

多需求点间车辆调度模型及优化算法混合求解研究

王素欣¹,熊琚恺^{2†},王雷震^{1,2},卢福强³,温恒⁴,司马聪²

(1. 东北大学秦皇岛分校 控制工程学院,河北 秦皇岛 066004;

2. 东北大学 信息科学与工程学院,辽宁 沈阳 110819;

3. 燕山大学 经济管理学院,河北 秦皇岛 066004;

4. 吉林大学 计算机科学与技术学院,吉林 长春 130012)

摘要:为解决多需求点间同时集送货问题,建立考虑需求拆分和转运的车辆路径模型.在模型中,加入车辆装载量动态变化约束、节点可多次访问约束和需求可拆分转运约束,提高问题的普遍性.在模型的优化算法中,算术、蚁群优化算法混合求解.通过算术蚁群算法嵌套优化模式,外层算术优化算法得到配送车辆的任务量,内层蚁群算法优化路径,并将结果反馈给外层算法继续更新求解,直至达到终止条件.同时,添加概率系数、增加算子位置更新公式和更新动态禁忌矩阵对混合算术蚁群算法改进,增加解的多样性,提高算法的求解效率.最后通过实例验证并与混合鲸鱼算法等比较,改进的算法解决本文问题效果更好.

关键词:路径规划;同时集送货问题;需求可拆分;随机转运点;算术优化算法;蚁群算法

中图分类号:N945.1

文献标志码:A

Research on Hybrid Solution of Vehicle Scheduling Model and Optimization Algorithm among Multi-demand Points

WANG Suxin¹, XIONG Junkai^{2†}, WANG Leizhen^{1,2}, LU Fuqiang³, WEN Heng⁴, SIMA Cong²

(1. School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China;

2. School of Information Science & Engineering, Northeastern University, Shenyang 110819, China;

3. School of Economics and Management, Yanshan University, Qinhuangdao 066004, China;

4. School of Computer Science and Technology, Jilin University, Changchun 130012, China)

Abstract: To solve simultaneous pickup and delivery among multiple demand points, a vehicle routing model that incorporates demand splitting and transfer is established. The constraint of dynamic variation of vehicle load, the constraint of multiple node access and the constraint of demand split transport are added in the model to improve the universality of the problem. In the optimization algorithm of the model, a hybrid approach combining arithmetic and ant colony optimization algorithm is employed to solve the problem. The algorithm follows a nested optimization structure, where the outer arithmetic optimization algorithm gets the task quantity of the delivery vehicle. The inner

* 收稿日期:2022-10-09

基金项目:国家重点研发计划项目(2020YFB1712802), National Key R&D Program of China(2020YFB1712802);国家自然科学基金资助项目(71401027), National Natural Science Foundation of China(71401027);河北省高等学校人文社会科学研究项目(SQ202002), Humanities and Social Sciences Funds for Hebei Universities(SQ202002)

作者简介:王素欣(1976—),女,河北唐山人,东北大学秦皇岛分校副教授,博士

† 通信联系人, E-mail:2855788162@qq.com

ant colony algorithm then optimizes the path, and provides feedback to the outer algorithm to continue to update and solve until the termination condition is met. At the same time, several enhancements are introduced to the hybrid arithmetic ant colony algorithm, such as incorporating probability coefficient, adding operator position update formula and updating dynamic tabu matrix. These enhancements aim to increase the diversity of solutions and improve the efficiency of the algorithm. Finally, the improved algorithm is verified by an example and compared with the hybrid whale algorithm and other algorithms to solve the problem in this paper.

Key words: path planning; simultaneous pickup and delivery problem; split delivery; random transshipment point; arithmetic optimization algorithm; ant colony optimization

随着经济的发展,人们对多需求点间的物流配送有更高要求,提出同时集送货的多对多的车辆调度问题(many-to-many vehicle routing problem with simultaneous pickup and delivery, M-M-VRPSD)。

国内外多需求点间货物配送及其延伸问题的研究现状如下。

1)多对多物流问题中, Lim等^[1]提出用于研究多车型、多仓库车辆路径模型; Zhang等^[2]通过节点处重新分配模式解决多商品多车型路径问题; Hornstra等^[3-5]研究货物装箱或自然灾害背景下的多需求点问题。

2)综合集送货问题中, Berahhou等^[6]研究同时集送货的动态车辆路径问题; 范厚明等^[7]构建集货需求随机的两阶段模型; Dehghan等^[8-10]考虑外界干扰风险、车辆超载的交叉取送货问题。

3)需求可拆分问题中, Abdi等^[11-12]针对绿色供应链效率和特殊商品的兼容等约束提出需求拆分、分批送货模式; Bortfeldt等^[13]针对需求强制与选择拆分两种情况,建立三维装箱路径优化模型。

4)算法研究中, Wang等^[14]提出两阶段混合算法; Avci等^[15-16]利用短期禁忌表来扩大算法的搜索范围; Bolanos等^[17]提出重组方法和变异算子提高遗传算法能力。

当前研究中,多需求点间配送、同时取货送货、需求可拆分、需求可转运、多车型这5个子问题没被考虑。现有研究存在如下问题。

1)车辆调度模式单一。一次工作中车辆仅取货或送货,装载量利用率低。

2)模型普适性差。没有考虑节点间不同货物对流约束、节点多次访问约束。

3)转运点为固定节点。转运点的数量、位置固定,全局寻优能力差,最优解为局部最优解。

4)混合算法局部优化。混合算法间无参数变量的直接联系,子算法分别优化子问题,局部最优间结合不等于全局最优。

针对当前研究不足,为扩大算法迭代范围、加快收敛速度,在 Abualigah等^[18]提出的算术优化算法(arithmetic optimization algorithm, AOA)上增加概率系数、算子位置更新公式,提出改进的算术优化算法(improved arithmetic optimization algorithm, IAOA)。在 Dorigo等^[19]提出的蚁群算法(ant colony optimization, ACO)上增加动态禁忌矩阵,提出改进的蚁群算法(improved ant colony optimization, IACO)。

通过优化算法混合求解方式,提出双算法嵌套优化模式,利用外层算法获得车辆配送任务,利用内层算法获得车辆路径。对比了改进的算术-蚁群混合求解(improved arithmetic optimization algorithm-improved ant colony optimization, IAOA-IACO)与算术-蚁群混合求解(arithmetic optimization algorithm-ant colony optimization, AOA-ACO),也对比了根据鲸鱼优化算法^[20](whale optimization algorithm, WOA)提出的鲸鱼-蚁群混合求解(whale optimization algorithm-ant colony optimization, WOA-ACO)与鲸鱼-鲸鱼混合求解(whale optimization algorithm-whale optimization algorithm, WOA-WOA),这些对比说明利用改进后的算法解决本文问题时效果最佳。

1 构建车辆调度模型

1.1 问题描述

本文解决多需求点间配送问题的描述如下:节点间有配送需求,有多车型车辆在节点间进行同时集送货服务。

节点和货物流通特征如图 1 所示,箭头流通方向代表货物的流通方向,节点间的特点如下:

- 1) 仅有货物流入或流出. 如节点 6、节点 5.
- 2) 同时存在货物流出、流入. 如节点 9.
- 3) 货物对流. 如节点 2 与节点 10 间.
- 4) 无货物流入、流出. 如节点 8.

