

多中心联合配送模式下集货需求随机的 VRPSDP 问题

范厚明¹ 刘鹏程¹ 刘浩¹ 侯登凯¹

摘要 针对多中心联合配送模式下集货需求随机的同时配集货车车辆路径问题 (MDVRPSDDSPJD), 构建了两阶段 MDVRPSDDSPJD 模型. 预优化阶段基于随机机会约束机制以及车载量约束为客户分配车辆, 生成预优化方案; 重优化阶段采用失败点重优化策略对服务失败点重新规划路径. 根据问题特征, 设计了自适应邻域文化基因算法 (Adaptive memetic algorithm and variable neighborhood search, AMAVNS), 针对文化基因算法易早熟、局部搜索能力弱等缺陷, 将变邻域搜索算法的深度搜索能力运用到文化基因算法的局部搜索策略中, 增强算法的局部搜索能力; 提出自适应邻域搜索次数策略和自适应劣解接受机制平衡种群进化所需的广度和深度. 通过多组算例验证了提出模型及算法的有效性. 研究成果不仅深化和拓展了 VRP (Vehicle routing problem) 相关理论研究, 也为物流企业制定车辆调度计划提供一种科学合理的方法.

关键词 车辆路径问题, 多中心, 同时配集货, 随机需求, 文化基因算法, 变邻域搜索算法

引用格式 范厚明, 刘鹏程, 刘浩, 侯登凯. 多中心联合配送模式下集货需求随机的 VRPSDP 问题. 自动化学报, 2021, 47(7): 1646–1660

DOI 10.16383/j.aas.c190519

The Multi-depot Vehicle Routing Problem With Simultaneous Deterministic Delivery and Stochastic Pickup Based on Joint Distribution

FAN Hou-Ming¹ LIU Peng-Cheng¹ LIU Hao¹ HOU Deng-Kai¹

Abstract A two-stage model is constructed to solve the multi-depot vehicle routing problem with simultaneous deterministic delivery and stochastic pickup based on joint distribution (MDVRPSDDSPJD). In pre-optimization stage, the pre-optimization scheme is generated, and the vehicles is assigned to the customers based on the stochastic chance constrained and vehicle capacity constraints. In re-optimization stage, the failure points rescheduling strategy is used to re-plan the path of the service failure points. According to the characteristics of the problem, a metaheuristic combined adaptive memetic algorithm and variable neighborhood search (AMAVNS) is designed to solve the problem. In this metaheuristic, aiming at the shortcomings of memetic algorithm such as fast convergence speed and weak local search capability, the deep search ability of the variable neighborhood search algorithm is applied to the local search strategy of the memetic algorithm to enhance the local search ability of the algorithm. The strategies include self-adaptive number of neighborhood iterations and adaptive mechanism of selective acceptance of inferior solutions are used to balance the diversification and exploitation in the algorithm iteration process. Numerical results show that the model and metaheuristic we proposed are rather effective. The research results not only deepen and expand the vehicle routing problem (VRP) theory research, but also provide a scientific and reasonable method for logistics enterprises to draw up the vehicle scheduling plan.

Key words Vehicle routing problem, multi-depot, simultaneous delivery and pickup, stochastic demand, memetic algorithm, variable neighborhood search

Citation Fan Hou-Ming, Liu Peng-Cheng, Liu Hao, Hou Deng-Kai. The multi-depot vehicle routing problem with simultaneous deterministic delivery and stochastic pickup based on joint distribution. *Acta Automatica Sinica*, 2021, 47(7): 1646–1660

车辆路径问题 (Vehicle routing problem, VRP)

收稿日期 2019-07-08 录用日期 2019-11-16

Manuscript received July 8, 2019; accepted November 16, 2019
国家自然科学基金 (61473053), 辽宁省社会科学规划基金 (L19BGL006), 辽宁省重点研发计划指导计划 (2018401002) 资助
National Natural Science Foundation of China (61473053),
Liaoning Social Science Planning Fund (L19BGL006), the Key
Research Plan Guidance Plan of Liaoning Province (2018401002)

