



自动化学报

Acta Automatica Sinica

ISSN 0254-4156, CN 11-2109/TP

## 《自动化学报》网络首发论文

题目: 结合聚类分解的增强蚁群算法求解复杂绿色车辆路径问题  
作者: 胡蓉, 李洋, 钱斌, 金怀平, 向凤红  
DOI: 10.16383/j.aas.c190872  
收稿日期: 2019-12-22  
网络首发日期: 2020-06-17  
引用格式: 胡蓉, 李洋, 钱斌, 金怀平, 向凤红. 结合聚类分解的增强蚁群算法求解复杂绿色车辆路径问题[J/OL]. 自动化学报. <https://doi.org/10.16383/j.aas.c190872>



**网络首发:** 在编辑部工作流程中, 稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定, 且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式(包括网络呈现版式)排版后的稿件, 可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定; 学术研究成果具有创新性、科学性和先进性, 符合编辑部对刊文的录用要求, 不存在学术不端行为及其他侵权行为; 稿件内容应基本符合国家有关书刊编辑、出版的技术标准, 正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性, 录用定稿一经发布, 不得修改论文题目、作者、机构名称和学术内容, 只可基于编辑规范进行少量文字的修改。

**出版确认:** 纸质期刊编辑部通过与《中国学术期刊(光盘版)》电子杂志社有限公司签约, 在《中国学术期刊(网络版)》出版传播平台上创办与纸质期刊内容一致的网络版, 以单篇或整期出版形式, 在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊(网络版)》是国家新闻出版广电总局批准的网络连续型出版物(ISSN 2096-4188, CN 11-6037/Z), 所以签约期刊的网络版上网络首发论文视为正式出版。

# 结合聚类分解的增强蚁群算法求解复杂绿色车辆路径问题

胡蓉<sup>1,2</sup> 李洋<sup>1,3</sup> 钱斌<sup>1,2</sup> 金怀平<sup>1</sup> 向凤红<sup>1</sup>

**摘要** 本文针对带时间窗的低能耗多车场多车型车辆路径问题 (low-energy-consumption multi-depot heterogeneous-fleet vehicle routing problem with time windows, LMHFVPR.TW), 提出一种结合聚类分解策略的增强蚁群算法 (enhanced ant colony algorithm based on clustering decomposition, EACO\_CD) 进行求解. 首先, 由于该问题具有强约束、大规模和NP-Hard等复杂性, 为有效控制问题的求解规模并合理引导算法在优质解区域搜索, 根据问题特点设计两种基于K-means的聚类策略, 将LMHFVPR.TW合理分解为一系列带时间窗的低能耗单车场单车型车辆路径子问题 (low-energy-consumption vehicle routing problem with time windows, LVRP.TW); 其次, 本文提出一种增强蚁群算法 (enhanced ant colony optimization, EACO) 求解分解后的各子问题 (LVRP.TW), 进而获得原问题的解. EACO不仅引入信息素挥发系数控制因子进一步动态调节信息素挥发系数, 从而有效控制信息素的挥发以提高算法的全局搜索能力, 而且设计基于4种变邻域操作的两阶段变邻域局部搜索 (two-stage variable neighborhood search, TVNS) 来增强算法的局部搜索能力. 最后, 在不同规模问题上的仿真和对比实验验证了所提EACO\_CD的有效性.

**关键词** 低能耗车辆路径问题; 多车场多车型; 时间窗; 聚类分解; 增强蚁群算法

**引用格式** TP399 文献标识码: A

**DOI** 10.16383/j.aas.c190872

## An enhanced ant colony algorithm combined with clustering decomposition for solving complex green vehicle routing problem

HU Rong<sup>1,2</sup> LI Yang<sup>1,3</sup> QIAN Bin<sup>1,2</sup> JIN Huai-Ping<sup>1</sup> XIANG Feng-Hong<sup>1</sup>

**Abstract** In this paper, an enhanced ant colony algorithm combined with clustering decomposition strategy (EACO\_CD) is proposed for solving the low-energy-consumption multi-depots heterogeneous-fleet vehicle routing problem with time windows (LMHFVPR.TW). Firstly, since the considered problem is a complex one with strong constraints, large scale and NP-hardness, In order to control the scale of problem and reasonably guide the algorithm to search in the high-quality solution region, two kinds of clustering method based on K-means strategies are designed to reasonably decompose it into a series of subproblems (i.e., the low-energy-consumption vehicle routing problems with time windows, LVRP.TW) by utilizing the problem characteristics. Secondly, an enhanced ant colony optimization (EACO) to solve the decomposed subproblems (LVRP.TW) for obtaining the solution of the original problem is proposed. Not only a control factor of pheromone decay parameter in EACO is added to adjust the pheromone decay parameter dynamically so as to control the volatilization of pheromone effectively and improve the global search ability of ACO, but also a two-stage variable neighborhood search (TVNS) is designed based on four variable neighborhood operations to enhance its the local search ability. Finally, simulation experiments and comparisons on instances with different scales demonstrate the effectiveness of proposed EACO\_CD.

**Key words** Low-energy-consumption vehicle routing problem (LVRP); multiple depots and heterogeneous fleet; time windows; cluster of decomposition; enhanced ant colony algorithm

**Citation** HU Rong, LI Yang, QIAN Bin, JIN Huai-Ping, XIANG Feng-Hong. An enhanced ant colony algorithm combined with clustering decomposition for solving complex green vehicle routing problem. *Acta Automatica Sinica*, 201X, XX(X): X-X

收稿日期 2019-12-22 录用日期 2020-05-03  
Manuscript received December 22, 2019; accepted May 03, 2020  
国家自然科学基金项目 (61963022, 51665025); 云南省应用基础研究计划重点项目资助.

Foundation item: Supported by National Natural Science Foundation of China (61963022, 51665025) and Applied Basic Research Key Project of Yunnan Province.

本文责任编辑 王鼎

Recommended by Associate Editor WANG Ding

1. 昆明理工大学信息工程与自动化学院 昆明 650500
1. School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500
2. 昆明理工大学 云南省人工智能重点实验室 昆明 650500

传统的车辆路径问题 (vehicle routing problem, VRP) 由Dantzig和Ramser于1959年首次提出<sup>[1]</sup>. 该问题主要描述为在满足车辆载重、容积、行驶里程及客户服务要求等条件的同时, 合理调度车辆出行数量、行车路线、出行时间, 使得总运费用最优化. 随

2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology, Kunming 650500
3. 云南电力技术有限责任公司 昆明 650051
3. Yunnan Electric Power Technology Co., LTD, Kunming 650051

着社会经济的高速发展,跨区域多车队联合车辆配送需求明显增加<sup>[2-4]</sup>.同时,当今环保要求越来越严格,考虑燃油消耗和碳排放等因素的低能耗车辆配送已开始受到重视<sup>[5]</sup>.此外,日益激烈的市场竞争逼迫企业注重降低物流配送成本和提高客户满意度<sup>[6]</sup>.在此背景下,研究带时间窗的低能耗多车场多车型车辆路径问题(low-energy-consumption multi-depot heterogeneous-fleet vehicle routing problem with time windows, LMHFVPR\_TW),具有十分重要的现实意义.由于VRP为NP-hard问题,而VRP可归约为LMHFVPR\_TW,故LMHFVPR\_TW也属于NP-hard问题,对其开展研究亦具有较大理论价值.

低能耗车辆路径问题近年已开始受到重视,但相关研究仍较为有限.在低能耗多车场VRP(low-energy-consumption multi-depot VRP, LMVRP)方面,Jabir等<sup>[7]</sup>在问题中考虑碳排放等因素,利用低能耗车辆路径问题近年已开始受到重视,但相关研究仍较为有限.在低能耗多车场VRP(low-energy-consumption multi-depot VRP, LMVRP)方面,Jabir等<sup>[7]</sup>在问题中考虑碳排放等因素,利用蚁群算法与变邻域搜索(Variable Neighborhood Search, VNS)相结合进行求解.仿真实验表明该算法可有效求解小规模和大规模问题.Kaabachi等<sup>[8]</sup>在问题中考虑距离、碳排放和燃油消耗等因素,并在蚁群算法中加入插入、交换和2-opt等邻域搜索策略以提高算法局部搜索能力,取得良好的效果.在低能耗多车型VRP(low-energy-consumption heterogeneous-fleet VRP, LHFVRP)方面,Xiao等<sup>[9]</sup>在问题中考虑交通拥堵和碳排放等因素,并设计融合变邻域搜索和整数规划的迭代搜索算法进行求解.Kwon等<sup>[10]</sup>在问题中考虑碳排放交易机制(即各国之间通过碳排放交易市场相互买卖碳排放量,以保证该国的碳排放量在其规定的配额内),然后采用混合禁忌搜索算法进行求解.目前尚无LMHFVPR\_TW的相关研究.

对于多车场VRP(multi-depot VRP, MVRP)、多车型VRP(heterogeneous-fleet VRP, HFVRP)和多车场多车型VRP(multi-depot heterogeneous-fleet VRP, MHFVPR)这几类复杂VRP,智能算法已有一定的研究.现有研究大多对问题进行整体编码和求解,但由于这些问题比单车场单车型VRP(即传统VRP)具有更多的决策变量和约束条件,使得解空间扩大且编码解码较为繁琐,这时仅靠智能算法本身已难以将搜索快速引导至目标值整体较优的不同区域(即较优区域)执行,容易导致算法实际搜索效率下降.因此,近年部分学者在所提算法中,先采用某种分解策略将问题合理分解为多个相对简

单的子问题,从而将算法搜索尽量限定于解空间中部分存在优质解的较优区域之内,然后再用智能算法等求解各子问题并获得原问题的解,取得良好效果.在MVRP方面,主要考虑地理条件、客户需求和交货时间等因素,并采用聚类算法把MVRP分解为一系列单个车场VRP,然后采用PSO算法<sup>[11,12]</sup>、遗传算法<sup>[13]</sup>等求解.在HFVRP方面,主要考虑车辆载重量、车辆运输费用和客户需求等因素,并采用路径分割法把HFVRP分解为一系列单车场VRP(即traveling salesman problem, TSP),进而采用禁忌搜索算法<sup>[14]</sup>等求解.在MHFVRP方面,由于该问题复杂度高,为HFVRP和MVRP的综合,故大都先采用分解策略将MHFVRP分解为一系列HFVRP(即单车场异构车辆的VRP),然后采用路径分割法将各HFVRP进一步分解为多个TSP.譬如,Donato等<sup>[15]</sup>先采用一种启发式聚类算法把全部客户分成若干类,然后利用路径分割算法分配车辆,从而将原问题转化为多个小规模TSP后再利用混合整数规划求解器进行求解.Tang等<sup>[16]</sup>先利用最近邻方法把客户聚类到相应的车场,然后再用扫描法对每个车场中的客户进行路径分割,进而把原问题转化为多个TSP,最后采用蚁群算法求解.上述文献中的仿真实验和算法对比验证了分解策略和智能算法结合的必要性.然而,对于现实物流配送中大量存在且更为复杂的LMHFVPR\_TW,尚无结合分解策略的智能求解算法,故开展相关研究意义重大.

