

超启发式分布估计算法求解带软时间窗的同时 取送货车车辆路径问题

张烜荧, 胡 蓉[†], 钱 斌

(昆明理工大学 信息工程与自动化学院 自动化系, 云南 昆明 650500;

昆明理工大学 云南省人工智能重点实验室, 云南 昆明 650500)

摘要: 本文针对带软时间窗的同时取送货车车辆路径问题(VRSPDSTW), 以最小化车辆行驶总里程和最大化服务准时率为优化目标, 提出一种超启发式分布估计算法(HHEDA)进行求解. 全局搜索阶段, 首先, 提出3种启发式规则生成初始个体, 以确保初始种群的质量和分散性; 其次, 根据问题特点, 构造3个概率矩阵分别学习和积累优质解的排序信息、客户间的距离信息和捆绑信息, 并通过采样概率矩阵生成新个体, 以增强算法全局搜索发现解空间中优质区域的能力. 局部搜索阶段, 将11种邻域操作组成备选集合, 进而设计学习型超启发式局部搜索(LHHLS), 用于动态选择备选集合中的部分邻域操作构成多种新的有效启发式算法, 以执行对解空间中优质区域的深入搜索. 最后, 仿真实验和算法比较验证了HHEDA的有效性.

关键词: 同时取送货车车辆路径问题; 软时间窗; 多目标优化; 超启发式分布估计算法

引用格式: 张烜荧, 胡蓉, 钱斌. 超启发式分布估计算法求解带软时间窗的同时取送货车车辆路径问题. 控制理论与应用, 2021, 38(9): 1427 – 1441

DOI: 10.7641/CTA.2021.00408

Hyper-heuristic estimation of distribution algorithm for solving vehicle routing problem with simultaneous pickup and delivery and soft time windows

ZHANG Xuan-ying, HU Rong[†], QIAN Bin

(Faculty of Information Engineering and Automation, Kunming University of Science and Technology,
Kunming Yunnan 650500, China;

Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology,
Kunming Yunnan 650500, China)

Abstract: This paper proposes a hyper-heuristic estimation of distribution algorithm (HHEDA) for the vehicle routing problem with simultaneous pickup and delivery and soft time windows (VRSPDSTW), whose optimization objectives are to minimize the total mileage of vehicles and maximize the on-time rate of service at the same time. In the global search stage. Firstly, three heuristic rules are presented to generate initial individuals to ensure the quality and diversity of the initial population. Secondly, according to the characteristics of the problem, three probability matrices are constructed to learn and accumulate the ordinal information of high-quality solutions, the distance information and the binding information among customers, respectively, and new individuals are generated by sampling these probability matrixes. By this way, the algorithm's global search ability of finding the high-quality regions in solution space can be enhanced. In the local search stage, eleven kinds of neighborhood operations are used to form the alternative set, and then a learning hyper-heuristic local search (LHHLS) is designed to dynamically select some neighborhood operations in the alternative set to form a variety of new effective heuristic algorithms, so as to execute the in-depth searches from the high-quality regions in solution space. Finally, simulation experiments and algorithm comparisons verify the effectiveness of the proposed HHEDA.

Key words: vehicle routing problem with simultaneous pickup and delivery; soft time windows; multi-objective optimization; hyper-heuristic estimation of distribution algorithm

Citation: ZHANG Xuanying, HU Rong, QIAN Bin. Hyper-heuristic estimation of distribution algorithm for solving vehicle routing problem with simultaneous pickup and delivery and soft time windows. *Control Theory & Applications*, 2021, 38(9): 1427 – 1441

收稿日期: 2020-07-02; 录用日期: 2021-05-18.

[†]通信作者. E-mail: ronghu@vip.163.com.

本文责任编辑: 柯良军.

国家自然科学基金项目(61963022, 51665025)资助.

Supported by the National Natural Science Foundation of China (61963022, 51665025).

1 引言

随着“制造强国战略”的实施,在推动先进制造业迅速发展的同时,现代物流业的高质量发展也成为大势所趋.在物流系统中,配送是核心部分,而车辆路径问题(vehicle routing problem, VRP)是其中的重要一环.传统的VRP主要描述为组织一个车队,安排适当的行车路线为一定数量的客户提供送货服务,并能在满足车载量、行车里程等约束条件下,达到诸如车辆行驶总里程最短,成本最小,耗时最少等目的.随着资源需求的增加与供给双向流通方式的快速发展,以及物流企业对降低成本、保护环境的注重,考虑逆向物流的车辆路径问题开始受到重视.此外,日益激烈的市场竞争逼迫企业在满足客户需求的同时要考虑提高客户满意度.在此背景下,研究带软时间窗的同时取送货车车辆路径问题(vehicle routing problem with simultaneous pickup and delivery and soft time windows, VRPSPDSTW)具有重要意义.由于带时间窗的同时取送货车车辆路径问题(vehicle routing problem with simultaneous pickup and delivery and time windows, VRPSPDTW)为NP-hard问题^[1],而VRPSPDTW可归约为VRPSPDSTW,故VRPSPDSTW也属于NP-hard问题,对其展开研究具有较大理论价值.

VRPSPDSTW的求解算法主要分为两类.一类是运筹学算法,包括分支定价、动态规划和拉格朗日松弛等^[2-8],基本都是用于求解单目标问题.此类算法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式的结构信息构造搜索,可在几分钟至几十分钟内获取较小规模问题(客户数小于等于20)的最优解.但是,VRPSPDSTW具有非凸特性,其几何结构和最优解间的关系仍属开放问题,不存在多项式时间获取最优解的算法,故对应的运筹学算法均是靠遍历或部分遍历解空间来确保求解质量,这使其需要数小时至数天才可获得较大规模问题(客户数在80~200之间)的最优解或满意解.另一类是智能算法,包括遗传算法、禁忌搜索算法、蚁群算法等^[9-29],既用于求解单目标问题,也用于求解多目标问题.此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、解码规则中来处理,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不同规模问题的满意解.因此,设计智能算法以实现VRPSPDSTW的快速有效求解,是合理且必要的.

近年来,智能算法求解VRPSPDSTW逐渐受到重视.针对单目标VRPSPDTW, Lai等^[9]提出改进差分进化算法(improved differential evolution algorithm, IDE)求解问题,取得较好的效果. Wang等^[11]提出一种基于最小代价插入法的协同进化遗传算法(genetic algorithm, GA)进行求解,实验结果表明所提算法的有效

性.王等^[10]提出模拟退火算法(simulated annealing, SA),并在模拟退火过程中融合局部搜索方法,在可接受的计算时间内取得了较好的优化结果.王等^[11]提出一种离散布谷鸟算法(discrete cuckoo search, DCS),其Lévy飞行机制协调局部搜索和全局搜索,实验结果更新了国际最优解.韩等^[12]针对带折线型软时间窗的车辆路径问题(vehicle routing problem with soft time windows, VRPSTW),以最小化运输总成本为目标,提出一种超启发式遗传算法(hyper-heuristic genetic algorithm, HHGA),以GA作为上层搜索算法, CW节约法、MJ插入法、Kilby插入法作为底层搜索规则,并通过加入排序规则和解内部调整算法以进一步提高解的质量.实验结果验证了HHGA的可行性和有效性.王等^[13]提出一种回溯搜索优化算法(backtracking search optimization algorithm, BSA),使用启发式规则产生初始种群,通过多种搜索算子更新局部最优解,实验验证了BSA的有效性. Qin等^[14]进一步考虑了低碳需求,提出一种自适应遗传爬山算法(adaptive genetic hill-climbing algorithm, AGHCA)进行求解,同时研究了碳税对总成本和碳排放的影响.

对于多目标VRPSPDSTW (multi-objective VRPSPDSTW, MOVSPDSTW)及相关问题,李等^[15]针对VRPSPDTW,以最小化总车辆数、运输总时间、车辆行驶总里程的线性组合为优化目标,提出一种基于路线集合划分的分解迭代算法,将路线集合分解为路线子集合,再用记录更新法求解子集合,不断迭代逐渐逼近最优解,实验结果表明了算法的有效性. Banos等^[16]针对考虑负载均衡的带硬时间窗的车辆路径问题(vehicle routing problem with time windows, VRP-TW),以同时最小化各车辆间最大行驶距离之差和车辆间最大负载之差为优化目标,提出一种结合模拟退火算法的多起点多目标进化算法(multi-start multi-objective evolutionary algorithm with simulated annealing, MMOEASA).仿真实验和对比结果验证了算法的有效性. Yang等^[17]针对带软时间窗的车辆路径问题(VRPSTW),以最小化总运营成本、时间窗偏离总量、碳排放为优化目标,提出一种混合遗传算法(hybrid genetic algorithm, HGA),通过实验验证了算法的有效性,并通过帕累托分析,说明了目标之间的关系. Sumaiya等^[18]针对VRPSTW,以最小化总行驶距离、违反时间窗的服务次数、车辆数为优化目标,提出一种人工蜂群算法(artificial bee colony, ABC),结合车辆内和车辆间的两步局部搜索进行求解,实验结果表明ABC能在合理的计算时间内获得高质量的解. Zhang等^[19]针对时间依赖型同时取送货车车辆路径问题(time dependent vehicle routing problem with simultaneous delivery and pickup, TDVRPSDP),以最小化碳排放和旅行时间为优化目标,提出一种多目标