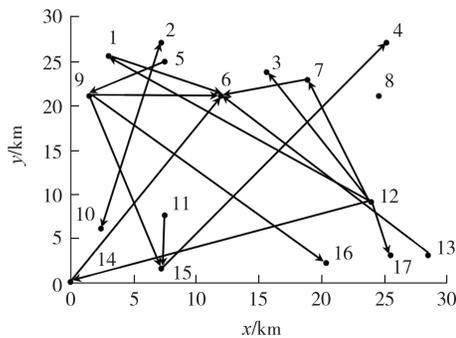


图 1 货物相关节点和流通方向

Fig.1 Cargo related nodes and flow direction

1.2 考虑需求拆分和转运的同时集送货的多需求点问题模型

1.2.1 问题假设

建立如下的假设:1)随机拆分车辆空载装载时超载的货物;2)货物可转运,为简化运算设定货物最多转运一次.

1.2.2 模型符号说明

模型符号及其说明如表 1 所示.

表 1 模型符号及其说明

Tab.1 Model symbols and description

符号	说明
N	客户节点集合
P	车型种类集合
K	可提供服务车辆总数
i, j, u	集合 N 中第 i, j, u 个客户节点
p_k	车型为 p 的第 k 辆车
l_{ij}	节点 i 与节点 j 间距离
q_{ij}	节点 i 与节点 j 间需求量
$m_{ij}^{p_k}$	车型为 p 的第 k 辆车从节点 i 直接到节点 j 时节点 i 处 q_{ij} 的取货量
$x_{ij}^{p_k}$	决策变量, 车型为 p 的第 k 辆车由节点 i 直接到节点 j 服务时为 1, 否则为 0
$y_{ij}^{p_k}$	决策变量, 车型为 p 的第 k 辆车由节点 i 到节点 j 需要转运服务时为 1, 否则为 0
$Q_{ij}^{p_k}$	车型为 p 的第 k 辆车离开节点 i 直接到节点 j 之前的剩余装载量
Q^p	车型为 p 的车辆最大装载量

1.2.3 数学模型

数学模型如下:

$$\min z = \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} \sum_{p \in P} x_{ij}^{p_k} l_{ij} \quad (1)$$

$$\text{s.t. } 0 \leq Q_{ij}^{p_k} \leq Q^p, \forall i, j \in N, \forall p \in P, \forall k \in K \quad (2)$$

$$\sum_{i \in N} \sum_{j \in N} \sum_{k \in K} \sum_{p \in P} m_{ij}^{p_k} x_{ij}^{p_k} - \sum_{i \in N} \sum_{j \in N} \sum_{k \in K} \sum_{p \in P} m_{ij}^{p_k} y_{ij}^{p_k} = \sum_{i \in N} \sum_{j \in N} q_{ij} \quad (3)$$

$$Q_{ij}^{p_k} + \sum_{i \in N} x_{ij}^{p_k} m_{ij}^{p_k} - \sum_{u \in N} x_{ju}^{p_k} m_{ju}^{p_k} = Q_{ij}^{p_k}, \quad (4)$$

$$\forall i, j, u \in N, \forall p \in P, \forall k \in K$$

$$1 \leq k \leq \frac{\sum_{i \in N} \sum_{j \in N} q_{ij}}{\max\{Q^1, Q^2, \dots, Q^p, \dots, Q^P\}}, \quad (5)$$

$$\forall p \in P, \forall k \in K$$

$$\sum_{i \in N} \sum_{j \in N} m_{ij}^{p_k} x_{ij}^{p_k} \leq \sum_{i \in N} \sum_{j \in N} q_{ij}, \forall p \in P, \forall k \in K \quad (6)$$

$$\sum_{i \in N} \sum_{j \in N} x_{ij}^{p_k} \geq 0, \forall p \in P, \forall k \in K \quad (7)$$

$$\sum_{k \in K} \sum_{p \in P} y_{ij}^{p_k} \geq 0, \forall i, j \in N \quad (8)$$

$$\sum_{k \in K} \sum_{p \in P} x_{ij}^{p_k} \geq 1, \forall i, j \in N \quad (9)$$

$$\sum_{i \in N} \sum_{j \in N} m_{ij}^{p_k} x_{ij}^{p_k} \geq Q^p, \forall p \in P, \forall k \in K \quad (10)$$

式(1)~式(5)是目标函数和基本车辆调度模型. 式(1)为目标函数,表示车辆总行驶的路径最短;式(2)表示车辆不超载;式(3)表示节点间的集送货需求均被满足;式(4)体现车辆访问节点前后货物装卸过程的装载量变化;式(5)根据节点的总需求量和车辆的最大装载量选择合理的可使用车辆数量范围.

式(6)~式(10)是体现需求拆分和转运的同时集送货问题特征约束. 式(6)表示车辆一次服务节点时配送量不大于节点需求,即货物可拆分;式(7)表示仅有部分车辆被使用;式(8)表示货物可转运;式(9)表示每位客户可多次访问;式(10)表示车辆配送过程中当前配送过的货物总量可大于车辆最大装载量.

模型的特点如下.

1) 转运点的随机性. 与固定转运点不同,根据当前货物信息随机选择转运节点以全局寻优. 此特点由式(8)体现.

2) 货物流通的多样性. 节点间、节点与车辆间为多对多关系,与一般调度模型相比,增加了节点间不同货物的对流约束、单个或多个车辆多次访问节点的约束,增强模型普适性. 此特点由式(9)体现.

3) 车辆运输的高效率性. 车辆在节点处有集货、送货和同时集送货模式. 货物随时装卸使车辆配送

的总货物量可大于车辆最大装载量,充分利用车辆装载量.此特点由式(10)体现.

