

控制与决策

Control and Decision

考虑载重影响耗电率的电动车车辆路径问题

吴廷映, 孙灏

引用本文:

吴廷映, 孙灏. 考虑载重影响耗电率的电动车车辆路径问题[J]. *控制与决策*, 2023, 38(2): 483–491.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2021.1050>

您可能感兴趣的其他文章

Articles you may be interested in

基于自适应大规模邻域搜索算法的多车辆与多无人机协同配送方法

The cooperative delivery of multiple vehicles and multiple drones based on adaptive large neighborhood search

控制与决策. 2023, 38(1): 201–210 <https://doi.org/10.13195/j.kzyjc.2021.2268>

基于配送收益均衡的多目标绿色车辆路径优化算法

A multi-objective green vehicle routing optimization algorithm based on delivery benefit balance

控制与决策. 2023, 38(2): 365–371 <https://doi.org/10.13195/j.kzyjc.2021.0669>

考虑时变速度的多车场绿色车辆路径模型及优化算法

Multi-depot green vehicle routing model and its optimization algorithm with time-varying speed

控制与决策. 2022, 37(2): 473–482 <https://doi.org/10.13195/j.kzyjc.2020.1427>

面向冷链物流配送路径优化的知识型蚁群算法

Knowledge based ant colony algorithm for cold chain logistics distribution path optimization

控制与决策. 2022, 37(3): 545–554 <https://doi.org/10.13195/j.kzyjc.2021.0160>

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>

考虑载重影响耗电率的电动车车辆路径问题

吴廷映[†], 孙 灏

(上海大学 管理学院, 上海 200444)

摘要: 随着新能源和绿色物流等政策的出台, 电动车逐渐成为物流配送的主要运输工具. 考虑到电动车的电池容量、充电时间、耗电率和充电站位置等因素, 研究载重影响耗电率的电动车车辆路径问题, 建立以总成本最小化为目标的混合整数规划模型. 结合禁忌搜索算法的思想, 设计改进的自适应大邻域搜索算法对其求解, 在该算法中, 开发多种基于模型特性的破坏算子和修复算子以提高求解效率. 通过算例求解验证模型和算法的有效性, 为物流企业电动车配送方案的规划提供一定的决策依据.

关键词: 物流配送; 电动车车辆路径问题; 载重影响耗电率; 自适应大邻域搜索算法; 禁忌搜索

中图分类号: TP18; U492.2; F252

文献标志码: A

DOI: 10.13195/j.kzyjc.2021.1050

开放科学(资源服务)标识码(OSID):



引用格式: 吴廷映, 孙灏. 考虑载重影响耗电率的电动车车辆路径问题[J]. 控制与决策, 2023, 38(2): 483-491.

Electric vehicle routing problem with time window and linear weight-related discharging

WU Ting-ying[†], SUN Hao

(School of Management, Shanghai University, Shanghai 200444, China)

Abstract: With the introduction of new energy and green logistics policies, electric vehicles have gradually become the main means of logistics distribution. Taking into account the battery capacity, charging time, power consumption rate and charging station location of electric vehicles, this paper studies the electric vehicle routing problem with the impact of time window and weight-related discharging. The hybrid integer programming model with the goal of minimizing the total cost is established. Combined with the idea of the tabu search algorithm, an improved adaptive large neighbourhood search algorithm is designed to solve it. In this algorithm, a variety of destroy operators and repair operators based on the characteristics of the model are developed to improve the efficiency of the solution. The effectiveness of the model and the algorithm is verified through the calculation examples, and it provides a certain decision basis for the planning of the electric vehicle distribution plan of the logistics enterprise.

Keywords: logistics distribution; electric vehicle routing problem; weight-related discharging; adaptive large neighbourhood search algorithm; tabu search

0 引言

随着科技进步和电子商务的迅速发展, 网上购物走进了人们生活并成为重要的购物方式之一, 使得网上购物需求量大幅增加, 零散、小件的包裹数量激增. 据国家邮政局统计, 尽管受新冠疫情的影响, 但 2020 年我国快递业务量仍高达 833.6 亿件, 同比增长 31.2%, 连续 7 年位居世界第一, 占全球六成以上^[1]. 物流包裹量的增多对物流企业的揽收、仓储、中转、分包和配送等各环节的处理能力提出了巨大挑战.

以小批量多批次为特点的末端物流配送车辆给城市内的交通和空气环境造成较大影响, 车辆排放

问题愈发受到国家重视. 随着碳排放、绿色物流及新能源汽车政策的提出, 纯电动物流车逐渐成为物流配送的主流交通工具. 郭戈等^[2]强调交通电气化能够解决资源和环境等问题, 但仍需要通过车辆调度策略缓解交通拥堵, 电动车的续航和充电问题是必要考虑因素. 纯电动车辆路径问题(electric vehicle routing problem, EVRP)与传统燃油车车辆路径问题(vehicle routing problem, VRP)不同, 除了考虑车辆容量、客户位置和时间窗等, 受电池容量和充电方式的影响, 还要考虑充电站位置、充电距离及充电时间等不可忽视的因素, 使得 EVRP 比 VRP 更为复杂.

自 1959 年 Dantzig 等^[3]提出了 VRP 问题, 便受到

国内外许多学者的关注. 随着绿色低碳的理念逐渐得到重视, 国内外学者开始研究绿色车辆路径问题 (green vehicle routing problem, GVRP), 其中 Erdoğn 等^[4]的 GVRP 模型对之后的 EVRP 有重要的指导意义. 随着技术的更新, 纯电动车在物流行业的应用逐渐普遍, EVRP 逐渐成为研究热点. Schneider 等^[5]基于 Solomon 实例生成的 EVRPTW 算例为后续研究提供了参考依据, 并通过可变邻域搜索算法与禁忌搜索算法相结合的方法求解; 此后, EVRP 相关研究逐渐扩展, Verma^[6]研究了快充、慢充和换电等多种方式选择的 EVRP; Erdelić 等^[7]对比了一条配送路径采取单次充电和多次充电策略的区别; Hiermann 等^[8]研究了混合车型和不同车队规模的 EVRP; Keskin 等^[9]用排队论处理充电桩数量有限及非线性分段充电率的 EVRP; 杨珺等^[10]、Schiffer 等^[11]研究了考虑充电站或换电站选址的 EVRP.