蚁群算法(ant colony optimization, ACO)是一种模拟蚂蚁寻找食物的群智能优化算法,最早由Dorigo等<sup>[17]</sup>提出并首次用于求解TSP.ACO借鉴蚂蚁觅食的群体智能行为,利用信息素浓度矩阵(以下简称信息素矩阵)学习和保留蚂蚁在各自行驶路径(即问题解)上留下的信息素浓度信息,并基于该矩阵构建路径转移概率矩阵(以下简称概率矩阵),然后通过对概率矩阵采样生成新种群来实现路径搜索(即解空间搜索)并引导搜索方向.ACO采用正反馈机制使得较短行驶路径(较优解)的优良信息得以更多积累,可引导算法在较短时间内到达解空间中存在优质解的区域.这使得算法具有较强的全局搜索能力,从而在多种VRP上得到成功应用<sup>[7,8,18-24]</sup>.从ACO在VRP系列问题上的研究现状来看,采用合理的信息素浓度计算策略可控制全局搜索的方向,引入基于有效邻域操作的局部搜索能进一步提高算法性能,两者是设计有效ACO的关键.文献调研表明,ACO求解LMHFVPR\_TW的研究处于空白状态.

本文研究LMHFVPR\_TW的建模与求解.在建模方面,给出包含经济费用、燃油消耗费用和客户满意度费用的运输总费用计算模型,并建立以最小



化运输总费用为优化目标的LMHFVPR\_TW. 在求解方面, 考虑到LMHFVPR\_TW这类问题的解空间庞大且整体编码解码复杂, 直接采用智能算法难以在较短时间内实现有效搜索, 故提出一种结合聚类分解的增强蚁群算法 (enhanced ant colony algorithm based on clustering decomposition, EACO\_CD) 进行求解.

EACO\_CD的结构如图1所示 (以2个车场2类车型为例). 由图1 可知, EACO\_CD由问题分解阶段和子问题求解阶段组成. (1)在问题分解阶段, 为确保分解后的各子问题区域尽量覆盖解空间中的优质解区域, 设计两层基于K-means的聚类算法对问题进行逐步分解. 第1层聚类算法为所提的一种改进平衡K-means 聚类算法 (improved balanced K-means algorithm, IBKA), 用于给每个车场分配一定数量的客户, 从而形成一系列带时间窗的低能耗单车场多车型VRP(LHFVRP with time windows, LHFVRP\_TW). 第2层聚类算法为所提的一种带粒子群优化的混合K-means聚类算法 (hybrid K-means clustering algorithm with PSO, HKMA), 用于给每个LHFVRP\_TW中的每类车型合理分配

对每个分解后子问题 (即LVRP\_TW) 的解空间区域进行搜索, 进而将子问题的解合并后得到原问题的解. 在EACO\_CD中, 采用动态的信息素浓度挥发系数 (以下简称信息素挥发系数), 并进一步加入信息素挥发系数控制因子以调节其取值, 可避免算法过早收敛并进一步引导算法全局搜索到达更多的不同区域; 同时, 设计基于4种变邻域操作的两阶段局部搜索策略, 用于对全局搜索发现的优质解区域进行细致且高效的搜索, 以平衡全局和局部搜索并增强算法性能. 最后, 通过仿真实验和算法比较验证所提算法的有效性.

## 1 LMHFVPR\_TW建模与分析

本节建立LMHFVPR\_TW模型并对问题特点和求解算法进行分析. LMHFVPR\_TW为带软时间窗约束的多车场多车型车辆配送问题, 优化目标为最小化运输总费用 (即经济费用、燃油消耗费用和客户满意度费用之和). 该问题在已知客户位置坐标、车场位置坐标、客户货物需求量、客户时间窗、各车场车型特征 (表1 中给出相应的符号定义) 的情况下, 希望找到各车场中车辆的最佳配送路线 (由1.3节问题模型中的决策变量具体取值确定), 从而使总运输费用达到最优. LMHFVPR\_TW的应用场景很多, 譬如电商和连锁超市的商品配送、战区和疫区的紧急物资配送、石化企业成品油的一次和二次配送等.

### 1.1 问题假设和符号定义

LMHFVPR\_TW满足如下假设:

- (1) 每个车场当中的每一类车型都参与配送任务.
- (2) 每个车场所含的车型种类相同.
- (3) 车辆在一次任务分配中仅进行一次配送.
- (4) 车辆从车场出发, 完成服务后需回到原车场.
- (5) 每个客户都被服务, 被服务的次数有且仅有一次.
- (6) 车辆配送中的货物需尽量在每个客户要求的到达时间段内送达.

同时, LMHFVPR\_TW的相关符号定义如表1所示.

### 1.2 问题优化目标计算模型

LMHFVPR\_TW的优化目标为运输总费用  $Z_{Total}$ , 该优化目标由三部分组成: 经济费用  $Z_{Emission\ cost}$ 、排放费用  $Z_{Emission\ cost}$  及客户满意度费用  $Z_{Customer\ Satisfaction}$ . 其中, 经济费用由运输距离费用  $F_1$  和车辆固定成本  $F_2$  组成, 排放费用设定为燃油消耗费用  $F_3$ , 客户满意度费用由软时

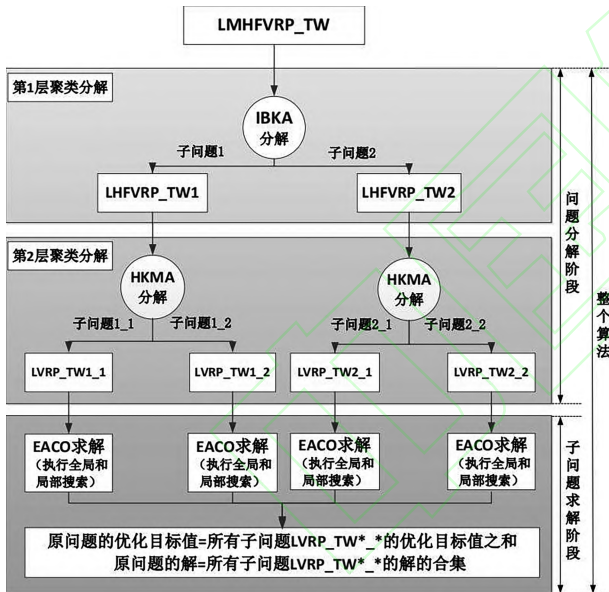


图1 EACO\_CD (EACO\_IBKA\_HKMA) 结构

Fig. 1 Framework of EACO\_CD

一定数量的客户, 进而得到一系列带时间窗的低能耗VRP (low-energy-consumption VRP with time windows, LVRP\_TW). 两层聚类分解算法将原问题最终分解为子问题LVRP\_TW, 而不是分解为比LVRP\_TW 规模更小的带时间窗的低能耗TSP, 这使其子问题的解空间区域相对较大, 有利于算法对原问题解空间中更多区域进行搜索, 可望获得更好的解. (2) 在子问题求解阶段, 采用所提的增强蚁群算法 (enhanced ant colony algorithm, EACO)

间窗惩罚费用  $F_4$  构成.  $F_1$ 、 $F_2$ 、 $F_3$ 、 $F_4$  的计算如下:

$$F_1 = \sum_{p=1}^{P_t} \sum_{M=1}^{M_t} \sum_{k=1}^{H_{PM}} \sum_{i=0}^N \sum_{j=0}^N C_{M1} * x_{PMijk} * d_{ij} \quad (1)$$

$$F_2 = \sum_{p=1}^{P_t} \sum_{M=1}^{M_t} \sum_{k=1}^{H_{PM}} C_{M2} * x_{PMijk} \quad (2)$$

$$F_3 = \sum_{p=1}^{P_t} \sum_{M=1}^{M_t} \sum_{k=1}^{H_{PM}} \sum_{i=0}^N \sum_{j=0}^N C_{M2} * FU_{Mij} * x_{PMijk} \quad (3)$$

$$F_4 = \sum_{i=1}^N \max \{C_1 * (ET_i - t_i), 0, C_2 * (t_i - LT_i)\} \quad (4)$$

表1 符号及定义

Table 1 Symbols and definitions

符号	释义	符号	释义
$F_1$	运输距离费用	$H_{PM}$	表示车场有 $H_{PM}$ 辆 $M$ 类型的车
$F_2$	车辆固定成本	$r(A)$	完成客户子集 $A$ 中所有客户的配送需要的最少车辆数
$F_3$	燃油消耗费用	$N$	总共有 $N$ 个客户
$F_4$	时间窗惩罚费用	$V$	$V$ 表示客户编号集合 $\{0, 1, 2, \dots, N\}$ ( $0$ 表示车场)
$C_{M1}$	第 $M$ 类型车的距离费用系数	$M_t$	表示共有 $M_t$ 种类型的车
$C_{M2}$	第 $M$ 种类型车的固定发车费用系数	$x_{PMijk}$	表示车场 $P$ 车型 $M$ 的第 $k$ 辆车从客户 $i$ 到客户 $j$ 的决策变量
$C_{M3}$	第 $M$ 种类型车的燃油费用系数	$k$	表示车辆编号 $k$
$C_1$	配送车辆提前到达的单位惩罚费用	$d_{ij}$	$d_{ij}$ 表示客户 $i$ 到客户 $j$ 的距离
$C_2$	配送车辆迟到的单位惩罚费用	$ET_i$	客户 $i$ 要求的最早到达时间
$i$	表示客户点 $i$	$LT_i$	客户 $i$ 要求的最晚到达时间
$j$	表示客户点 $j$	$S_i$	客户 $i$ 要求的卸货时间
$P_s$	表示全部车场集合 $\{1, 2, \dots, P_t\}$	$q_i$	客户 $i$ 要求的货物需求量
$P_t$	表示总共有 $P_t$ 个车场	$t_i$	车辆到达客户 $i$ 的时间
$P$	表示车场编号 $P$	$Q_M$	第 $M$ 种车型的最大载重量
$M_s$	表示全部车型集合 $\{1, 2, \dots, M_t\}$	$M$	表示车型编号 $M$
$H_{PMS}$	表示车场 $P$ 中车型 $M$ 的全部车辆集合 $\{1, 2, \dots, H_{PM}\}$	$FU_{Mij}$	车型为 $M$ 的车辆从客户 $i$ 到客户 $j$ 之间的耗油量

备注: 综合燃油消耗模型中的其他相关参数设定参考文[25]

其中, 式(3)中燃油消耗量  $FU_{Mij}$  的计算基于文中[30]的综合燃油消耗模型和参数设置, 并假设各类车型的油耗参数相同, 可得到  $FU_{Mij}$  的如下计算

公式:

$$FU_{Mij} = \left( \frac{\omega_1^M}{v_{ij}^M} + \omega_2^M + \omega_3^M * G_{ij}^M + \overline{v_{ij}^M}^2 * \omega_4^M \right) * d_{ij} \quad (5)$$

式(5)中  $G_{ij}^M$  为车型  $M$  的车辆从客户  $i$  到客户  $j$  途中的车辆总重量,  $\overline{v_{ij}^M}$  为车型  $M$  从客户  $i$  到客户  $j$  的平均速度,  $\omega_1^M$ 、 $\omega_2^M$ 、 $\omega_3^M$ 、 $\omega_4^M$  为常量. 此外, 式(4)中车辆到达时间  $t_j$  的计算如下:

$$t_j = \sum_{i=0}^N x_{PMijk} * \left( t_i + \frac{d_{ij}}{v_{ij}^M} + s_i \right) \quad (6)$$

