

# 控制与决策

Control and Decision

## 三阶段拉格朗日启发式算法求解带同时取送货的绿色车辆路径问题

李熠胥, 胡蓉, 吴绍云, 于乃康, 钱斌

引用本文:

李熠胥, 胡蓉, 吴绍云, 于乃康, 钱斌. 三阶段拉格朗日启发式算法求解带同时取送货的绿色车辆路径问题[J]. 控制与决策, 2023, 38(12): 3525–3533.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2022.1825>

---

### 您可能感兴趣的其他文章

#### Articles you may be interested in

#### [带不相关并行机和有限缓冲MHFS调度的混合启发式算法](#)

Hybrid heuristic algorithm for multi-stage hybrid flow shop scheduling with unrelated parallel machines and finite buffers  
控制与决策. 2021, 36(3): 565–576 <https://doi.org/10.13195/j.kzyjc.2019.0835>

#### [超启发式交叉熵算法求解模糊分布式流水线绿色调度问题](#)

Hyper-heuristic cross-entropy algorithm for green distributed permutation flow-shop scheduling problem with fuzzy processing time  
控制与决策. 2021, 36(6): 1387–1396 <https://doi.org/10.13195/j.kzyjc.2019.1681>

#### [基于改进NSGA-II算法求解多目标资源受限项目调度问题](#)

An improved NSGA-II algorithm for multi-objective resource-constrained project scheduling problem  
控制与决策. 2021, 36(3): 669–676 <https://doi.org/10.13195/j.kzyjc.2019.0906>

#### [基于动态行为选择的和声搜索算法](#)

Harmony search algorithm based on dynamic behavior selection  
控制与决策. 2021, 36(3): 577–588 <https://doi.org/10.13195/j.kzyjc.2019.0597>

#### [考虑卸载顺序约束的成品油二次配送车辆路径问题](#)

Vehicle routing problem of refined oil secondary distribution considering unloading sequence constraints  
控制与决策. 2020, 35(12): 2999–3005 <https://doi.org/10.13195/j.kzyjc.2018.1756>

# 三阶段拉格朗日启发式算法求解带同时取送货的 绿色车辆路径问题

李熠胥<sup>1</sup>, 胡蓉<sup>1†</sup>, 吴绍云<sup>2</sup>, 于乃康<sup>3</sup>, 钱斌<sup>1</sup>

(1. 昆明理工大学 信息工程与自动化学院, 昆明 650504; 2. 玉溪思润印刷有限公司, 云南 玉溪 653100; 3. 昆明理工大学 机电工程学院, 昆明 650504)

**摘要:** 针对带同时取送货的绿色车辆路径问题, 以最小化带碳排放费用的配送成本为优化目标, 建立混合整数规划模型, 并提出一种结合数学规划方法与启发式算法的三阶段拉格朗日启发式算法进行求解. 第 1 阶段, 利用拉格朗日松弛技术得到该问题的拉格朗日对偶模型; 第 2 阶段, 设计一种改进的次梯度算法迭代求解该对偶模型, 同时引入修复机制, 将每次迭代所得下界对应的解修复为原问题较高质量的可行解, 并在下次迭代中利用该可行解更新次梯度方向和步长; 第 3 阶段, 设计一种启发式局部搜索算法, 对第 2 阶段得到的可行解进行优化, 进一步改进解的质量, 以得到原问题的近似最优解. 实验表明, 所提出算法能够获得问题的一个优质解, 同时提供一个紧致下界, 用以定量评估解的质量.

**关键词:** 绿色车辆路径问题; 同时取送货; 拉格朗日启发式; 可行解修复; 松弛技术; 问题下界

中图分类号: TP18 文献标志码: A

DOI: 10.13195/j.kzyjc.2022.1825

**引用格式:** 李熠胥, 胡蓉, 吴绍云, 等. 三阶段拉格朗日启发式算法求解带同时取送货的绿色车辆路径问题[J]. 控制与决策, 2023, 38(12): 3525-3533.

## Three-stage Lagrangian heuristic algorithm for solving green vehicle routing problem with simultaneous pickup and delivery

LI Yi-xu<sup>1</sup>, HU Rong<sup>1†</sup>, WU Shao-yun<sup>2</sup>, YU Nai-kang<sup>3</sup>, QIAN Bin<sup>1</sup>

(1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650504, China; 2. Yuxi Sirun Printing Co., Ltd, Yuxi 653100, China; 3. Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650504, China)

**Abstract:** For the green vehicle routing problem with simultaneous pickup and delivery, a mixed integer programming model is developed with the optimization objective of minimizing the distribution cost with carbon emission cost, and a three-stage Lagrangian heuristic algorithm combining mathematical programming methods and heuristics is proposed for solving the problem. In the first stage, the Lagrangian dual model of the problem is obtained using Lagrangian relaxation technique. In the second stage, an improved sub-gradient algorithm is designed to solve the dual model iteratively, and a repair mechanism is introduced to repair the solution corresponding to the lower bound obtained in each iteration to a high-quality feasible solution of the original problem. The feasible solution is then used to update the sub-gradient direction and step size in the next iteration. In the third stage, a heuristic local search algorithm is designed to optimize the feasible solution obtained in the second stage and further improve the quality of the solution to obtain the approximate optimal solution of the original problem. Experiments show that the proposed algorithm can obtain a high-quality solution to the problem and provide a compact lower bound to evaluate the quality of the solution.

**Keywords:** green vehicle routing problem; simultaneous pickup and delivery; Lagrangian heuristic; feasible solution repair; relaxation technique; lower bound

## 0 引言

车辆路径问题(vehicle routing problem, VRP)由 Dantzig 等<sup>[1]</sup>于 1959 年首次提出, 此后得到了广泛

关注, 成为组合优化领域的前沿研究热点. 面对多元化的运输需求, 基本送达服务已无法满足企业发展和大众的需求, 例如在实际生活中, 快递员既要

收稿日期: 2022-10-22; 录用日期: 2023-04-14.

基金项目: 国家自然科学基金项目(61963022, 62173169); 云南省基础研究重点项目(202201AS070030).

责任编辑: 刘士新.

<sup>†</sup>通讯作者. E-mail: ronghu@vip.163.com.

将快递送达客户,又要从客户处带回需要邮寄的快递.因此,考虑同时取送货车车辆路径问题(VRP with simultaneous pickup and delivery, VRPSPD)的研究是迫切且必要的.此外,随着温室气体排放量急剧增加,环境污染问题引起社会高度关注.据统计,以二氧化碳为主的污染物很大一部分来源于货物运输<sup>[2]</sup>.在此背景下,考虑油耗和碳排放的绿色车辆路径问题(green VRP, GVRP)<sup>[3]</sup>一经提出,便得到了广泛关注.带同时取送货的绿色车辆路径问题(green VRPSPD, GVRPSPD)综合了VRPSPD和GVRP的约束条件,在满足客户取送货需求的同时,降低了运输过程中的能源消耗,减少了碳排放量.因VRP问题属于NP-hard问题,而VRP又可归类为GVRPSPD,故GVRPSPD也属于NP-hard<sup>[4]</sup>问题.因此,研究GVRPSPD的建模及其求解算法具有重要的理论价值和实际意义.