协同进化量子算法(multi-objective cooperative quantum evolutionary algorithm, MCQEA), 将种群分为两个亚种群, 使用非劣解集和亚种群中的解共同产生新种群, 实验结果验证了MCQEA的有效性. Li等^[20]针对MOVRPSPDSTW, 以最小化车辆数、车辆固定成本、行驶总距离、旅行时间、服务成本为优化目标, 使用基于分解的多目标化学优化算法(chemical reaction optimization, CRO), 将复杂的MOVRPSPDSTW转换为多个单目标子问题, 并同时在CRO中对其进行优化, 实验结果验证了算法的有效性. 陈等^[21]针对多目标同时取送货车辆路径问题(multi-objective vehicle routing problem with simultaneous pickup and delivery, MOVRPSPD), 以最小化总路径成本、各车辆最大行驶里程差为优化目标, 提出一种嵌入禁忌表、且具有贪婪转移准则的多目标改进蚁群算法(improved ant colony optimization, IACO), 实验结果表明IACO可求得权衡各优化目标且使单一优化目标近似最优的Pareto解. 柴等^[22]考虑路径具有双重属性, 建立基于双属性的多目标VRPTW, 进而提出一种蚁群算法(ant colony algorithm, ACO), 在算法中定义了一种综合考虑各优化目标、时间窗和信息素等启发信息的状态转移概率公式. 实验结果表明, 所提算法在运行时间、收敛性和群体多样性方面具有优势. Wang等^[23]针对带时间窗的周期性车辆路径问题(periodic vehicle routing problem with time windows, PVRPTW), 以同时最小化车辆行驶总距离、最小化车辆总数、最大化客户访问频率为优化目标, 提出一种多目标模拟退火蚁群优化算法(multi-objective simulate annealing-ant colony optimization, MOSAACO), 在算法中设计了基于欧氏距离的优质解接受方式和可生成不同初始解的参数控制策略, 并加入了有效局部搜索策略. 实验表明, 该算法在求解PVRPTW问题上性能良好. 由文献调研可知, 虽然VRPSPDSTW已得到一定研究, 但尚无人考虑客户满意度(服务准时率)这一重要优化目标. 此外, 上述算法在全局搜索时基本都直接采用算法原有搜索机制执行, 未考虑结合问题特性做合理更改来增强算法发现优质解的能力, 且局部搜索时大多迭代执行较为单一的插入、变换操作, 难以实现对复杂问题解空间中优质解区域的深入搜索. 显然, 针对VRPSPDSTW这类复杂问题, 如何设计能获取高质量解的有效算法是需要进一步研究的课题.

分布估计算法(estimation of distribution algorithm, EDA)是一种新兴的基于统计学原理的随机优化算法. EDA采用统计学习手段从群体宏观角度建立概率模型, 用来学习和积累优质解在搜索空间的分布信息, 然后对此概率模型采样产生新种群, 如此反复实现种群进化^[24]. 此进化方式可一定程度上避免传统进

化算法中交叉、变异等操作所带来的模式破坏问题, 从而使其搜索具有更好的引导性^[25]. 目前, 使用EDA求解VRP的研究较为有限. Wan等^[26]针对VRPSTW, 提出一种混合粒子群算法的混合EDA, 通过粒子群算法的快速更新策略修改了EDA的概率矩阵, 实验结果验证了算法的有效性. 孟等^[27]针对带硬时间窗的车辆路径问题(VRPTW), 提出一种混合种群增量学习算法(hybrid population-based incremental learning algorithm, HPBIL), 为每辆车建立2维EDA概率模型, 同时设计了基于插入和两点邻域交换的两阶段局部搜索来增强算法的局部开发能力, 实验结果表明混合后的算法求解效果优良. Wang等^[28]针对客户需求和运输成本不确定的车辆路径问题(uncertain capacitated vehicle routing problems, UCVRP), 提出一种改进EDA, 不仅利用概率模型学习优质解信息并引导全局搜索, 而且引入插入操作和2-opt操作来提高局部搜索能力, 同时采用多场景技术, 增强算法求解不确定问题的鲁棒性. Ji等^[29]针对考虑碳排放的多目标联合运输VRP, 提出一种混合EDA(hybrid estimation of distribution algorithm, HEDA), 采用扩展的2维概率矩阵学习各阶段运输模式对应的优质非支配解信息, 同时设计了基于部分交换操作的局部搜索, 对问题进行了有效求解. 从EDA在VRP相关问题上的研究现状来看, 采用基于问题的概率模型可以合理引导全局搜索的方向, 引入有效的局部搜索策略能进一步提高算法的性能, 两者的协同搜索是算法设计的关键. 据文献调研可知, 针对VRPSPDSTW, 尚无EDA求解的报道, 故十分必要开展相关研究.

综上, 本文研究多目标VRPSPDSTW的建模与求解. 建模方面, 建立同时以最小化总行驶路径和最大化服务准时率为优化目标的VRPSPDSTW. 求解方面, 在对问题模型特点分析的基础上, 提出一种超启发式分布估计算法(hyper-heuristic estimation of distribution algorithm, HHEDA). HHEDA在全局搜索阶段执行分布估计算法EDA, 在局部搜索阶段执行学习型超启发式局部搜索(learning hyper-heuristic local search, LHHLS). 其中, EDA利用结合问题特性所设计的3类概率矩阵协同学习和积累优质解(即可更新当前非劣解集的解)信息并引导搜索尽快到达优质解区域; LHHLS采用学习型评分机制和11种有效邻域操作构造一系列启发式算法或搜索, 然后分别对当前种群的部分个体和当前非劣解集中的解逐一执行相应的启发式搜索, 可进一步加大搜索深度, 从而增大算法在较短时间内获取高质量解的机率. 最后, 通过对不同测试问题的仿真实验和算法比较, 验证了HHEDA的有效性.

可更新当前非劣解集的解)信息, 然后通过采样这些概率矩阵以生成较高质量的新个体或解, 并更新非劣解集和概率矩阵, 可较快引导算法到达解空间中存在优质解的区域. 在HHEDA的局部搜索阶段, 执行学习型超启发式局部搜索LHHLS, 以增强算法的局部寻优能力. 在LHHLS中, 低层问题域采用11种有效邻域操作组成备选集合, 高层控制域设计改进的学习型评分机

制动态评价底层不同邻域操作在搜索过程中的贡献(即对两个优化目标的改进程度), 进而依据各邻域操作的贡献率, 对全局搜索阶段EDA种群中部分个体和非劣解集中的解, 逐一执行由4种邻域操作(用轮盘赌方法确定)构成的启发式算法, 以实现较深入的局部搜索并更新当前非劣解集和概率矩阵.

本节后续对HHEDA各环节进行介绍.

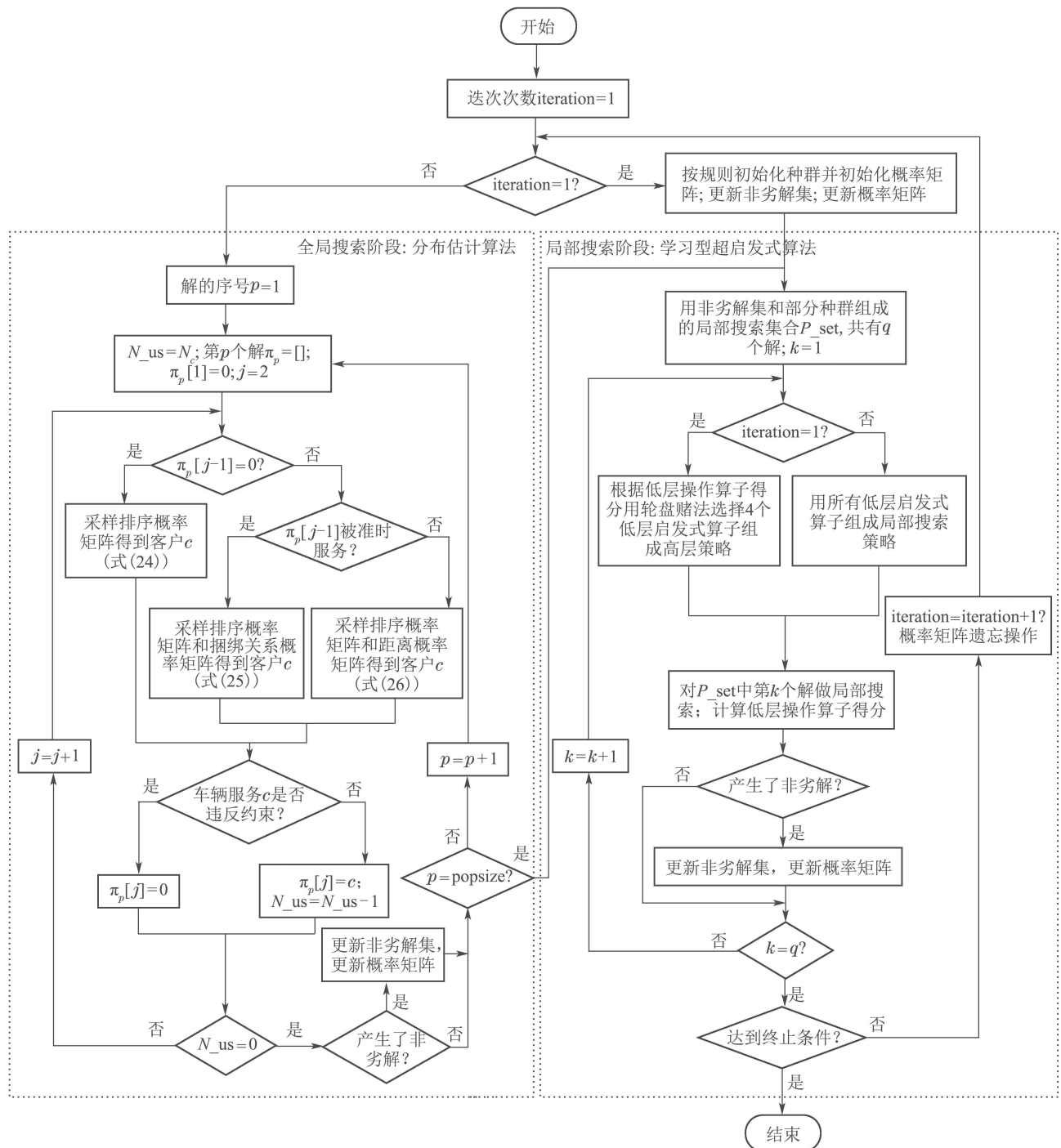


图2 HHEDA流程图

Fig. 2 HHEDA's flow chart

3.1 解的表示

令解 π 的长度为 L , $\pi = (\pi[1], \pi[2], \dots, \pi[L])$, 其中 $N_c + 2 \leq L \leq 2 \times N_c + 1$. 配送中心编号为0, 客户编号为大于0的整数. 车辆从配送中心开出前往客户处, 服务一定数量的客户后返回配送中心. 如图1所示, 编码为 $\pi = (0, 1, 2, 3, 4, 0, 5, 6, 7, 0, 8, 9, 10, 0)$ 的解表示车辆1从配送中心开出, 依次服务客户1、客户2、客户3、客户4后返回配送中心. 车辆2、车辆3的服务顺序释义以此类推.