本文责任编辑 王鼎

Recommended by Associate Editor WANG Ding

1. 大连海事大学交通运输工程学院 大连 116026

1. College of Transportation Engineering, Dalian Maritime University, Dalian 116026

是物流配送企业的核心问题, 自 Dantzig 等^[1] 在 1959 年首次提出之后, 众多学者对该问题展开了研究, 并衍生出随机需求车辆路径问题 (Vehicle routing problem with stochastic demand, VRPSD)、多中心车辆路径问题 (Multi-depot vehicle routing problem, MDVRP) 以及同时配集货车车辆路径问题 (Vehicle routing problem with simultaneous delivery and pickup, VRPSDP) 等众多 VRP 拓展问题. 本文研究的多中心联合配送模式下集货需求随机的

同时配集货车车辆路径问题 (Multi-depot vehicle routing problem with simultaneous deterministic delivery and stochastic pickup based on joint distribution, MDVRPSDDSPJD) 是综合考虑了 VRPSD、MDVRP、VRPSDP 及联合配送特点的科学问题。

近年来, 有关 MDVRP、MDVRPJD (Multi-depot vehicle routing problem based on joint distribution)、VRPSD 问题已成为 VRP 拓展问题研究的热点。葛显龙等^[2]对基于联合配送的多中心开放式动态车辆路径问题进行研究, 针对跨区域多中心联合配送的情况, 建立“多对多”的网络配送机制, 车辆配送完成后可就近返回任意配送中心, 设计云遗传算法对问题进行求解。杨翔等^[3]研究了带有模糊时间窗的 MDVRP 问题, 假设虚拟配送中心对多配送中心问题进行处理, 设计了改进的蚁群算法对问题进行求解。Gruler 等^[4]对带有随机需求的 MDVRP 问题进行研究, 设计了结合迭代局部搜索算法的模拟优化算法对问题进行求解。Alinaghian 等^[5]研究了多车厢多类型产品的 MDVRP 问题, 车辆从配送中心出发最终返回原配送中心, 在配送过程中, 针对同一种产品不允许拆分服务, 设计了结合自适应大邻域搜索算法和变邻域搜索算法的混合算法对问题进行求解。葛显龙等^[6]研究了需求可拆分模式下的集配一体化 MDVRP 问题, 设计改进遗传算法进行求解。陈呈频等^[7]对多车型 MDVRP 问题进行研究, 车辆从配送中心出发对客户进行服务后, 最终返回原配送中心, 设计多染色体遗传算法对问题进行求解。Wang 等^[8]研究了时变网络下的多中心车辆路径问题, 提出了新的交通资源共享策略, 结合节约算法、扫描算法和多目标粒子群算法设计了混合元启发式算法对问题进行求解。Li 等^[9]研究了带时间窗限制的多目标绿色 MDVRP 问题, 设计改进蚁群算法对问题进行求解。可见, 众多学者从时间窗、多车型、需求可拆分、随机需求以及配集货等方面进一步深化了对 MDVRP 问题的研究。

多中心同时配集货车车辆路径问题 (Multi-depot vehicle routing problem with simultaneous delivery and pickup, MDVRPSDP) 是集成了 MDVRP 和 VRPSDP 问题而展开的研究。该问题涉及多个配送中心的物流配送企业对同时具有配货需求和集货需求的客户点进行配集货服务。Li 等^[10]研究了闭环式 MDVRPSDP 问题, 车辆从配送中心出发, 服务完客户后返回原配送中心, 设计基于迭代局部搜索的元启发式算法对问题进行求解。Kachitvich-

yanukul 等^[11]研究了多中心多种配集货需求的车辆路径问题, 针对不同的客户配集货需求, 车辆可从任一位置出发, 且无须返回原配送中心, 设计基于粒子群的改进算法对问题进行求解。Soriano 等^[12]对两个区域多中心配集货车车辆路径问题进行研究, 要求车辆服务完成后返回原配送中心, 设计了自适应大邻域搜索算法对问题进行求解。Ben Alaia 等^[13]对闭环式多车辆多中心配集货车车辆路径问题进行研究, 采用遗传算法和粒子群算法对问题进行求解。Bouanane 等^[14]研究了带有库存限制的 MDVRPSDP 问题, 并设计了一种结合最近仓库启发式算法、扫描算法和最远插入启发式算法的混合遗传算法进行求解。