2 算术-蚁群混合求解车辆调度模型

2.1 算术优化算法求解货物分配问题

2.1.1 编码产生车辆调度方案

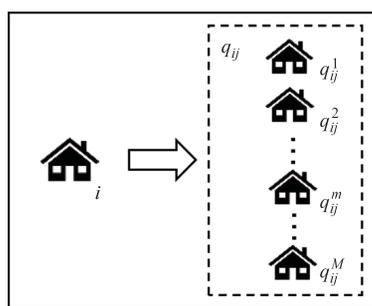
为解决多需求点间车辆调度的编码复杂问题,提出构建虚拟需求和设定 AOA 编码方式辅助算法产生调度方案.

1)虚拟需求的提出.车辆空载时节点需求超载时,随机拆分需求满足配送车辆不超载.节点的需求流通有 3 种情况:①仅流入;②仅流出;③既流入又流出.在有流出需求的节点处构建虚拟需求,解决拆分后的多个需求有相同起点和终点问题.

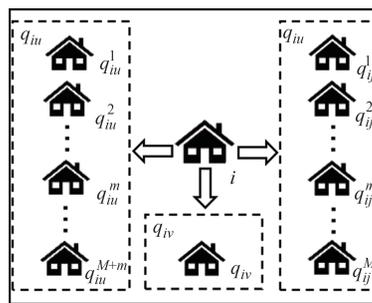
如图 2(a)所示,节点仅有单个流出需求时,虚拟需求个数与需求拆分份数相等,即需求 q_{ij} 拆分 M 份时生成 M 个虚拟需求.如图 2(b)所示,节点有多个流出需求时,生成虚拟需求个数与多个拆分需求份数和无需拆分需求份数之和相等.每个虚拟需求的起点和终点与拆分前相同,即需求 $q_{iu}^1, q_{iu}^2, \dots, q_{iu}^m, \dots, q_{iu}^{M+m}$ 的起点和终点与拆分前相同,分别为节点 i 和节点 u .

2)AOA 编码方式. AOA 的每个初始解为三行多列矩阵,矩阵列数为拆分后的需求总数.矩阵的每行分别为转运点、转运前车辆、转运后车辆,矩阵的列表示拆分后的配送任务.

如图 3 所示,当车型为 p 的车辆空载时,仅装载



(a) 单个需求流出



(b) 多个需求流出

图 2 虚拟需求

Fig.2 Virtual demand

需求 q_{ij} 超载,需求 q_{ij} 随机拆分 M 份,得到需求 $q_{ij}^1, q_{ij}^2, \dots, q_{ij}^m, \dots, q_{ij}^M$. 需求 $q_{(i-3)(j+4)}$ 等满足车辆空载不超载条件无需拆分.其中,需求 q_{ij}^1 与需求 q_{ij}^m 转运点分别为其对应的起点 i 与终点 j ,为无效转运点,需求转运前后的车辆相同,分别为 p_{k+2} 和 $(p+1)_k$,为无效转运车辆,故货物不转运.需求 $q_{(i+3)(j+2)}$ 与需求 $q_{(i-3)(j+4)}$ 的转运点分别为 $i-2$ 与 $i-4$,不是需求对应的起点和终点,为有效转运点,需求转运前后车辆不同,为有效转运车辆,故货物转运.

节点间需求	$q_{(i-3)(j+4)} \leq Q^p$	$q_{(i+5)(j-1)} \leq Q^{(p+2)}$	$q_{ij} > Q^p$						$q_{(i+3)(j+2)} \leq Q^{(p-5)}$
需求转运点	$i-4$	$i+5$	i	$i+1$	\dots	j	\dots	i	$i-2$
转运前车辆	p_{k-1}	$(p+2)_{k+1}$	p_{k+2}	p_{k+5}	\dots	$(p+1)_k$	\dots	p_k	$(p-5)_{k+3}$
转运后车辆	$(p-1)_k$	$(p+2)_{k+1}$	p_{k+2}	p_{k-3}	\dots	$(p+1)_k$	\dots	p_k	$(p-5)_{k-1}$

图 3 AOA 编码方式

Fig.3 AOA encoding mode

2.1.2 算术优化算法的更新迭代

AOA 更新迭代求解有 3 个阶段.

1)通过加速函数选择迭代策略.若函数 M 小于 0 到 1 之间的随机数 r_1 ,则全局探索;否则选择局部开发.

加速函数为

$$M(t) = M_1 + t \left(\frac{M_2 - M_1}{T} \right) \quad (11)$$

式中: t 为当前迭代次数; M_1 为加速函数当前最小值; M_2 为加速函数当前最大值; T 为最大迭代次数.

2)利用乘除法运算特点进行解的探索,其位置更新公式为

$$X(t+1) = \begin{cases} P[(U_B - L_B)\mu + L_B]X_b(t), & 0 \leq r_2 \leq 0.5 \\ X_b(t)/(P + \xi) \{[(U_B - L_B)\mu + L_B]\}, & 0.5 < r_2 \leq 1 \end{cases} \quad (12)$$

$$P(t+1) = 1 - \left(\frac{t}{T}\right)^{1/\alpha} \quad (13)$$

式中: $X_b(t)$ 为当前迭代目标函数最大值; P 为数学优化器概率; ξ 为极小值; U_B 为搜索值域的上限; L_B 为搜索值域的下限; μ 为控制参数; α 为敏感参数.

3) 利用加减法运算进行解的开发, 其位置更新公式为

$$X(t+1) = \begin{cases} X_b(t) + P[(U_B - L_B)\mu + L_B], & 0 \leq r_3 \leq 0.5 \\ X_b(t) - P[(U_B - L_B)\mu + L_B], & 0.5 < r_3 \leq 1 \end{cases} \quad (14)$$

利用 AOA 解决问题时, 其乘法、除法算子的高分散性特点赋予算法寻优范围, 加法、减法算子的精确性特点赋予算法寻优精度. 但由于乘除加减算子位置更新公式的参数设定使得每次搜索迭代时, 部分区域不在算法的搜索范围内, 勘探范围不全面导致了算法搜索效率低, 易陷入局部最优.

2.1.3 算术优化算法迭代策略的改进

针对 2.1.2 总结的 AOA 不足之处提出改进策略以达到扩大勘探范围的效果. 改进原则如下.

1) 对称原则扩大解的搜索范围. 通过增加 0 到 1 之间的概率系数 $r_4 \sim r_7$, 参考当前最优解位置并利用对称原则增加乘除加减算子的选择空间, 解决原始 AOA 部分区域不在算法搜索范围内的问题.

2) 二等分原则选择搜索空间. 利用概率系数 $r_4 \sim r_7$ 对搜索空间概率二等分, 保证当前算子的搜索空间与其对称搜索空间的选择概率相同.