近年来,针对VRPSPD、GVRP等各类复杂VRP,得到了较多的关注与研究.Min<sup>[5]</sup>于1989年首次提出VRPSPD,随后,诸多学者致力于建立VRPSPD的数学模型<sup>[6-8]</sup>并设计各种有效的求解算法<sup>[9-11]</sup>.GVRP目前主要有3种类型,即污染路径问题(pollution-routing problem, PRP)、能源最小化车辆路径问题(energy minimizing VRP, EMVRP)和低碳车辆路径问题(low-carbon VRP, LCVRP).其中:PRP由Bektaş等首次提出<sup>[12]</sup>,该问题在减少行驶距离、时间等运营成本的基础上,同时考虑了降低温室气体排放量、燃料消耗等环境成本<sup>[13-15]</sup>;EMVRP由Kara等<sup>[16-17]</sup>首次提出,该问题可以视为特定情况下的PRP,其优化目标是使所有车辆的负载与行驶距离乘积之和最小,即利用车辆负载来最小化燃料消耗或碳排放<sup>[18]</sup>;LCVRP考虑车辆行驶过程中的碳排放,以最小化碳排放成本、行驶成本等组成的总运输成本为优化目标,许多GVRP都可归类为LCVRP.Demir等<sup>[19]</sup>综述并比较了多个油耗和碳排放模型<sup>[20-23]</sup>,验证了车辆负载和行驶速度是对碳排放量影响最大的关键因素.后续针对LCVRP的研究大都在考虑这两个因素的基础上进行扩展<sup>[24-28]</sup>.

通过文献调研可知,针对各类复杂VRP模型的求解算法大致可分为启发式算法<sup>[5,24,26]</sup>、智能算法<sup>[9-10,11,14-15,28]</sup>和数学规划方法<sup>[7-8,12-13,16-18,25,27]</sup>.启发式算法基于经验、规则和邻域等来构造问题的解,可在短时间内获取较优解.智能算法包括遗传算法、微粒群算法、蚁群算法和模拟退火算法等,该类算法采用某种拟人、拟物机制不断生成新的可行个体或

解,从而实现对问题解空间的搜索,可在合理时间内获得问题的优质解.数学规划方法包括列生成、分支定界和拉格朗日松弛等,该类算法利用数学规划模型的优化目标函数和约束条件式的代数几何结构特性进行设计,以获取问题的解(列生成和分支定界)或下界(拉格朗日松弛).其中,列生成和分支定界通过构造遍历或部分遍历解空间的搜索来获取问题的解,可在一定时间内获取较小规模问题的近似最优或最优解,但求解较大规模问题时往往时间很长(求解时间随问题规模线性增大而指数增长).拉格朗日松弛用于获取问题的下界,经过合理设计,可较快获取不同规模问题的较紧下界.

上述算法中,启发式算法和智能算法求解问题的时间较短且可控,但这两类算法所得解的质量无法被定量评价.因此,为有效求解复杂组合优化问题并客观评价求解质量,可将拉格朗日松弛与启发式算法结合.拉格朗日启发式算法就是在这一背景下形成的一种混合算法.该算法既能获得下界用以评价解的质量,又能在较短时间内获得高质量可行解.目前,拉格朗日启发式算法已成功地用于求解包含VRP在内的多种组合优化问题<sup>[29-31]</sup>.但由文献调研可知,此类算法求解复杂VRP的相关研究仍较少,且还存在一些开放问题.例如:如何有效实现下界对应的非可行解(以下简称下界解)到原问题可行解之间的转换;如何利用下界解中的有效信息帮助启发式算法获取更高质量的可行解.因此,本文尝试设计一种高效的拉格朗日启发式算法,以获取较紧下界并实现下界解到可行解的转换,同时利用下界解中的有效信息来提升启发式搜索的效率.

本文在考虑碳排放和同时取送货的情况下,以最小化带碳排放费用的配送成本为优化目标,构建GVRPSPD的混合整数规划模型(mixed integer programming, MIP),并提出一种三阶段拉格朗日启发式算法(three-stage Lagrangian heuristic algorithm, TSLHA)进行求解.首先,在第1阶段采用改进拉格朗日松弛技术获取原问题的拉格朗日对偶模型;然后,在第2阶段设计一种改进次梯度算法迭代更新该对偶模型的拉格朗日乘子,得到较紧的下界,同时引入修复机制,将下界解修复为可行解,并在后续次梯度算法迭代中利用可行解更新次梯度方向和步长;最后,在第3阶段设计一种高效的启发式局部搜索算法,对第2阶段得到的可行解进行改进,以得到原问题的近似最优解.通过对不同测试问题的仿真实验和算法比较验证了TSLHA的有效性.

# 1 问题与模型

## 1.1 问题描述和符号定义

GVRPSPD描述如下: 整个配送网络有一个配送中心及若干具有不同取送货需求的客户, 一批同质车辆从配送中心出发对所有客户进行服务, 满足客户的取送货需求, 服务完成后返回配送中心<sup>[32]</sup>. GVRPSPD的相关符号释义如表1所示.

表1 符号及释义

符号	释义
$V$	节点集, $V = \{0, 1, \dots, n\}$ (0表示仓库)
$V_0$	客户点集, $V_0 = \{1, 2, \dots, n\}$
$K$	车辆集, $K = \{1, 2, \dots, m\}$
$dist_{ij}$	点 <i>i</i> 与点 <i>j</i> 之间的欧氏距离; $i \in V, j \in V, i \neq j$
$P_i$	点 <i>i</i> 处的取货量, $i \in V_0$
$D_i$	点 <i>i</i> 处的送货量, $i \in V_0$
$c$	单位距离行驶成本
$Q$	车辆最大载重
$b_i$	用于消除子环约束的变量, $i \in V$
$y_{ijk}$	车辆 <i>k</i> 的在途载货量; $i, j \in V, k \in K$
$x_{ijk}$	决策变量, 车辆 <i>k</i> 从客户 <i>i</i> 到客户 <i>j</i> 时为1, 否则为0

## 1.2 碳排放模型

在本文问题中, 车辆需要在客户处取货, 故车辆在行驶全程均接近满载, 车辆载重与碳排放之间关联较大, 因此, 采用文献[33]的碳排放计算方法. 碳排放主要包含车辆在运输过程中自身产生的二氧化碳和由于车辆负载产生的二氧化碳. 具体计算公式如下:

$$Z_{CO_2} = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} dist_{ij} \omega_c E_{fuel} p \frac{A_{veh}}{\mu} x_{ijk} + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} dist_{ij} \omega_c E_{fuel} p y_{ijk}. \quad (1)$$

各参数具体取值如表2所示.