综上所述文献梳理可见, 在以下几个方面有待更深入的研究。1) 现有对于 MDVRP 问题的研究通常采用先分组后路径的方式, 将客户划分到不同的区域, 每个配送中心仅负责该区域内的客户, 将 MDVRP 问题转化为多个独立的单中心 VRP 进行求解。该方法未能考虑到多个配送中心联合配送的优势, 客户资源不能共享, 难以对整个配送网络进行全局优化。2) 未考虑多个配送中心之间车辆资源共享并设计适合其特点的车辆返回规则。在对闭环式 MDVRP 问题的研究中, 仅考虑了各配送中心对于客户资源共享, 而对于车辆资源共享方面研究不足, 导致车辆在完成配集货服务后仍需返回原配送中心, 造成路径成本的增加。在对开放式 MDVRP 问题的研究中, 车辆服务完客户后, 通常采用就近返回任意配送中心的规则。在配送中心车辆数量有限的情况下, 虽然就近返回配送中心规则会降低本次车辆调度的成本, 但未能根据配送中心的需求进行合理的车辆分配, 造成车辆的无序分配, 使得下一次规划车辆路径受到限制。即需要先进行配送中心间的车辆调度, 增加了额外的路径成本。3) 针对 MDVRPSDP 方面的研究, 所涉及的客户集货量和配货量多为确定型, 然而在许多实际配送服务中, 常存在客户配货量已知、集货量不确定的情况。例如, 在一个地区含有多个配送中心的玻璃制造公司对该地区客户进行服务, 其中配货量可以通过客户订单来确定, 而对于从客户手中收集的玻璃原材料, 客户使用过程中损坏、剩余的废旧玻璃以及玻璃残次品的回收量无法得知, 仅能根据以往的经验得知其服从的某种概率分布。以往的研究缺乏对于不确定环境下的 MDVRPSDP 的研究。

MDVRPSDDSPJD 问题是集成上述三个方面而展开的研究, 已有方法无法适用于 MDVRPS-

DDSPJD 问题的原因在于以下两方面. 1) 已有文献对于车辆返回规则的算法设计无法应用于本文提出的适合车辆共享特点的车辆返回规则. 2) 在已有研究中, 对于 VRPSDP 问题的算法设计未考虑随机需求对于车辆实时载重量的影响以及随机需求引起的对于服务失败点的二次配送. 在 VRPSD 问题的研究中, 未考虑由于客户双重需求引起的车辆初始装载量的设计. 即传统对于 VRPSD 问题的研究中, 车辆从配送中心出发时, 车辆初始装载量为满载 (配货的情况) 或者空载 (集货的情况). 然而针对 MDVRPSDDSPJD 问题的研究, 由于集货需求的随机性, 必须考虑车辆的初始装载量情况, 不能同传统 VRPSD 问题一般, 满载 (或空载) 出发对客户进行服务.

本文针对 MDVRPSDDSPJD 问题展开研究, 阐述 MDVRPSDDSPJD 的衍生进程, 并构建了两阶段 MDVRPSDDSPJD 模型. 预优化阶段基于随机机会约束机制以及车载量约束为客户分配车辆, 生成预优化方案; 重优化阶段针对服务失败的客户点采用重优化策略生成重优化路径. 根据问题特征设计自适应邻域文化基因算法对问题进行求解, 并通过三组算例对本文的模型及算法进行验证. MDVRPSDDSPJD 问题之所以能应用到含有多个配送中心的物流企业实施多中心联合配送问题中, 是由于含有多个配送中心的物流企业中有唯一的决策者或各决策者之间思想与行为高度协调一致, 且各配送中心之间不存在利益冲突或建有利益分配机制. 因此, 本文对 MDVRPSDDSPJD 进行研究不仅进一步拓展了 VRP 问题的理论研究, 也有利于提高物流配送活动的效率, 为制定科学合理的物流企业车辆调度计划提供理论依据, 具有重要的理论和实践价值.