3.2 全局搜索

3.2.1 种群初始化

为了保证种群的多样性和分散性, 使用规则和随机方法产生规模为popsize的第一代种群. 使用规则1和规则2产生2个解, 剩下popsize-2个解使用规则3产生. 令未被服务的客户集合为 C . 具体规则如下:

规则1 最近邻规则.

步骤 如果车辆从配送中心出发, 计算配送中心与 C 中每个客户的距离, 选择最近的一个客户进行服务, 如果有多个最近客户, 随机选择其中一个进行服务, 并将这个客户从 C 中去除; 如果车辆从客户 i 出发, 计算 i 与 C 中客户的距离, 选择最近的一个客户进行服务, 如果有多个最近客户, 随机选择其中一个进行服务, 如果车辆继续服务该客户不会违反约束, 则将这个城市从 C 中去除, 否则, 车辆放弃服务该客户并返回配送中心.

规则2 准时率最高规则.

步骤 如果车辆从配送中心出发, 计算 C 中每个客户的到达时间, 选择在时间窗内到达的客户进行服务, 如果有多个可在时间窗内到达的客户, 选择其中最近的进行服务, 如果没有可在时间窗内到达的客户, 在 C 中选择距离最近的客户进行服务, 将选中的客户从 C 中去除; 如果车辆从客户 i 出发, 计算车辆从 i 到 C 中每个客户的时间, 选择在时间窗内到达的客户进行服务, 如果有多个可在时间窗内到达的客户, 选择其中最近的进行服务, 如果车辆继续服务该客户不会违反约束, 则将这个城市从 C 中去除, 否则, 车辆放弃服务该客户并返回配送中心, 如果没有可在时间窗内到达的客户, 车辆返回配送中心.

规则3 随机规则.

步骤 在 C 中随机选取客户进行服务, 如果车辆服务该客户不会违反约束条件, 那么将这个客户从 C 中去除, 否则, 车辆放弃服务该客户并返回配送中心.

3.2.2 概率矩阵更新

在设计概率矩阵时, 通过研究问题的特性, 引入问题的特定信息和知识, 是提高算法搜索能力的有效途径. 对于VRPSDSTW, 为了最小化车辆行驶总里程, 车辆总是倾向于先服务那些离当前位置比较近的客

户, 所以在车辆行驶总里程较小的解中, 存在一些由客户间距离决定的位置信息; 为了最大化服务准时率, 车辆总是倾向于按照时间窗先后顺序服务客户, 若客户 i, j 的时间窗和距离满足式(13)的关系, 那么当车辆连续服务客户 i, j 时, 只要车辆在时间窗内到达 i , 那么 j 一定能在时间窗内被服务, 即 ij 满足捆绑关系. 所以, 在服务准时率较高的解中, 客户间由于时间窗和距离较为匹配, 存在一些捆绑关系信息. 在HHEDA中, 用 ρ, λ, τ 三个矩阵分别学习优质解的排序信息、客户间的距离信息和捆绑关系.

概率矩阵初始化阶段, ρ 和 λ 为 $N_c \times N_c$ 的全零矩阵, τ 为 $N_c \times N_c$ 的矩阵, 用式(14)对 τ 中元素值进行初始化. 在概率矩阵更新时, 选取非劣解集中新增加的个体集合 Ω 按式(16)–(20)更新概率矩阵. 其中, 在每张车辆服务顺序中, $\rho_{j'j}$ 表示客户 j 出现在位置 j' 的次数; K_{ij} 为决策变量, 当客户 i 与 j 被连续服务时为1, 否则为0; $I_{j'j}$ 为决策变量, 当客户 j 出现在位置 j' 时为1, 否则为0; γ_{ij} 为学习速率, 其值为客户 ij 之间距离的倒数; L_{ij} 为决策变量, 当客户 i 与 j 被连续服务并且车辆到达 i, j 的时间都在时间窗内时其值为1, 否则为0.

$$\begin{cases} a_i \leq t_i \leq b_i, \\ a_j \leq t_i + s_i + l_{ij} \leq b_j, \end{cases} \quad (13)$$

$$\tau_{ij} = \begin{cases} 1, & \text{客户 } i, j \text{ 满足捆绑关系,} \\ 0, & \text{否则,} \end{cases} \quad (14)$$

$$I_{j'j} = \begin{cases} 1, & \text{客户 } j \text{ 出现在位置 } j', \\ 0, & \text{否则,} \end{cases} \quad (15)$$

$$\rho_{j'j} = \rho_{j'j} + \sum_{\Omega} I_{j'j}, \quad (16)$$

$$K_{ij} = \begin{cases} 1, & \text{客户 } i, j \text{ 被连续服务,} \\ 0, & \text{否则,} \end{cases} \quad (17)$$

$$\lambda_{ij} = \lambda_{ij} + \sum_{\Omega} \gamma_{ij} K_{ij}, \quad (18)$$

$$L_{ij} = \begin{cases} 1, & \tau_{ij} \text{ 不为0且 } ij \text{ 在时间窗内被服务,} \\ 0, & \text{否则,} \end{cases} \quad (19)$$

$$\tau_{ij} = \tau_{ij} + \sum_{\Omega} L_{ij}. \quad (20)$$

3.2.3 概率矩阵遗忘操作

在迭代过程中, 新产生的非劣解会支配掉部分原非劣解集中的解, 为了使从新非劣解中学习到的信息在概率矩阵中占较大比重, 在算法每一次迭代结束后对概率矩阵按式(21)–(23)执行遗忘操作, 遗忘因子为 α ($\alpha < 1$)

$$\rho_{ij} = \alpha \times \rho_{ij}, \quad (21)$$

$$\lambda_{ij} = \alpha \times \lambda_{ij}, \quad (22)$$

$$\tau_{ij} = \alpha \times \tau_{ij}. \quad (23)$$

3.2.4 对概率模型采样产生种群

算法的第iteration (iteration ≥ 2)代, 新种群由精英保留机制和采样概率矩阵产生的解共同构成. 随机选择 m 个非劣解($m = \min(\text{ps}, \lceil \text{popsize} \times 15\% \rceil)$), ps 为非劣解集中解的个数)作为精英解集, 剩余 $\text{popsize} - m$ 个解由概率矩阵采样生成.

采样过程中, 根据式(24)–(26), 考虑客户排序信息、距离信息和捆绑关系, 产生较优个体. 其中 C 为未被服务的客户集合, $(i-1)'$ 表示在车辆中被第 $i-1$ 个被服务的客户, p_{ij} 为客户 j 在车辆中被第 i 个服务的概率, 每次采样均使用轮盘赌操作, 在一定程度上保证了种群的多样性.

当产生位置 $i=1$ 的客户时, 由于是车辆第1个服务的客户, 所以仅考虑客户在解中的排序位置, 由式(24)计算 C 中客户被选中的概率; 当产生位置 $i(i \geq 2)$ 的客户时, 若客户 $(i-1)'$ 被准时服务, 则在计算 C 中客户被选中的概率时, 着重考虑客户之间的捆绑关系, 以最大化服务准时率为侧重优化目标, 由客户排序信息和捆绑关系综合计算概率, 计算公式如式(25)所示; 若客户 $(i-1)'$ 没有被准时服务, 着重考虑客户之间的距离关系, 以最小化总距离为侧重优化目标, 由客户排序信息和距离关系综合计算概率, 计算公式如式(26)所示:

$$p_{1j} = \frac{\rho_{1j}}{\sum_{l \in C} \rho_{1l}}, \quad (24)$$

$$p_{ij} = \frac{1}{2} \times \left(\frac{\rho_{ij}}{\sum_{l \in C} \rho_{ij}} + \frac{\tau_{(i-1)'j}}{\sum_{l \in C} \tau_{(i-1)'l}} \right), \quad (25)$$

$$p_{ij} = \frac{1}{2} \times \left(\frac{\rho_{ij}}{\sum_{l \in C} \rho_{ij}} + \frac{\lambda_{(i-1)'j}}{\sum_{l \in C} \lambda_{(i-1)'l}} \right). \quad (26)$$

综上, 算法任意代数的种群初始化的伪代码如表2所示.