表2 碳排放计算相关参数

参数	定义	取值
$\omega_c$	每升燃油所含二氧化碳的重量/t	0.0027
$E_{fuel}$	车辆单位重量单位距离耗油量/(L/t/km)	0.01653
$A_{veh}$	整车质量/t	16.2
$\mu$	车辆质量利用系数	5.0175
$p$	排放单位二氧化碳产生的成本	20

## 1.3 模型构建

建立GVRPSPD的MIP模型如下:

$$\min Z = \sum_{i \in V_0} \sum_{j \in V_0} \sum_{k \in K} dist_{ij} x_{ijk} c + Z_{CO_2}. \quad (2)$$

$$\text{s.t.} \sum_{i \in V} \sum_{k \in K} x_{ijk} = \sum_{i \in V} \sum_{k \in K} x_{jik} = 1, \forall j \in V_0; \quad (3)$$

$$\sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} = 0, i = j; \quad (4)$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0, \forall j \in V_0, k \in K; \quad (5)$$

$$\sum_{j \in V_0} x_{0jk} = \sum_{i \in V_0} x_{i0k} = 1, \forall k \in K; \quad (6)$$

$$b_j \geq b_i + 1 - (n - 1) \left( 1 - \sum_{k \in K} x_{ijk} \right),$$

$$0 \leq i \leq n, 1 \leq j \leq n + 1; \quad (7)$$

$$y_{ijk} \leq x_{ijk} Q, \forall i \in V, j \in V, k \in K; \quad (8)$$

$$\sum_{i \in V} \sum_{j \in V_0} D_j x_{ijk} = \sum_{j \in V_0} y_{0jk}, \forall k \in K; \quad (9)$$

$$\sum_{i \in V} y_{ijk} + (P_j - D_j) \sum_{i \in V} x_{ijk} = \sum_{i \in V} y_{jik},$$

$$\forall j \in V_0, k \in K; \quad (10)$$

$$x_{ijk} \in \{0, 1\}, \forall i \in V, j \in V, k \in K; \quad (11)$$

$$y_{ijk} \geq 0, \forall i \in V, j \in V, k \in K. \quad (12)$$

其中: 式(2)为模型的目标函数, 以最小化带碳排放费用的总配送成本为优化目标; 式(3)确保各客户点仅由一辆车进行服务; 式(4)防止车辆在一个节点处循环; 式(5)为流量平衡约束; 式(6)确保每辆车都从配送中心出发, 并在配送完成后返回配送中心; 式(7)为子环消除约束; 式(8)表明当车辆*k*经过弧(*i*, *j*)时(即  $x_{ijk} = 1$ ), 其载重  $y_{ijk}$  不能超过车辆最大载重  $Q$ ; 式(9)确保车辆从配送中心出发时的载重等于其行驶路线上所有客户的送货需求之和; 式(10)表示车辆服务完某一客户后的载重量; 式(11)表示决策变量为0-1变量; 式(12)表示变量  $y_{ijk}$  非负.

## 2 三阶段拉格朗日启发式算法

本文TSLHA分为3个阶段: 基于拉格朗日松弛的对偶模型获取(2.1节); 基于对偶模型的下界及可行解获取(2.2节); 基于局部搜索的可行解优化(2.3节).

### 2.1 基于拉格朗日松弛的对偶模型获取

目前, 获取下界的主流方法主要有拉格朗日松弛和线性松弛. 通过文献调研可知, 采用拉格朗日松弛得到的下界通常比线性松弛得到的下界更紧<sup>[34-35]</sup>, 因此, 采用改进拉格朗日松弛技术对1.3节中的模型进行松弛. 在原模型中, 部分约束被称为“复杂约束”, 这些约束将所有的车辆信息结合在一起, 使得模型很难求解. 对于GVRPSPD, 由于车辆在为各个客户点送货的同时还要从客户点处取货, 车辆在行驶全程载货量均接近满载. 车辆在对各个客户点进行取送货服务时, 需考虑取货之后的载货量是否满足载重约束. 由此可见, 载重约束对目标函数的求解影响较大, 基于此, 本节将原模型中的载重约束(式(8))通过拉格朗日乘子  $\lambda$  以惩罚项的形式添加到目标函数中, 得到拉格朗日对偶模型如下:

$$D_2 = \max_{\lambda \in R_+^m} \min_{x \in X} Z(\lambda) = \max \left( \min Z + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} \lambda^{ijk} (y_{ijk} - x_{ijk} Q) \right).$$

s.t. 式(3) ~ (7), (9) ~ (12). (13)

## 2.2 基于对偶模型的下界及可行解获取

本阶段获取原问题下界及可行解. 设计一种改进次梯度算法迭代求解2.1节拉格朗日对偶模型, 获取原问题的拉格朗日下界. 同时引入修复机制, 设计一种启发式修复算法, 将每次迭代得到的下界解修复为可行解, 并在后续迭代中利用可行解更新次梯度方向和步长, 从而驱动算法搜索过程向可行域逼近.

### 2.2.1 改进次梯度算法

本节设计一种改进的次梯度算法, 在迭代过程中利用动态的可行解更新次梯度方向和步长.

设  $x_\lambda$  是拉格朗日松弛问题  $Z(\lambda)$  的最优解, 则  $\xi = y - x_\lambda Q$  是  $Z(\lambda)$  在  $\lambda$  处的次梯度. 通过如下方式进行迭代:

$$\lambda^{k+1} = \lambda^k + s_k \xi^k. \quad (14)$$

其中:  $\lambda^{k+1}$  是在第  $k+1$  次迭代时, 拉格朗日乘子的值;  $s_k$  表示沿次梯度方向的一个步长因子;  $\xi^k$  是  $Z(\lambda)$  在  $\lambda^k$  处的次梯度.

为防止迭代路线上下波动(即 Zig-Zag 现象<sup>[36]</sup>), 采用偏转次梯度  $\varphi^k$  代替  $\lambda^k$  点的次梯度  $\xi^k$ <sup>[37]</sup>, 有

$$\varphi^k = \xi^k + \theta_k \varphi^{k-1}, \quad \theta_k \geq 0. \quad (15)$$

其中:  $\theta_k$  是偏转系数, 当前的迭代方向  $\varphi^k$  是次梯度  $\xi^k$  与上次迭代方向  $\varphi^{k-1}$  的线性组合. 新的迭代计算公式如下:

$$\lambda^{k+1} = \lambda^k + s_k \varphi^k. \quad (16)$$

为使  $\lambda^k$  尽可能接近最优解  $\lambda^*$ , 受梯度算法中最优一维搜索的启示, 搜索方向需与上次搜索方向正交, 采用文献[38]中偏转系数  $\theta_k$  的取值, 有

$$\theta_k = \frac{(\xi^k)^T \varphi^{k-1}}{\|\varphi^{k-1}\|^2} = \frac{(Z(\lambda^*) - Z(\lambda^k))^T \varphi^{k-1}}{\|\varphi^{k-1}\|^2}, \quad (17)$$

其中公式推导过程见文献[38].