1 问题描述及模型建立

1.1 问题提出

经典的 MDVRP 通常指在一个区域中含有多个配送中心的物流配送企业, 车辆从某一配送中心出发, 对客户进行服务后返回原配送中心. 如图 1 (a) 所示, 车辆 K_1 从配送中心 V_1 出发, 对客户 1、2、6 进行服务后返回配送中心 V_1 , 车辆 K_2 同理.

MDVRP 问题考虑的是客户仅具有一种需求, 然而, 在实际中存在客户同时具有配货和集货需求的情况, 车辆在对客户进行服务时, 需要同时满足客户的配、集货需求. 仅具有配货 (或集货) 的单一需求的情况, 可以看作是同时配集货问题的特例, 即集货 (或配货) 量为 0 的情况. 由此, 原问题便拓展成了 MDVRPSDP 问题. 由于配、集货量的加入, 车辆的装载量不再是单一的增加 (集货) 或者减少 (配货), 需要考虑在行驶过程中车辆装载量的波动情况, 车辆的实时装载量应始终小于车辆容量, 因此该问题较单一需求的问题更加复杂. 如图 1 (b) 所示, 车辆 K_1 从配送中心 V_1 出发, 对客户 1、2 进行服务后, 由于车辆剩余空间无法满足客户 6 的集货量需求, 因此车辆 K_1 返回配送中心 V_1 ; 车辆 K_2 从配送中心 V_2 出发, 经检验, 发现车辆装载量可以满足客户 4、5、3 的配、集货需求, 车辆 K_2 对客户 4、5、3 服务后, 返回配送中心 V_2 ; 派出车辆 K_3 从配送中心 V_1 出发, 对客户 6 服务完成后, 返回配送中心 V_1 .

在 MDVRPSDP 问题的基础上, 考虑到客户配货量确定、集货量为随机变量以及多中心之间联合配送的情况, 便形成了 MDVRPSDDSPJD 问题. 多中心联合配送机制打破了传统的分区域配送模式, 从全局的角度出发对客户进行联合配送服务. 此外, 由于随机变量的引入, 可能产生在某个客户