EVPR 是 VRP 的扩展, 属于 NP-hard 问题, 对这类问题的求解方法分为精确算法和启发式算法两类. 揭婉晨等^[12-13]使用分支定价方法研究了单车型和多车型 EVRP 问题; Desaulniers 等^[14]使用分支定价求解了一次充电、多次充电、满充和部分充电等不同策略的 EVRP 方案; Felipe 等^[15]使用局部搜索和模拟退火算法求解部分充电的 EVRP; Shao 等^[16]构建了混合遗传算法求解 EVRP; Cortés-Murcia 等^[17]构造了局部迭代搜索元启发式框架和变邻域下降算法求解部分充电的 EVRP, 提出合理利用充电时间, 如通过步行等方式配送距充电站较近的客户; Macrina 等^[18]考虑了加速减速制动阶段的能耗问题, 使用大邻域搜索算法求解; 周鲜成等^[19]使用蚁群算法求解了综合油耗、碳排放、车辆使用成本等最小化为目标的模型; 李国明等^[20]设计了改进的混合禁忌搜索算法求解随机 VRP. 还有一部分研究验证了自适应大邻域搜索算法 (adaptive large neighborhood search, ALNS) 求解此类问题的效果, 如文献 [21-23], ALNS 运用破坏算子和修复算子最大限度搜索解空间, 依据一定准则接收新解和自适应的选择算子, 具有搜索空间大、效率高等特点.

在以往的 EVRP 研究中, 很少有文章考虑载重量对电量消耗率的影响. 然而, 在同等行驶条件下载重量对耗电量有较大影响, 改变车辆的续航里程, 从而影响客户节点的服务顺序、充电站的选择及充电时间等. 忽略载重对电量消耗的影响, 会降低方案在实际问题中的适用性. Kancharla 等^[24-25]曾对该问题展开研究, 考虑载重与耗电量之间的线性关系并构建了能耗最小为目标的数学模型, 对比以距离

最小和能耗最小为目标时产生的能耗值, 但是没有深入定义载重影响的耗电量与其他变量之间的数学关系和取值, 没有综合考虑多因素影响的问题决策. 本研究结合实际情况, 聚焦于载重影响耗电率的电动车车辆路径问题 (EVPR with time window and linear weight-related discharging, EVRPTW-LWD), 将在耗电量非线性变化、车辆容量有限、客户时间窗以及充电站固定的情况下, 分析电动车载重量与耗电量间的数学关系, 建立总成本最小化的数学模型, 设计改进的 ALNS 算法 (hybrid ALNS and tabu search algorithm, HALNS-TS) 决策客户配送方案和车辆充电方案, 从而为降低物流公司运营成本, 改善城市交通拥堵和车辆碳排放引起的环境污染问题提供一定的参考依据.

本文的创新点如下: 1) 首次以固定用车成本、路径行驶成本及车辆充电成本为优化目标, 将载重影响耗电率的复杂现实情况考虑其中, 构建耗电量与载重之间的数学关系以及相应的 EVRPTW-LWD 数学模型; 2) 设计 HALNS-TS 算法, 加入禁忌搜索算法 (tabu search, TS) 的思想, 基于问题特性构造多个破坏算子和修复算子, 通过算例求解说明问题模型的有效性, 验证载重相关的耗电量对路径规划的重要影响以及算法的效果和稳定性.

1 数学模型构建

1.1 问题描述

本文构造的配送场景如图 1 所示, 假设仅有一个配送中心, 由同类型的电动车为客户提供服务, 电动车保持匀速行驶, 从配送中心出发, 完成任务后返回配送中心. 车辆载货量不得超过其最大容量, 到达客户点时的运载量应满足客户的需求. 到达客户点的时间应满足客户时间窗的要求, 本文将时间窗设置为客户可接受服务的硬性时间约束, 早于客户时间窗到达需进行等待, 晚于时间窗到达则不得为客户提供配送服务. 电动车为固定充电率的线性充电方式, 耗电量由固定耗电率和载重影响的耗电量两部分组成, 车辆行驶过程中需保证充足的电量到达下一节点. 充电站仅有一种充电方式, 能同时为多辆电动车提供服务, 且每次充电后, 电动车达到满电状态.

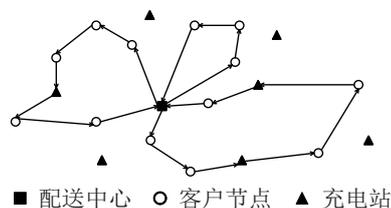


图 1 配送场景网络

模型中的相关参数和变量定义如下.

$D = \{0, N + 1\}$: 起始和终止配送中心集合;

$F = \{f_1, f_2, \dots, f_m\}$: 充电站集合;

$EV = \{ev_1, ev_2, \dots, ev_k\}$: 电动车集合;

B_0 : 电动车的固定用车成本;

B_1 : 电动车单位运距的行驶成本;

B_2 : 电动车单位时间的充电成本;

C : 电动车最大载货量;

v : 电动车匀速行驶的速度;

Q : 电动车电池最大容量;

g : 电动车固定充电率(单位电量的充电时间);

h : 电动车固定耗电率(单位运距的耗电量);

p : 电动车载重影响的耗电率(单位运距运量的耗电量);

$V = \{v_1, v_2, \dots, v_n\}$: 客户集合;

q_i : 客户点 i 的需求;

d_{ij} : 客户点 i 与 j 之间的距离;

$[e_i, l_i]$: 客户可接受服务的时间窗范围;

x_{ij} : 0-1 变量, 电动车经过弧 (i, j) 时为 1;

u_i : 电动车到达客户点 i 的载货量;

t_i : 电动车到达客户点 i 的时间;

y_i : 电动车到达客户点 i 时的剩余电量;

s_j : 电动车在充电站 j 的充电时间.

此外, 基于以上定义, 模型使用的集合分别为 $V_0 = 0 \cup V, V_p = F \cup V, V_{N+1} = V \cup N + 1, V_{0p} = 0 \cup F \cup V, V_{pN+1} = F \cup V \cup N + 1, V_{0pN+1} = 0 \cup F \cup V \cup N + 1$.

1.2 车辆载重与耗电率的关系

本文参考文献[26]对车辆载重与耗电率间的数学关系进行定义. 电动车的重量由车辆自重 W_0 和载重 W_1 两部分组成, 假设其与耗电率之间为线性关系, 则电动车的耗电率与车辆重量之间的函数关系可定义为 $P(W_1) = a(W_0 + W_1) + b$.

电动车满载或空载行驶时, 得到以下方程组:

$$\begin{cases} P^* = a(W_0 + W) + b, \\ P_0 = aW_0 + b. \end{cases}$$

其中: P^* 为车辆满载时耗电率, P_0 为车辆空载时耗电率, W 为车辆额定的最大载重. 综合方程组求解可得到 a 和 b 的值分别为 $a = (P^* - P_0)/W, b = P_0 - W_0(P^* - P_0)/W$.