此外, 在迭代过程中, 算法的步长也会影响算法性能, 步长较大导致算法稳定性较差, 步长较小则会导致算法过早收敛. 设计一种步长更新方式如下:

$$s_k = \frac{\alpha_k (Z(\lambda^*) - Z(\lambda^k))}{\|\varphi^k\|^2}, \quad (18)$$

其中  $\alpha_k$  是一个适当的正参数, 取  $1 \leq \alpha_k < 2$ . 在算法迭代时, 若  $Z(\lambda)$  上升, 则  $\alpha_k$  取值不变; 若  $Z(\lambda)$  在给定

的若干次迭代没有变化, 则取其一半.  $Z(\lambda^*)$  表示原问题最优解, 即  $Z(\lambda)$  的上界.

一般情况下, 上界  $Z(\lambda^*)$  是未知的, 因此, 在算法第1次迭代时, 取 Gurobi 求解器在规定时间内得到的解近似替代. 而在后续迭代中, 为避免次梯度方向偏离可行域方向, 利用下面2.2.3节中修复操作得到的解近似替代  $Z(\lambda^*)$ . 因每次迭代过程中, 修复操作得到的可行解不同, 故次梯度的方向和步长会随可行解的变化而做出相应调整, 从而驱动算法搜索不断地向可行域逼近. 改进次梯度算法步骤如下.

step 1: 令次梯度  $\xi^{ijk} = y_{ijk} - x_{ijk} Q$ , 取 Gurobi 求解器在有效时间内获得的原问题解近似代替  $Z(\lambda^*)$ , 拉格朗日乘子  $\lambda^1$  等于  $(0, 0.1)$  中的随机数,  $k = 1$ .

step 2: 利用式(15)和(17)求迭代方向, 利用式(18)求步长, 利用式(16)更新拉格朗日乘子.

step 3: 获取下界, 并取2.2.3节中修复操作获得的可行解替代  $Z(\lambda^*)$ .

step 4: 判断是否满足次梯度算法的终止条件. 若满足, 则输出最终下界; 否则,  $k = k + 1$ , 转至 step 2.

### 2.2.2 数学规划解到排序解的转换

在引入修复机制将下界解转换为可行解之前, 需对下界的数学规划解进行编/解码操作转换为排序解. 假设所求数学规划解  $x_{0j_1k_1} = 1, x_{j_1j_2k_1} = 1, x_{j_2j_3k_1} = 1, x_{j_30k_1} = 1$ , 则对应的排序解为  $\pi_1 = \{0, j_1, j_2, j_3, 0\}$ , 表示车辆  $k_1$  从配送中心出发, 依次服务完客户  $j_1, j_2, j_3$  后, 返回配送中心.

### 2.2.3 修复操作

本节将2.2.1节中每一次改进次梯度算法迭代后获得的下界解修复为原问题可行解. 上文中所得拉格朗日下界解是通过松弛载重约束求得, 由于缺少载重约束, 在使用次梯度算法对对偶模型的目标函数进行优化时, 优化过程不再考虑车辆是否超载, 这导致各点之间的距离成为优化过程中最直接的影响因素, 从而使得拉格朗日下界对应的解中相邻两点间的距离可能较短. 针对这一特点, 并结合同时取送货问题的特征, 提出一种启发式修复算法, 在修复过程中尽可能保留下界解中的优质序列, 使其进入原问题可行域后, 目标值尽可能靠近下界目标值, 以保证解的质量. 设  $K_f$  为拉格朗日下界的排序解中可行车辆的集合,  $K_{inf}$  为不可行车辆的集合,  $V_{inf}$  为一个空集, 启发式修复算法步骤如下.

step 1: 对集合  $K_{inf}$  中的所有车辆进行如下操作: 将当前车辆配送路线中的最后一个点拿出, 并判断约束. 若满足约束, 则将其放入集合  $K_f$ ; 否则, 重复此操

作,直到该车辆满足约束; $k = 1$ .

step 2: 将 $V_{inf}$ 中最后一个点拿出并放入 $K_f$ 中第 $k$ 辆车的最后一个位置,并判断约束.若满足约束,则重复step 2,直到该车辆载重饱和;否则, $k = k + 1$ ,并重复step 2,直到 $k = m$ .

step 3: 判断 $V_{inf}$ 是否为空集.若 $V_{inf}$ 为空集,则输出可行解,算法结束;否则,执行step 4.

step 4: 依次对 $K_f$ 中初始载货量小于最大载重的车辆进行操作:将送货需求大于取货需求的点插入该车辆配送序列的初始位置,再将 $V_{inf}$ 中的最后一个点插入该车辆最后一个位置并判断约束.若满足约束,则转至step 3;否则,切换下一辆车,并重复step 4.

### 2.3 基于局部搜索的可行解优化

本阶段设计一种启发式局部搜索算法,对2.2节中得到的所有可行解中质量最高的解进行优化.

#### 2.3.1 邻域算子

本节提出6种邻域搜索算子如下.

1) shift(1.0): 将第 $k$ 辆车配送路线中的第 $i$ 个点插入到第 $l$ 辆车配送路线中的第 $j$ 个位置.

2) shift(2.0): 将第 $k$ 辆车配送路线中的第 $i$ 个点及其相邻点插入到第 $l$ 辆车配送路线中的第 $j$ 个位置及其相邻位置.

3) insert: 将第 $k$ 辆车配送路线中的第 $i$ 个点插入到该路线的第 $j$ 个位置.

4) swap: 将第 $k$ 辆车配送路线中的第 $i$ 个点与第 $l$ 辆车配送路线中的第 $j$ 个点交换位置.

5) exchange: 将第 $k$ 辆车配送路线中的第 $i$ 个点与该路线的第 $j$ 个点交换位置.

6) 2-opt: 将第 $k$ 辆车配送路线中的第 $i$ 个点与第 $j$ 个点及它们之间的所有点逆序.

此外,上述搜索算子由于对邻域结构的影响较小,算法在到达局部最优后难以跳出,容易陷入局部最优解.为了使算法具备跳出局部最优的能力,利用下界解中的部分启发式信息,提出一种基于下界的扰动算子 lower bound-based search (LBS).如2.2.3节所述,拉格朗日下界解中相邻两点间的距离可能较短,根据这一信息,定义LBS算子如下.

从当前解 $\pi$ 中的每条配送路线中各选出一个客户点放入节点池 $U_{LBS}$ 内,得到解 $\pi'$ .然后将 $U_{LBS}$ 内的所有节点按如下规则插入到 $\pi'$ 中:对于 $U_{LBS}$ 内的节点 $j$ ,若其在下界解中的序列为 $i - j - k$ ,则将其插入到 $\pi'$ 中 $i$ 的后一个位置及 $k$ 的前一个位置中使目标值增量较小的位置,并判断约束;若插入到最佳位置后不满足约束,则插入到次优位置;若两个位置插入后

都不满足约束,则采用贪婪插入,将其插入到 $\pi'$ 中使目标值最小化且满足约束的位置.