综上, 运输总费用  $Z_{Total}$  的计算如下:

$$Z_{Total} = F_1 + F_2 + F_3 + F_4 \quad (7)$$

### 1.3 问题模型

问题优化目标:

$$\min Z_{Total} = F_1 + F_2 + F_3 + F_4 \quad (8)$$

问题约束条件:

$$\sum_{M=1}^{M_t} \sum_{j=1}^N \sum_{k=1}^{H_{PM}} x_{PM0jk} = \sum_{M=1}^{M_t} \sum_{j=1}^N \sum_{k=1}^{H_{PM}} x_{PMj0k} \quad (\forall P \in P_s) \quad (9)$$

$$\sum_{P=1}^{P_t} \sum_{M=1}^{M_t} \sum_{i=1}^N \sum_{k=1}^{H_{PM}} x_{PMijk} = 1 \quad (i \neq j, \forall j \in V \setminus \{0\}) \quad (10)$$

$$\sum_{P=1}^{P_t} \sum_{M=1}^{M_t} \sum_{j=1}^N \sum_{k=1}^{H_{PM}} x_{PMijk} = 1 \quad (i \neq j, \forall i \in V \setminus \{0\}) \quad (11)$$

$$\sum_{j=1}^N x_{PM0jk} = \sum_{j=1}^N x_{PMj0k} = 1, \quad (12)$$

$$(\forall P \in P_s, \forall M \in M_s, \forall k \in H_{PMS})$$

$$\sum_{P=1}^{P_t} \sum_{M=1}^{M_t} \sum_{k=1}^{H_{PM}} \sum_{i \notin A} \sum_{j \in A} x_{PMijk} \geq r(A) \quad (13)$$

$$(\forall A \subseteq V \setminus \{0\}, A \neq \emptyset)$$

$$\sum_{j=1}^N \sum_{i=1}^N x_{PMijk} * q_i \leq Q_M, \quad (14)$$

$$(i \neq j, \forall k \in H_{PMS}, \forall P \in P_s, \forall M \in M_s)$$

决策变量:

$$x_{PMijk} = \begin{cases} 1, & \text{if vehicle } k \text{ of type } M \text{ in depot } P \\ & \text{travels from node } i \text{ to } j \\ 0, & \text{otherwise} \end{cases}$$

$$(i \neq j, (j, i) \in V, \forall k \in H_{PM}, \forall P \in P_s, \forall M \in M_s) \quad (15)$$

其中, 式 (9) 要求从某车场出发的车辆数目和回到该车场的车辆数目相等; 式 (10) 和式 (11) 要求每个客户只能被一辆车服务1次; 式 (12) 要求所有车辆均需参与配送, 且配送完后返回原车场式 (13) 要求生成的配送方案或问题解不出现不含车场的子回路; 式 (14) 要求车辆配送过程中不能超过其对应车型的额定载重量。

#### 1.4 问题特点分析

由1.3节可知, 本文的LMHFVPR\_TW为具有非线性和大规模约束的0-1整数规划问题. 相对于传统VRP, 式 (1) 至式 (4) 使优化目标不同且更加复杂, 同时式 (4) 使优化目标具有非线性; 多车场和多车型的引入, 使得决策变量由传统VRP的3维变量 $x_{ijk}$ 变为5维变量 $x_{PMijk}$ , 导致决策变量和约束的数量增多, 这时问题解空间更加庞大和复杂. 此外, 相对于常规0-1整数规划问题, VRP系列问题均含子回路消除约束 (见式 (13)), 该类约束的数量为 $2^{N+1} - 2$ , 这种指数级的约束数量限制了问题的求解规模. 虽然可以通过增加连续决策变量使式 (13) 的约束个数变为客户数 $N$ 的多项式, 从而扩大问题的求解规模, 但这时原问题近似转化为同时含离散和连续决策变量的混合0-1整数规划问题, 该近似问题的求解难度并未降低<sup>[26]</sup>.

#### 1.5 问题求解算法分析

VRP系列问题 (包括本文的LMHFVPR\_TW) 均可建模为0-1整数或混合整数规划模型, 其求解算法分为两类. 一类为运筹学算法, 包括分支定界、分支切割、动态规划等<sup>[27]</sup>. 该类算法主要以线性代数和几何分析为基本工具, 利用问题优化目标函数和约束条件式的结构信息构造遍历或部分遍历解空间的搜索, 可在几分钟至几十分钟内求解较小规模问题 (客户数小于等于20), 同时经过合适设计, 也可用于求解较大规模问题 (客户数大于等于80), 但往往求解时间很长<sup>[27-29]</sup>. 不同于线性规划问题, 0-1规划问题具有非凸特性, 其内在几何结构与最优解间的关系仍属开放问题, 目前仅在无约束0-1二次规划等少数具有特殊结构的简单问题上存在有效多项式时间求解算法<sup>[6,28,30]</sup>, 尚无通用的最优解多项式

时间求解算法. 对于LMHFVPR\_TW这类复杂的0-1规划问题, 采用已有的运筹学算法框架虽也可设计相应的求解算法, 但算法难以在较短时间获取较大规模问题的满意解. 另一类是智能优化算法 (以下简称智能算法), 包括蚁群优化算法、遗传算法、禁忌搜索算法等<sup>[27]</sup>. 该类算法不依赖于问题结构, 而是将问题的约束条件隐式地包含在所设计的解的编码、解码规则中处理, 并利用某种拟人、拟物的机制不断生成新的可行个体或解, 从而引导算法执行搜索, 往往可在几秒或十几秒内就能获得各类VRP的满意解<sup>[6,27,31]</sup>.

目前, 各类文献对所提智能算法使用少量个体一般种群规模为20~100) 运行少量代数 (一般进化代数为100~1000代) 后, 为何都可获得问题的满意解, 基本都只从算法本身的机制进行解释. 这导致现有研究过于追求新方法层面的创新. 实际上, 智能算法为何有效, 是由其编码解码规则和智能算法自身机制共同决定的. 智能算法并不直接对0-1变量编码, 而是对客户序号排列进行编码解码规则的设计. 这样的排序编码解码, 容易将解中元素的取值限定在满足约束的可行范围, 从而避免繁杂的约束处理, 可明显提升算法搜索效率. 同时, 排序编码解码规则所确定的解空间极为扁平, 问题目标值的变化范围远远小于排序模型解空间的规模, 这非常有利于算法短时间内获取满意解. 具体来说, 譬如优化目标为最小化总行驶距离的单车VRP (即TSP), 假如有100个客户, 任意两个客户间的距离为在区间[1, 1000]上均匀分布的随机数, 且要求车辆从某个客户驻地 (车场) 出发, 服务完客户后返回该驻地, 则其目标值变化范围在(1, 1000000)之内 (1000000为各客户间的距离之和的上限, 实际的最大目标值小于此值), 而解空间规模为99! (车辆出发位置的客户需扣除), 平均每一个具体目标值对应约 $9 \times 10^{149}$ 个不同排列或解, 这表明数量巨大的不同解具有相同的目标值. 对于其他更为复杂的VRP (包括本文的LMHFVPR\_TW), 也存在这一情况. 另外, 离散问题本身解之间没有梯度, 相邻解之间的目标值差别可能很小, 也可能较大. 排序编码解码规则所确定解空间的“极为扁平”性和“无梯度”性, 使得各种智能算法即使只用几秒 (CPU为2.6GHZ的主流PC在1秒内可搜索并评价含100个客户的传统VRP问题的数万个解) 搜索解空间中极小的区域 (数万相对于99! 类似于1根针或更小的面积相对于体育场), 也可能到达较广的目标值区域, 同时其内在的寻优机制可驱动算法到达目标值较优的不同区域搜索, 从而获得满意解. 智能算法这种通过对解空间极小区域的搜索来实现对目标值较广且较优区域的搜索, 是其有效的本质原因,



也是现有运筹学算法这类偏遍历或部分遍历解空间的算法难以做到的. 因此, 设计智能算法以实现 LMHFVPR\_TW 快速有效求解, 是合理且必要的.

由1.4节和1.5节的分析可知, 对于LMHFVPR\_TW这类非凸、非线性问题, 直接采用运筹学方法在短时间内获取满意解难度很大, 故本文设计一种智能算法(即结合聚类分解的增强蚁群算法EACO\_CD)进行求解.

## 2 问题求解算法EACO\_CD

本文所提的求解算法EACO\_CD包含问题分解阶段和子问题求解阶段(见图1). 通过两层聚类分解, 原问题LMHFVPR\_TW转换成为 $P_t \times M_t$ 个单车场单车型子问题LVRP\_TWs, 然后采用本文设计的子问题求解算法EACO依次对每个LVRP\_TW进行求解, 进而可获得原问题的解和优化目标值. 2.1节介绍EACO\_CD中问题分解阶段的细节, 包含两层聚类算法的细节和复杂度分析; 2.2节介绍EACO\_CD中子问题求解阶段的细节, 包含EACO的细节和复杂度分析; 2.3节介绍EACO\_CD的总体框架并分析整体复杂度.

### 2.1 EACO\_CD的问题分解阶段

EACO\_CD的问题分解阶段执行两层聚类算法, 第1层聚类算法为IBKA, 用于将原问题LMHFVPR\_TW分解为 $P_t$ 个单车场子问题LHFVRP\_TWs; 第2层聚类算法为HKMA, 用于把每个单车场问题LHFVRP\_TW进一步分解为个单车场单车型子问题LVRP\_TWs(见图1). 在对数值型数据进行聚类时, 大都采用K-means、模糊聚类(fuzzy cluster)等算法<sup>[14,32-34]</sup>. 而针对混合型数据, 可采用k-prototypes算法或先将混合型数据转换为数值型数据<sup>[35]</sup>. 由于本文问题中的客户属性数据均为值型, 不需要对数据类型进行转换, 同时K-means具有线性计算复杂度已被成功用于VRP系列问题<sup>[12,14,33,36,37]</sup>, 故本文在IBKA和HKMA中采用K-means作为基本分解策略.

#### 2.1.1 第1层聚类算法 (IBKA)

本文问题的优化目标(见式(8))由四部分组成, 其中有两部分与车辆运输距离有关. 当车辆距客户较远时, 配送所产生的燃油消耗费用和距离费用都随之增加(见1.2节模型), 故IBKA采用客户之间的欧式距离作为评价标准, 把客户聚成和车场数目相同的几类, 使得每个车场服务其中一类客户. 但是, 随着客户数量逐渐增大, 聚类后会出现每个车场服务的客户数量相差较大. 为确保每个车场的客户资源能均衡分配, 设计IBKA进行聚. 由于文<sup>[37]</sup>

提出的K-means平衡算法仅适用于2类客户之间的平衡( $P_t=2$ ), 本节把文<sup>[37]</sup>提出的K-means平衡算法推广客户群, 同时设计车场分配规则, 可更为合理地确定各车场服务的客户群, 进而获得 $P_t$ 个单车场子问题LHFVRP\_TWs. IBKA的步骤如下:

**步骤1:** 对全体客户进行K-means聚类. 初始化聚类重心为每个车场的坐标位置, 初始化聚类数量为车场数量, 以欧式距离作为评价指标, 以迭代次数作为终止条件.

**步骤2:** 平衡客户群. 完成上述步骤后进行 $k$ 类客户群之间的平衡计算, 具体如图2所示, 按每类客户群数目由大到小排序, 根据2类客户间的边缘算法<sup>[37]</sup>把客户较多的客户群依次向其余客户较少的客户群移动, 循环操作直到各类客户群中客户数量数目达到平衡.