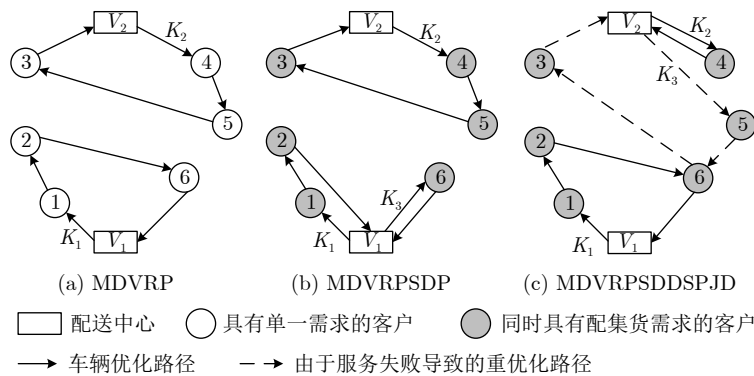


图 1 MDVRPSDDSPJD 的衍生进程
Fig.1 Derivation of MDVRPSDDSPJD

点不能满足客户的需求, 导致服务失败, 车辆需返回配送中心后重新对服务失败的客户点进行服务. 如图 1 (c) 所示, 假设车辆的预优化方案为 $V_1-1-2-6-V_1$, $V_2-4-5-3-V_2$. 车辆 K_1 从配送中心 V_1 出发, 按照预优化方案, 对客户 1、2 服务后, 在客户 6 处, 由于客户随机集货需求的影响, 经检验, 车辆剩余空间无法满足其集货需求, 车辆 K_1 返回配送中心 V_1 , 客户 6 为失败点; 车辆 K_2 从配送中心 V_2 出发, 按照预优化方案, 对客户 4 服务完成后, 经过预判断发现车辆剩余空间无法满足客户 5 的需求, 则车辆不再对客户 5 进行服务, 车辆 K_2 提前返回配送中心 V_2 ; 对于未完成服务的客户 3、5、6 重新规划路径, 由车辆 K_3 从配送中心 V_2 出发, 按照重优化路径 $V_2-5-6-3-V_2$ 对未完成服务的客户点进行服务, 服务完成后返回配送中心 V_2 .

本文综合考虑多个配送中心联合配送、客户资源共享、车辆资源统一调度、客户同时具有配集货需求以及客户集货需求的随机性, 对 MDVRPSD-DSPJD 问题进行研究, 构建两阶段求解方案. 预优化阶段, 车辆从配送中心出发按照预优化路径对客户进行服务; 重优化阶段, 对于预优化阶段服务失败的客户点进行重新优化. 车辆服务完成后, 按照车辆返回规则返回配送中心.

1.2 问题描述

本文研究的 MDVRPSDDSPJD 问题描述如下. 配送网络有完备的有向图 $G = (V, E)$; 所有节点集合 $V = V_c \cup V_d \cup \{0\}$, 客户节点集合 $V_c = \{1, 2, \dots, n\}$, 配送中心集合 $V_d = \{n+1, n+2, \dots, n+m\}$, 节点 0 为引入的虚拟配送中心编号, 虚拟配送中心同其他实际配送中心相连接且同各个实际配送中心间的距离为 0; 边集合 $E = \{(i, j) | i, j \in V\}$, 两节点 i, j 之间的路径成本为 c_{ij} ; k 为可用配送车辆集合 $K = \{1, 2, \dots, \varphi\}$ 的任一车辆, 最大车辆数为 φ , 车辆容量为 Q ; 客户节点 $i (i \in V_c)$ 的配货量为 d_i , 选用泊松分布对客户点的集货需求信息进行表征^[15], 客户点 i 的集货量为随机变量 ξ_i , 相互独立并且服从泊松分布, 客户点 i 的集货量为 q 的概率为 $P_{iq} = \Pr(\xi_i = q)$, $q = 0, 1, 2, \dots; 0 \leq q \leq Q, i \in V_c$, 期望值 $E[\xi_i]$ 和方差 $\text{Var}[\xi_i]$ 均为 λ ; 假设客户无配送时间要求. 决策变量 x_{ijk} 表示车辆 k 是否直接从点 i 到达点 j , 是为 1, 否为 0; y_{ik} 表示客户点 $i (i \in V_c)$ 是否由车辆 k 服务, 是为 1, 否为 0. 需要解决的问题是车辆从配送中心出发, 对客户进行配集货服务后, 返回配送中心, 要求规划车辆行驶路径, 使得总路径成本最低.

1.3 预优化阶段

在预优化阶段, 考虑到客户点集货需求的随机性, 本文引入随机机会约束机制对是否为客户点服务进行预判断. 假设某一条预优化路径 $V_k = \{s, t, e, w\}$, $V_k \subseteq V_c$, 为车辆服务的客户序列集合, 车辆从某一配送中心出发, 按照预优化路径进行配送服务. 当车辆对客户 s, t, e 服务完成后, 其实时车载量可表达为 $Q_{ewk} = \sum_{j \in V_c} Q_{0jk} - d_s - d_t - d_e + \xi_s + \xi_t + \xi_e$, Q_{ewk} 为车辆服务完客户 e 后而在访问客户 w 之前的车辆装载量, $\sum_{j \in V_c} Q_{0jk}$ 为车辆从配送中心出发时的初始装载量. 在车辆访问客户点 w 前, 引入随机机会约束机制对是否为客户点 w 提供服务进行检验, 预先设置风险偏好值 α , $\alpha \in (0, 1]$. 此时客户点 w 集货量与车辆服务完客户 e 后实时车载量应满足 $\xi_w \leq Q - Q_{ewk} + d_w$, 当随机约束以预设的偏好值水平 α 成立, 则可以对客户点 w 进行配集货服务. 即 $\Pr\{\xi_w \leq Q - Q_{ewk} + d_w\} \geq \alpha$, α 的设定取决于决策者在随机环境下基于风险偏好的喜好程度, α 值越大, 代表决策者的保守性越强, 更重视服务的成功率; α 值越小, 代表决策者的冒险性越强, 更希望用一辆车配送更多的客户.