#### 2.3.2 搜索策略

由于最终搜索到的解的质量很大程度取决于修复解的质量,且修复解很可能为当前局部最优解,算法从搜索宽度及搜索深度两个方向对修复解进行搜索.为增强算法的搜索宽度,本节使用上述6种算子 shift(1.0)、shift(2.0)、insert、swap、exchange 和 2-opt 作为搜索算子对修复解进行变邻域搜索,并使用 LBS 算子作为扰动算子对陷入局部最优的解进行扰动.为增强算法的搜索深度,对每一种搜索算子,均在当前邻域随机生成 $2n$  ( $n$ 表示客户点数)个解,将满足约束的解保留,从中选择最优解.设修复解为序列 $\pi$ ,启发式局部搜索算法步骤如下.

step 1: counter = 1.

step 2: 依次使用上述6种搜索算子对当前解 $\pi$ 进行 $2n$ 次操作,将满足约束的解保留,并保优.

step 3: 判断是否满足局部搜索算法的终止条件.若满足,则输出最优解,算法结束;否则,执行step 4.

step 4: 判断当前最优解是否优于历史最优.若是,则 counter = 1,并转至step 2;否则,counter = counter + 1,执行step 5.

step 5: 若 counter = 3,则使用LBS算子对 $\pi$ 进行扰动,并转至step 1;否则,转至step 2.

## 3 算法测试及对比

本文实验均在同一实验环境中进行,CPU主频率为1.30 GHz,内存为16 GB,操作系统为64位 Windows 10,编程语言为Python3.8.本文选取并改进标准VRP实例库(www.bernabe.dorronsoro.es/vrp)中的24个实例进行测试.这24个实例的客户点数量范围为[10, 100],配送中心的车辆数范围为[3, 20],各客户点取送货的货物重量范围为 $P_i, D_i \in [1, 5]$ ,车辆最大载重 $Q$ 为20单位及50单位.所有测试算法的终止条件均设置为相同的算法运行时间.

### 3.1 改进次梯度算法有效性测试

为验证本文所提出改进次梯度算法的有效性,将本文改进次梯度算法所得下界与一般次梯度算法<sup>[39]</sup>所得下界进行对比.为有效评估下界紧致程度,选取10个中小规模实例进行测试.各算法运行时间为20s.以Gap作为评价指标,有

$$\text{Gap} = \frac{\text{解} - \text{下界}}{\text{下界}} \times 100\%, \quad (19)$$

用来评价下界与可行解之间的偏离程度.每个问题对应的最好结果用粗体表示,对比结果如表3所示.

表3 不同次梯度算法测试结果

Q	n_m	Gurobi	一般次梯度		改进次梯度	
			下界	Gap/%	下界	Gap/%
20	10_3	290.317	283.117	2.54	<b>288.295</b>	<b>0.70</b>
	15_3	300.778	291.191	3.29	<b>297.986</b>	<b>0.94</b>
	20_5	407.858	393.377	3.68	<b>404.435</b>	<b>0.92</b>
	30_5	432.307	411.014	5.18	<b>427.536</b>	<b>1.13</b>
	40_10	785.948	763.946	2.88	<b>783.184</b>	<b>0.35</b>
	50_10	859.888	807.115	6.54	<b>812.394</b>	<b>4.75</b>
50	40_5	488.662	437.859	11.60	<b>449.040</b>	<b>8.82</b>
	50_5	522.171	471.261	10.80	<b>486.738</b>	<b>7.28</b>
	55_10	856.200	769.754	11.23	<b>785.412</b>	<b>9.01</b>
	60_10	842.237	790.361	6.56	<b>798.978</b>	<b>5.41</b>

由表3可知,改进次梯度算法所得下界对应的Gap优于一般次梯度算法所得下界对应的Gap,表明改进次梯度算法所得下界更紧.原因在于:一般次梯度算法仅利用初始解引导次梯度方向,且始终不改变初始解,当次梯度方向偏离可行域方向时,算法无法做出有效调整,从而导致其收敛速度较慢;而改进次梯度算法在每次迭代过程中均利用修复操作得到的可行解来动态引导次梯度方向,可不断驱动算法的搜索过程向可行域逼近,因此,该算法收敛速度较快,有

助于最终获得更紧或质量更高的下界.

### 3.2 TSLHA 有效性测试

本节验证TSLHA的有效性.因TSLHA基于数学规划方法和启发式算法,故分别将其与Gurobi求解器和启发式算法进行比较.

#### 3.2.1 与Gurobi求解器比较

为评估TSLHA与Gurobi求解器的性能,将其各自所得最优解与相同下界(即TSLHA所得的拉格朗日下界)间的Gap作为评价指标.显然,Gap越小,表明所对应算法的性能越好,其所得解的质量越高.

设定每种算法在相同运行时间下均进行20次独立实验.同时,为验证算法在不同终止条件下的鲁棒性,设置算法运行时间分别为2ns、5ns、10ns、20ns及50ns.其中,TSLHA中局部搜索的执行时间设置为总时间的10%,其余时间均执行改进次梯度算法以获取下界.限于篇幅,仅保留算法运行时间为20ns的实验结果,如表4所示,其余实验数据可在(<https://pan.baidu.com/s/1g0w7HSimSSgA3RdqPjYodQ> 提取码:8xei)中下载.表4中给出了不同规模问题下20次重复实验所得最优结果的Gap值,每个规模问题对应的最好结果用粗体表示.

表4 TSLHA与Gurobi求解器实验结果对比

Q	n_m	下界	Gurobi	Gap/%	TSLHA	Gap/%	Q	n_m	下界	Gurobi	Gap/%	TSLHA	Gap/%
20	10_3	288.295	<b>290.317</b>	<b>0.70</b>	<b>290.317</b>	<b>0.70</b>	50	40_5	449.040	<b>488.662</b>	<b>8.82</b>	492.137	9.60
	15_3	297.986	<b>300.778</b>	<b>0.94</b>	<b>300.778</b>	<b>0.94</b>		50_5	486.738	522.171	7.28	<b>522.038</b>	<b>7.25</b>
	20_5	404.435	<b>407.858</b>	<b>0.92</b>	<b>407.858</b>	<b>0.92</b>		55_10	785.412	856.200	9.01	<b>817.563</b>	<b>4.09</b>
	25_5	412.577	<b>416.966</b>	<b>1.08</b>	<b>416.966</b>	<b>1.08</b>		60_10	798.978	845.000	5.76	<b>841.843</b>	<b>5.36</b>
	30_5	427.536	<b>432.307</b>	<b>1.13</b>	<b>432.307</b>	<b>1.13</b>		65_10	826.102	890.132	7.75	<b>884.904</b>	<b>7.12</b>
	35_5	481.391	514.611	6.90	<b>514.490</b>	<b>6.88</b>		70_10	879.438	1857.740	111.24	<b>947.476</b>	<b>7.74</b>
	40_10	783.184	<b>785.948</b>	<b>0.35</b>	<b>785.948</b>	<b>0.35</b>		75_10	915.508	1922.818	110.03	<b>941.656</b>	<b>2.86</b>
	50_10	812.394	859.888	4.75	<b>848.242</b>	<b>4.41</b>		80_10	978.398	-	-	<b>1000.867</b>	<b>2.30</b>
	55_10	820.350	1786.944	117.83	<b>891.486</b>	<b>8.67</b>		85_10	969.080	-	-	<b>1054.433</b>	<b>8.81</b>
	60_15	1127.497	1280.095	11.53	<b>1210.215</b>	<b>7.34</b>		90_10	1011.422	-	-	<b>1098.049</b>	<b>8.56</b>
65_15	1174.951	1705.337	45.14	<b>1221.568</b>	<b>4.36</b>	95_15	1296.095	-	-	<b>1408.774</b>	<b>8.69</b>		
75_15	1269.337	-	-	<b>1322.830</b>	<b>4.21</b>	100_20	1619.454	2974.300	83.66	<b>1698.087</b>	<b>4.86</b>		