**步骤3:** 计算每个车场到各聚类重心的距离. 列出距离矩阵 $\mathbf{A}$ , 设 $\mathbf{A}$ 为 $P_t(\text{行}) \times P_t(\text{列})$ 的二维矩阵, 矩阵中的每一个元素表示某一车场坐标到某聚类重心的距离(详见式(16)).

$$\mathbf{A} = \begin{pmatrix} d(1,1) & \dots & d(1,P_t) \\ \vdots & \ddots & \vdots \\ d(P_t,1) & \dots & d(P_t,P_t) \end{pmatrix} \quad (16)$$

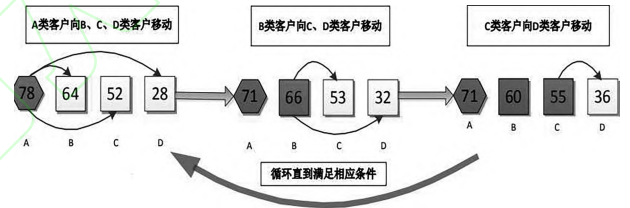


图2 4类客户平衡移动示意图

Fig. 2 Diagram of balanced movement for four customer groups

其中,  $d(i,j)$ 为车场坐标 $i$ 到聚类重心坐标 $j$ 的欧式距离

**步骤4:** 按设计的分配规则对车场所服务的客户群进行分配. 选取矩阵 $\mathbf{A}$ 中不同行(车场不同)不同列(客户群重心不同)的 $P_t$ 个元素并相加, 共有 $P_t!$ 种不同的组合及对应的累加和( $S_1, \dots, S_{P_t!}$ ), 选出其中最小的 $S_{min}$ (见式(17))所对应的组合中每个元素 $d(i,j)$ 的行、列标号即为车场 $i$ 所服务的客户群 $j$ . 采用 $S_{min}$ 确定车场服务的客户群有利于整体减小车辆运输距离.

$$S_{min} = \min(S_1, S_2, \dots, S_l, \dots, S_{P_t!}) \quad (17)$$

其中,  $S_l$ 表示矩阵 $\mathbf{A}$ 中对不同行不同列元素的第 $l$ 种

累加和。

对于K-means迭代次数为 $gen1_K$ 、平衡迭代次数为 $gen_b$ 、问题规模为 $N \cdot P_t$  (客户数量·车场数量)的问题, IBKA的计算复杂度(以下简称复杂度) $T_{IBKA}$  分别由K-means聚类复杂度 $O(gen1_K * P_t * N)$ 、客户平衡复杂度 $O(gen_b * (P_t * N)^2)$  (最坏情况)、获取方阵A的复杂度 $O(P_t^2)$  和获取 $S_{min}$ 的复杂度 $O(P_t * P_t!)$ 决定. 因此,  $T_{IBKA} = O(gen1_K * P_t * N + gen_b * (P_t * N)^2 + P_t^2 + P_t * P_t!)$ .

针对3车场和144客户的本文问题LMHFVPR\_TW, 采用IBKA 进行车场划分, 列出距离矩阵 $\mathbf{A}_{(3 \times 3)}$ , 分别列出矩阵 $\mathbf{A}_{(3 \times 3)}$ 中不同行不同列的 $P_t$ 个元素加和值的全部组合方式, 并计算3!次(详见式(18)), 若  $S_{min} = S_1$ , 即车场1 服A类客户, 车场2服务B类客户, 车场3服务C类客户, 划分结果如图3所示, 图中上、下图分别为未采用和采用平衡聚类结果, 从中可看出采用平衡聚类可使各车场服务的客户数量更加接近, 有利于提高各车场的整体效率和控制子问题规模。

$$\begin{aligned} S_1 &= d(1, 1) + d(2, 2) + d(3, 3) \\ S_2 &= d(1, 1) + d(2, 3) + d(3, 2) \\ &\dots\dots\dots \\ S_6 &= d(1, 3) + d(2, 2) + d(3, 1) \end{aligned} \quad (18)$$

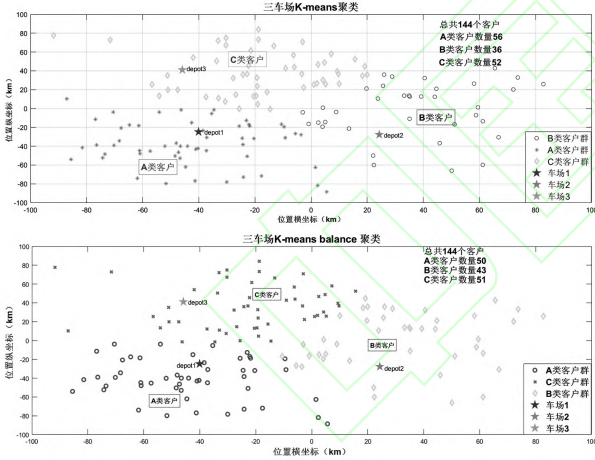


图3 三车场K-means未平衡聚类与平衡聚类比较

Fig. 3 The comparison of unbalanced K-means cluster and balanced K-means cluster of three depots

### 2.1.2 第2层聚类算法 (HKMA)

客户的时间窗、客户的货物需求重量、客户与车场间的距离是影响本文问题优化目标 (见式 (8)) 的重要因素, 故HKMA 将其作为客户的3类属性, 采用K-means对客户聚类后再为各类客户分配相应的车型K-means聚类时是把每个客户作为一个对象 (质点), 多属性客户进行K-means 聚类时, 需将各属性做加权和后才可进行. 然而, 上述3类属性对

优化目标 (见式 (8)) 的影响程度无法直接确定, 故需对各属性权重系数进行优化, 从而提高子问题分解的合理性。

HKMA为内外两层结构, 外层执行微粒群优化 (particle swarm optimization, PSO) 算法, 用于优化属性权重系数向量 $(\lambda = (\lambda_1, \lambda_2, \lambda_3))$ 且 $\lambda_1 + \lambda_2 + \lambda_3 = 1$ , 内层对基于每个 $\lambda_i$ 的聚类分解效果进行评价, 并将评价值 $f_i(\lambda_i)$  作为PSO 种群个体的适应度值. HKMA的工作机制如图4所示, 其中PSO种群 $P_{PSO}$ 的每个 $\lambda_i$  ( $i=1, \dots, M1$ )表示一组权重系数取值, 基于 $\lambda_i$ 对单车场问题LHFVRP\_TW做K-means 聚类可得 $M_t$  类客户群, 将 $M_t$ 类车型和 $M_t$  类客户群逐一配对得到单车场单车型子问题LVRP\_TW (记为 $\mathbf{A}_{i,j}$ ,  $i$ 和 $j$  分别为车型和客户群编号,  $i, j = 1, \dots, M_t$ ),  $f(\mathbf{A}_{i,j})$  表示对问题 $\mathbf{A}_{i,j}$  采用3.1节的编码规则随机生成10 个解的评价值 (即子问题总运输费用) 的平均值,  $\mathbf{F}_i$  为 $f(\mathbf{A}_{i,j})$  组成的方阵,  $Sum\_f_k$  ( $k = 1, \dots, M_t!$ ) 表示第 $k$ 个由 $\mathbf{F}_i$  中 $M_t$  个相互不同行 (车型不同) 不同列 (客户群不同) 元素累加所得的值,  $f_i\text{-min}$ 取所有 $Sum\_f_k$  中的最小值并赋值给 $f_i(\lambda_i)$ . 由于 $M_t$  个车型和客户群均不同的 $\mathbf{A}_{i,j}$ 可构成一个LHFVRP\_TW, 故 $Sum\_f_k$  实际上是对第 $k$  种子问题分解方式合理性的评估, 同时 $f_i\text{-min}$ 用于标识相对最为合理的子问题分解方式。

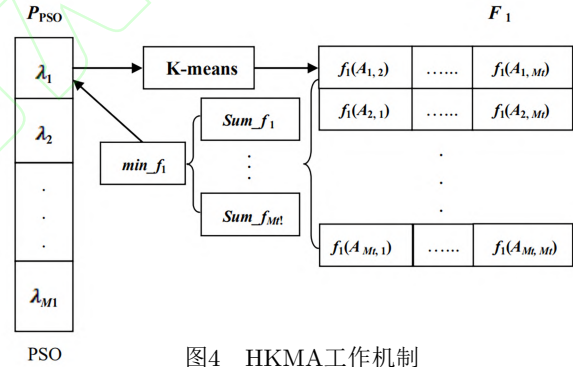


图4 HKMA工作机制

Fig. 4 The HKMA's running mechanism

令 $Z_{Total}(\mathbf{A}_{i,j})$  (见式(8))为个体 $\mathbf{A}_{i,j}$ 的目标函数值,  $\lambda_{opt}$ 为 $P_{PSO}$ 的历史最优个体 $sub P(\lambda_{opt})$ 为 $f(\lambda_{opt})$ 对应的 $M_t$ 个单车场单车型问题LVRP\_TW的集合 (具体见上一段), Maxgen(PSO)为HKMA中PSO算法的运行代数,  $gen2_K$ 为HKMA 中每次执行K-means算法的运行代数. HKMA的具体步骤如下:  
步骤1: 令 $gen = 1$ , 随机生成 $P_{PSO}$ 中的每个 $\lambda_i$ .  
步骤2 (内层): 确定 $P_{PSO}$ 中所有 $\lambda_i$ 的适应度值 (具体见上一段).  
步骤3 (外层): 若 $gen > 1$ , 则将当代 $P_{PSO}$ 和父代 $P_{PSO}$  的个体进行冒泡排序并择优保留前 $M1$ 个组成 $P_{PSO}$ . 更新 $\lambda_{opt}$ 和 $sub P(\lambda_{opt})$ .



步骤4 (外层):  $P_{PSO}$ 通过PSO算法进化后得到新的种群 $P_{PSO}$ , 也即得到 $M1$ 组新的权重系数向量.

步骤5: 令 $gen = gen + 1$ , 若 $gen \leq Maxgen(PSO)$ , 则返回步骤2, 否则输出 $sub P(\lambda_{opt})$ .