在以往的闭环式 MDVRP 问题的研究中, 车辆服务完成后返回原配送中心, 虽然保证了配送中心含有的车辆数平衡, 考虑了配送中心之间的客户资源可以共享, 但忽略了车辆资源的共享, 造成路径成本的增加. 在开放式 MDVRP 问题的研究中, 车辆服务完成后可就近返回任意配送中心, 但就近返回规则存在着不合理性, 因为在对配送中心进行选址时, 其辐射范围、客户等均已在考虑在内, 并根据客户需求为其配备适量的车辆, 因此配送中心所应含有的车辆数是受到限制的. 然而车辆就近返回任意配送中心的规则会导致配送中心车辆数目的混乱, 造成配送中心对车辆的需求数与所拥有的车辆数的不平衡, 即车辆返回哪一个配送中心会影响到下一次路径优化时各配送中心含有的初始车辆数. 若仍然按照以往的车辆就近返回规则, 可能会产生额外的配送中心之间车辆调度的成本, 随着规划次数的增加, 各配送中心的车辆数将产生更大的无序性, 甚至可能出现某配送中心无车辆可用的情况. 综合考虑两种车辆返回规则的优缺点, 并基于多中心联合配送的特点, 本文提出新的车辆返回规则: 车辆从配送中心出发, 可返回任意配送中心, 但应满足每个配送中心车辆的进出平衡, 即配送中心驶离的车辆数应等于返回该配送中心的车辆数, 保证每一次路径优化完毕后, 各配送中心的车辆数不变.

1.3.1 模型建立

相应的 MDVRPSDDSPJD 模型如下:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} c_{ij} \tag{1}$$

s.t.

$$\Pr\{\xi_j \leq Q - \sum_{i \in V} x_{ijk} Q_{ijk} + d_j\} \geq \alpha, \tag{2}$$

$$\forall j \in V_c, k \in K$$

$$\sum_{i \in V} \sum_{k \in K} Q_{ipk} + \sum_{i \in V} \sum_{k \in K} x_{ipk} (\xi_p - d_p) = \sum_{j \in V} \sum_{k \in K} Q_{pjk}, \forall p \in V_c \tag{3}$$

$$0 \leq Q_{ijk} \leq Q, \forall i, j \in V, k \in K \tag{4}$$

$$\sum_{j \in V_c} Q_{0jk} = \sum_{i \in V} \sum_{j \in V_c} d_j x_{ijk}, \forall k \in K \tag{5}$$

$$\sum_{i \in V} \sum_{k \in K} x_{ijk} = 1, \forall j \in V_c \tag{6}$$

$$x_{ijk} = 0, \forall i = j, i, j \in V, k \in K \tag{7}$$

$$\sum_{i \in V} x_{ipk} = \sum_{j \in V} x_{pjk}, \forall p \in V_c, k \in K \tag{8}$$

$$\sum_{j \in V} x_{ijk} = y_{ik}, \forall i \in V_c, k \in K \tag{9}$$

$$\sum_{i \in V} x_{ijk} = y_{jk}, \forall j \in V_c, k \in K \tag{10}$$

$$\sum_{j \in V_c} x_{0jk} = \sum_{j \in V_c} x_{j0k} \leq 1, \forall k \in K \tag{11}$$

$$\sum_{j \in V} \sum_{k \in K} x_{0jk} \leq \varphi \tag{12}$$

$$\sum_{j \in V_c} \sum_{k \in K} x_{ijk} = \sum_{j \in V_c} \sum_{k \in K} x_{jik}, \forall i \in V_d \tag{13}$$

$$x_{ijk} \in \{0, 1\}, y_{ik} \in \{0, 1\}, \forall i, j \in V, k \in K \tag{14}$$