由表4可知,当求解小规模问题时,Gurobi求解器和TSLHA所得解对应的Gap相差不大,且均稳定在10%以内.这表明二者均能在短时间内获取小规模问题的优质可行解.当求解大规模问题时,Gurobi求解器所得解对应的Gap较大,且部分问题未能在限定时间内输出解,而TSLHA所得解对应的Gap始终保持在10%以内.这表明在合理时间内,Gurobi求解器仅适用于求解小规模问题,而TSLHA可有效求解不同规模的问题.需指出的是,从上传网络的其他运行时间的实验数据亦可得出如上结论.

实际上,Gurobi求解器是采用基于分支定界框架的通用算法来求解(混合)整数规划问题.“通用分支定界算法”(非多项式时间算法)仅能识别一些(混合)整数规划问题中的“共性结构”,无法确保一定能识别和利用本文问题的特定结构来“分支剪切”掉该问题解空间中绝大部分不存在最优解的区域,故Gurobi求解器获得本文较大规模问题的优质解所需的运行时间并不可控,往往需要很长时间(参见引言第4段对分支定界算法特点的分析).这使得Gurobi求解器在本文设定的较短运行时间内所得解的质量较差,对

应的Gap较大. 因此,对于各种面向实际的组合优化问题,仍需自行设计有效的求解算法.

### 3.2.2 与启发式算法比较

为验证TSLHA中启发式算法的性能,将TSLHA与4种国际期刊中求解相同或相近问题的有效算法进行比较. 其中:HH-ILS<sup>[40]</sup>结合迭代局部搜索和变邻域下降,是求解GVRPSD的有效算法;VNS-BSTS<sup>[41]</sup>结合变邻域搜索和双结构禁忌搜索,是求解VRPSD的有效算法;ALNS<sup>[42]</sup>和TS<sup>[43]</sup>则是求解各类复杂VRP公认的高效算法. 为确保对比的公平性,将TS的初始解方式(随机生成)改为与其他对比算法一致,即每条路线的第1个客户随机选择,同时其

余客户均采用贪婪插入. 由于TSLHA大部分时间用于获取下界,TSLHA从次梯度算法结束开始计时,每种算法的运行时间为2n s. 表5中给出了每种算法在相同时间内获得的不同规模实例的平均值和最小值. 每个实例对应的最优结果用粗体表示. 由表5可知,TSLHA在绝大部分实例下的测试结果都明显优于其他对比算法.

为评估TSLHA中启发式算法的收敛特性,选用2个不同规模实例对TSLHA中启发式算法和4种对比算法进行测试. 各算法收敛曲线如图1所示. 由图1(a)可知,4种对比算法在小规模问题上的收敛性差异不大,TSLHA可较快获取优质解. 由图1(b)可

表5 TSLHA与HH-ILS、VNS-BSTS、ALNS、TS实验结果对比

Q	n_m	HH-ILS		VNS-BSTS		ALNS		TS		TSLHA	
		平均值	最小值	平均值	最小值	平均值	最小值	平均值	最小值	平均值	最小值
20	10_3	<b>290.317</b>	<b>290.317</b>	<b>290.317</b>	<b>290.317</b>	290.707	<b>290.317</b>	<b>290.317</b>	<b>290.317</b>	<b>290.317</b>	<b>290.317</b>
	15_3	300.790	<b>300.778</b>	<b>300.778</b>	<b>300.778</b>	301.174	<b>300.778</b>	300.860	<b>300.778</b>	<b>300.778</b>	<b>300.778</b>
	20_5	<b>407.858</b>	<b>407.858</b>	407.861	<b>407.858</b>	407.983	<b>407.858</b>	408.025	<b>407.858</b>	<b>407.858</b>	<b>407.858</b>
	25_5	417.097	<b>416.966</b>	417.280	<b>416.966</b>	417.429	<b>416.966</b>	417.061	<b>416.966</b>	<b>417.024</b>	<b>416.966</b>
	30_5	<b>432.351</b>	<b>432.307</b>	432.544	<b>432.307</b>	432.949	432.333	432.392	<b>432.307</b>	432.355	<b>432.307</b>
	35_5	<b>519.664</b>	<b>514.490</b>	521.367	<b>514.490</b>	522.972	517.855	519.889	514.508	520.348	<b>514.490</b>
	40_10	787.659	785.959	788.707	786.032	792.290	786.657	789.036	786.015	<b>786.691</b>	<b>785.948</b>
	50_10	874.891	850.987	879.859	858.361	871.787	854.006	<b>868.445</b>	853.314	870.880	<b>848.242</b>
	55_10	932.013	891.954	933.908	892.248	931.673	903.083	939.017	<b>889.599</b>	<b>930.263</b>	891.486
	60_15	1234.574	1214.992	1240.903	1213.209	1244.997	1218.354	1234.421	1216.722	<b>1227.061</b>	<b>1210.215</b>
50	65_15	1254.941	1227.888	1261.806	1236.711	1263.161	1237.289	1260.162	1241.518	<b>1252.026</b>	<b>1221.568</b>
	75_15	1372.718	1343.290	1397.428	1372.331	1425.857	1385.703	1415.039	1375.890	<b>1344.909</b>	<b>1322.830</b>
	40_5	496.154	490.887	502.528	492.357	499.149	493.938	499.656	<b>490.852</b>	<b>494.018</b>	492.137
	50_5	560.265	524.889	562.565	535.824	561.170	530.177	557.964	530.317	<b>557.499</b>	<b>522.038</b>
	55_10	828.377	820.371	834.055	829.555	835.208	825.976	830.697	826.947	<b>825.679</b>	<b>817.563</b>
	60_10	866.971	848.765	876.009	854.671	879.918	849.041	<b>865.299</b>	850.776	866.842	<b>841.843</b>
	65_10	896.187	888.925	908.081	891.323	913.346	890.922	902.895	889.171	<b>890.095</b>	<b>884.904</b>
	70_10	987.771	953.081	995.247	968.021	994.377	965.750	989.618	971.263	<b>978.511</b>	<b>947.476</b>
	75_10	961.187	945.382	970.835	947.103	978.830	951.242	961.334	943.436	<b>955.361</b>	<b>941.656</b>
	80_10	1027.725	1005.654	1031.088	1012.391	1032.691	1011.823	1029.073	1007.363	<b>1016.008</b>	<b>1000.867</b>
85_10	1077.989	1061.529	1087.386	1072.673	1082.544	1065.904	1085.877	1062.816	<b>1070.716</b>	<b>1054.433</b>	
90_10	1150.745	1112.018	1163.143	1130.568	1172.776	1135.812	1154.725	1126.538	<b>1137.349</b>	<b>1098.049</b>	
95_15	1492.327	1433.906	1513.889	1457.215	1519.264	1465.929	1507.238	1451.376	<b>1470.547</b>	<b>1408.774</b>	
100_20	1740.628	1714.359	1771.674	1726.720	1780.182	1718.547	1769.049	1723.628	<b>1717.678</b>	<b>1698.087</b>	