HKMA的复杂度 $T_{HKMA}$ 由步骤2至步骤5的复杂度决定. 对于单车场问题LHFVRP-TW, 问题规模为 $N_i \cdot M_t$  (客户数量\_车型数量), 其中 $\sum_{i=1}^{P_t} N_i = N$ . 步骤2的复杂度由K-means聚类复杂度 $O(gen2_K * M_t * N_i)$ 、获取方阵 $F_i$ 的复杂度 $O(M_t^2 * N_i)$ 、获取 $f_i \cdot min$ 的复杂度 $O(M_t * M_t!)$ 组成成为 $O(M1 * (gen2_K * M_t * N_i + M_t^2 * N_i + M_t * M_t!))$ ; 步骤3的复杂度为 $O(M1^2)$ ; 步骤4的复杂度为 $O(M1)$ . 因此,  $T_{HKMA} = O(Maxgen(PSO) * (M1 * gen2_K * M_t * N_i + M1 * M_t^2 * N_i + M1 * M_t * M_t! + M1^2))$ . 由于需采用HKMA对 $P_t$ 个单车场子问题LHFVRP-TWs进行分解, 故第2层分解总的复杂度为:

$$\begin{aligned} \text{total } T_{HKMA} = & O(Maxgen(PSO) \\ & * (M1 * gen2_K * M_t * N + M1 * M_t^2 \\ & * N + P_t * M1 * M_t * M_t! + P_t * M1^2)) \end{aligned}$$

带权重的多维k-means聚类

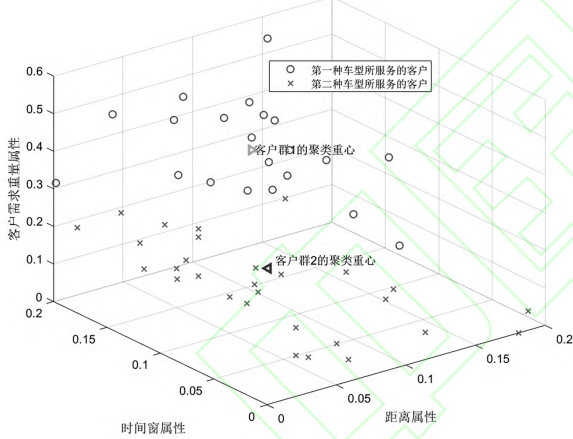


图5 HKMA三维聚类效果

Fig. 5 The 3D clustering results of HKMA

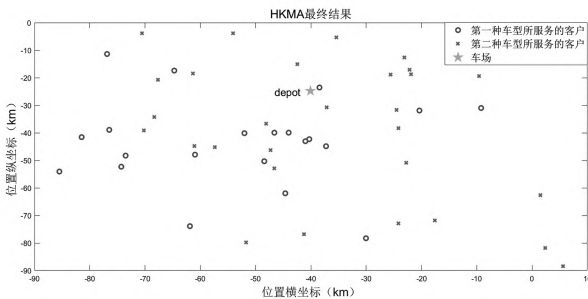


图6 HKMA二维结果

Fig. 6 The 2D results of HKMA

在本文算法EACO\_CD中, HKMA的参数设置为 $M1 = 10$ 、 $M2 = 10$ 、 $Maxgen(PSO) = 10$ 和 $gen2_K = 20$ , PSO中的惯性权重和加速因子的设置和文[38]相同, 具体设置为惯性权重 $\omega = 0.9$ , 速度因子 $c_1 = 1.2$ ,  $c_2 = 1.5$ . 针对2.1节中示例问题的车场1和其服务的A类客户 (即IBKA划分得到一个单车场子问题LHFVRP-TW), 采用HKMA进行车型划分, 可得如图5、图6. 由图5和图6可见, A类客户依据车型和客户的3类属性得以进一步划分, 从而将1个LHFVRP-TW分解为规模更小的2个单车场单车型子问题LVRP-TWs.

## 2.2 EACO\_CD的子问题求解阶段

通过上一阶段的两层聚类分解后, 原问题LMHFVPR-TW转换为 $P_t * M_t$ 个单车场单车型子问题LVRP-TWs, 从而本阶段可采用本文设计的子问题求解算法EACO依次对每个LVRP-TW进行求解, 进而可获得原问题的解和优化目标值 (见图1). 2.2.1节至2.2.4节介绍EACO的细节, 2.2.5节给出EACO的复杂度分析.

### 2.2.1 编码与解码规则

EACO对各车辆服务的所有客户进行整体编码. 每只蚂蚁代表多辆车, 每只蚂蚁的行驶路径 (即子问题LVRP-TW的 $l$ 个解) 为对应各车辆的总行驶路径. 譬如对于第 $l$ 只蚂蚁服务10个客户, 可记为:  $\pi_{ant(l)} = [1, 4, 9, 5, 10, 2, 3, 6, 7, 8]$ .

EACO采用文[39]中的方法对每只蚂蚁的行驶路径进行解码, 即逐一选择车场中的每辆车, 将 $\pi_{ant(l)}$ 中客户从左往右依次加入当前车辆, 如果当前车辆的载重量大于其服务客户的总需求量, 则继续往该车辆中加入客户, 否则选择下一辆车继续依次加入剩余客户. 譬如, 对于上一段的 $\pi_{ant(l)}$ , 假设需两辆车进行服务, 解码后可得 $\pi_{ant(l)} = [\pi_{vehicle(1)}, \pi_{vehicle(2)}] = [1, 4, 9, 5], [10, 2, 3, 6, 7, 8]$ .

显然, 当采用上述编码解码规则时, 式(13)的子回路消除约束自然得到满足, 同时也不违反其他约束.

### 2.2.2 全局搜索

#### 2.2.2.1 初始化

首先, 利用扫描法构造1只蚂蚁的行驶路径<sup>[40]</sup>, 并采用该路径对信息素浓度 $\tau_{ij}$ 进行初始化:

$$\tau_{ij} = \begin{cases} P_m, & \text{if } i, j \text{ in the route} \\ & \text{constructed by SWA} \\ 1, & \text{otherwise} \end{cases} \quad (19)$$

其中,  $i, j$ 为客户编号,  $P_m$ 为信息素浓度值的初始值 (设置为大于1的值). 为防止算法过早陷入局部

最优<sup>[41]</sup>, 设置信息素浓度 $\tau_{ij}$ 的最大值和最小值为:

$$\tau_{ij\_min} = \frac{Q_m}{2 \sum_{i=1}^N d_{Di}}, 0 < \tau_{ij\_min} < 1 \quad (20)$$

$$\tau_{ij\_max} = \frac{Q_m}{\sum_{i=1}^N d_{Di}}, \tau_{ij\_max} > P_m \quad (21)$$

其中,  $N$ 为车场需服务的全部客户总数,  $d_{Di}$ 为车场到客户 $i$ 的距离,  $Q_m$ 为常数( $\sum_{i=1}^N d_{Di} < Q_m < 2 \sum_{i=1}^N d_{Di}$ ).

然后, 设置信息素浓度增量 $\Delta\tau_{ij} = 0$ 和种群大小 $popsiz = (2/3) * N$ .

### 2.2.2.2 蚂蚁行驶路径搜索

EACO每代通过对当前种群中各蚂蚁行驶路径的确定来实现对问题解空间的全局搜索. 每只蚂蚁均利用式(22)确定其行驶路径.

$$P_{iu}^l = \begin{cases} \frac{(\tau_{iu})^\alpha * (\eta_{iu})^\beta}{\sum_{s \notin tabu} ((\tau_{is})^\alpha * (\eta_{is})^\beta)}, & \text{if } u \notin tabu \\ 0, & \text{otherwise} \end{cases} \quad (22)$$

其中,  $i$ 为当前客户,  $u$ 下一客户,  $P_{iu}^l$ 为第 $l$ 只蚂蚁从 $i$ 到 $u$ 的转移概率,  $s$ 为蚂蚁 $l$ 还未服务过的客户,  $\tau_{iu}(\tau_{is})$ 为 $i$ 和 $u(s)$ 间的信息素浓度,  $\eta_{iu}(\eta_{is})$ 为 $u(s)$ 之间距离的倒数,  $\alpha$ 和 $\beta$ 为重要程度权重,  $tabu$ 为蚂蚁已经搜索过的客户地点集合. 对 $\{P_{iu}^l | u \notin tabu\}$ 进行轮盘赌选择, 即可得到 $u$ 的取值, 从而确定蚂蚁 $l$ 从客户 $i$ 出发需到达的下一客户.

### 2.2.2.3 信息素浓度更新

EACO每代对每只蚂蚁的信息素浓度采用式(23)-(26)依次进行更新:

$$\gamma = \begin{cases} \gamma * (1 - \frac{M_a}{T}), & \text{The } L_b \text{ is unchanged for } M_a \text{ iterations} \\ \gamma * (1 + \frac{M_b}{T}), & \text{The } L_b \text{ is changed for } M_b \text{ iterations} \\ \gamma, & \text{otherwise} \end{cases} \quad (23)$$

$$\rho^{(t+1)} = \begin{cases} \gamma * \rho^t, & \text{if } \rho_{min} \leq \gamma * \rho^t \leq \rho_{max} \\ \rho_{min} \text{ or } \rho_{max}, & \text{otherwise} \end{cases} \quad (24)$$

$$\Delta\tau_{ij}^t = \begin{cases} \frac{W}{L_b}, & \text{if } \text{edge}(i, j) \in L_b \\ 0, & \text{otherwise} \end{cases} \quad (25)$$

$$\tau_{ij}^{(t+1)} = \rho^t * \tau_{ij}^t + \Delta\tau_{ij}^t, \rho^t \in [\rho_{min}, \rho_{max}] \quad (26)$$

其中,  $\gamma$ 为信息素挥发系数控制因子(初始设置为1),  $M_a$ 和 $M_b$ 为迭代次数系数(均设置为5),  $T$ 为基准值(设置为50),  $L_b$ 为当前最好解对应的目标函数值,  $t$ 表示算法进化的第 $t$ 代(即算法第 $t$ 次迭代),  $\rho^t$ 为第 $t$ 代的信息素挥发系数(初始设置为0.9),  $\rho_{min}$ 和 $\rho_{max}$ 分别为的最小值和最大值(分别设置为0和1),  $\Delta\tau_{ij}^t$ 为第 $t$ 代的信息素浓度增量,  $W$ 为信息素增量常数(设置为500),  $\tau_{ij}^t$ 为第 $t$ 代的信息素浓度.  $\gamma$ 根据算法每代的实际运行效果动态调整其取值, 可在运行效果好(差)时增加(减小)信息素浓度, 有利于增强算法合理引导搜索的能力.

### 2.2.3 局部搜索

本节在基本变邻域搜索(variable neighborhood search, VNS)<sup>[42]</sup>基础上提出一种两阶段变邻域局部搜索策略TVNS(two-stage variable neighborhood search, TVNS), 对EACO每代发现的当前最好解进一步执行高效且细致的邻域操作, 以提高解的质量.

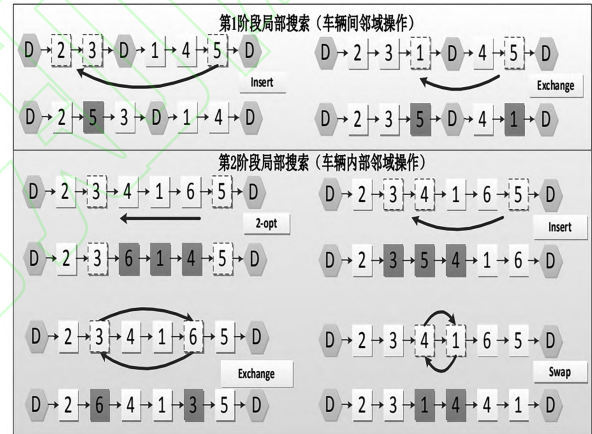


图7 局部搜索策略

Fig. 7 Local search strategy

TVNS为在车辆间和车辆内部分别执行多种邻域操作(见图7)的两阶段局部搜索, 两个阶段中各邻域操作均在当前最好解上执行且每次执行的最大次数均设置为 $gen_{local} = 20$ . 首先, 第1阶段随机选择两辆车, 依次对车辆间的客户执行“Insert”和“Exchange”邻域操作, 如果在执行操作期间发现更好解, 则更新当前最好解并立刻进入第2阶段, 否则各邻域操作均执行至最大次数 $gen_{local}$ 后进入第2阶段. 然后, 第2阶段逐一选择每辆车, 对每辆车的内部客户依次执行“2-opt”、“Insert”、“Exchange”和“Swap”邻域操作, 如果在执行操作期间发现更好解, 则更新当前最好解并立刻对下一辆车执行操作, 否则各邻域操作均执行至最大次数 $gen_{local}$ 后再对下一辆车执行操作.

第2阶段结束后完成本次局部搜索.

### 2.2.4 EACO终止条件

设定终止条件为算法的运行时间, 如果满足运行时间要求, 则输出服务所有客户的每辆车的行驶路径以及相应的行驶总费用.

