徐岚、姜姗、金鹏展、刘世伟、金典聪、桂升、徐孜立、李翊君、王晓荣、杨鹏飞. 感谢应用所的徐汾同学做我的答辩秘书.

最后我要感谢我的父母. 感谢你们多年的养育之恩. 因为有你们的辛苦工作我才能过上衣食无忧的生活. 因为有你们的默默支持我才能走完这 21 年的求学之路. 每当我沮丧、气馁的时候, 是你们的关怀和鼓励使我有继续前进的动力. 感谢刘喆, 你的包容和陪伴是我最宝贵的财富. 希望未来我们一切都好.