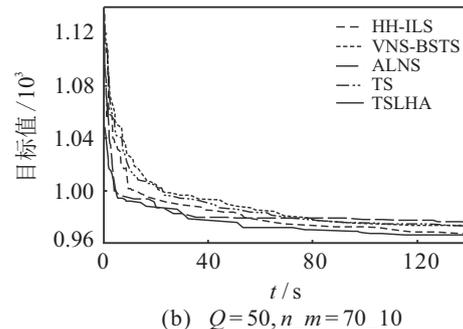
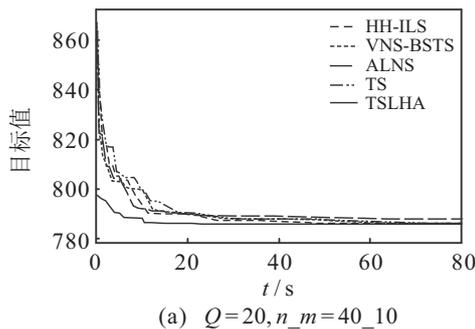


图1 算法收敛曲线

知, TSLHA 在运行初期能以较快的速度(仅稍慢于 ALNS)改进解的质量,且在运行中后期对应的目标值曲线呈现“阶梯状”下降(见图 1(b))并一直保持在最低位置.这是由于 TSLHA 在局部搜索中使用所设计的 LBS 算子,有助于引导算法跳出当前局部极小,并跳入可能存在优质解的区域继续搜索.此外, TSLHA 的修复操作合理利用了所得较紧下界解中的部分优质序列信息,从而能构造具有较高质量的初始可行解(对应解空间中的优质区域).因此, TSLHA 中的启发式算法可直接从优质区域开始执行搜索,这使得算法能更快地发现高质量解.

## 4 结论

本文的主要结论如下: 1) TSLHA 的第 1 阶段针对同时取送货的特点,提出了一种改进拉格朗日松弛技术,通过合理松弛车辆载重约束来获取对偶模型; 2) TSLHA 的第 2 阶段设计了改进次梯度算法,迭代求解对偶模型并获取较紧下界,同时设计了修复机制以合理利用较紧下界对应解中的部分优质序列信息来构造较高质量的原问题可行解,从而确保后续阶段的搜索可直接从优质解区域开始; 3) TSLHA 的第 3 阶段设计了启发式局部搜索算法,对第 2 阶段得到的可行解进行优化,并利用下界解中的有效信息,设计基于下界的搜索算子来提升局部搜索算法的搜索效率.

综上, TSLHA 在求解不同规模的问题时,既能获得问题下界来定量评估算法所得解的质量,又能获得高质量可行解.其综合了数学规划方法和启发式算法的优势,对设计求解 GVRSPD 及类似问题的高效算法具有一定的借鉴价值.后续工作将把 TSLHA 拓展应用于更加复杂的多级车辆路径问题.

## 参考文献(References)

- [1] Dantzig G B, Ramser J H. The truck dispatching problem[J]. *Management Science*, 1959, 6(1): 80-91.
- [2] Moghdani R, Salimifard K, Demir E, et al. The green vehicle routing problem: A systematic literature review[J]. *Journal of Cleaner Production*, 2021, 279: 123691.
- [3] Erdoğan S, Miller-Hooks E. A green vehicle routing problem[J]. *Transportation Research Part E: Logistics and Transportation Review*, 2012, 48(1): 100-114.
- [4] Solomon M M. Algorithms for the vehicle routing and scheduling problems with time window constraints[J]. *Operations Research*, 1987, 35(2): 254-265.
- [5] Min H. The multiple vehicle routing problem with simultaneous delivery and pick-up points[J]. *Transportation Research Part A: General*, 1989, 23(5): 377-386.
- [6] Koç Ç, Laporte G, Tükenmez İ. A review of vehicle routing with simultaneous pickup and delivery[J]. *Computers & Operations Research*, 2020, 122: 104987.
- [7] DellAmico M, Righini G, Salani M. A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection[J]. *Transportation Science*, 2006, 40(2): 235-247.
- [8] Rieck J, Zimmermann J. Exact solutions to the symmetric and asymmetric vehicle routing problem with simultaneous delivery and pick-up[J]. *Business Research*, 2013, 6(1): 77-92.
- [9] Shi Y, Zhou Y J, Boudouh T, et al. A lexicographic-based two-stage algorithm for vehicle routing problem with simultaneous pickup-delivery and time window[J]. *Engineering Applications of Artificial Intelligence*, 2020, 95: 103901.
- [10] Liu W J, Zhou Y T, Liu W, et al. A hybrid ACS-VTM algorithm for the vehicle routing problem with simultaneous delivery & pickup and real-time traffic condition[J]. *Computers & Industrial Engineering*, 2021, 162: 107747.
- [11] Öztaş T, Tuş A. A hybrid metaheuristic algorithm based on iterated local search for vehicle routing problem with simultaneous pickup and delivery[J]. *Expert Systems With Applications*, 2022, 202: 117401.
- [12] Bektaş T, Laporte G. The pollution-routing problem[J]. *Transportation Research Part B: Methodological*, 2011, 45(8): 1232-1250.
- [13] Fukasawa R, He Q, Song Y J. A disjunctive convex programming approach to the pollution-routing problem[J]. *Transportation Research Part B: Methodological*, 2016, 94: 61-79.
- [14] Rauniyar A, Nath R, Muhuri P K. Multi-factorial evolutionary algorithm based novel solution approach for multi-objective pollution-routing problem[J]. *Computers & Industrial Engineering*, 2019, 130: 757-771.
- [15] Qiu R, Xu J P, Ke R M, et al. Carbon pricing initiatives-based bi-level pollution routing problem[J]. *European Journal of Operational Research*, 2020, 286(1): 203-217.
- [16] Kara M, Kara B Y, Yetis M K. Energy minimizing vehicle routing problem[C]. *International Conference on Combinatorial Optimization and Applications*. Berlin, 2007: 62-71.
- [17] Kara M, Yeti B, Kadri M. Cumulative vehicle routing problems[C]. *Vehicle Routing Problem*. InTech, 2008: 85-98.
- [18] Fukasawa R, He Q, Song Y J. A branch-cut-and-price algorithm for the energy minimization vehicle routing problem[J]. *Transportation Science*, 2016, 50(1): 23-34.
- [19] Demir E, Bektaş T, Laporte G. A comparative analysis of several vehicle emission models for road freight transportation[J]. *Transportation Research Part D: Transport and Environment*, 2011, 16(5): 347-357.
- [20] Kent J H, Post K, Tomlin J A. Fuel consumption and emission modelling in traffic links[C]. *The*