### 2.2.5 EACO复杂度分析

EACO为EACO\_CD中求解由聚类分解(见2.1节)得到的单车场单车型子问题LVRP-TW的算法. 令EACO种群规模为 $popsiz_E$ , 全局搜索运行代数 $gen_{global}$ , 局部搜索每迭代次数 $gen_{local}$ , 局部搜索(含第1和第2阶段)执行时实际使用各邻域的平均总次数 $Z$ (即 $Z \leq 2 + 4 * \max \{H_{PM} | P = 1, 2, \dots, P_t, M = 1, 2, \dots, M_t\}$ ), 第 $i$ 个车场的LVRP-TW的客户数量为 $N_{i,x}$ , ( $\sum_{x=1}^{M_t} N_{i,x} = N_i$ ,  $\sum_{i=1}^{P_t} \sum_{x=1}^{M_t} N_{i,x} = N$ ). EACO的算法复杂度 $T_{EACO}$ 分别由EACO的全局搜索复杂度和局部搜索复杂度构成. 在每一代中, 全局搜索复杂度由种群初始化复杂度 $O(popsiz_E * N_{i,x})$ 、信息素矩阵和概率矩阵更新复杂度 $O(N_{i,x}^2)$ 、采样概率矩阵生成蚂蚁路径(即个体)的复杂度 $O(popsiz_E * N_{i,x}^2)$ 、种群评价复杂度 $O(popsiz_E * N_{i,x})$ 组成, 为 $O(popsiz_E * N_{i,x}^2)$ ; 局部搜索复杂度为 $O(gen_{local} * Z * N_{i,x})$ (最坏情况). 因此,  $T_{EACO} = O(gen_{global} * (popsiz_E * N_{i,x}^2 + gen_{local} * Z * N_{i,x}))$ . 由于需采用EACO对 $P_t * M_t$ 个单车场单车型子问题LVRP-TWs进行求解, 故采用EACO求解所有LVRP-TWs总的复杂度 $totalT_{EACO} = O(gen_{global} * (popsiz_E * \sum_{i=1}^{P_t} \sum_{x=1}^{M_t} N_{i,x}^2 + gen_{local} * Z * \sum_{i=1}^{P_t} \sum_{x=1}^{M_t} N_{i,x})) = O(gen_{global} * (popsiz_E * \log N * N + gen_{local} * Z * N))$ . 这里用 $\log N$ (而非默认以2为底数的 $\log N$ )表示低于 $N$ 的亚线性复杂度.

## 2.3 EACO\_CD整体结构及复杂度分析

### 2.3.1 EACO\_CD整体结构

整个算法EACO\_CD的结构如图1所示(以2个车场2类车型为例). 由图1可知, EACO\_CD对个子问题LVRP-TWs依次执行EACO进行求解, 最终将获得的各子问题解合并得到原问题的解, 同时将各子问题解的优化目标值累加得到原问题的优化目标值, 从而实现对原问题的求解.

### 2.3.2 EACO\_CD复杂度分析

EACO\_CD是对本文问题LMHFVPR-TW的整体求解, 其复杂度 $T_{EACO\_CD}$ 由分解算法IBKA的复杂度 $T_{IBKA}$ (见2.1.1节)、分解算法HKMA总的复杂度 $totalT_{HKMA}$ (见2.1.2节)、求解算法EACO总的复杂度 $totalT_{EACO}$ 组成(见2.2.5节),

即 $T_{EACO\_CD} = T_{IBKA} + totalT_{HKMA} + totalT_{EACO}$ . 虽然 $T_{IBKA}$ 、 $totalT_{HKMA}$ 和 $totalT_{EACO}$ 中的变量较多, 但除 $N$ (客户数量)、 $t$ (车场数量)、 $M_t$ (车型数量)与问题规模相关,  $Z$ (局部搜索每次迭代搜索的邻域或个体)与具体问题相关,  $popsiz_E$ 设置为 $2/3 * N$ (见2.2.2.1节)以外, 其余变量均与算法相关且实际使用中都设置为常数. 因此, 可得简化的 $T_{IBKA} = O((P_t * N)^2 + P_t * P_t!)$ 、 $totalT_{HKMA} = O(M_t^2 * N + P_t * M_t * M_t!)$ 、 $totalT_{EACO} = O(\log N * N^2 + Z * N)$ . 显然, 所有变量的多项式次数不超过2, 影响 $T_{EACO\_CD}$ 的是 $T_{IBKA}$ 中的 $O(P_t!)$ 和 $totalT_{HKMA}$ 中的 $O(P_t * M_t!)$ . 然而, 实际问题中 $P_t$ 和 $M_t$ 一般都不大于10, 而单计算10!用CPU为2.6GHZ的主流PC可在1秒左右完成计算, 故EACO\_CD对大多数实际问题可在较短时间内结束运算. 另外, 如果 $P_t(M_t)$ 很大(譬如为50), 可将IBKA的步骤4(HKMA的步骤2)中穷举不同行不同列的组合改为采样 $P_t(M_t)$ 的多项式次数的组合, 从而扩大EACO\_CD求解问题的规模.

## 3 实验比较与分析

本节所有算法的测试均采用: 英特尔I7处理器(3.2GHz)、8G内存、Win10操作系统、matlab2018a编程环境.

### 3.1 EACO参数设定

在第1节所提出的目标函数中(式(7)), 距离费用系数设定参考文[7], 车辆固定费用参考文[9], 燃油费用系数设定参考文[43], 时间窗惩罚费用系数参考文[16], 具体取值见表2.

表2 目标函数中的相关系数

Table 2 Coefficients in the object function

符号	数值
$C_{M1}$	1.5元/kM
$C_{M2}$	300-800元/辆
$C_{M3}$	7.6元/L
$C_1$	15元/H
$C_2$	20元/H

表3 主要参数与水平

Table 3 Main parameters and level

主要参数	水平设置			
	1	2	3	4
$\alpha$	1.25	1.5	1.75	2.0
$\beta$	10	1.5	2.0	2.5
$P_m$	1.1	1.2	1.3	1.4
$W$	500	1000	1500	2000



表4 参数设置的正交表

Table 4 Orthogonal table of parameter settings

组合编号	水平设置				AVR (元)
	$\alpha$	$\beta$	$P_m$	$W$	
1	1	1	1	1	9677
2	1	2	2	2	9625
3	1	3	3	3	9613
4	1	4	4	4	9541
5	2	1	2	3	9745
6	2	2	1	4	9624
7	2	3	4	1	9602
8	2	4	3	2	9593
9	3	1	3	4	9836
10	3	2	4	3	9703
11	3	3	1	2	9654
12	3	4	2	1	9612
13	4	1	4	2	9865
14	4	2	3	1	9689
15	4	3	2	4	9656
16	4	4	1	3	9672

EACO的残留信息素重要程度 $\alpha$ , 启发信息素重要程度 $\beta$ , 初始化信息素浓度参数 $P_m$ , 信息素增量常数 $W$ 为个主要的参数, 为确定合适的参数组合, 对这4个参数进行正交实验, 各参数设置水平如表3所示. 由于聚类分解后原问题LMHFVPR-TW转变为 $P_t * M_t$ 个LVRP-TWs, 故采用Solomon标准VRPTW数据集集中的问题c101 (100个客户)进行测试. 每组参数组合下的EACO在c101上独立运行20次, 取20次的平均最小运输总费用值作为平均响应值ARV, 参数设置的正交表见表4, 参数的平均响应值和影响力见表5. 由表4和表5可得EACO的参数设置为:  $\alpha = 1.25, \beta = 2.5, P_m = 1.1, W = 500$ .

表5 各参数不同水平下的平均响应值和影响力

Table 5 Average response values and influences table at different levels of each parameter

水平	水平设置			
	$\alpha$	$\beta$	$P_m$	$W$
1	<b>9614</b>	9780	9656	9645
2	<b>9641</b>	9660	9659	9684
3	<b>9701</b>	9631	9683	9683
4	<b>9720</b>	9604	9677	9664
极差	<b>106</b>	176	27	39
影响力排名	<b>2</b>	1	4	3

## 3.2 实验设计及测试问题选取

### 3.2.1 实验设计

EACO\_CD由IBKA、HKMA和EACO组成. 为验证EACO\_CD的有效性, 先将这3种算法分别与国际期刊上的相近算法进行比较, 最后再进行总体比较.

首先, 为验证EACO的有效性, 在多车场单车型问题LMVRP-TW上, 将EACO1和国际期刊中的有效算法DHACO<sup>[37]</sup>进行对比. EACO1为EACO\_CD不含两层聚类的算法, 而DHACO为不含分解且直接求解整个问题的一类蚁群算法. EACO1采用DHACO的编码. 测试结果见表7.

其次, 为单独验证IBKA的有效性, 在多车场单车型问题LMVRP-TW上, 将EACO\_IBKA和EACO\_KM、EACO\_NNA进行对比. 这3种算法为在EACO1中分别加入3类第1层聚类算法 (即IBKA、K-means<sup>[12,14]</sup>、NNA<sup>[11]</sup>)后得到. 测试结果见表7.

然后, 为单独验证HKMA的有效性, 在单车场多车型问题LHFVPR-TW上, 将EACO\_HKMA和EACO\_RDA、EACO\_RAA、EACO\_KEW、EACO\_2、TSA\_RDA<sup>[10]</sup>进行对比. 前4种算法为在EACO1中分别加入4类第2层聚类算法 (即HKMA、RDA<sup>[10]</sup>、RAA、KEW)后得到, 其中RDA根据客户需求量优先选择最大载重量的车型, 进而将问题分解为一系列单车LVRPs (即TSPs), RAA随机将客户划分给某种车型, KEW每类客户属性分配相同权重后用K-means划分车型 (即简化的HKMA), RAA和KEW均将问题分解为一系列单车LVRP-TWs; EACO2为EACO1改用多车型编码后得到, 用于直接求解整个问题; TSA\_RDA为国际期刊中的有效算法, 该算法在禁忌搜索算法中加入RDA. 测试结果见表8.

最后, 为验证EACO\_CD的有效性, 在本文问题LMHFVPR-TW上, 将EACO\_CD和国际期刊中的有效算法IACO\_CD<sup>[16]</sup>和IHGA<sup>[13]</sup>进行对比. IACO\_CD为两层分解算法, 该算法第2层将问题分解为单车LVRP (即TSPs), 而IHGA通过加入车型染色体后对问题进行整体求解. 测试结果见表9.

### 3.2.2 测试问题选取

本文的测试数据数据来源于网站 (<http://neo.lcc.uma.es/vrp/>) 中的MDVRP数据集 (数据中不存在明显聚类特征的客户位置分布). 其中包括问题pr01 (48个客户), pr02 (96个客户), pr03 (144个客户), pr04 (192个客户), pr05 (240个客户), pr06 (288个客户), p22

(360个客户). 由于该数据中未含有车型数据, 故在HKMA和EACO\_CD的有效性验证时加入车型数据(见表6),

表6 4种不同车型相关参数设置

Table 6 Related parameter settings for four different vehicle types

车型列表	车型参数				
	载重量(kg)	空车重量(kg)	平均速度(km/h)	固定费用(元)	最大承货物载数(件)
Type 1	200	1600	60-80	300-400	20
Type 2	500	2700	50-70	400-500	30
Type 3	600	3500	40-60	500-600	40
Type 4	800	5000	30-50	600-800	50

进而构成多车场多车型问题. 各比较算法在每个问题上独立运行20次, 每个算法设置相同运行时间 $T$ (单位: s). 性能指标为各算法20次运行中的最优值best、平均值average、最差值worst和标准差SD. 各种带分解策略的比较算法每次运行时间为问题分解算法与问题求解算法运行时间的总和, 譬如EACO\_CD每次运行的时间为EACO、IBKA和HKMA的运行时间总和, 表7-9中粗体为算法运行结果\*的较小值. 需要说明的是, 为节约空间, 表7-9中仅提供问题pr02(96个客户), pr04(192个客户), pr06(288个客户), p22(360个客户)的相关测试结果, 更多问题和性能指标下的测试结果可网上下载\*. 表7-9中的测试结果和网上测试结果具有相近统计性能, 均可对各比较算法性能做出较客观判断.