将 a 和 b 代回等式 $P(W_1) = a(W_0 + W_1) + b$ 中, 得到 $P(W_1) = P_0 + W_1(P^* - P_0)/W$, 即考虑载重影响耗电率时, 电动车耗电率与 P_0, P^* 及 W 有关. 结合模型的参数定义, 可得空载时的固定耗电率 $h = P_0$, 载重影响的耗电率 $p = (P^* - P_0)/W$.

1.3 EVRPTW-LWD 数学模型

参考 Kancharla 等^[24] 研究的 EVRP 模型, 本文构建的 EVRPTW-LWD 模型如下:

$$\min B_0 \sum_{j \in V_p} x_{0j} + B_1 \sum_{i \in V_{0p}} \sum_{j \in V_{pN+1}} d_{ij} x_{ij} + B_2 \sum_{i \in V_{0p}} \sum_{j \in F} s_j x_{ij}. \quad (1)$$

$$\text{s.t.} \quad \sum_{i \in V_{0p}, i \neq j} x_{ij} - \sum_{i' \in V_{pN+1}, i' \neq j} x_{ji'} = 0, \quad \forall j \in V_p, i \neq i'; \quad (2)$$

$$\sum_{j \in V_{pN+1}, i \neq j} x_{ij} = 1, \quad \forall i \in V; \quad (3)$$

$$\sum_{j \in V_{pN+1}, i \neq j} x_{ij} \leq 1, \quad \forall i \in F; \quad (4)$$

$$q_j \leq u_j \leq u_i - x_{ij} q_i + C(1 - X_{ij}), \quad \forall i \in V_{0p}, j \in V_{pN+1}, i \neq j; \quad (5)$$

$$0 \leq u_0 \leq C; \quad (6)$$

$$\max(t_i, e_i) + x_{ij} \left(s_i + \frac{d_{ij}}{v} \right) - l_0(1 - x_{ij}) \leq t_j \leq l_j, \quad \forall i \in V_0, j \in V_{pN+1}, i \neq j; \quad (7)$$

$$\max(t_i, e_i) + x_{ij} \frac{d_{ij}}{v} - (Q - y_i)g - (l_0 + gQ)(1 - x_{ij}) \leq t_j \leq l_j, \quad \forall i \in F, j \in V_{pN+1}, i \neq j; \quad (8)$$

$$0 \leq y_j \leq y_i - d_{ij}(hx_{ij} + pu_j) + Q(1 - x_{ij}), \quad \forall i \in V, j \in V_{pN+1}, i \neq j; \quad (9)$$

$$0 \leq y_j \leq Q - d_{ij}(hx_{ij} + pu_j), \quad \forall i \in F \cup 0, j \in V_{pN+1}, i \neq j. \quad (10)$$

目标函数(1)表示最小化总成本, 各项分别为电动车的固定成本、路径行驶成本、充电成本, 其中充电成本与电动车在充电站的充电时间有关; 约束(2)表示客户节点或充电站节点进出流量平衡; 约束(3)表示每个客户节点均只被访问一次, 离开每个客户节点的车辆只有一辆且只有一个方向; 约束(4)表示单个充电站可同时为多个车辆提供充电服务; 约束(5)和(6)为车辆容量约束, 即从配送中心出发时的载货量既需要满足路径总需求又不能超过车辆容量限制, 到达各节点时的载货量应满足节点需求; 约束(7)和(8)为时间约束, 到达时间与前一节点的到达时间、服务时间及该段路径长度有关, 电动车可早于客户时间窗到达, 但应在时间窗内为客户提供配送服务; 约束(9)和(10)为车辆电量约束, 剩余电量与前一节点剩余电量、路径固定耗电量和载重耗电量有关, 既要满足车辆电池容量限制, 又要保证有足够的电量到达下一节点, 且电动车从配送中心出发或途中经过充电站后的

电量为满电状态.

2 改进的自适应大邻域搜索算法

EVRPTW-LWD为EVRP问题的扩展,属于NP-hard问题,精确算法和CPLEX等优化求解器难以在可接受的时间内求解该问题的大规模实例,因此,本文针对问题设计了HALNS-TS算法.

HALNS-TS算法框架如图2所示.首先,构造 k 阶时间最近邻启发式算法(K -time oriented nearest neighbor heuristic, K -TONNH)生成较高质量的初始解;其次,设计随机破坏、最差节点破坏和邻域破坏3个破坏算子,以及随机修复、最优节点修复两个修复算子,运行中随机选择破坏和修复算子对当前解进行操作;再次,结合禁忌搜索算法中禁忌表的思想,改进新解的判断和记录过程,避免过早陷入局部最优解;最后,构建合适的接收准则,自适应地调整算子权重并更新当前解currentSol和最优解bestSol,设置最大迭代次数、最长运行时间等作为算法的终止条件.

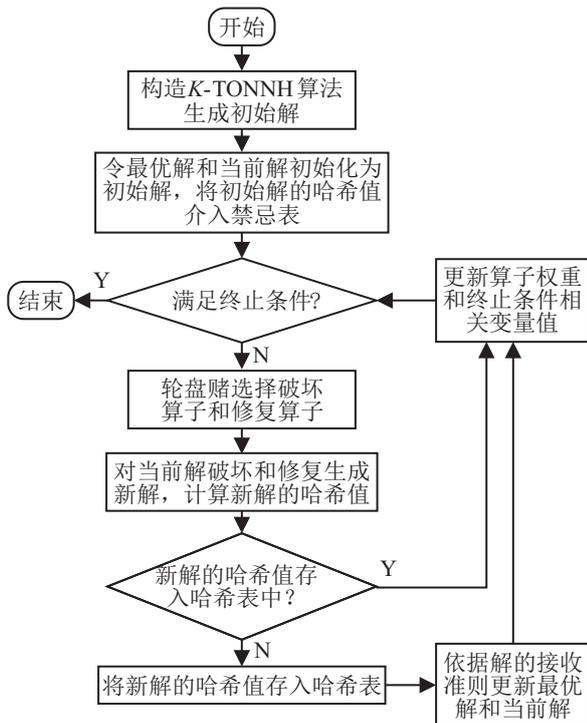


图2 HALNS-TS算法框架

下面详细介绍初始解构造算法、算子设计、禁忌操作、解的接收准则和自适应过程的具体实现.