- 2nd Conference on Traffic, Energy and Emissions, Melbourne, 1982: 1-10.
- [21] Bowyer D P, Akçelik R, Biggs D C. Guide to fuel consumption analyses for urban traffic management[M]. Australia: Australian Road Research Board, 1984.
- [22] Barth M, Boriboonsomsin K. Real-world carbon dioxide impacts of traffic congestion[J]. Transportation Research Record: Journal of the Transportation Research Board, 2008, 2058(1): 163-171.
- [23] Hickman J. Methodology for calculating transport emissions and energy consumption[J]. Transport Research Fourth Framework Programme Strategic Research, 1999, 381.
- [24] Tiwari A, Chang P C. A block recombination approach to solve green vehicle routing problem[J]. International Journal of Production Economics, 2015, 164: 379-387.
- [25] Yu Y, Wang S H, Wang J W, et al. A branch-and-price algorithm for the heterogeneous fleet green vehicle routing problem with time windows[J]. Transportation Research Part B: Methodological, 2019, 122: 511-527.
- [26] 周鲜成, 吕阳, 贺彩虹, 等. 考虑时变速度的多车场绿色车辆路径模型及优化算法[J]. 控制与决策, 2022, 37(2): 473-482.  
(Zhou X C, Lv Y, He C H, et al. Multi-depot green vehicle routing model and its optimization algorithm with time-varying speed[J]. Control and Decision, 2022, 37(2): 473-482.)
- [27] Wen M Y, Sun W, Yu Y, et al. An adaptive large neighborhood search for the larger-scale multi depot green vehicle routing problem with time windows[J]. Journal of Cleaner Production, 2022, 374: 133916.
- [28] Liu Y, Yu Y, Zhang Y, et al. Branch-Cut-and-Price for the time-dependent green vehicle routing problem with time windows[J]. INFORMS Journal on Computing, 2023, 35(1): 14-30.
- [29] 潘常春, 杨根科. 奖励收集斯坦利最小树的混合拉格朗日与分散搜索算法[J]. 控制与决策, 2007, 22(12): 1341-1346.  
(Pan C C, Yang G K. Hybrid Lagrange and scatter-search algorithm for prize-collecting Steiner tree problem[J]. Control and Decision, 2007, 22(12): 1341-1346.)
- [30] Bouzid M C, Ait Haddadene H, Salhi S. An integration of Lagrangian split and VNS: The case of the capacitated vehicle routing problem[J]. Computers & Operations Research, 2017, 78: 513-525.
- [31] Parchami Afra A, Behnamian J. Lagrangian heuristic algorithm for green multi-product production routing problem with reverse logistics and remanufacturing[J]. Journal of Manufacturing Systems, 2021, 58: 33-43.
- [32] Park H, Son D, Koo B, et al. Waiting strategy for the vehicle routing problem with simultaneous pickup and delivery using genetic algorithm[J]. Expert Systems with Applications, 2021, 165: 113959.
- [33] Jabir E, Panicker V V, Sridharan R. Design and development of a hybrid ant colony-variable neighbourhood search algorithm for a multi-depot green vehicle routing problem[J]. Transportation Research Part D: Transport and Environment, 2017, 57: 422-457.
- [34] Fisher M L. The Lagrangian relaxation method for solving integer programming problems[J]. Management Science, 1981, 27(1): 1-18.
- [35] 孙小玲, 李端. 整数规划[M]. 北京: 科学出版社, 2010. (Sun X L, Li D. Integer programming[M]. Beijing: Science Press, 2010.)
- [36] Sherali H D, Ulular O. A primal-dual conjugate subgradient algorithm for specially structured linear and convex programming problems[J]. Applied Mathematics and Optimization, 1989, 20(1): 193-221.
- [37] Camerini P M, Fratta L, Maffioli F. On improving relaxation methods by modified gradient techniques[C]. Nondifferentiable Optimization. Berlin: Springer, 1975: 26-34.
- [38] 何方国. 拉格朗日松弛对偶问题的一个改进次梯度算法[J]. 长江大学学报: 自科版, 2016, 13(4): 1-5. (He F G. An improved subgradient algorithm for Lagrangian relaxation dual problem[J]. Journal of Yangtze University: Natural Science Edition, 2016, 13(4): 1-5.)
- [39] Xing T, Zhou X S. Finding the most reliable path with and without link travel time correlation: A Lagrangian substitution based approach[J]. Transportation Research Part B: Methodological, 2011, 45(10): 1660-1679.
- [40] Olgun B, Koç Ç, Altıparmak F. A hyper heuristic for the green vehicle routing problem with simultaneous pickup and delivery[J]. Computers & Industrial Engineering, 2021, 153: 107010.
- [41] Shi Y, Zhou Y J, Boudouh T, et al. A lexicographic-based two-stage algorithm for vehicle routing problem with simultaneous pickup-delivery and time window[J]. Engineering Applications of Artificial Intelligence, 2020, 95: 103901.
- [42] Yu Z X, Zhang P, Yu Y, et al. An adaptive large neighborhood search for the larger-scale instances of green vehicle routing problem with time windows[J]. Complexity, 2020, 2020: 1-14.
- [43] Qiu M, Fu Z, Eglese R, et al. A Tabu Search algorithm for the vehicle routing problem with discrete split deliveries and pickups[J]. Computers & Operations Research, 2018, 100: 102-116.

### 作者简介

李熠胥(1998—), 男, 硕士生, 从事数学规划与优化调度的研究, E-mail: 980837615@qq.com;

胡蓉(1974—), 女, 教授, 博士生导师, 从事优化方法、决策支持系统等研究, E-mail: ronghu@vip.163.com;

吴绍云(1978—), 男, 副教授, 硕士生导师, 从事智能算法与优化调度的研究, E-mail: hongtawsy@163.com;

于乃康(1993—), 男, 博士生, 从事数学规划与优化调度的研究, E-mail: 240004982@qq.com;

钱斌(1976—), 男, 教授, 博士生导师, 从事智能优化调度理论与方法等研究, E-mail: bin.qian@vip.163.com.