3.3 实验结果比较和分析

3.3.1 验证EACO有效性

由表7可知, EACO的最优值和平均值均优于DHACO. 在算法全局搜索部分, DHACO和EACO的复杂度无明显差异, 但EACO在全局阶段引入信息素挥发系数控制因子进一步动态调节信息素挥发系数, 可有效控制信息素的挥发从而提高了算法的全局搜索能力. 在算法局部搜索部分, 相比DHACO的VNS策略, EACO的TVNS能有效搜索更多不同区域, 具有更强的局部搜索能力.

3.3.2 验证IBKA有效性

由表7可知, EACO\_IBKA总体优于EACO\_KM和EACO\_NNA. 这表明IBKA通过聚类 and 平衡客户群, 可更合理地分配车场并较好实现子问题解耦. 同时, EACO\_IBKA也总体优于不含问题分解策略

的整体求解算法EACO1和DHACO. 这说明采用有效的问题分解策略为算法提前确定需搜索的较优区域, 可在一定程度上避免过多的低效搜索, 有利于增强算法性能. 此外, 测试结果显示带分解策略的算法更适用于较大规模问题, 原因在于这些问题的解空间更加庞大, 仅依靠智能算法自身进化机制难以快速引导算法发现优质解区域.

3.3.3 验证HKMA有效性

由表8可知, EACO\_HKMA总体优于其他比较算法. 这表明HKMA综合考虑客户的3类属性并为各属性设定合适的权重, 可更合理地划分子问题. 具体而言, EACO\_HKMA明显优于EACO\_RDA和TSA\_RDA, 这说明RDA将问题分解为更小规模的单车辆LVRPs(即TSPs)会将算法实际搜索空间限定得过小, 从而遗漏较多存在优质解的区域; EACO\_HKMA明显优于EACO\_RAA(除N=96的1个较小规模问题)和EACO\_KEW, 这说明RAA随机划分子问题的方式和KEW给各属性分配相同权重后再划分子问题的方式, 均无法获取真正优质的搜索区域; EACO\_HKMA也明显优于整体求解算法EACO2, 这再次验证了将问题先合理分解可把算法搜索直接限定在较优搜索区域进行, 有利于提高算法效率. 此外, 与3.3.2节类似, 本节的分解策略同样对较大规模问题更有效.

3.3.4 验证EACO\_CD有效性

由表9可知, EACO\_CD总体优于IACO\_CD和IHGA. 具体而言, EACO\_CD的最优值、平均值、最差值均明显优于IACO\_CD, 且标准差整体较小, 这说明IACO\_CD将问题分解为更小规模的单车辆LVRPs(即TSPs)会遗漏较多存在优质解的区域, 同时IACO\_CD自身算法的搜索能力有限, 从而导致该算法的性能和鲁棒性较差; EACO\_CD的最优值、平均值、最差值、标准差明显优于IHGA(除N=96的2个较小规模问题), 这表明随着车场、车型和客户的增多, 本文问题LMHFVPR\_TW的可行解空间呈指数倍增加, 仅采用智能算法(如IHGA)难以在较短时间内完成大范围搜索以获得原问题的优质解, 而EACO\_CD依靠合理的问题分解能将EACO的搜索限定在较优区域之内, 进而利用EACO较强的搜索能力可短时间内从较优区域中获得优质解, 从而使其性能和鲁棒性均较强.

综上所述, 结合IBKA和KHMA的EACO\_CD是求解本文问题LMHFVPR\_TW的有效鲁棒算法.

\*文章中表7至表9的完整测试结果可在: <https://pan.baidu.com/s/19sqBboZHLcGfQtZS7Id-Q> 提取码3cv6下载. 若无法下载可联系通讯作者.

表7 EACO\_IBKA与其他算法对比结果

Table 7 Comparison results of EACO\_IBKA with other algorithms

$N \cdot P_t$	EACO_IBKA		EACO_KM		EACO_NNA		EACO1		DHACO		$T(s)$
	best	average	best	average	best	average	best	average	best	average	
96_2	17483	18155	18037	18379	16768	17675	<b>15371</b>	<b>17009</b>	16011	17559	19
192_2	<b>27522</b>	<b>28546</b>	28457	29482	28758	29560	28379	29838	28524	30863	38
288_2	<b>39217</b>	<b>41028</b>	40412	42363	41179	42142	41283	43164	42606	43930	58
360_2	<b>53748</b>	<b>56268</b>	54672	57402	55251	57847	56276	58653	56965	59409	72
96_3	16638	17278	17066	17491	15957	16821	<b>14627</b>	<b>16582</b>	15236	17186	29
192_3	<b>26199</b>	<b>27175</b>	27090	28066	27376	28140	27016	28405	27153	29381	58
288_3	<b>37337</b>	<b>39062</b>	38476	40333	39205	41123	39305	41096	40565	41826	86
360_3	<b>51176</b>	<b>53576</b>	52056	54656	52608	55080	53584	55848	54240	56568	108
96_4	16062	16943	16810	17146	<b>15871</b>	<b>16425</b>	16012	16907	16391	16991	38
192_4	<b>25690</b>	<b>26612</b>	26528	27235	26846	27620	26443	27896	26623	28850	77
288_4	<b>36519</b>	<b>38266</b>	37624	39493	38365	39338	38442	40278	39692	40974	115
360_4	<b>50480</b>	<b>52904</b>	51344	53960	51904	54392	52088	55184	53584	55904	144
平均值	<b>33173</b>	<b>34651</b>	34048	35501	34174	35514	34069	35905	34799	36620	/

表8 HKMA与其他划分算法的对比结果

Table 8 Comparison results of HKMA and the other dividing algorithms

$N \cdot M_t$	EACO_HKMA		EACO_RDA		EACO_RAA		EACO_KEW		EACO2		TSA_RDA		$T(s)$
	best	average	best	average	best	average	best	average	best	average	best	average	
96_2	16299	<b>16759</b>	17570	19461	<b>16238</b>	17661	16754	17704	16543	17838	17746	19656	29
192_2	<b>24847</b>	<b>25998</b>	25983	26905	26000	27201	26039	26918	26520	27745	26243	27174	57
288_2	<b>32225</b>	<b>33356</b>	33783	34713	32740	35242	32540	35156	34050	36652	34121	35060	87
360_2	<b>48344</b>	<b>50050</b>	50695	52088	49129	52874	48828	52743	51094	54989	51202	52609	108
96_3	<b>15777</b>	<b>15929</b>	16704	18494	16011	16792	15932	16839	16171	16960	16871	18679	44
192_3	<b>23624</b>	<b>24718</b>	24704	25570	24714	25850	24748	25588	25455	26626	24951	25826	87
288_3	<b>30621</b>	<b>31700</b>	32106	32984	31111	33493	30929	33414	32667	35168	32427	33314	129
360_3	<b>45933</b>	<b>47554</b>	48172	49502	46691	50237	47401	50118	49026	52749	48654	49997	162
96_4	<b>14214</b>	<b>14349</b>	15047	16655	14422	15118	14346	15171	14566	15269	15197	16822	57
192_4	<b>21273</b>	<b>22251</b>	22237	23018	22257	23268	22282	23034	22925	23966	22459	23248	116
288_4	<b>27563</b>	<b>28542</b>	28902	29689	28012	30149	27842	30088	29413	31656	29191	29986	173
360_4	<b>41350</b>	<b>42809</b>	43360	44558	42028	45218	41776	43111	44129	47479	43794	45004	216
平均值	<b>28506</b>	<b>29501</b>	29939	31136	29113	31092	29118	30824	30213	32258	30238	31448	/



表9 EACO\_CD 性能验证  
Table 9 EACO\_CD' s performance verification

$N.P_t.M_t$	EACO_CD				IACO_CD				IHGA				$T(s)$
	(EACO_IBKA_HKMA)				(IACO_NNA_SWA)								
	best	average	worst	SD	best	average	worst	SD	best	average	worst	SD	
96_2_2	16370	16526	16904	295	17800	18071	18381	<b>272</b>	<b>15852</b>	<b>16011</b>	<b>16376</b>	285	31
192_2_2	<b>23664</b>	<b>24760</b>	<b>25650</b>	<b>556</b>	26040	27250	28218	576	24863	26010	26948	590	71
288_2_2	<b>30690</b>	<b>31768</b>	<b>32227</b>	<b>724</b>	33772	34962	35457	803	35311	36550	37076	824	123
360_2_2	<b>39903</b>	<b>41303</b>	<b>41906</b>	<b>951</b>	43907	45448	46111	1059	45507	46098	47781	1197	162
96_3_2	16883	<b>17869</b>	<b>18993</b>	358	18568	20115	20999	<b>343</b>	<b>16800</b>	18283	19011	363	48
192_3_2	<b>24257</b>	<b>25492</b>	<b>26408</b>	<b>668</b>	26692	28059	29065	671	26688	28052	29053	746	109
288_3_2	<b>30258</b>	<b>31572</b>	<b>32860</b>	<b>907</b>	34814	36319	37797	1017	35414	36956	38450	1092	183
360_3_2	<b>39347</b>	<b>41061</b>	<b>42734</b>	<b>1141</b>	43291	45187	47020	1263	47228	49291	51297	1389	243
96_3_3	<b>15867</b>	<b>17082</b>	<b>17860</b>	388	17470	18911	19664	<b>368</b>	16514	17874	18584	478	62
192_3_3	<b>22816</b>	<b>23977</b>	<b>24832</b>	<b>631</b>	25108	26383	27320	708	26242	27588	28567	743	145
288_3_3	<b>28458</b>	<b>29693</b>	<b>30892</b>	<b>828</b>	32742	34157	35530	957	35582	37120	38622	1048	245
360_3_3	<b>37006</b>	<b>38610</b>	<b>40166</b>	<b>1079</b>	40722	42474	44186	1198	47081	49427	51523	1399	288
平均值	<b>27127</b>	<b>28309</b>	<b>29286</b>	711	30077	31445	32479	770	31090	32438	33607	846	/

4 结论

针对多车场多车型绿色车辆路径问题LMHFVRP\_TW, 本文提出一种结合聚类分解策略的增强蚁群算法(EACO\_CD) 进行求解. 这是首次采用智能优化算法求解该问题. EACO\_CD包括两个阶段. 第一阶段为问题分解, 该阶段设计两种聚类分解策略IBKA和HKMA, 用于对LMHFVRP\_TW 进行分解, 从而有效制问题规模并快速引导算法在较优区域内搜索. 第二阶段为子问题求解, 该阶段采用EACO对每个分解后的子问题LVRP\_TW 进行求解. 在EACO的全局搜索中, 通过加入信息素挥发系数控制因子动态调节信息素挥发系数的取值, 可较好避免算法出现过早收敛, 进而引导算法搜索达到解空间中更多的不同区域. 同时, 为增强EACO的局部搜索能力, 设计一种两阶段变邻域搜索对全局搜索获得的优质解区域进行细致且高效的搜索. 仿真实验和算法比较验证了所提EACO\_CD是求解LMHFVRP\_TW的有效算法.对于HFVRP和MHFVRP, 已有的问题分解方法大都将其最终分解为一系列最为简单的单车VRP(即标准TSP), 这一分解方式是否合理且唯一, 尚无人探讨. 本文所提的IBKA和HKMA把LMHFVRP\_TW 有效分解为一系列单车VRP (即多车辆TSP), 使其子问题具有较大的可行解区域进而适当扩大了算法的实际搜索空间, 取得了更好的效果. 这表明合理选取子问题可在一定程度上提高算法求解性能, 也对复杂车辆路径问题如何进行分解优化设计具有借鉴意义. 后续研究将把EACO\_CD和随机处理技术结合, 用于

求解动态复杂车辆路径问题.