乘法运算的位置更新公式改进为

$$X(t+1) = \begin{cases} P[(U_B - L_B)\mu + L_B]X_b(t), & 0 \leq r_2 \leq 0.5, 0.5 < r_4 \leq 1 \\ 2X_b(t) - P[(U_B - L_B)\mu + L_B]X_b(t), & 0 \leq r_2 \leq 0.5, 0 \leq r_4 \leq 0.5 \end{cases} \quad (15)$$

除法运算的位置更新公式改进为

$$X(t+1) = \begin{cases} X_b(t)/(P + \xi) \{[(U_B - L_B)\mu + L_B]\}, & 0.5 < r_2 \leq 1, 0.5 < r_5 \leq 1 \\ 2X_b(t) - X_b(t)/(P + \xi) \{[(U_B - L_B)\mu + L_B]\}, & 0.5 < r_2 \leq 1, 0 \leq r_5 \leq 0.5 \end{cases} \quad (16)$$

加法运算的位置更新公式改进为

$$X(t+1) = \begin{cases} X_b(t) + P[(U_B - L_B)\mu + L_B], & 0 \leq r_3 \leq 0.5, 0.5 < r_6 \leq 1 \\ X_b(t) - P[(U_B - L_B)\mu + L_B], & 0 \leq r_3 \leq 0.5, 0 \leq r_6 \leq 0.5 \end{cases} \quad (17)$$

减法运算的位置更新公式改进为

$$X(t+1) = \begin{cases} X_b(t) - P[(U_B - L_B)\mu + L_B], & 0.5 < r_3 \leq 1, 0.5 < r_7 \leq 1 \\ X_b(t) + P[(U_B - L_B)\mu + L_B], & 0.5 \leq r_3 \leq 1, 0 \leq r_7 \leq 0.5 \end{cases} \quad (18)$$

乘法算子以当前最优解为基准, 根据式(15)进行左右对称运算, 使获得的结果更分散, 保证算法前期的充分搜索. 同理, 除法算子以式(16)进行左右对称运算. 加法算子以当前迭代最优解为基准, 根据式(17)进行上下对称运算, 保证算法后期的局部优化精度. 同理, 减法算子以式(18)进行上下对称运算, 扩大搜索空间, 某次迭代 AOA 的加减乘除算子的位置更新范围见图 4 (a)^[21], IAOA 相关算子的位置更新范围见图 4(b).

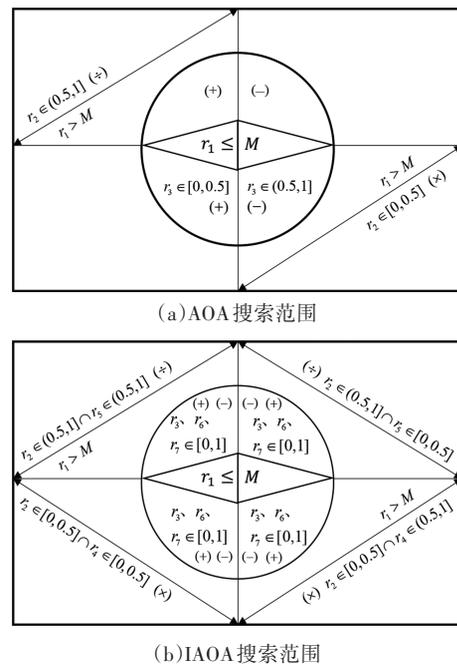


图4 AOA 与 IA OA 搜索范围

Fig.4 The search scope of AOA and IA OA

2.2 蚁群算法求解路径优化问题

2.2.1 蚁群算法节点能见度策略的改进

利用 ACO 解决 TSP 问题时, 节点访问一次后变为禁止访问状态. 所以解决本文节点多次访问问题

时,能见度矩阵失去指导车辆选择可服务节点的作用,导致算法陷入局部最优而多次无效求解,计算效率低.

动态禁忌矩阵代替能见度矩阵可解决上述问题,指导算法进入高效的搜索空间.当车辆处于当前节点时,将其剩余装载量与未配送的需求整合,利用满足装载条件的需求的起点、终点以及转运点更新当前禁忌矩阵.将不满足装载条件的需求的相关节点的禁忌值设定为 0,满足条件的节点的禁忌值计算公式为

$$\eta_{ij} = \frac{1}{l_{ij}(t)} \quad (19)$$

如图 5 所示,车辆空载不超载时,货物 q_{ij} 无需拆分,节点 i 到节点 j 只访问一次,无需考虑两点间能见度变化.车辆空载超载时,货物 q_{ij} 被拆分为 $q_{ij}^1, q_{ij}^2, \dots, q_{ij}^m, \dots, q_{ij}^M$, 节点 i 到节点 j 访问多次.改进前 q_{ij}^1 配送完成而 q_{ij}^2 等未完成时,起点与终点间能见度变为 0,导致车辆无法配送 q_{ij}^2 等货物;改进后,通过货物状态和车辆状态更新禁忌矩阵,根据节点间禁忌值更新能见度, q_{ij}^1 配送完成而 q_{ij}^2 等货物未完成时,起点与终点间能见度不变,车辆可继续配送.

	未拆分货物	改进前				改进后							
		q_{ij}				q_{ij}							
货物访问终点前后节点间能见度		$\frac{1}{l_{ij}}$	0	$\frac{1}{l_{ij}}$	0	$\frac{1}{l_{ij}}$	0	$\frac{1}{l_{ij}}$	0				
被拆分货物		q_{ij}^1	q_{ij}^2	\dots	q_{ij}^m	\dots	q_{ij}^M	q_{ij}^1	q_{ij}^2	\dots	q_{ij}^m	\dots	q_{ij}^M
货物起点与终点间能见度		$\frac{1}{l_{ij}}$	0	\dots	0	\dots	0	$\frac{1}{l_{ij}}$	$\frac{1}{l_{ij}}$	\dots	$\frac{1}{l_{ij}}$	\dots	$\frac{1}{l_{ij}}$

图 5 ACO 改进前后需求能见度变化

Fig.5 Change in demand visibility before and after ACO improvement

2.2.2 蚁群算法求解路径

IACO 求解路径有两个阶段.

1) 针对每辆配送车辆,都有 k 只蚂蚁随机选择节点出发,每只蚂蚁在节点 i 有两种选择:装载货物的终点或满足剩余装载量的货物起点.节点 j 的概率为

$$P_{ij}^k = \begin{cases} \frac{[\tau_{ij}(t)]^v [\eta_{ij}(t)]^\omega}{\sum_{j \in a_k} [\tau_{ij}(t)]^v [\eta_{ij}(t)]^\omega}, & j \in a_k \\ 0, & j \notin a_k \end{cases} \quad (20)$$

式中: τ_{ij} 为节点 i 与节点 j 之间的信息素浓度; v 与 ω 表示期望启发因子; a_k 表示满足访问条件的节点集合.