2.1 初始解的构造

初始解的质量对后续算法的优化运算具有较大影响,较优的初始解能够以较大的概率获得满意解.本文借鉴文献[7,15]的思路,构建改进的 K -TONNH算法.该算法目标函数 $cost = \theta_1 d_{ij} + \theta_2 t_{rec}$ ($\theta_1 + \theta_2 = 1$)同时考虑增加的行驶距离和充电时间两个不同维度,保证解可行的前提下,在目标

值较小的前 k 个插入方案中随机选择,从而生成不同于其他算法的高质量初始解.

算法1: 初始解的构造 (K -TONNH)

```

初始化 Sol, tempSol = Sol,  $C_{nI} = V, C_s = \emptyset$ ;
为 Sol 开辟一条新路径 Sol.insertNewRoute(0,0), 令 pos = 1;
while  $C_{nI} \neq \emptyset$ 
for  $C_{nI}$  中各客户点  $c$ 
令  $r$  为 tempSol 最后一条路径, 将客户  $c$  插入  $r$  位置 pos 上;
if tempSol.isFeasibleCapacityAndTimeWindowAndElectricity()
 $C_s$ .pushback( $c, 0, cost$ );
else if tempSol.isFeasibleCapacityAndTimeWindow()
寻找末位充电站后满足电量约束的充电站  $f$  及  $pos'$ , 插入  $r$ ;
if tempSol.isFeasibleCapacityAndTimeWindowAndElectricity()
 $C_s$ .pushback( $c, 1, cost, f, pos'$ );
else if  $r.size()=3$ 
 $C_s$ .pushback( $c, 2, cost, c.nearestStation()$ ); //  $c$  前后各插 1 个充电站
end if
end if
tempSol = Sol;
end for
if  $C_s \neq \emptyset$ 
按 cost 值对集合  $C_s$  升序排列, 随机选取前  $k$  个方案中的 1 个;
更新 Sol, 使 pos 指向 Sol 最后一条路径的最后 1 个客户点;
else
为 Sol 开辟一条新路径, 令 pos = 1;
end if
清除  $C_{nI}$  中刚被插入 Sol 的客户, 清空集合  $C_s$ ;
end while
    
```

2.2 算子设计

本文针对所提出的EVRPTW-LWD问题设计了随机破坏、最差节点破坏、邻域破坏3个破坏算子和随机修复、最优节点修复两个修复算子,在迭代过程中用于currentSol的改进.每进行一次破坏或修复操作,需要对currentSol中冗余的充电站进行移除.

破坏算子: 依据客户总数随机确定破坏的客户个数 $k = rand(customerNum)$, 破坏当前解currentSol中 k 个客户,将被破坏的客户存入未被插入客户集合 C_{nI} ,并移除无效路径和多余的充电站.破坏操作能增大解的不确定性,有效跳出局部最优解,深度搜索解空间,为后续的修复算子生成新解提供更大可能性.

1) 随机破坏算子,对当前解currentSol路径上的客户点随机破坏.

算法2: 随机破坏

```

初始化  $i = 0$ ;
while  $i < k$ 
随机选择要破坏的路径  $r$ , 在路径  $r$  上随机破坏 1 个客户点  $c$ ;
将客户点  $c$  插入待插入客户集  $C_{nI}$ ;
若路径  $r$  无客户节点, 删除路径  $r$ ;
 $i++$ ;
end while
    
```

2) 最差客户点破坏算子,依次破坏总成本增加最大的客户点.

算法3: 最差节点破坏

```

初始化 cost, cost', Δcost = cost - cost', Ctemp, i = 0;
while i < k
  for currentSol 中的各客户节点
    计算单独移除该客户节点的 Δcost;
    将客户编号 id、路径 r、位置 pos 和 Δcost 存入 Ctemp;
  end for
  按 Δcost 值对 Ctemp 降序排列, 前 k 个中随机移除客户 c;
  将客户点 c 插入待插入客户集 CNI;
  若路径 r 无客户节点, 则删除路径 r;
  清空集合 Ctemp;
  i++;
end while

```

3) 邻域破坏算子, 随机确定一个中心客户点, 同距离该客户点最近的 $k - 1$ 个客户点一起被破坏。

算法4: 邻域破坏

```

初始化 i = 0, CnearCust;
在 currentSol 中随机选择一个中心客户节点 c;
距 c 最近 k - 1 个客户及路径 r、位置 pos 存入 CnearCust;
for CnearCust 中的客户点 j
  从 currentSol 中移除客户 j, 将 j 存入 CNI;
  若 j 所在的路径 r 无其他客户点, 删除路径 r;
end for

```

修复算子: 将未被插入客户集 C_{NI} 中的客户点按修复算子的规则依次插入路径中. 修复算子旨在进一步对当前解进行优化, 为后续解的优化实现更多选择。

1) 随机修复, 对集合 C_{NI} 中的客户点依次操作, 在已破坏的解中随机选择一个满足车辆容量和客户时间窗条件的位置插入, 并对电量不满足约束的情况插入充电站进行修复。

算法5: 随机修复

```

GenericStep1: r = rt, 从 CNI 中删除客户 c;
GenericStep2: currentSol 开辟新路径插入 c, CNI 中删除 c;
while CNI ≠ ∅
  取集合 CNI 中的第 1 个客户点 c;
  随机选择插入路径 r 和位置 pos;
  令 rt = r;
  在路径 rt 的位置 pos 插入客户 c;
  if rt.isFeasibleCapacityAndTimeWindowAndElectricity()
    GenericStep1;
  else if rt.isFeasibleCapacityAndTimeWindow()
    寻找路径成本增加最少且满足电量约束的充电站 f 及位置 pos', 插入路径 rt;
  if rt.isFeasibleCapacityAndTimeWindowAndElectricity()
    GenericStep1;
  else;
    GenericStep2;
  end if;
  GenericStep2;
end if
end while

```

2) 最佳节点修复, 依次将集合 C_{NI} 中的客户点插入满足容量和时间窗约束以及充电站修复成本最少的位置。

算法6: 最佳节点修复

```

初始化 Δcost = cost - cost', Ctemp, tempSol = currentSol;
while CNI ≠ ∅
  取集合 CNI 中的第 1 个客户点 c;
  for currentSol 的各路径 r 的各位置 pos
    客户 c 插入 tempSol 的路径 r 的位置 pos;
  if tempSol.isFeasibleCapacityAndTimeWindowAndElectricity()
    客户 c、路径 r、位置 pos 和成本差 Δcost 存入 Ctemp;
  else if tempSol.isFeasibleCapacityAndTimeWindow()
    寻找使距离增加最少且满足电量约束的充电站 f 和位置 pos',
    并将 f、pos'、c、r、pos 和 Δcost 存入集合 Ctemp;
    若无合适的充电站则开辟新路径, 将 c、r、pos、Δcost 及必要的充电站 f 和 pos' 存入集合 Ctemp;
  else
    开辟新路径, 将 c、r、pos、Δcost 及必要的充电站 f 和 pos' 存入集合 Ctemp;
  end if
  tempSol = currentSol;
end for
按 Δcost 值对集合 Ctemp 升序排序, 并选择第 1 个客户 c 方案;
从集合 CNI 中删除客户 c, 清空集合 Ctemp;
end while