References

- 1 Dantzig G B, Ramser J H. The truck dispatching problem. *Management Science*. 1959, **1**(6): 80–81.
- 2 Anbuudayasankar S P, Ganesh K, Mohapatra S. *Models for Practical Routing Problems in Logistics*. Springer Press, 2014.
- 3 Li H Q, Yuan J L, LV T, et al. The two-echelon time-constrained vehicle routing problem in linehaul-delivery systems considering carbon dioxide emissions. *Transportation Research Part D*. 2016(**49**): 231–245.
- 4 Zhao Y W, Zhang J L, Wang W L. *Vehicle Routing Optimization Method for Logistics Distribution*. Beijing: Science Press, 2014.
- 5 Sbihi A, Eglese R. Combinatorial optimization and green logistics. *4OR: A Quarterly Journal of Operations Research*. 2007, **2**(5): 97–116.
- 6 Chen D S, Batson R, Dang Y. *Applied integer Programming*. New York: Wiley Press, 2010.
- 7 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. *Transportation Research Part D: Transport and Environment*. 2017,**57**: 422–457.
- 8 Kaabachi I, Jriji D, Krichen S. An Improved Ant Colony Optimization for Green MultiDepot Vehicle Routing Problem with Time Windows. *IEEE*, 2017.
- 9 Xiao Y, Konak A. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. *Transportation Research Part E: Logistics and Transportation Review*. 2016, **88**: 146–166.

- 10 Kwon Y J, Choi Y J, Lee D H. Heterogeneous fixed fleet vehicle routing considering carbon emission. *Transportation Research Part D*. 2013, **23**: 81–89.
- 11 Geetha S. Metaheuristic approach for the multi-depot vehicle routing problem. *Applied Artificial Intelligence: An International Journal*. 2012, **26**: 878–901.
- 12 Geetha S, Poonthalir G, Vanathi P T. Nested particle swarm optimization for multi-depot vehicle routing problem. *Operational Research*. 2013, **16**(3): 112–128.
- 13 Ho W, Ho G T S, Ji P, et al. A hybrid genetic algorithm for the multi-depot vehicle routing problem. *Engineering Applications of Artificial Intelligence*. 2008, **21**: 548–557.
- 14 Wang Y, Kevin Assogba, Liu Y. Two-echelon location-routing optimization with time windows based on customer clustering. *Expert System with Application*. 2018, **104**: 244–260.
- 15 Dondo R, Cerdá J. A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. *European Journal of Operational Research*. 2007, **176**: 1478–1507.
- 16 Tang Y L, Cai Y G, Yang Q J. Improved ant colony optimization for multi-depot heterogeneous vehicle routing problem with soft time windows. *Journal of Southeast University (English Edition)*. 2015, **31**(1): 94–99.
- 17 Dorigo M, Maniezzo V, Colnari A. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*. 1996, **26**(1): 29.
- 18 Wang Su Xin, GAO Li, CUI Xiao Guang, CAO Hong Mei. Study on multi-requirement points vehicle scheduling model and its swarm mix algorithm. *ACTA AUTOMATICA SINICA*. 2008, **34**: 102–104.  
(王素欣, 高利, 催小光, 曹宏美. 多需求点车辆调度模型及其群体智能混合求解. 自动化学报. 2008, **34**: 102–104.)
- 19 Lee C Y, Lee Z J, Lin S W, et al. An enhanced ant colony optimization (EACO) applied to capacitated vehicle routing problem. *Applied Intelligence*. 2010, **32**(1): 88–95.
- 20 Yu B, Yang Z Z. An ant colony optimization model: The period vehicle routing problem with time windows. *Transportation Research Part E*. 2011, **47**(2): 166–181.
- 21 Ding Q L, Hu X P, Sun L J, et al. An improved ant colony optimization and its application to vehicle routing problem with time windows. *Neurocomputing*. 2012, **98**: 101–107.
- 22 Yan F L. Autonomous vehicle routing problem solution based on artificial potential field with parallel ant colony optimization (ACO) algorithm. *Pattern Recognition Letters*. 2018, **116**: 195–199.
- 23 X Chen, D Hu, Q Yang, et al. An improved ant colony algorithm for multi-objective vehicle routing problem with simultaneous pickup and delivery. *Control Theory & Applications*. 2018, **35**(09): 1347–1356.  
(陈希琼, 胡大伟, 杨倩倩, 等. 多目标同时取送货车辆路径问题的改进蚁群算法. 控制理论与应用. 2018, **35**(09): 1347–1356.)
- 24 Xu H T, Pu P, Duan F. Dynamic vehicle routing problems with enhanced ant colony optimization. *Discrete Dynamics in Nature and Society*. 2018: 1–13.
- 25 Demir E, Bektas T, Laporte G. An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*. 2012, **223**: 346–359.
- 26 Bektas T, Gouveia L. Gouveia Requiem for the Miller-Tucker-Zemlin subtour elimination constraints. *European Journal of Operational Research*. 2014, **236**: 820–832.
- 27 Toth P, Vigo D. *The Vehicle Routing Problem*. Beijing: Tsinghua University Press, 2012.
- 28 Wolsey L A. *Integer Programming*. New York: Wiley Press, 1998.
- 29 Schneider M, Stenger A, Goeke D. The electric vehicle-routing problem with time windows and recharging stations. *Transportation Science*. 2014: 1–21.
- 30 Gu S S. A polynomial time solvable algorithm to linearly constrained binary quadratic programming problems with Q being a tri-diagonal matrix. *Advances in Information Sciences and Service Sciences*. 2011, **3**(6): 65–72.
- 31 Meng X, Li J, Zhou M C, et al. Population-based incremental learning algorithm for a serial colored traveling salesman problem. *IEEE Transactions on Systems Man & Cybernetics: Systems*. 2018, **48**(2): 277–288.
- 32 Wang Y, Ma X L, Lao Y T, et al. A fuzzy-based customer clustering approach with hierarchical structure for logistics network optimization. *Expert Systems with Applications*. 2013, **40**(7): 78–91.
- 33 Wang Y, Zhang J, Kevin Assogba, et al. Collaboration and transportation resource sharing in multiple centers vehicle routing optimization with delivery and pickup. *Knowledge-Based Systems*. 2018, **160**: 296–310.
- 34 Wang B D, Miao Y W, Zhao H Y, et al. A biclustering-based method for market segmentation using customer pain points. *Engineering Applications of Artificial Intelligence*. 2015, **06**: 5–14.
- 35 Ji J C, Pang W, Zhou C G, et al. A fuzzy k-prototype clustering algorithm for mixed numeric and categorical data. *Knowledge-Based Systems*. 2012, **30**: 129–135.
- 36 Yong Wang, Xiaolei Ma, Mingwu Liu, et al. Cooperation and profit allocation in two-echelon logistics joint distribution network optimization. *Applied Soft Computing*. 2017, **02**: 25–67.
- 37 He R, Xu W, Sun J, et al. Balanced K-Means Algorithm for Partitioning Areas in Large-Scale Vehicle Routing Problem. *IEEE*, 2009.
- 38 Wang D F, Li Meng. Performance analysis and parameter selection of PSO algorithms. *ACTA AUTOMATICA SINICA*. 2016, **10**: 1552–1561.  
(王东风, 孟丽. 粒子群优化算法的性能分析和参数选择. 自动化学报. 2016, **10**: 1552–1561.)
- 39 Beasley J E. Route first—Cluster second methods for vehicle routing. *OMEGA*. 1983, **11**(4): 403–408.

- 40 E G B, Miller Leland R. A heuristic algorithm for the vehicle-dispatch problem. *Operations Research*. 1974, **22**(2): 340-349.
- 41 Yua B, Yao B. An improved ant colony optimization for vehicle routing problem. *European Journal of Operational Research*. 2009, **196**: 171-176.
- 42 Mladenovic M, Hansen P. Variable neighborhood search. *Computers and Operations Research*. 1997, **24**: 1097-1100.
- 43 Li J, Fu P. Heterogeneous fixed fleet low-carbon routing problem and algorithm. *Computer integrated manufacturing systems*. 2013, **06**: 1351-1362.  
(李进, 傅培华. 具有固定车辆数的多车型低碳路径问题及算法. 计算机集成制造系统. 2013, **06**: 1351-1362.)



**胡 蓉** 昆明理工大学信息工程与自动化学院副教授, 2004年获得清华大学自动化系硕士学位, 主要研究方向为调度理论与方法, 智能计算, 决策支持系统.

E-mail: ronghu@vip.163.com

(**HU Rong** Associate professor at the School of Information Engineering and Automation, Kunming University of Science and Technology. She received her Master degree from Tsinghua University in 2004. Her research interests include scheduling theory and method, intelligent computation, and decision support system.)



**李 洋** 昆明理工大学信息工程与自动化学院硕士研究生, 2009年获得昆明理工大学电力工程学院学士学位. 主要研究方向为调度理论与智能优化算法.

E-mail: Yang.L.Liam@hotmail.com

(**LI Yang** Postgraduate at the School of Information Engineering and Automation, Kunming University of Science and Technology. He received his bachelor degree from Kunming University of Science and Technology in 2009. His research interests include scheduling methods and intelligent optimization algorithms.)

Technology. He received his bachelor degree from Kunming University of Science and Technology in 2009. His research interests include scheduling methods and intelligent optimization algorithms.)



**钱 斌** 昆明理工大学信息工程与自动化学院副教授, 2009年获得清华大学自动化系博士学位, 主要研究方向为调度理论与方法, 以及智能优化.

E-mail: bin.qian@vip.163.com

(**QIAN Bin** Professor at the School of Information Engineering and Automation, Kunming University of Science and Technology. He received his Ph.D. degree from Tsinghua University in 2009. His research interests include scheduling theory and method, and intelligent optimization. Corresponding author of this paper.)



**金怀平** 昆明理工大学信息工程与自动化学院副教授, 2016年获得北京理工大学博士学位, 主要研究方向为智能计算和软测量方法.

E-mail:

(**JIN Huai-Ping** Associate professor at the School of Information Engineering and Automation, Kunming University of Science and Technology. He received his Ph. D. from Beijing Institute of Technology in 2016. His research interests include intelligent computation and soft sensor methods.)



**向凤红** 昆明理工大学信息工程与自动化学院教授, 2002年获得昆明理工大学博士学位, 主要研究方向为智能优化与控制.

E-mail:

(**XIANG Feng-Hong** Professor at the School of Information Engineering and Automation, Kunming University of Science and Technology. He received his Ph. D. from Kunming University of Science and Technology in 2002. His research interests include intelligent optimization and control.)