3.3 局部搜索

对于多目标组合优化问题, 全局最优非劣解集中的任何解(即全局最优非劣解)在所有邻域结构下均为局部最优非劣解, 而接近全局最优非劣解的优质解往往在多种邻域结构下均为局部最优非劣解. 采用单一邻域操作(对应一种邻域结构)的迭代搜索容易较早达到并陷入该邻域结构的局部最优非劣解, 但该解的质量大都一般; 而采用多种有效邻域操作(对应多种邻域结构)的迭代搜索可在到达多种邻域结构共同的局部最优非劣解前一直持续进行搜索, 从而容易获取优质非劣解. 此外, 多目标组合优化问题的解空间十分复杂, 大量优质非劣解分散在庞大的“峡谷型”解空间中靠近底部的多个区域^[30]. 因此, 为能对HHEDA全局搜索发现的优质解区域(即非劣解集)进行有效搜索,

本文设计学习型超启发式局部搜索(LHHLS), 用于执行丰富的多种变邻域搜索, 以增强算法局部搜索的深度.

表2 种群初始化

Table 2 Population initialization

Algorithm 1: 种群初始化

```

输入: iteration, c_set, m, popsize
输出: 种群
1 if iteration=1 then
2    $n = 1$ 
3   repeat
4     if  $n = 1$  then
5       使用规则1产生一个解
6     else if  $n = 2$  then
7       使用规则2产生一个解
8     end
9   else
10    使用规则3产生一个解
11  end
12  if 产生非劣解 then
13    更新非劣解集; 用产生的解更新概率矩阵
14  end
15   $n = n + 1$ 
16 until  $n = \text{popsize}$ ;
17 else
18   随机选择 $m$ 个非劣解作为一部分种群
19   repeat
20      $k = 1$ 
21     repeat
22       customer_set = c_set
23       if 车辆 $k$ 从车场处出发 then
24         用式(24)计算customer_set中每个客户的概率;
25         使用轮盘赌法选择客户 $C_j$ 
26       else if 车辆 $k$ 从客户 $C_i$ 处出发 then
27         if  $C_i$ 在时间窗内到达 then
28           用式(25)计算customer_set中每个客户的概率;
29           使用轮盘赌法选择客户 $C_j$ 
30         else
31           用式(26)计算customer_set中每个客户的概率;
32           使用轮盘赌法选择客户 $C_j$ 
33         end
34       end
35     end
36     if 车辆 $k$ 继续服务 $C_j$ 违反约束 then
37       车辆 $k$ 放弃服务 $C_j$ , 返回车场;
38        $k = k + 1$ 
39     else
40       车辆继续服务客户 $C_j$ ;
41       将 $C_j$ 从未服务客户集合customer_set中移除
42     end
43      $m = m + 1$ 
44     if 产生非劣解 then
45       更新非劣解集; 用产生的解更新概率矩阵
46     end
47   until customer_set = [];
48 until  $m = \text{popsize}$ ;
49 end

```

超启发算法为两层结构, 底层问题域包含一组直接搜索问题解空间的启发式算法(low-level heuristics, LLHs), 其中每个LLH可为启发式算法, 也可以是规则或邻域操作; 高层策略域包含确定底层不同LLH组合的某种算法. 高层策略域的算法(high-level strategy, HLS)通过动态控制底层问题域的LLHs来不断组合生产新的启发式算法, 进而使用这些新的启发式算法在底层搜索问题解空间. 对于本文的LHHLS, 底层问题域设计11种LLH, 每种LLH为一类有效邻域操作; 高层策略域的算法由改进的学习型评分机制和轮盘赌方法共同构成. 前者用于定量地评估和统计每个LLH在搜索过程中的执行效果; 后者依据所有LLHs的评分(即贡献)对其进行轮盘赌, 以生成一系列新的启发式算法(即LLH的组合).

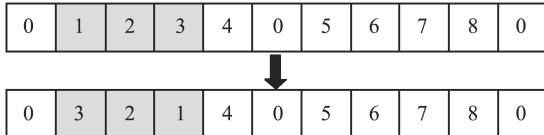
显然, 搜索能力较强的LLH能以较大概率被选中, 其他LLH也能以一定概率被选中, 这有利于生成多种新的有效启发式算法.

下面对LHHLS各环节进行介绍.

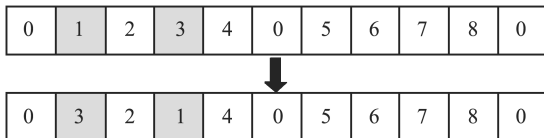
3.3.1 LLH设计

LLH要根据具体问题进行设计. 本文设计了2类共11种邻域变换操作作为低层启发式算子, LLH₁至LLH₅为车辆内路径局部搜索操作, LLH₆–LLH₁₁为车辆间路径局部搜索操作.

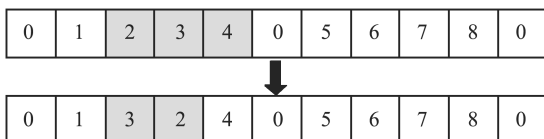
1) LLH₁: 车辆内客户顺序逆转操作. 选择被同一张车连续服务的几个客户, 进行逆转.



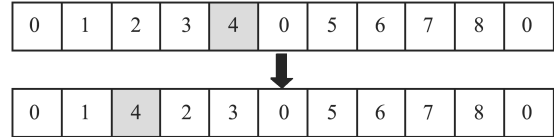
2) LLH₂: 车辆内不相邻客户交换操作. 选择被同一张车服务的两个不连续的客户, 进行交换.



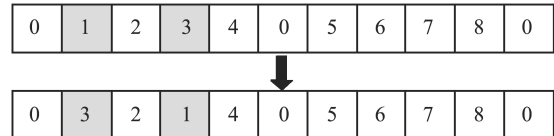
3) LLH₃: 车辆内相邻客户交换操作. 选择被同一张车服务的两个相邻的客户, 进行交换.



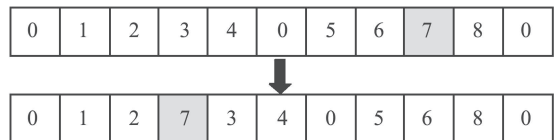
4) LLH₄: 车辆内客户插入操作. 选择一张车辆的服务路线 r_i , 在 r_i 中随机选择一个城市 c , 将其插入到 r_i 中的其他位置.



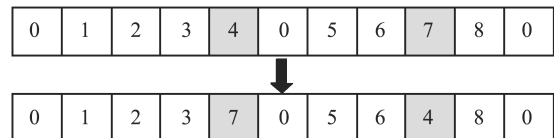
5) LLH₅: 车辆内客户基于准时性的交换操作. 选择一张车辆的服务路线 r_i , 在 r_i 中选择车辆晚于时间窗下限到达的城市 c_1 和车辆早于时间窗上限到达的城市 c_2 , 交换 c_1 和 c_2 .



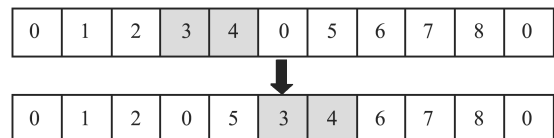
6) LLH₆: 车辆间客户插入操作. 选中两张车辆的服务路线 r_i 和 r_j , 在 r_i 中随机选择一个城市 c , 将其插入到 r_j 中.



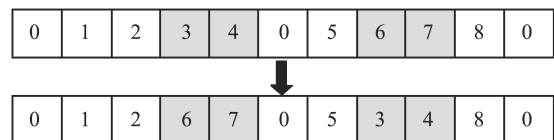
7) LLH₇: 车辆间客户交换操作. 选中两张车辆的服务路线 r_i 和 r_j , 在 r_i 中随机选择一个城市 c_1 , 在 r_j 中随机选择一个城市 c_2 , 交换 c_1 和 c_2 .



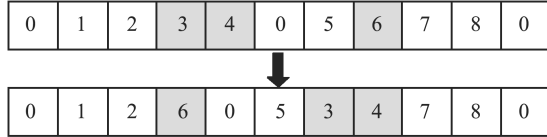
8) LLH₈: 车辆间连续客户插入操作. 选中两张车辆的服务路线 r_i 和 r_j , 在 r_i 中随机选择两个被连续服务的城市 c_1 和 c_2 , 将 c_1 和 c_2 插入到 r_j 中.



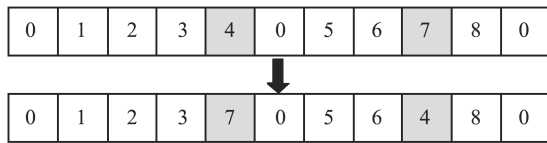
9) LLH₉: 车辆间连续客户交换操作. 选中两张车辆的服务路线 r_i 和 r_j , 在 r_i 中随机选择两个被连续服务的城市 c_1 和 c_2 , 在 r_j 中随机选择两个被连续服务的城市 c_3 和 c_4 , 交换 c_1 , c_2 和 c_3 , c_4 .



10) LLH_{10} : 车辆间客户交换操作. 选中两张车辆的服务路线 r_i 和 r_j , 在 r_i 中随机选择两个连续服务的城市 c_1 和 c_2 , 在 r_j 中随机选择一个城市 c_3 , 交换 c_1 , c_2 和 c_3 .



11) LLH_{11} : 车辆间客户基于准时性的交换操作. 选择两张车辆服务路线 r_i 和 r_j , 在 r_i 中选择车辆晚于时间窗下限到达的城市 c_1 , 在 r_j 中选择车辆早于时间窗上限到达的城市 c_2 , 交换 c_1 和 c_2 .