```

2.3 禁忌操作

禁忌操作参考了禁忌搜索算法中禁忌表的思想, 通过设置禁忌表存储出现过的解的哈希值, 并将其运用在新解的判断步骤, 旨在有效筛查出真正的新解, 避免对出现过的解重复优化而陷入局部最优. 本文设定禁忌表中哈希值的计算与解的路径集中的节点排列顺序以及节点编号有关, 计算公式为 $\text{hash} = \sum_{i \in R} \sum_{j \in r} (\text{hash} \times 1009 + \text{ID}_{ij})$. 对每进行一组破坏和修复操作生成的新解先进行哈希值计算, 与禁忌表记录对比, 确认新解从未出现才继续下一步操作。

2.4 解的接收准则

接收准则应用于评估禁忌操作过滤后的新解 newSol, 将 newSol 与 currentSol 和 bestSol 进行对比, 依据 newSol 的优劣程度自动选取参与下一轮迭代的解, 并更新最优解. 最后为算子权重的自适应调整提供相应的调节参数, 其中, 算法7中权重值的大小关系为 $\text{weight}_1 > \text{weight}_2 > \text{weight}_3 > \text{weight}_4$.

算法7: 解的接收准则

```

初始化 w = 0; //w 为算子权重更新的依据
if newSol < bestSol
  bestSol = newSol, currentSol = newSol;
  //新解作为最优解和当前解参与下一次的解的改进
  w = weight1;
else if newSol > bestSol && newSol < currentSol
  currentSol = newSol;
  //新解为下次破坏和修复的初始解
  w = weight2;
else if newSol > bestSol && newSol > currentSol
  若 rand() > p, currentSol = newSol;
  w = weight3; //一定概率下接收新解
else
  w = weight4; //仍使用 currentSol 作为下次迭代的初始解
end if

```

2.5 自适应设计

1) 算子权重的更新. 算子权重表示算子被选中的概率, 权重越大的算子说明在以往迭代过程中表现越好, 有更大的可能性探索到未出现过的解, 其被选中的概率越大. 破坏算子和修复算子的初始权重平均分配, 迭代过程中依据接受准则传回的 $weight_x$ 值对上轮算子作适应性调整, 即 $w_n = \lambda_1 w_n + \lambda_2 weight_x$ (其中 $\lambda_1 + \lambda_2 = 1$), 标准化的算子权重为 $w_{n'} = w_n / \sum_{n \in N} w_n$.

2) 轮盘赌法则选择算子. 轮盘赌法则应用于迭代中对破坏和修复算子的选择, 随机生成 $0 \sim 1$ 的概率数 P , 当算子累计权重大于 P 时, 即 $\sum_{n=1}^N w_n \geq P$ 时选择第 n 个算子.

3 算例分析

为验证 HALNS-TS 算法的求解效率和问题的实际应用意义, 本文做了以下 3 部分实验: 1) 使用 CPLEX 和 HALNS-TS 算法对 5、10、15 个客户点的 36 个小规模算例求解, 验证模型和算法的有效性; 2) 对载重影响的耗电率做灵敏度分析, 对比载重对耗电率不同影响情况下的车辆路径规划方案成本组成, 说明其在路径规划中的重要影响; 3) 使用 HALNS-TS 算法对 100 个客户点的 56 个大规模算例求解, 验证算法在大规模算例中的稳定表现.

3.1 算例介绍与参数设置

本文的 EVRPTW 相关算例是基于文献 [7] 在 Solomon 算例基础上生成的. 该组算例有聚类分布 C、随机分布 R 及随机聚类分布 RC 三种类型, 且每类算例都具有两个子类, 即窄客户时间窗 ($c1^{**}, r1^{**}, rc1^{**}$) 和宽客户时间窗 ($c2^{**}, r2^{**}, rc2^{**}$). 在算例实验中, 模型的常数系数设置如下: $B_0 = 100, B_1 = 1, B_2 = 1, h = P_0 = 1, P^* = 2$. 载重相关的耗电率 $p = (P^* - P_0) / W$ 与不同算例中车型的额定载重量有关, 将算例的额定容量 C 视为额定载重量 W . 算法中更新权重的参数设置为 $\lambda_1 = 0.8, \lambda_2 = 0.2$, CPLEX 最大运行时间为 3 600 s, 大规模算例设置算法终止条件为最大迭代次数 300 次, 最长运行时间 1 000 s, 最优解连续未改进次数为 30 次.

3.2 实验结果与对比分析

本文实验计算机配置为 Inter Core I7-8550U 2 GHz、4 核、8 GB 内存, IDE 采用 Visio Studio 2019 和 CPLEX 12.8, 使用 C++ 语言. 在表 1 ~ 表 5 中, 将数据表列示的信息简写为: 算例名称 AC、电池容量 Q 、充电率 g 、路径条数 RN、总成本 TC、运行时间 RT.

3.2.1 模型和算法的验证

将 CPLEX 在 3 600 s 内求得的解与 HALNS-TS 结果对比以验证 HALNS-TS 算法的求解效果. 其中 $Gap = (TC(HALNS-TS) - TC(CPLEX)) / TC(CPLEX)$ 为负表示 HALNS-TS 算法求解效果优于 CPLEX.

CPLEX 和 HALNS-TS 对 36 个小规模算例求解的结果如表 1 所示, 粗体表示两个求解方法对比下更优的运行结果.