目标函数式 (1) 为最小化运输总成本; 式 (2) 为随机容量机会约束, 保证为客户 j 服务成功的概率大于等于预设的风险偏好值 α ; 式 (3) 表示车辆 k 访问客户 p 前后的车辆负载平衡约束; 式 (4) 表示车辆在访问客户 i 之后的车载量约束; 式 (5) 为车辆初始装载量; 式 (6) 表示每个客户只被一辆车服务; 式 (7) 表示相同节点之间没有路径; 式 (8) 为车辆进出平衡约束; 式 (9) 和式 (10) 保证客户点被车辆服务时一定有路径与其连接; 式 (11) 表示车辆被使用时只有一条服务路径且从虚拟配送中心出发, 并

最终回到虚拟配送中心; 式 (12) 保证了从虚拟配送中心出发的车辆数不超过总车辆数; 式 (13) 为从任意一个配送中心出发的车辆数等于返回该配送中心的车辆数; 式 (14) 为决策变量取值约束.

1.3.2 随机机会约束确定型等价处理

式 (2) 为对单个客户点的随机机会约束, 由单点随机机会约束式可以得到每条配集货路径的随机机会约束式, 如式 (15):

$$\Pr\left\{\xi_j \leq Q - \sum_{i \in V} x_{ijk} Q_{ijk} + d_j\right\} = \Pr\left\{\sum_{i \in V} \sum_{j \in V_c} x_{ijk} \xi_j \leq Q - \sum_{p \in V_c} Q_{0pk} + \sum_{i \in V} \sum_{j \in V_c} x_{ijk} d_j\right\} \geq \alpha, \tag{15}$$

$$\forall k \in K$$

即在预优化方案中, 对于任意的车辆 k ($k \in K$), 每条路径中客户点的集货量之和不超过车辆装载量的概率大于等于预设的风险偏好值 α . 将随机机会约束式 (15) 转化为确定型的等价形式, 进行预优化路径客户点的选取, 可以在保证得到理论上的最优解时降低计算量.

根据中心极限定理, 对随机机会约束机制进行确定型等价处理. 由于客户点集货量 ξ_j 为相互独立的随机变量, 由中心极限定理可知, 每一条路径的客户的集货量之和近似服从正态分布, 期望值 $M_k = \sum_{i \in V} \sum_{j \in V_c} x_{ijk} E[\xi_j]$, 标准差 $S_k = \left(\sum_{i \in V} \sum_{j \in V_c} x_{ijk}^2 \text{Var}[\xi_j]\right)^{\frac{1}{2}} = \sqrt{M_k}$.

令 $Q' = Q - \sum_{p \in V_c} Q_{0pk} + \sum_{i \in V} \sum_{j \in V_c} x_{ijk} d_j$, 则式 (15) 可转化为:

$$\Pr\left\{\frac{\sum_{i \in V} \sum_{j \in V_c} x_{ijk} \xi_j - M_k}{S_k} \leq \frac{Q' - M_k}{S_k}\right\} \geq \alpha, \forall k \in K \tag{16}$$

存在常数 τ 满足 $\tau = \Phi^{-1}(\alpha)$ 时, (τ 为 α 分位点), 式 (16) 等价于:

$$\Pr\left\{\frac{\sum_{i \in V} \sum_{j \in V_c} x_{ijk} \xi_j - M_k}{S_k} \leq \tau\right\} = \alpha, \forall k \in K \tag{17}$$