在对解执行任意一种 LLH_i ($1 \leq i \leq 11$)操作后, 若产生的新解为可行解且支配原解, 则用新解替代原解, 否则, 不接受该新解.

3.3.2 基于改进程度的学习型评分机制

对于双目标优化问题, 当前解进行 LLH_i ($1 \leq i \leq 11$)操作后得到的可行解存在3种可能: 1) 新解所有的目标都得到改进或一个目标得到改进, 另一个目标保持不变; 2) 新解的一个目标得到改进, 另一个目标变差了; 3) 新解的两个目标都变差了. 每次使用 LLH_i 对解 π 进行局部搜索后, 采用所设计的评分机制(式(27)–(29))定量学习和积累 LLH_i 的贡献率或得分信息, 并依此采用轮盘赌方法构造新的启发式算法HLS (即局部搜索策略), 用于对下一代的各解进行局部搜索. LLH_i 初始得分为 S_{start} , 最高得分为 S_{upper} , 最低得分为 S_{lower} . $l(\pi)$ 为解 π 的行驶总里程, $otr(\pi)$ 为解 π 的时间窗准时率. 若 $otr(\pi) \neq 0$, 则 LLH_i 得分更新公式如式(27)所示, 否则用式(28)更新 LLH_i 得分.

$$S_i^j = S_i^{j-1} + \frac{l(\pi)}{l(\pi')} + \frac{otr(\pi')}{otr(\pi)} - 2, \quad (27)$$

$$S_i^j = S_i^{j-1} + \frac{l(\pi)}{l(\pi')} - 1, \quad (28)$$

其中: S_i^j 为第 LLH_i 第 j 次使用过后的得分, π 为局部搜索前的解, π' 为局部搜索后得到的可行解. 当 S_i^j 大于 S_{upper} 或小于 S_{lower} 时, 按式(29)对其进行取值.

$$S_i^j = \begin{cases} S_{upper}, & S_i^j > S_{upper}, \\ S_{lower}, & S_i^j < S_{lower}. \end{cases} \quad (29)$$

3.3.3 算法高层选择策略

为增强算法的局部搜索能力, 保证非劣解集周围解空间得到充分搜索, 需要对其进行细致的局部搜索

以保证搜索的深度; 同时, 为了保证局部搜索的广度, 需要对一部分种群做局部搜索, 充分利用种群来寻找更多的非劣解. 为了在可接受时间内平衡算法局部搜索的深度与广度, 通常对非劣解集和1/5到1/4的种群做局部搜索^[30].

当算法运行第1代, 对非劣解集和部分种群做局部搜索时, 为了在同样使用次数下全面评价各个 LLH 的搜索性能, 并且搜索出更多的非劣解供概率矩阵学习优质信息, 对非劣解集和部分种群使用所有 LLH 构成的启发式算法HLS进行局部搜索, 并在搜索过程中计算 LLH 得分.

当算法运行至第 i ($i > 1$)代, 对非劣解集和部分种群做局部搜索时, 按照 LLH 得分, 采用轮盘赌法在 LLH_1 至 LLH_{11} 中选择4种组成启发式算法HLS, 对非劣解集和种群中 $1/\theta$ 的个体做局部搜索, 并更新 LLH_i 得分. 每产生一个可行解, 就更新一次 LLH 的得分.

综上, 表3为对一个可行解做局部搜索操作的伪代码.

表3 对一个解的局部搜索策略

Table 3 Local search strategy for a solution

Algorithm 2: 局部搜索

```

输入: iteration, 每个 $LLH_i$ 的重复执行次数
repeat_times,  $\pi$ 
输出:  $\pi'$ 
1 if iteration = 1 then
2   HLS = [ $LLH_1$   $LLH_2$   $LLH_3$   $\dots$   $LLH_{11}$ ];
3   count1 = 11
4 else
5   根据得分, 使用轮盘赌法选出4个 $LLH_i$ , 组成HLS
6   HLS = [ $LLH_{i1}$   $LLH_{i2}$   $LLH_{i3}$   $LLH_{i4}$ ];
7   count1 = 4
8 end
9 repeat
10  count2 = 0;  $n = 1$ 
11  repeat
12    对 $\pi$ 执行HLS中第count1个 $LLH$ 操作后产生
    的解为 $\pi'$ ; 更新 $LLH_n$ 得分;
13    count2 = count2 + 1
14    if  $\pi'$ 支配 $\pi$  then
15       $\pi = \pi'$ 
16      if  $\pi$ 为非劣解 then
17        更新非劣解集; 用 $\pi$ 更新概率矩阵
18      end
19    end
20  until count2 = repeat_times;
21  count1 = count1 - 1
22 until count1 = 0;
  
```

本节根据局部搜索操作的范围, 将 LLH 分为2类共11种局部搜索操作(车辆内的局部搜索操作和车辆

间的局部搜索操作),采用LHHLS,用对解的改进程度来评价LLH的搜索性能,并对其评分;在算法高层选择LLH阶段,依据LLH得分使用轮盘赌法选择部分算子,在选择LLH时,轮盘赌操作使得搜索能力较强的LLH能以较大概率被选中,其他LLH也能以一定概率被选中.组合选中的部分算子产生启发式算法HLS,对可行解进行高效多样的局部搜索.

综上所述,HHEDA整体伪代码如表4所示.

表4 HHEDA伪代码
Table 4 HHEDA pseudo code

Algorithm 3: HHEDA	
输入:	算法运行截止时间time
输出:	非劣解集
1	当前迭代次数为iteration = 1
2	repeat
3	初始化种群(表2)
4	用种群中1/5的解和非劣解集组成局部搜索集local_search_set; 该解集中共有n个解
5	for $i = 1$ to n do
6	对解local_search_set[i]进行局部搜索(表3)
7	end
8	对概率矩阵执行遗忘操作(第3.2.3节); 获取当前算法运行时间t
9	until $t \geq \text{time}$;

4 实验设计与分析

4.1 实验设置

本文测试算例采用文献[1]中测试算例集的部分算例.该算例集由在VRPTW测试算例集(即著名的所罗门测试算例集)^[31]基础上生成的6种VRPPDTW测试算例集组成,是目前唯一通用的VRPPDTW测试算例集.各算法采用Python3.7编程实现,操作系统为windows10,电脑内存为16G, CPU为Intel i7-1065G7,主频1.30 GHz.

4.2 参数设置

HHEDA的关键参数为:种群的局部搜索比例 $1/\theta$ 、概率矩阵遗忘因子 α 、局部搜索策略中每种LLH重复的次数 n .为考察参数设置对算法效果的影响,本文针对中等规模的测试算例rdp101 ($N_c = 100$),采用实验设计方法(design of experiment, DOE)^[32]进行实验分析.各关键参数设置水平表如表4所示.HHEDA的其他参数设置为:popsize = 100, $S_{\text{start}} = 50$, $S_{\text{lower}} = 1$, $S_{\text{upper}} = 200$.

针对各参数的水平设置,选择规模为 L_{16} ($2^1 \times 4^2$)的正交实验,采用Ishibuchi等^[33]提出的方法对非支配解集进行评价,评价指标如式(30)所示:

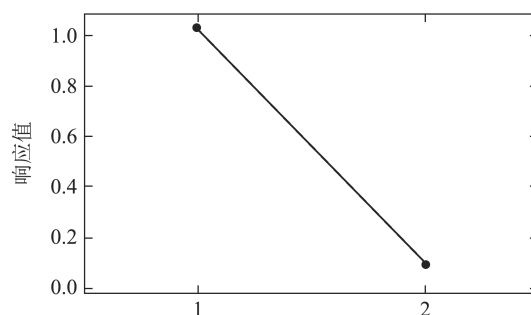
$$R_NDS(S_j) =$$

$$\frac{|S_j - \{x \in S_j | \exists y \in S : y \prec x\}|}{|S_j|}, \quad (30)$$

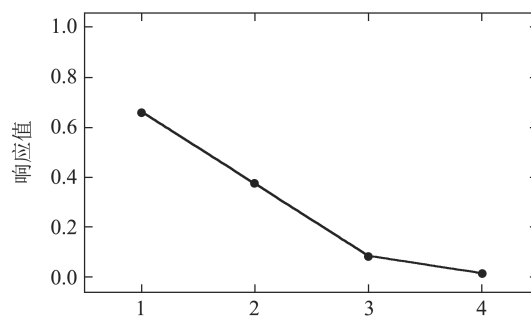
其中: S 表示在不同算法获得的非劣解集合并后的总非支配解; $y \prec x$ 表示解 x 被 y 支配; $|S_j|$ 表示第 j 种算法获得的非支配解集中解的个数.以R_NDS的平均值AVG作为平均响应值,AVG越大意味着这组参数下的算法性能越强,参数设置的正交表如表5所示,参数响应值如表6所示,各参数的响应趋势如图3所示.

表5 参数水平表
Table 5 Parameter level table

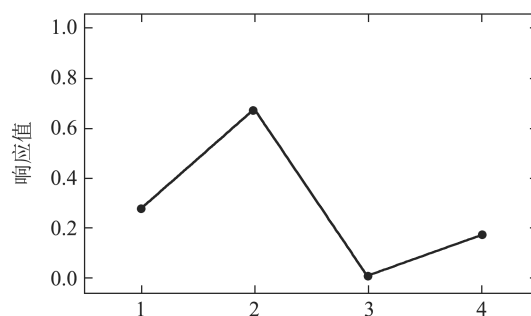
参数	水平			
	1	2	3	4
$1/\theta$	1/5	1/4	—	—
α	0.8	0.85	0.9	0.95
n	2	3	4	5



水平
(a) α 对算法性能的影响



水平
(b) $1/\theta$ 对算法性能的影响



水平
(c) n 对算法性能的影响

图3 各参数响应趋势

Fig. 3 The influence trend of each parameter

由表4–6和图3可知, $1/\theta$, α 和 n 的不同组合方式对算法性能会产生较大影响. 为了发挥算法的最佳性能, 对算法进行了参数实验. 基于实验结果, HHEDA的参数设置为: $1/\theta = 1/5$, $\alpha = 0.8$ 和 $n = 3$, 此时, 算法可表现出良好性能. 因此, 后文将基于此参数设置开展进一步的算法比较研究.