2) 蚂蚁由节点 i 到节点 j 过程中,节点 i 的信息素浓度会挥发进而减少,节点 j 的信息素浓度会增加,信息素浓度表达式为

$$\tau_{ij}(t+1) = (1-R)\tau_{ij}(t) + \Delta\tau_{ij}(t, t+1) \quad (21)$$

$$\Delta\tau_{ij}(t, t+1) = \begin{cases} \frac{W}{L_k}, & \text{蚂蚁经过路径 } ij \\ 0, & \text{其他情况} \end{cases} \quad (22)$$

式中: R 为信息素浓度挥发系数; $\Delta\tau_{ij}$ 表示从节点 i 到节点 j 的信息素增加量; W 表示信息素量; L_k 表示蚂蚁 k 的行驶路径长度.

IACO 中多只蚂蚁利用配送货物的起始点集合随机选择不同节点出发,每选择下一经过节点时,利用式(19)更新当前禁忌矩阵获得节点的能见度,利用式(21)、式(22)更新当前信息素浓度,同时利用式(20)更新不同节点的选择概率选择下一节点直至获得最优路径.

2.3 算法混合求解的优化思路与编码方式

IAOA-IACO 混合求解的优化与编码思路如下.

1) 利用 IAOA 获得需求的拆分情况与车辆的配送任务.种群内个体编码方式如图 3 所示,通过三行多列矩阵,整合每辆配送车辆的配送矩阵.

2) 配送车辆路径的优化. IACO 的输入为上述 1) 中配送矩阵,利用 IACO 进行路径优化.

3) 对模型混合求解. IAOA-IACO 混合求解的流程见图 6,通过双层算法嵌套优化的方式全局寻优得到最优解.混合算法外层 IAOA 根据内层算法 IACO 的反馈更新迭代,通过双算法整体大循环模式、局部小循环模式得到全局最优解.

3 实验仿真与结果分析

3.1 多需求点间车辆调度实验仿真

3.1.1 案例说明

目前没有 M-M-VRPSPD 问题的标准案例,故本文考虑具有不同需求流向特点的节点生成测试案例.有装载量为 3 t 和 3.5 t 的车辆可供选择,需求流向及其关系见图 1,节点坐标见表 2,选择 17 个客户点为例进行分析求解,其需求的起点、终点以及需求量见表 3,如 $q_{(1)(6)}$ 表示货物起点、终点分别为地点 1 与地点 6.

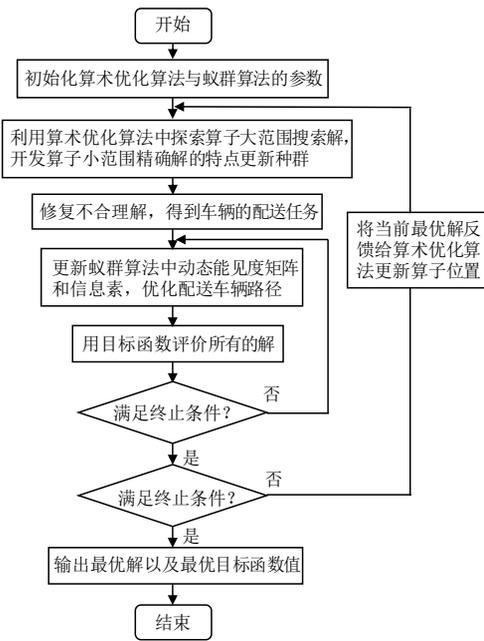


图6 IAOA-ACO 混合求解流程
Fig.6 IAOA-ACO hybrid solution process

表2 客户节点信息

Tab.2 Customer node information

节点标号	横坐标/km	纵坐标/km	节点标号	横坐标/km	纵坐标/km
1	3	25.5	10	2.4	6
2	7.2	27	11	7.5	7.5
3	15.6	23.7	12	24	9
4	25.2	27	13	28.5	3
5	7.5	24.9	14	0	0
6	12	21	15	7.2	1.5
7	18.9	22.8	16	20.4	2.1
8	24.6	21	17	25.5	3
9	1.5	21			

表3 货物信息

Tab.3 Cargo information

货物标号	货物质量/t	货物标号	货物质量/t	货物标号	货物质量/t
$q_{(1)(6)}$	2	$q_{(12)(14)}$	2	$q_{(12)(7)}$	1.4
$q_{(9)(6)}$	2.8	$q_{(2)(10)}$	1.4	$q_{(12)(3)}$	0.6
$q_{(14)(6)}$	2.6	$q_{(10)(2)}$	2.8	$q_{(12)(1)}$	1.8
$q_{(13)(6)}$	3.5	$q_{(15)(4)}$	1.6	$q_{(9)(16)}$	2
$q_{(7)(6)}$	2	$q_{(11)(15)}$	2.1	$q_{(5)(9)}$	2.8
$q_{(12)(17)}$	4.2	$q_{(9)(15)}$	1.2		

3.1.2 结果与分析

运用4种优化算法解决多需求点问题,各算法参数见表4.每种算法实验30次,最大迭代次数为500次. WOA-WOA、WOA-ACO和AOA-ACO混合

优化的特点如下.

- 1)WOA-WOA 调整参数少、收敛快.
- 2)WOA-ACO 鲁棒性强. 解决混合问题时易于实现.
- 3)AOA-ACO 全局搜索能力强. 较单一算法,求解空间更大.

IAOA-ACO 混合求解相较其他算法,其优势如下.

- 1)搜索空间大、收敛速度快. IAOA-ACO 较AOA-ACO,算法外层利用对称原则扩大了搜索范围,算法内层通过动态禁忌矩阵减少寻找合理解迭代次数.
- 2)跳出局部最优能力强. IAOA-ACO 较WOA-WOA和WOA-ACO,通过概率系数增加算法全局搜索的概率,减少算法陷入局部开发阶段的概率.

表4 各算法参数

Tab.4 Parameters for each algorithm

算法	参数
WOA-WOA	螺状常数 $b = 1$
WOA-ACO	螺状常数 $b = 1$,信息素量 $Q = 1$,信息素挥发系数 $R = 0.5$,启发因子 $v = 1, \omega = 5$
AOA-ACO	信息素量 $Q = 1$,信息素挥发系数 $R = 0.5$,启发因子 $v = 1, \omega = 5$
IAOA-ACO	控制参数 $\mu = 0.499$,敏感参数 $\alpha = 5$,信息素量 $Q = 1$,信息素挥发系数 $R = 0.5$,启发因子 $v = 1, \omega = 5$

实例验证结果见表5. IAOA-ACO的最优解、平均最优解都优于其他算法,且方差最小,算法的稳定性最好. 每种算法的最优解迭代收敛速度曲线见图7,IAOA-ACO收敛速度较快.