表 1 小规模算例求解结果对比

AC	CPLEX			HALNS-TS			Gap/%
	RN	TC	RT	RN	TC	RT	
c101-5	3	831.21	30.38	4	831.82	0.291	0.07
c103-5	3	484.50	2.70	3	484.50	0.60	0.00
c206-5	4	757.11	36.84	4	757.11	1.29	0.00
c208-5	1	657.93	143.78	1	657.93	0.08	0.00
r104-5	2	378.08	1453.48	2	378.08	0.11	0.00
r105-5	2	449.35	3 600	3	498.52	0.09	10.94
r202-5	1	287.89	3 600	1	287.89	6.46	0.00
r203-5	1	354.81	3 600	1	354.81	10.26	0.00
rc105-5	3	561.42	3 600	2	513.76	0.82	-8.49
rc108-5	3	685.99	3 600	2	521.23	0.69	-24.02
rc204-5	1	321.17	3 600	1	332.72	0.51	3.60
rc208-5	1	315.92	3 600	1	315.92	0.63	0.00
c101-10	4	1 613.01	3 600	5	1 550.21	4.99	-3.89
c104-10	6	1 201.37	3 600	3	1 163.37	7.35	-3.16
c202-10	4	846.99	3 600	3	814.21	30.74	-3.87
c205-10	3	814.64	3 600	3	814.64	10.72	0.00
r102-10	4	737.11	3 600	4	717.90	17.16	-2.61
r103-10	3	545.07	3 600	2	468.15	27.36	-14.11
r201-10	2	503.11	3 600	2	503.11	7.21	0.00
r203-10	1	498.04	3 600	1	470.82	4.23	-5.47
rc102-10	6	1 143.86	3 600	6	1 127.24	3.56	-1.45
rc108-10	4	863.81	3 600	3	804.38	2.07	-6.88
rc201-10	2	615.84	3 600	2	645.53	6.12	4.82
rc205-10	2	607.14	3 600	2	607.14	74.72	0.00
c103-15	6	1 767.01	3 600	5	1 444.30	2.43	-18.26
c106-15	5	1 019.86	3 600	5	1 019.86	5.82	0.00
c202-15	4	1 357.02	3 600	4	1 346.07	7.15	-0.81
c208-15	3	1 149.69	3 600	3	1 149.69	67.40	0.00
r102-15	6	1 155.93	3 600	6	1 110.25	1.82	-3.95
r105-15	5	1 005.27	3 600	4	909.25	4.07	-9.55
r202-15	3	816.28	3 600	3	789.47	138.99	-3.28
r209-15	2	634.32	3 600	2	625.41	53.82	-1.40
rc103-15	5	1 047.71	3 600	4	886.60	4.43	-15.38
rc108-15	5	1 090.70	3 600	4	1 004.33	8.94	-7.92
rc202-15	2	774.92	3 600	2	729.02	76.15	-5.92
rc204-15	2	671.97	3 600	2	603.71	76.22	-10.16

从表 1 可以看出, HALNS-TS 的求解效果较好: 有 21 个算例的结果比 CPLEX 结果更优, 成本相差最大达 24.02%; 有 11 个算例的结果与 CPLEX 结果相同; 仅有 4 个算例的求解结果略差于 CPLEX, 目标函数值最低偏差为 0.07%. 就求解时间而言, HALNS-TS 的运行时间远远小于 CPLEX.

由于 EVRPTW-LWD 是 NP-hard 问题, 随着算例规模的增大, 精确算法求解时间呈指数增长.

HALNS-TS能不断地在大规模邻域中寻找更优的可行解,使用禁忌表保存更优的解,通过轮盘赌法选择算子并自适应调整算子权重.此外HALNS-TS加入了扰动机制,在一定概率下接受较差的新解,避免算法较快陷入局部最优解.因此,HALNS-TS既能够高效地对解空间进行搜索,又能够生成更优的可行解.从小规模算例的运算结果可以看出,HALNS-TS算法较CPLEX更优的方案对应的路径条数更少,说明其有效优化了路径数量,即模型中对应的固定用车

成本.因而,HALNS-TS在该问题的求解中表现较好,能够有效解决此类问题.

3.2.2 载重耗电率灵敏度分析

本部分以100个客户点的c101、r101和rc101算例为例,分别计算耗电率为 $0.0 \times p$ 、 $1.1 \times p$ 、 $1.2 \times p$ 、 $1.3 \times p$ 等的配送方案成本组成,分析不同耗电率对不同类型算例路径规划的影响.

载重对耗电率不同影响程度下的配送组成成本如表2所示.

表2 不同的载重耗电率对路径规划的影响

p'	c101				r101				rc101			
	RN	DC	RC	TC	RN	DC	RC	TC	RN	DC	RC	TC
$0.0 \times p$	17	1410.13	763.44	3873.57	22	2087.99	462.34	4750.33	21	2227.78	356.37	4684.15
$1.0 \times p$	22	1812.16	1890.36	5902.52	25	2144.39	596.71	5241.10	23	2438.59	427.62	5166.21
$1.1 \times p$	26	1982.12	2323.31	6905.43	27	2207.69	519.64	5427.33	23	2494.86	555.19	5350.05
$1.2 \times p$	24	1929.17	2468.62	6797.79	25	2193.53	629.37	5322.90	24	2513.48	563.16	5476.64
$1.3 \times p$	26	2064.43	2798.50	7462.93	27	2263.59	592.19	5555.78	23	2561.98	628.30	5490.28
$1.4 \times p$	25	2118.97	2792.75	7411.72	27	2197.41	624.81	5522.21	24	2410.90	582.50	5393.40
$1.5 \times p$	29	2278.04	2381.06	7559.10	27	2259.88	646.85	5606.73	25	2506.00	583.36	5589.36
$1.6 \times p$	29	2297.35	2394.88	7592.23	28	2270.30	645.66	5715.96	25	2554.27	608.37	5662.64
$1.7 \times p$	30	2303.59	2406.11	7709.70	26	2354.36	844.11	5798.47	26	2601.21	669.87	5871.08
$1.8 \times p$	27	2351.99	3712.51	8764.50	29	2226.91	649.60	5776.51	26	2646.22	647.34	5893.55
$1.9 \times p$	25	2140.64	3585.39	8226.03	27	2384.73	823.67	5908.40	26	2612.16	720.88	5933.03
$2.0 \times p$	29	2318.75	3316.16	8534.91	29	2354.47	756.39	6010.87	26	2761.64	751.34	6112.98

在表2中: p' 为调整后的载重耗电率,DC为路径行驶成本,RC为充电成本.不考虑 p' 值的解决方案,其路径条数、充电成本、路径行驶成本都远小于考虑载重耗电率的解决方案.在不同类型的算例中,随着 p' 的增大,车辆载货状态下的续航里程减少,通过增加用车数量、充电次数以规划较优的方案,使得固定成本、路径行驶的距离成本和充电时间成本增加,方案总成本增大.其中,C类算例充电率较高,耗电率轻微变化对路径规划结果造成较大的影响.因此,在进行电动车路径规划时,考虑客户需求和车辆载重影响的耗电率是十分必要的.

3.2.3 大规模算例求解

100个客户点的算例测试结果如表3~表5所示,HALNS-TS能够对C、R、RC三种不同类型大规模算例进行有效求解,说明其能解决不同场景的大规模实例,在实际应用中具有一定的参考意义.