表6 正交表和AVG统计值

Table 6 Orthogonal table and AVG statistics

参数组合	参数水平			AVG
	θ	α	n	
1	1	1	1	0.276
2	1	1	2	0.298
3	2	1	3	0.003
4	2	1	4	0.084
5	1	2	1	0.0
6	1	2	2	0.368
7	2	2	3	0.001
8	2	2	4	0.003
9	2	3	1	0.0
10	2	3	2	0.003
11	1	3	3	0.003
12	1	3	4	0.079
13	2	4	1	0.003
14	2	4	2	0.006
15	1	4	3	0.0
16	1	4	4	0.003

4.3 仿真结果与比较

为评价算法性能, 采用式(30)–(31)所示的两个指

标对非支配解集评价. 第1个指标R_N如式(30)所示, 表示 S_j 中的解不被 S 中的任意解支配的解所占的比例, 衡量解的质量, 第2个指标N_N如式(31)所示, 评价的是 S_j 中的解不被 S 中的解支配的数量, 衡量算法获得非支配解的能力. 两个指标的值越大, 算法的性能越好.

$$\text{NDS_NUM}(S_j) =$$

$$|S_j - \{x \in S_j \mid \exists y \in S : y \prec x\}|. \quad (31)$$

表7 各参数平均响应值

Table 7 Average response value of each parameter

水平	参数		
	θ	α	n
1	1.027	0.661	0.279
2	0.103	0.372	0.675
3	—	0.085	0.007
4	—	0.012	0.169

4.3.1 验证LHHLS的有效性

为验证HHEDA中局部搜索LHHLS的有效性, 将其修改, 形成2种算法HHEDA_V1和HHEDA_V2, 并进行比较. HHEDA_V1仅采用算法运行第1代后11种LLH中评分最高的4个作为后续生成启发式算法HLS的备选集合, HHEDA_V2采用全部11种LLH的随机排序生成启发式算法HLS. 对于每个算例, 各算法均在相同时间($1.5 \times N_c$ s)下独立运行20次. 测试结果如表8所示, 其中最好指标值用粗体表示.

表8 LHHLS性能验证

Table 8 LHHLS performance verification

算例	客户数量	HHEDA_V1		HHEDA_V2		HHEDA	
		R_N	N_N	R_N	N_N	R_N	N_N
rcdp2501	25	0.118	1.1	0.139	1.8	0.876	9.8
rcdp2504	25	0.098	1.0	0.098	0.8	0.889	8.4
rcdp-2507	25	0.045	0.5	0.15	0.2	0.984	11.9
rcdp5001	50	0.203	3.9	0.175	4.6	0.825	21.8
rcdp5004	50	0.162	1.5	0.098	1.0	0.867	10.4
rcdp5007	50	0.444	5.8	0.121	1.6	0.750	12.8
rdp101	100	0.418	10.5	0.055	2.9	0.847	48.2
rdp201	100	0.297	7.7	0.038	2.0	0.868	43.5
cdp101	100	0.403	14.5	0.010	0.6	0.793	48.8
cdp201	100	0.381	9.2	0.110	5.2	0.86	42.2
rcdp101	100	0.506	12.5	0.032	1.6	0.84	52.3
rcdp201	100	0.517	19.4	0.018	1.0	0.733	45.3
C1.2.1	200	0.559	16.8	0.12	9.2	0.822	70.5
C2.2.1	200	0.400	18.6	0.039	3.3	0.839	72.5
R1.2.1	200	0.405	13.1	0.109	7.4	0.853	56.1
R2.2.1	200	0.605	18.1	0.046	3.9	0.837	75.1
RC1.2.1	200	0.451	18.5	0.052	4.0	0.837	66.4
RC2.2.1	200	0.608	31.7	0.028	2.7	0.789	86.3

由表8可知, HHEDA在各问题上均优于HHEDA_V1和HHEDA_V2. 这表明选择较大的备选集合可确保启发式算法HLS的多样性(对比HHEDA_V1), 同时验证了利用各LLH在搜索过程中的贡献率来动态选择LLH构造启发式算法HLS的必要性(HHEDA_V2). 因此, 对于本文VRPSPDSTW这类解空间十分复杂的问题, 采用LHHLS执行局部搜索是合理有效的.

4.3.2 验证HHEDA的有效性

为验证HHEDA的有效性, 本节将HHEDA与现有的5种有效算法进行比较, 包括ACO^[22], CRO^[20], HEDA^[29], MMOEASA^[16], MOSA-ACO^[23]. 对于每个测试算例, 所有算法均在相同时间(秒)下独立运行20次. 测试结果如表9所示, 其中最好指标值用粗体表示. 此外, 图4~9给出了各算法在部分算例上获得的非劣解集. 由于优化目标为最小化车辆行驶总里程(对应X轴)和最大化服务准时率(对应Y轴), 故帕累托前沿(即最优非劣解集)趋向于左上角(即坐标为(0, 1)的点).

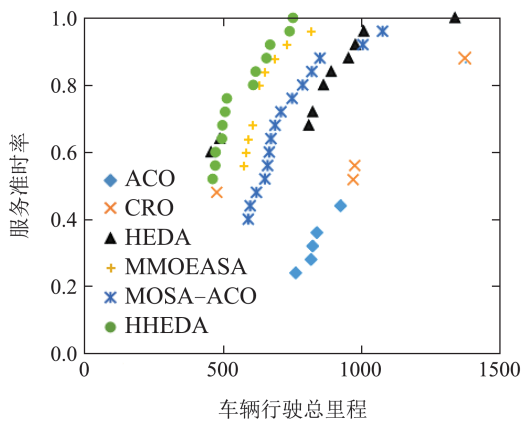


图4 算例rcdp2501非劣解集

Fig. 4 Example rcdp2501 non-inferior solution set

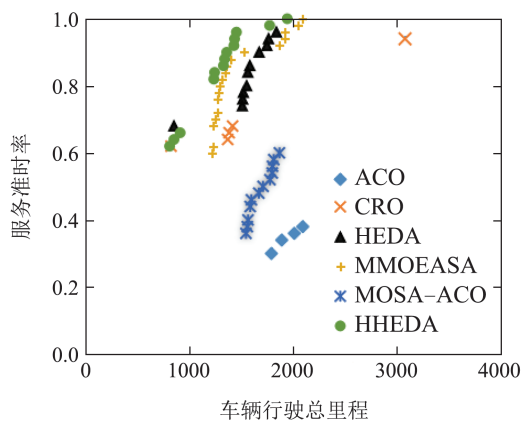


图5 算例rcdp5001非劣解集

Fig. 5 Example rcdp5001 non-inferior solution set

由表9可知, HHEDA在绝大多数问题上占优. 除算例rcdp5001之外, HHEDA获得的非劣解集中解的个数均多于其他算法, 同时HHEDA获得的非劣解集中

的解不被总非劣解集中的解支配的比例明显较高. 从图4~9可看出, HHEDA的分散性整体较好, 覆盖区域较广, 但HHEDA与HEDA, MMOEASA这3个整体性能最好的算法获得的非劣解集分布不均匀. 造成这一现象的原因在于本文问题的两个优化目标(最小化车辆行驶总里程和最大化服务准时率)间存在较明显冲突, 导致问题解空间中最优和优质非劣解的分布本身就不均匀, 从而使得性能较好算法的非劣解集中会出现“断裂”现象. 譬如, 图6~7中服务准时率约在0.3~0.4间且行驶总距离约在2000~3500间的区域为各算法均未能发现非劣解的“断裂”区域. 虽然问题解空间中最优和优质非劣解分布的不规则性增加了算法的搜索难度, 但HHEDA利用结合问题特点设计的初始解生成规则、全局搜索和局部搜索, 可实现对问题解空间更为有效的搜索, 进而能发现更多趋向于图中左上角的优质非劣解(非劣解越接近坐标为(0, 1)的左上角则其质量越好). 仍以图6~7为例, 服务准时率在0.3以下时HHEDA的非劣解基本支配了其他算法的非劣解, 同时服务准时率在0.4以上时HHEDA的非劣解个数和质量整体占优. 类似地, 从其他图中也可看出HHEDA的非劣解在总非劣解集S中所占比例最大, 覆盖区域最广.