表5 4种混合算法计算结果

Tab.5 Results of four hybrid algorithms

算法	最优解/km	平均最优解/km	方差/(km) ²	平均运行时间/s
WOA-WOA	573.15	693.01	50.74	778.33
WOA-ACO	469.90	512.55	21.30	971.53
AOA-ACO	469.03	507.81	17.39	918.66
IAOA-ACO	414.06	481.23	16.61	919.56

选择的配送车辆、车辆的最优路径以及装载货物的过程见图8.横坐标表示车辆经过节点的顺序,车辆在节点装货与卸货情况以及货物的起点、终点和重量都已标注.方框表示货物,加粗的方框表示被转运的货物,标有数字的方框代表被拆分的货物,标有相同数字的方框代表被拆分若干份的同一货物.

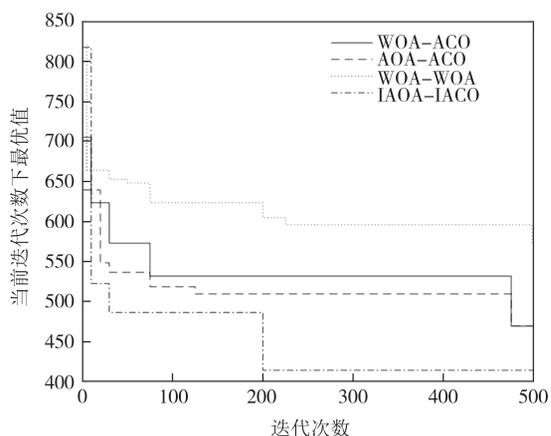


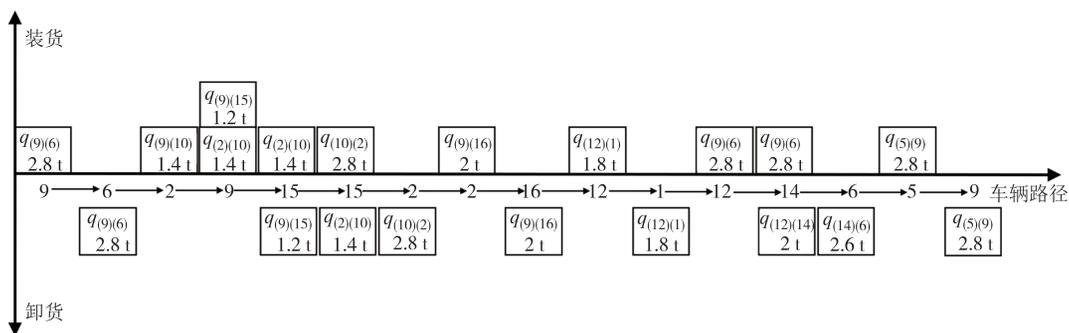
图 7 迭代收敛速度曲线

Fig.7 Iterative convergence rate curve

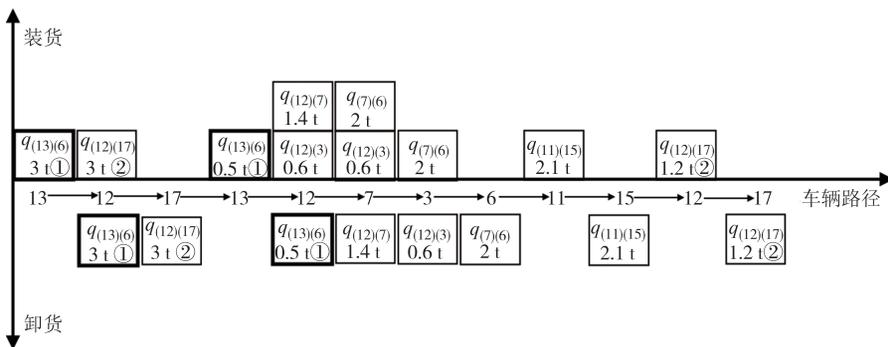
车辆 1、车辆 2、车辆 3 的最大装载量分别为 3.5 t、3 t、3.5 t。在图 8(a)中,由于车辆装载量限制,节点 2、节点 9、节点 12 被多次访问,车辆第二次经过节点 9 之前,装载了需求 $q_{(2)(10)}$,到达节点 9 时装载了需求 $q_{(9)(15)}$,到达节点 15 时需求 $q_{(9)(15)}$ 抵达目的地并对其卸货,车辆剩下需求 $q_{(2)(10)}$ 。

在图 8(b)中,标号①的方框代表需求 $q_{(13)(6)}$ 拆分成 3 t 和 0.5 t,标号②的方框代表需求 $q_{(12)(17)}$ 拆分成 3 t 和 1.2 t。车辆从节点 13 出发时因装载量不足拆分需求 $q_{(13)(6)}$,装载其一部分,到达转运节点 12 时卸货并装载需求 $q_{(12)(17)}$ 的一部分。

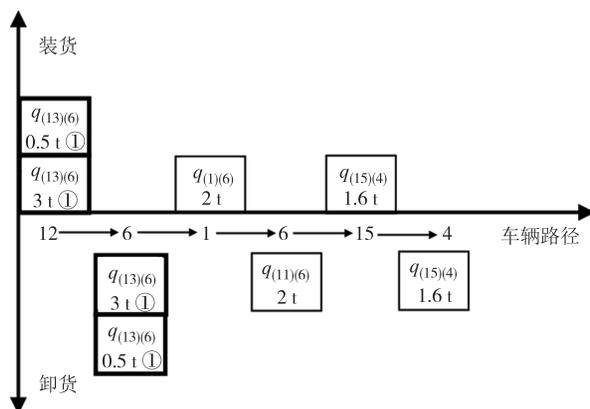
在图 8(c)中,车辆在节点 12 配送转运到此的需求 $q_{(13)(6)}$,并完成其余任务。



(a) 车辆 1 的行驶路径及货物装卸过程(234.03 km)



(b) 车辆 2 的行驶路径及货物装卸过程(91.64 km)



(c) 车辆 3 的行驶路径及货物装卸过程(88.39 km)

图 8 最优车辆配送流程

Fig.8 Optimal vehicle delivery process

3.2 基准函数测试仿真

3.2.1 案例说明

选择 Sphere、Rastrigin、Branin 这 3 个基准函数对 IAOA、AOA、IACO、ACO 和 WOA 进行测试与结果对比. 如图 9 所示, Sphere 函数是单峰函数, 仅有一个最值, 可检测算法收敛能力; Rastrigin 函数是多峰函数, 有多个局部最小值, 可检测算法跳出局部最优能力; Branin 函数是平滑函数, 可检测算法寻优能力,

更有代表性. 函数的表达式及其自变量的取值范围见表 6. 函数的最小值分别为 0、0、0.398.