类型C的17个算例的结果如表3所示,运行时间约200s~400s,其中c208运行时间仅有180.74s. HALNS-TS能够求解聚类的大规模客户群,说明HALNS-TS在实际应用中可针对区域内呈社区分布特征的实例求解.

类型R的23个算例的运行结果如表4所示,运行时间约为400s~500s,较长于类型C的运行时

间. HALNS-TS能够求解随机分布的大规模客户群,说明HALNS-TS在应用中可适用于零散分布的客户群或物流企业中转站之间的配送路径规划问题.

类型RC的16个算例的运行结果如表5所示,运行时间约400s~500s,与类型R相当,较长于类型C. HALNS-TS能够求解随机聚类型的大规模客户群.该结果表明,HALNS-TS可同时实现零散的客户以及社区型客户分布特征的路径规划.

表3 C类100个客户点算例求解结果

AC	Q	g	RN	TC	RT
c101	79.69	3.39	22	5902.5	260.77
c102	79.69	3.39	20	6262.8	284.51
c103	79.69	3.39	19	5456.6	275.21
c104	79.69	3.39	16	5565.7	310.97
c105	79.69	3.39	20	5685.0	293.82
c106	79.69	3.39	22	6296.7	471.17
c107	79.69	3.39	22	6847.4	251.41
c108	79.69	3.39	21	5947.8	375.72
c109	79.69	3.39	20	6276.8	323.51
c201	118.31	2.28	10	2093.9	309.27
c202	118.31	2.28	9	2287.8	311.79
c203	118.23	2.28	10	2145.6	387.84
c204	118.12	2.29	8	2227.6	348.31
c205	117.78	2.29	10	2562.6	383.11
c206	117.70	2.29	10	2371.9	400.81
c207	117.66	2.29	9	2428.7	409.07
c208	117.66	2.29	10	2561.0	180.74

表4 R类100个客户点算例求解结果

AC	Q	g	RN	TC	RT
r101	62.14	0.48	25	5 241.10	367.92
r102	62.14	0.48	25	5 244.18	375.21
r103	62.14	0.48	20	4 333.36	475.22
r104	67.15	0.45	18	3 694.07	368.30
r105	62.14	0.48	21	4 620.02	445.65
r106	62.60	0.48	21	4 790.95	491.61
r107	66.28	0.45	16	3 612.56	434.14
r108	64.06	0.47	15	3 557.25	471.99
r109	65.17	0.46	18	4 109.95	438.15
r110	67.12	0.45	18	3 703.88	496.91
r111	65.80	0.46	18	3 840.60	779.40
r112	65.48	0.46	16	3 684.71	493.24
r201	187.86	0.16	8	2 191.12	501.78
r202	238.34	0.13	5	1 726.11	415.45
r203	187.90	0.16	7	1 875.07	500.67
r204	247.66	0.12	5	1 643.68	452.51
r205	198.88	0.15	7	2 035.34	516.79
r206	181.23	0.17	6	1 901.01	507.66
r207	267.18	0.11	5	1 522.20	438.39
r208	218.03	0.14	4	1 327.47	449.63
r209	181.83	0.16	5	2 204.83	474.16
r210	187.87	0.16	7	2 048.78	575.68
r211	265.71	0.11	8	2 022.08	441.29

表5 RC类100个客户点算例求解结果

AC	Q	g	RN	TC	RT
rc101	79.69	0.38	23	5 166.21	501.74
rc102	79.69	0.38	20	4 848.60	553.32
rc103	79.69	0.38	18	4 204.15	487.83
rc104	79.69	0.38	17	4 009.65	510.63
rc105	79.69	0.38	20	4 767.51	563.13
rc106	79.69	0.38	19	4 622.19	478.99
rc107	79.69	0.38	19	4 556.72	425.02
rc108	79.69	0.38	16	3 936.07	499.21
rc201	211.04	0.14	10	2 801.00	384.00
rc202	273.13	0.11	8	2 337.43	463.06
rc203	209.92	0.14	6	2 237.95	629.98
rc204	159.68	0.19	5	1 765.62	394.44
rc205	194.58	0.15	7	1 991.69	431.37
rc206	229.26	0.13	6	1 903.61	579.14
rc207	212.23	0.14	6	2 174.41	432.88
rc208	165.63	0.18	6	2 251.57	507.01

综合表3~表5的数据可以看出,运行时间大都在400s左右,是实际问题中可被接受的运算速度,且相同类型的算例运算结果相差不大,HALNS-TS在大规模例子求解中表现稳定.此外,对比表3~表5的数据可以看出:1)窄客户时间窗算例(c1**,r1**,rc1**)的配送方案路径条数和总成本都远大于宽客户时间窗算例(c2**,r2**,rc2**),为满足客户时间窗约束需使用更多的电物流车,导致最终配送方案的总成本相对更高.2)类型C算例的电动车额定充电率较大,充电时间较长,产生的成本值大大高于其他类型的算例.3)车辆电池容量影响着电动车的使用数,电池容量越大的算例(c2**,r2**,rc2**)规划的路径数越少,较小的电池容量因行驶里程的限制使充电次数增多,产生较多额外的行驶距离和充电成本.同时电池容

量大的车辆更不容易受载重耗电率影响,其动态的耗电量对客户配送方案影响较小,在规划时具有更大的解空间.4)耗电率决定着配送和充电方案,影响充电时间和行驶路径,考虑与载重相关的耗电率使得问题更具有实际意义.5)类型C算例的运行时间小于类型R算例和类型RC算例,零散随机分布特征的算例计算时间较长,说明实际问题的优化中应关注随机分布客户群的配送规划,该特征影响了路径规划效率.

4 结论

本文以固定用车成本、路径行驶成本和充电成本最小化为目标,研究了载重影响耗电率的电动车车辆路径优化问题,构建了相应的混合整数规划模型.为高效率地求得模型的满意解,设计了改进的自适应大邻域搜索算法,通过求解36个小规模算例与CPLEX求解结果对比,验证了算法的效率,并通过56个大规模算例验证了算法的稳定性和有效性.对算例运行结果分析发现电动车耗电率、充电率、电池容量和客户时间窗等因素通过影响充电时间、行驶路径、用车数量进而影响配送总成本,而与载重相关的耗电率对路径规划的影响不容忽视,以总成本最小化为目标得到的物流路径规划方案具有实际的应用意义.HALNS-TS在小规模案例上表现较好,在大规模算例的求解中表现稳定,在实际应用中为提升宏观物流网络配送方案规划的效率以及解决末端环节客户数量多、分布分散的实际配送难题提供了一定的决策依据.