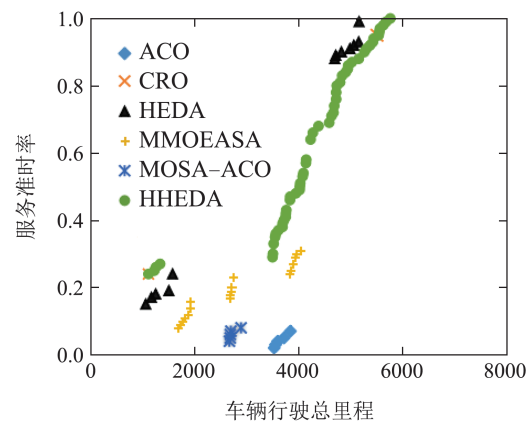


图6 算例cdp101非劣解集

Fig. 6 Example cdp101 non-inferior solution set

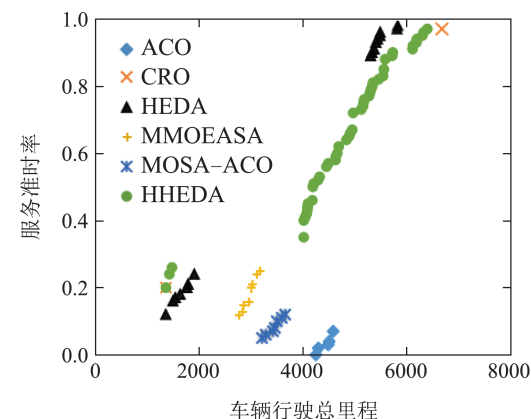


图7 算例rcdp101非劣解集

Fig. 7 Example rcdp101 non-inferior solution set

表9 算法比较结果

Table 9 Algorithm comparison results

算例	客户数	ACO		CRO		HEDA		MMOEASA		MOSA_ACO		HHEDA	
		R_N	N_N	R_N	N_N	R_N	N_N	R_N	N_N	R_N	N_N	R_N	N_N
rcdp2501	25	0.0	0.0	0.0	0.0	0.30	2.6	0.31	2.5	0.01	0.1	0.45	4.9
rcdp2504	25	0.0	0.0	0.0	0.0	0.23	1.9	0.28	2.4	0.04	0.4	0.55	5.3
rcdp2507	25	0.0	0.0	0.02	0.1	0.20	2.7	0.39	3.3	0.01	0.2	0.66	6.7
rcdp5001	50	0.0	0.0	0.02	0.1	0.67	19.7	0.15	2.1	0.0	0.1	0.38	13.1
rcdp5004	50	0.0	0.0	0.11	0.4	0.24	1.7	0.02	0.3	0.0	0.0	0.90	9.9
rcdp5007	50	0.0	0.0	0.17	0.7	0.14	2.8	0.16	2.3	0.0	0.0	0.91	15.6
cdp101	100	0.0	0.0	0.50	1.0	0.35	3.1	0.01	0.2	0.0	0.0	0.91	52.8
cdp103	100	0.0	0.0	0.45	1.0	0.45	5.2	0.09	1.2	0.0	0.0	0.88	28.0
cdp105	100	0.0	0.0	0.60	1.2	0.25	4.8	0.03	0.6	0.0	0.0	0.85	47.2
cdp201	100	0.0	0.0	0.40	0.8	0.70	8.6	0.05	0.7	0.0	0.0	0.91	51.5
cdp203	100	0.0	0.0	0.25	1.1	0.42	4.6	0.09	0.8	0.0	0.0	0.88	29.5
cdp205	100	0.0	0.0	0.52	1.1	0.72	8.2	0.09	1.3	0.0	0.0	0.90	52.0
rcdp101	100	0.0	0.0	0.55	1.1	0.61	8.7	0.12	1.3	0.0	0.0	0.67	36.3
rcdp103	100	0.0	0.0	0.50	1.0	0.46	3.5	0.0	0.0	0.0	0.0	0.91	25.3
rcdp105	100	0.0	0.0	0.65	1.3	0.35	4.8	0.02	0.3	0.0	0.0	0.80	31.6
rcdp201	100	0.0	0.0	0.35	2.7	0.69	13.9	0.23	3.7	0.0	0.0	0.66	43.1
rcdp203	100	0.0	0.0	0.32	2.5	0.48	8.1	0.27	2.9	0.0	0.0	0.58	20.4
rcdp205	100	0.0	0.0	0.46	4.1	0.26	6.2	0.15	2.4	0.0	0.0	0.75	44.2
rdp101	100	0.0	0.0	0.25	1.2	0.53	5.4	0.16	1.2	0.0	0.0	0.75	45.0
rdp103	100	0.0	0.0	0.05	0.1	1.0	5.9	0.0	0.0	0.0	0.0	0.70	23.0
rdp105	100	0.0	0.0	0.31	1.0	0.70	7.5	0.07	0.7	0.0	0.0	0.73	40.0
rdp201	100	0.0	0.0	0.40	0.8	0.70	8.6	0.05	0.7	0.0	0.0	0.91	51.5
rdp203	100	0.0	0.0	0.11	0.9	0.70	16.9	0.02	0.2	0.0	0.0	0.54	19.1
rdp205	100	0.0	0.0	0.48	1.0	0.69	10.5	0.10	1.4	0.0	0.0	0.85	48.8
S_C1_2_1	200	0.0	0.0	0.70	1.4	0.28	3.5	0.07	0.6	0.0	0.0	0.97	87.8
S_C1_2_3	200	0.0	0.0	0.50	1.0	0.39	5.2	0	0.0	0.0	0.0	0.98	47.5
S_C1_2_5	200	0.0	0.0	0.55	1.1	0.06	0.9	0.04	0.5	0.0	0.0	0.99	85.0
S_C2_2_1	200	0.0	0.0	0.02	0.2	0.79	9.1	0.23	1.4	0.0	0.0	0.67	60.2
S_C2_2_3	200	0.0	0.0	0.02	0.2	0.68	11.1	0.04	0.2	0.0	0.0	0.91	34.8
S_C2_2_5	200	0.0	0.0	0.18	1.1	0.74	12.3	0.0	0.0	0.0	0.0	0.83	71.6
S_R1_2_1	200	0.0	0.0	0.09	0.5	0.88	9.80	0.83	4.9	0.0	0.0	0.64	56.3
S_R1_2_3	200	0.0	0.0	0.50	1.0	0.78	6.80	0	0	0.0	0.0	0.76	37.6
S_R1_2_5	200	0.0	0.0	0.52	1.2	0.79	9.5	0.44	2.5	0.0	0.0	0.66	57.3
S_R2_2_1	200	0.0	0.0	0.18	1.2	0.31	5.50	0.03	0.1	0.0	0.0	0.71	70.7
S_R2_2_3	200	0.03	0.1	0.03	0.2	0.79	11.6	0.03	0.1	0.0	0.0	0.62	34.7
S_R2_2_5	200	0.0	0.0	0.18	1.2	0.32	7.0	0.15	0.8	0.0	0.0	0.68	64.3
S_RC1_2_1	200	0.0	0.0	0.70	1.4	0.72	15.2	0.0	0.0	0.0	0.0	0.81	71.4
S_RC1_2_3	200	0.0	0.0	0.60	1.2	0.34	4.2	0.0	0.0	0.0	0.0	0.94	44.4
S_RC1_2_5	200	0.0	0.0	0.60	1.2	0.22	2.7	0.16	1.5	0.0	0.0	0.98	95.4
S_RC2_2_1	200	0.0	0.0	0.04	0.3	0.94	28.1	0.03	0.1	0.0	0.0	0.68	70.8
S_RC2_2_3	200	0.0	0.0	0.01	0.1	0.89	16.1	0.37	2.5	0.0	0.0	0.68	41.5
S_RC2_2_5	200	0.0	0.0	0.03	0.2	0.62	20.1	0.16	1.3	0.0	0.0	0.74	75.3

具体到各算法的设计, ACO, CRO, HEDA, MMOEASA和MOSA_ACO侧重于简单采用算法自身进化机制执行全局搜索, 同时采用交叉、变异等邻域操作或者较为单一的客户交换操作执行局部搜索. HHEDA进一步考虑问题特点及其解空间的复杂性, 设计3种概率矩阵较全面有效地学习和积累优质解信息,

并通过对其采样生成新个体以实现全局搜索, 有助于尽快发现优质解区域, 同时在局部搜索过程中动态构造丰富的变邻域搜索(即启发式算法HLS), 有利于增加算法在优质解区域的搜索深度. 因此, HHEDA对搜索方向的引导更为合理, 能在相同时间内更有效地发现高质量的非劣解集.

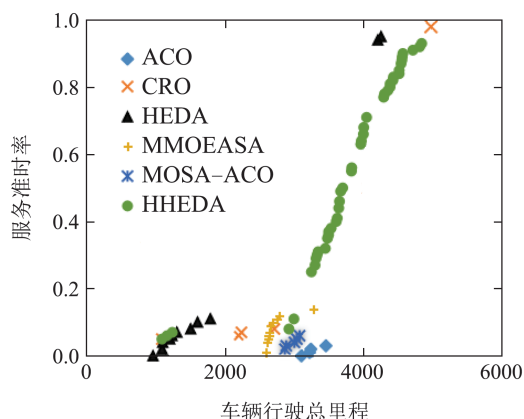


图8 算例rdp101非劣解集

Fig. 8 Example rdp101 non-inferior solution set

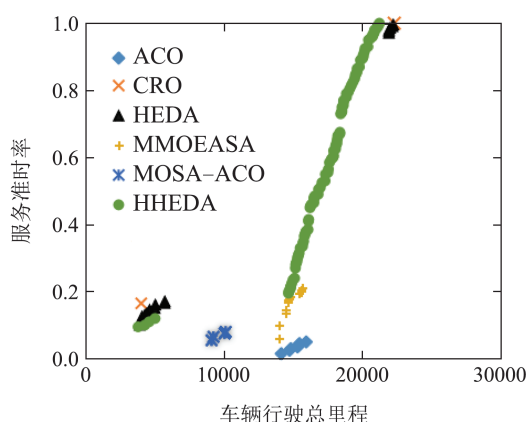


图9 算例S.C1.2.1非劣解集

Fig. 9 Example S.C1.2.1 non-inferior solution set

5 结语

本文针对带软时间窗的同时取送货车辆路径问题,在考虑车辆载重等约束条件的同时,构建了最小化车辆行驶总里程和最大化服务准时率的双目标模型,并提出一种超启发式分布估计算法 HHEDA 进行求解. HHEDA 具有如下优点: 1) 采用多种启发式规则共同生成初始解,可确保初始种群的质量和分散性; 2) 利用多个概率矩阵充分学习优质解中不同的有用信息,以推动全局搜索较快到达不同的优质解区域进行搜索; 3) 采用学习型超启发式局部搜索执行丰富的变邻域搜索,能有效增加搜索的深度,从而提升算法的性能. 仿真实验和算法比较验证了 HHEDA 是求解 VRP-SPDSTW 的有效算法. 后续研究将进一步考虑道路拥堵导致速度变化的问题,并设计有效求解算法.