3.2.2 结果与分析

算法的种群大小设置为 30, 问题维度为 30, 最大迭代次数为 1 000. 将不同函数运行 30 次, 其评价指标见表 7. 分别计算不同基准函数每次迭代的当前最优解的平均值, 绘制迭代收敛的半对数曲线, 如图 10 所示.

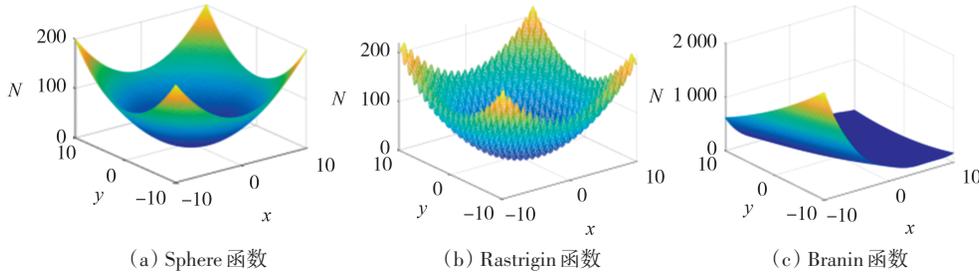


图 9 基准函数图像

Fig.9 Baseline function image

表 6 基准函数表

Tab.6 Table of benchmark function

函数名	表达式	x_n 的取值
Sphere	$f_1 = \sum_{d=1}^D x_d^2$	$[-100, 100]$
Rastrigin	$f_2 = \sum_{n=1}^N [x_n^2 - 10\cos(2\pi x_n) + 10]$	$[-5.12, 5.12]$
Branin	$f_3 = (\frac{-5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} + x_2 - 6)^2 + (10 - \frac{5}{4\pi})\cos(x_1) + 10$	$[5, 15]$

表 7 基准函数测试结果表

Tab.7 Table of benchmark function test results

函数	评价指标	算法				
		IAOA	AOA	IACO	ACO	WOA
Sphere	最优值	1.378e-60	1.451e-10	2.342e-10	1.212e-02	5.290e-04
	最差值	3.456e-17	5.372e-09	1.223e-04	0.190e+01	0.187e+01
	平均值	2.342e-18	3.435e-09	1.037e-04	0.101e+01	1.037e-01
	标准差	1.363e-17	2.563e-09	6.548e-05	6.793e-01	6.863e-01
Rastrigin	最优值	0.00e-00	0.00e-00	1.889e-09	7.331e-04	0.112e+01
	最差值	0.00e-00	0.00e-00	8.220e-02	1.413e-02	1.200e+01
	平均值	0.00e-00	0.00e-00	1.932e-04	1.401e-02	1.001e+01
	标准差	0.00e-00	0.00e-00	6.443e-04	9.131e-03	0.616e+01
Branin	最优值	3.989e-01	3.982e-01	5.671e-01	6.112e-01	5.778e-01
	最差值	3.899e-01	3.899e-01	0.121e+01	0.221e+01	0.234e+01
	平均值	3.980e-01	3.980e-01	8.931e-01	0.178e+01	0.163e+01
	标准差	4.023e-04	3.209e-04	5.544e-01	7.855e-01	6.010e-01

在 Sphere 函数中, IAOA 在 AOA 基础上扩大算法的搜索空间进行改进, 其中后期的收敛速度明显快于 AOA. 通过一定次数的迭代, IAOA 跳出局部最优的能力优势凸显. IACO 在 ACO 基础上缩小最优解

的范围进行改进, 减少了寻找最优解的迭代次数. WOA 较其余算法效果不明显.

在 Rastrigin 函数中, 函数最小值为 0, 由于 0 没有对数, 所以 IAOA 与 AOA 寻找到最小值时对数曲

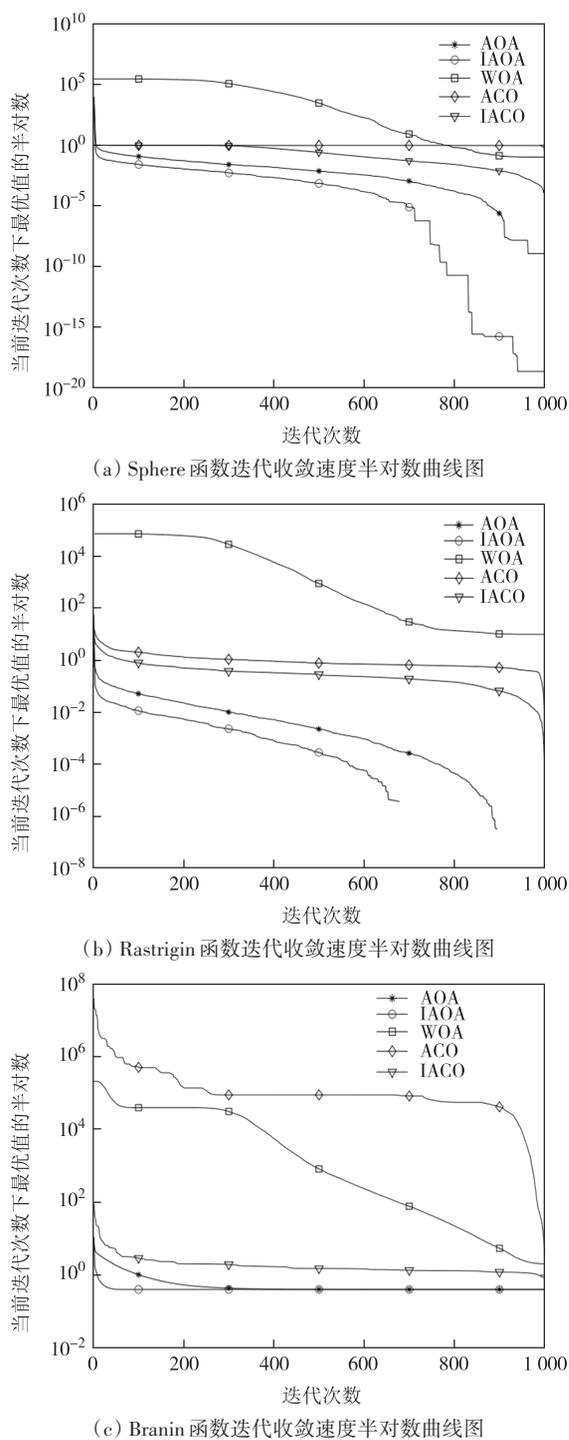


图 10 基准函数迭代收敛速度半对数曲线图

Fig.10 Semi-log curve of the rate of iterative convergence of the baseline function

线中断,IAOA 搜索到最优解的迭代次数明显小于 AOA. IACO 搜索的函数最小值优于 ACO.