基于本文的研究结果,后续研究将考虑更合实际的耗电模式,构建载重与耗电率间更现实的数学关系,考虑快递柜模式下的客户软时间窗服务形式等情况,解决更实际的电动车路径优化问题,为物流企业配送方案的制定提供更多的参考依据.

参考文献(References)

- [1] 2020年中国快递发展指数报告[R].北京:国家邮政局,2021:1-10.
(Report on China express development index in 2020[R]. Beijing: State Post Bureau, 2021: 1-10.)
- [2] 郭戈,徐涛,韩英华,等.电动汽车时代的电网-交通网协同优化综述[J].控制与决策,2021,36(9):2049-2062.
(Guo G, Xu T, Han Y H, et al. A survey of cooperative optimization of traffic-grid networks in the era of electric vehicles[J]. Control and Decision, 2021, 36(9): 2049-2062.)
- [3] Dantzig G B, Ramser J H. The truck dispatching problem[J]. Management Science, 1959, 6(1): 80-91.
- [4] Erdoğan S, Miller-Hooks E. A green vehicle routing problem[J]. Transportation Research Part E: Logistics and

- Transportation Review, 2012, 48(1): 100-114.
- [5] Schneider M, Stenger A, Goeke D. The electric vehicle-routing problem with time windows and recharging stations[J]. *Transportation Science*, 2014, 48(4): 500-520.
- [6] Verma A. Electric vehicle routing problem with time windows, recharging stations and battery swapping stations[J]. *EURO Journal on Transportation and Logistics*, 2018, 7(4): 415-451.
- [7] Erdelić T, Carić T, Erdelić M, et al. Electric vehicle routing problem with single or multiple recharges[J]. *Transportation Research Procedia*, 2019, 40: 217-224.
- [8] Hiermann G, Puchinger J, Ropke S, et al. The electric fleet size and mix vehicle routing problem with time windows and recharging stations[J]. *European Journal of Operational Research*, 2016, 252(3): 995-1018.
- [9] Keskin M, Laporte G, Çatay B. Electric vehicle routing problem with time-dependent waiting times at recharging stations[J]. *Computers & Operations Research*, 2019, 107: 77-94.
- [10] 杨珺, 冯鹏祥, 孙昊, 等. 电动汽车物流配送系统的换电站选址与路径优化问题研究[J]. *中国管理科学*, 2015, 23(9): 87-96.
(Yang J, Feng P X, Sun H, et al. Battery exchange station location and vehicle routing problem in electric vehicles distribution system[J]. *Chinese Journal of Management Science*, 2015, 23(9): 87-96.)
- [11] Schiffer M, Walther G. The electric location routing problem with time windows and partial recharging[J]. *European Journal of Operational Research*, 2017, 260(3): 995-1013.
- [12] 揭婉晨, 杨珺, 陆坚毅. 基于分支定价算法的电动汽车车辆路径问题[J]. *运筹与管理*, 2016, 25(4): 93-100.
(Jie W C, Yang J, Lu J Y. Electric vehicle routing problem based on a branch-and-price algorithm[J]. *Operations Research and Management Science*, 2016, 25(4): 93-100.)
- [13] 揭婉晨, 杨珺, 杨超. 多车型电动汽车车辆路径问题的分支定价算法研究[J]. *系统工程理论与实践*, 2016, 36(7): 1795-1805.
(Jie W C, Yang J, Yang C. Branch-and-price algorithm for heterogeneous electric vehicle routing problem[J]. *Systems Engineering—Theory & Practice*, 2016, 36(7): 1795-1805.)
- [14] Desaulniers G, Errico F, Irnich S, et al. Exact algorithms for electric vehicle-routing problems with time windows[J]. *Oper Research*, 2014, 64(6): 1388-1405.
- [15] Felipe Á, Ortuño M T, Righini G, et al. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges[J]. *Transportation Research Part E: Logistics and Transportation Review*, 2014, 71: 111-128.
- [16] Shao S, Guan W, Bi J. Electric vehicle-routing problem with charging demands and energy consumption[J]. *IET Intelligent Transport Systems*, 2018, 12(3): 202-212.
- [17] Cortés-Murcia D L, Prodhon C, Afsar H M. The electric vehicle routing problem with time windows, partial recharges and satellite customers[J]. *Transportation Research Part E: Logistics and Transportation Review*, 2019, 130: 184-206.
- [18] Macrina G, Laporte G, Guerriero F, et al. An energy-efficient green-vehicle routing problem with mixed vehicle fleet, partial battery recharging and time windows[J]. *European Journal of Operational Research*, 2019, 276(3): 971-982.
- [19] 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.)
- [20] 李国明, 李军华. 基于混合禁忌搜索算法的随机车辆路径问题[J]. *控制与决策*, 2021, 36(9): 2161-2169.
(Li G M, Li J H. Stochastic vehicle routing problem based on hybrid tabu search algorithm[J]. *Control and Decision*, 2021, 36(9): 2161-2169.)
- [21] Zhang S, Chen M Z, Zhang W Y, et al. Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations[J]. *Expert Systems With Applications*, 2020, 145: 113123.
- [22] Keskin M, Çatay B. Partial recharge strategies for the electric vehicle routing problem with time windows[J]. *Transportation Research Part C: Emerging Technologies*, 2016, 65: 111-127.
- [23] Goeke D, Schneider M. Routing a mixed fleet of electric and conventional vehicles[J]. *European Journal of Operational Research*, 2015, 245(1): 81-99.
- [24] Kancharla S R, Ramadurai G. An adaptive large neighborhood search approach for electric vehicle routing with load-dependent energy consumption[J]. *Transportation in Developing Economies*, 2018, 4(2): 1-9.
- [25] Kancharla S R, Ramadurai G. Electric vehicle routing problem with non-linear charging and load-dependent discharging[J]. *Expert Systems With Applications*, 2020, 160: 113714.
- [26] Xiao Y Y, Zhao Q H, Kaku I, et al. Development of a fuel consumption optimization model for the capacitated vehicle routing problem[J]. *Computers & Operations Research*, 2012, 39(7): 1419-1431.

作者简介

吴廷映(1982—), 男, 讲师, 博士, 从事运筹优化、鲁棒优化等研究, E-mail: tingyingwu@shu.edu.cn;

孙灏(1998—), 女, 硕士生, 从事车辆路径优化、智能优化算法的研究, E-mail: sun_shiny@shu.edu.cn.

(责任编辑: 闫妍)