参考文献:

- [1] WANG H, CHEN Y Y. A genetic algorithm for the simultaneous delivery and pickup problems with time window. *Computers & Industrial Engineering*, 2011, 62(1): 84 – 95.
- [2] MAHMOUDI M, ZHOU X S. Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: a dynamic programming approach based on state-space-time network representations. *Transportation Research Part B*, 2016, 89: 19 – 42.
- [3] VEACESLAV G, JEAN-FRANCOIS C, EMRAH D. Branch-and-price for the pickup and delivery problem with time windows and scheduled lines. *Transportation Science*, 2018, 52(5): 1191 – 1210.
- [4] ALYASIRY A M, FORBES M, BULMER M. An exact algorithm for the pickup and delivery problem with time windows and last-in-first-out loading. *Transportation Science*, 2019, 53(6): 1695 – 1705.
- [5] BETTINELLI A, CACCHIANI V, CRAINIC T G, et al. A branch-and-cut-and-price algorithm for the multi-trip separate pickup and delivery problem with time windows at customers and facilities. *European Journal of Operational Research*, 2019, 279(3): 824 – 839.
- [6] AZIEZ I, COTE J F, COELHO L C. Exact algorithms for the multi-pickup and delivery problem with time windows. *European Journal of Operational Research*, 2020, 284(3): 906 – 919.
- [7] SITEK P, WIKAREK J, RUTCZYNSKA W K, et al. Optimization of capacitated vehicle routing problem with alternative delivery, pickup and time windows: a modified hybrid approach. *Neurocomputing*, 2021, 423: 670 – 678.
- [8] LERA-ROMERO G, BRONT J J M, SOULIGNAC F J. Linear edge costs and labeling algorithms: the case of the time-dependent vehicle routing problem with time windows. *Networks*, 2020, 76(1): 24 – 53.
- [9] LAI M Y, CAO E B. An improved differential evolution algorithm for vehicle routing problem with simultaneous pickups and deliveries and time windows. *Engineering Applications of Artificial Intelligence*, 2009, 23(2): 188 – 195.
- [10] WANG Chao, MU Dong. Solving VRPSPDTW problem using simulated annealing algorithm. *Journal of System Simulation*, 2014, 26(11): 2618 – 2623.
(王超, 穆东. 基于模拟退火算法求解VRPSPDTW问题. 系统仿真学报, 2014, 26(11): 2618 – 2623.)
- [11] WANG Chao, LIU Chao, MU Dong, et al. VRPSPDTW problem solving by discrete cuckoo search. *Computer Integrated Manufacturing Systems Cims*, 2018, 24(3): 570 – 582.
(王超, 刘超, 穆东, 等. 基于离散布谷鸟算法求解带时间窗和同时取送货的车辆路径问题. 计算机集成制造系统, 2018, 24(3): 570 – 582.)
- [12] HAN Yajuan, PENG Yunfang, WEI Hang, et al. Super-heuristic genetic algorithm for vehicle routing problem with soft time window. *Computer Integrated Manufacturing System*, 2019, 25(10): 2571 – 2579.
(韩亚娟, 彭运芳, 魏航, 等. 超启发式遗传算法求解带时间窗的车辆路径问题. 计算机集成制造系统, 2019, 25(10): 2571 – 2579.)
- [13] WANG Chao, GAO Yang, LIU Chao, et al. Vehicle routing problem with simultaneous delivery and pick up problem solving by back tracking search optimization algorithm. *Computer Integrated Manufacturing Systems Cims*, 2019, 25(9): 2237 – 2247.
(王超, 高扬, 刘超, 等. 基于回溯搜索优化算法求解带时间窗和同时取送货的车辆路径问题. 计算机集成制造系统, 2019, 25(9): 2237 – 2247.)
- [14] QIN G Y, TAO F M, LI L X, et al. Optimization of the simultaneous pickup and delivery vehicle routing problem based on carbon tax. *Industrial Management & Data Systems*, 2019, 119(9): 2055 – 2071.
- [15] LI Jian, ZHANG Yong. The study on vehicle routing problem of simultaneous deliveries and pickups with time windows. *Information and Control*, 2009, 38(6): 752 – 758.
(李建, 张永. 带时间窗的同时集散货物路线问题研究. 信息与控制, 2009, 38(6): 752 – 758.)
- [16] BANOS R, ORTEGA J, Consolation Gil, et al. A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows. *Computers & Industrial Engineering*, 2013, 65(2): 286 – 296.
- [17] YANG B, HU Z, WEI C, et al. Routing with time-windows for multiple environmental vehicle types. *Computers & Industrial Engineering*, 2015, 89: 150 – 161.

- [18] IQBAL S, KAYKOBAD M, RAHMAN M S. Solving the multi-objective vehicle routing problem with soft time windows with the help of bees. *Swarm and Evolutionary Computation*, 2015, 24: 50 – 64.
- [19] ZHANG J L, ZHAO Y W, WANG H W, et al. Multi-objective cooperative QEA for low-carbon time dependent vehicle routing problem with simultaneous delivery and pickup. *International Journal of Wireless and Mobile Computing*, 2017, 12(4): 400 – 407.
- [20] LI H Y, WANG L, HEI X H, et al. A decomposition-based chemical reaction optimization for multi-objective vehicle routing problem for simultaneous delivery and pickup with time windows. *Memetic Computing*, 2018, 10(1): 103 – 120.
- [21] CHEN Xiqiong, HU Dawei, YANG Qianqian, et al. An improved ant colony algorithm for multi-objective vehicle routing problem with simultaneous pickup and delivery. *Control Theory & Applications*, 2018, 35(9): 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(9): 1347 – 1356.)
- [22] CHAI Huo, HE Ruichun, SU Jiangsheng, et al. Ant colony algorithm for solving dual-object vehicle routing problem with time window. *Transportation System Engineering and Information*, 2018, 18(4): 156 – 162.
(柴获, 何瑞春, 苏江省, 等. 求解双目标带时间窗车辆路径问题的蚁群算法. 交通运输系统工程与信息, 2018, 18(4): 156 – 162.)
- [23] WANG Y, WANG L, CHEN G C, et al. An improved ant colony optimization algorithm to the periodic vehicle routing problem with time window and service choice. *Swarm and Evolutionary Computation*, 2020, 55.
- [24] LARRANAGA P, LOZANO J A. *Estimation of Distribution Algorithms: a New Tool for Evolutionary Computation*. Boston: Kluwer Press, 2002.
- [25] YANG Haijun, LI Jianwu, LI Minqiang. *Research on the Mode, Emergence and Difficulties of Evolutionary Algorithms*. Beijing: Science Press, 2012.
(杨海军, 李建武, 李敏强. 进化算法的模式、涌现与困难性研究. 北京: 科学出版社, 2012.)
- [26] WAN S S, QIU D W. Vehicle routing optimization problem with time constraint using advanced PBIL algorithm. *Proceeding of the IEEE International Conference on Service Operations and Logistics, and Informatics*, 2008: 1394 – 1398.
- [27] MENG Xianghu, HU Rong, QIAN Bin. Effective hybrid population-based incremental learning algorithm for vehicle routing problem with time windows. *System Engineering Theory & Practice*, 2014, 34(10): 2701 – 2709.
(孟祥虎, 胡蓉, 钱斌. 求解带时间窗车辆路径问题的有效混合PBIL算法. 系统工程理论与实践, 2014, 34(10): 2701 – 2709.)
- [28] WANG J, TANG K, JOSE A L, et al. Estimation of the distribution algorithm with a stochastic local search for uncertain capacitated arc routing problems. *IEEE Transactions on Evolutionary Computation*, 2016, 20(1): 96 – 109.
- [29] JI S F, LUO R J. A hybrid estimation of distribution algorithm for multi-objective multi sourcing intermodal transportation network design problem considering carbon emissions. *Sustainability*, 2017, 9(7): 1 – 24.
- [30] QIAN B, WANG L, HUANG D X, et al. An effective hybrid DE-based algorithm for flow shop scheduling with limited buffers. *International Journal of Production Research*, 2009, 47(1): 1 – 24.
- [31] SOLOMON M M. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 1987, 35(2): 254 – 265.
- [32] MONTGOMERY D C. *Design and Analysis of Experiments*. Hoboken: John Wiley and Sons, 2005.
- [33] ISHIBUCHI H, YOSHIDA T, MURATA T. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE Transactions on Evolutionary Computation*, 2003, 7(2): 204 – 223.

作者简介:

张烜荧 硕士研究生, 目前研究方向为智能算法与优化调度, E-mail: 2575143735@qq.com;

胡蓉 副教授, 硕士生导师, 目前研究方向为智能优化调度、物流优化, E-mail: ronghu@vip.163.com;

钱斌 教授, 博士生导师, 目前研究方向为优化调度理论与方法、智能优化方法, E-mail: bin.qian@vip.163.com.