在 Branin 函数中,IAOA 较 AOA 能在较少迭代次数的情况下获得函数最小值. ACO 与 WOA 获得的函数最小解相近, IACO 获得的函数值优于 ACO 与 WOA.

4 结语

本文对多需求点问题研究的创新从模型优化、算法改进展开,模型与算法有以下不足.

1)模型中,以车辆总路径最短为目标函数,未考虑时间约束,可能会出现车辆为完成装载或转运货物而提前到达转运节点等待的现象,时间利用率低.

2)算法中,IACO 的改进策略是针对节点多次访问、装载量非线性变化问题提出的,增加了蚂蚁行走每一步筛选有效节点过程的计算,增加了运行时间.

通过对多需求点问题的研究,有以下潜在研究方向.

1)模型中,可考虑将车辆行驶速度与本文问题结合建模,建立车辆行驶时间最短与路径最短的双目标模型.

2)算法中,在迭代过程中向最优解学习,舍弃了差解,算法的改进中可考虑修复差解,利用差解寻找最优解.

参考文献

[1] LIM H, LEE G M, SINGGHI I K. Multi-depot split-delivery vehicle routing problem [J]. IEEE Access, 2021, 9: 112206-112220.

[2] ZHANG Z Z, CHEANG B, LI C S, et al. Multi-commodity demand fulfillment via simultaneous pickup and delivery for a fast fashion retailer [J]. Computers & Operations Research, 2019, 103:81-96.

[3] HORNSTRA R P, SILVA A, ROODBERGEN K J, et al. The vehicle routing problem with simultaneous pickup and delivery and handling costs [J]. Computers & Operations Research, 2020, 115:104858.

[4] 胡小兵,孟相至. 动态灾害环境下多对多物资配送路径规划方法 [J]. 计算机工程与应用, 2022, 58(8):297-306.

HU X B, MENG X Z. Many-to-many path planning method for material distribution under dynamic disaster environment [J]. Computer Engineering and Applications, 2022, 58(8):297-306. (in Chinese)

[5] 王素欣,温恒,卢福强,等. 货物三维装箱问题建模及其乌鸦搜索算法优化 [J]. 湖南大学学报(自然科学版), 2020, 47(8):21-30.

WANG S X, WEN H, LU F Q, et al. Modeling of 3D cargo loading problem and optimization of crow search algorithm [J]. Journal of Hunan University (Natural Sciences), 2020, 47(8):21-30. (in Chinese)

[6] BERAHOU A, BENADADA Y. Dynamic vehicle routing problem with simultaneous delivery and pickup: formulation and

- resolution [C]//2020 5th International Conference on Logistics Operations Management (GOL). Rabat, Morocco: IEEE, 2021: 1-8.
- [7] 范厚明,刘鹏程,刘浩,等. 多中心联合配送模式下集货需求随机的 VRPSDP 问题[J]. 自动化学报,2021,47(7):1646-1660. FAN H M, LIU P C, LIU H, et al. The multi-depot vehicle routing problem with simultaneous deterministic delivery and stochastic pickup based on joint distribution [J]. Acta Automatica Sinica, 2021, 47(7): 1646-1660. (in Chinese)
- [8] DEGHAN M, HEJAZI S R, KARIMI-MAMAGHAN M, et al. Capacitated location routing problem with simultaneous pickup and delivery under the risk of disruption [J]. RAIRO - Operations Research, 2021, 55(3): 1371-1399.
- [9] 熊浩,郭吴颖,鄢慧丽,等. 考虑取送交叉和多种扰动因素的外卖配送路径优化研究[J]. 湖南大学学报(自然科学版), 2022, 49(10):92-102. XIONG H, GUO H Y, YAN H L, et al. Research on real-time route optimization of takeaway delivery considering pickup and delivery cross and multiple disturbance factors [J]. Journal of Hunan University (Natural Sciences), 2022, 49(10): 92-102. (in Chinese)
- [10] WANG X F. The heterogeneous fleet location routing problem with simultaneous pickup and delivery and overloads [J]. Discrete & Continuous Dynamical Systems - S, 2019, 12(4/5): 1147-1166.
- [11] ABDI A, ABDI A, AKBARPOUR N, et al. Innovative approaches to design and address green supply chain network with simultaneous pick-up and split delivery [J]. Journal of Cleaner Production, 2020, 250: 119437.
- [12] CASAZZA M, CESELLI A, WOLFLER CALVO R. A branch and price approach for the split pickup and split delivery VRP [J]. Electronic Notes in Discrete Mathematics, 2018, 69: 189-196.
- [13] BORTFELDT A, YI J M. The split delivery vehicle routing problem with three-dimensional loading constraints [J]. European Journal of Operational Research, 2020, 282(2): 545-558.
- [14] WANG Y, RAN L Y, GUAN X Y, et al. Multi-depot pickup and delivery problem with resource sharing [J]. Journal of Advanced Transportation, 2021, 12: 112-122.
- [15] AVCI M, TOPALOGU S. A hybrid metaheuristic algorithm for heterogeneous vehicle routing problem with simultaneous pickup and delivery [J]. Expert Systems with Applications, 2016, 53: 160-171.
- [16] XIA Y K, FU Z. A tabu search algorithm for distribution network optimization with discrete split deliveries and soft time windows [J]. Cluster Computing, 2019, 22(6): 15447-15457.
- [17] BOLAÑOS R I, ESCOBAR J W, ECHEVERRI M G. A metaheuristic algorithm for the multi-depot vehicle routing problem with heterogeneous fleet [J]. International Journal of Industrial Engineering Computations, 2018, 9(4): 461-478.
- [18] ABUALIGAH L, DIABAT A, MIRJALILI S, et al. The arithmetic optimization algorithm [J]. Computer Methods in Applied Mechanics and Engineering, 2021, 376: 113609.
- [19] DORIGO M, MANIEZZO V, COLORNI A. Ant system: optimization by a colony of cooperating agents [J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 1996, 26(1): 29-41.
- [20] MIRJALILI S, LEWIS A. The whale optimization algorithm [J]. Advances in Engineering Software, 2016, 95: 51-67.