进而有:

$$M_k + \tau \cdot S_k \leq Q', \forall k \in K \tag{18}$$

将期望值和方差代入式 (18):

$$\sum_{i \in V} \sum_{j \in V_c} x_{ijk} \cdot E[\xi_j] + \tau \cdot \left(\sum_{i \in V} \sum_{j \in V_c} x_{ijk}^2 \cdot \text{Var}[\xi_j] \right)^{\frac{1}{2}} \leq Q', \quad \forall k \in K \quad (19)$$

由于集货量服从泊松分布, 期望值和方差相等且为 λ , 因而存在一个常数 \bar{Q}' , 使得式 (15) 转化如下:

$$\sum_{i \in V} \sum_{j \in V_c} x_{ijk} \cdot E[\xi_j] \leq \bar{Q}' \quad (20)$$

$$\bar{Q}' = \frac{2Q' + \tau^2 - \tau \sqrt{\tau^2 + 4Q'}}{2}, \quad \tau = \Phi^{-1}(\alpha) \quad (21)$$

$$Q' = Q - \sum_{p \in V_c} Q_{0pk} + \sum_{i \in V} \sum_{j \in V_c} x_{ijk} d_j \quad (22)$$

1.4 重优化阶段

由于客户点集货量为随机变量, 可能会导致车辆在到达某一个客户点时, 车辆实时剩余空间无法满足客户的集货需求而造成路径失败, 对于服务失败的客户点称为失败点. 由于预优化方案并不一定能完全实施, 因此需要采取重优化策略对失败点进行合理规划.

在以往对随机需求问题的研究中, 对于失败点的处理方法多为失败点返回策略、失败点前序点返回策略以及分离配集货策略三种^[16-19]. 图 2 给出了一个简单的示例, 对不同的失败点重优化策略进行分析, 实线代表预优化方案路径, 虚线代表由重优化策略得出的路径. 假设有两条预优化路径 V_1 -1-2-6- V_1 - V_2 -4-5-3- V_2 (如图 2 (a)), V_1 、 V_2 为配送中

心, 节点 i 和 j 之间的路径成本为 c_{ij} , 客户 2 和客户 5 为路径失败点.

1) 失败点返回策略 (如图 2 (b)). 当车辆在客户 2 处发生服务失败时, 车辆返回配送中心后, 再按照预优化方案继续对客户 2 及其后续客户进行服务; 客户 5 同理.

2) 失败点前序点返回策略 (如图 2 (c)). 在对客户 2 进行服务前, 对车辆剩余空间能否满足客户 2 需求的期望值 (或通过计算预选返回点得到路径方案的总体期望成本等方法) 进行预判断, 当车辆剩余空间大于等于该客户的需求期望时对其进行服务, 否则车辆返回配送中心后再按照预优化方案对客户 2 及其后续客户点进行服务; 客户 5 同理.

3) 分离配集货策略 (如图 2 (d)). 若到达客户 2 时, 车辆剩余空间无法满足客户 2 的全部集货需求, 则仅满足部分集货需求后, 继续对其后续客户点进行服务; 客户 5 同理. 最后将未完成服务的客户 2 和客户 5 看作经典的 VRP 问题进行求解.

上述三种策略中, 失败点返回策略会造成路径的往返, 导致路径成本的大量增加, 而且失败点返回策略一定劣于失败点前序点返回策略, 原因在于由三角形三边原理可知 $c_{12} + c_{20} > c_{10}$ (客户 5 同理); 失败点前序点返回策略对点选择的要求严格, 选取合适的返回点较为困难, 对客户点不当的预判断易导致多余的返回, 引起路径成本的增加; 分离配集货策略缺乏对是否服务客户的预判断, 可能产生多次服务客户的情况, 降低客户的满意度并造成路径的往返, 增加路径成本. 鉴于以上三种方案存在的不足, 本文采取文献 [20] 中所提出的路径失败点重优化策略并作出改进 (如图 2 (e)). 当车辆未能满足客户需求造成路径失败 (客户 2 处的失败原因) 或通过预判断不对客户进行服务 (客户 5 处的

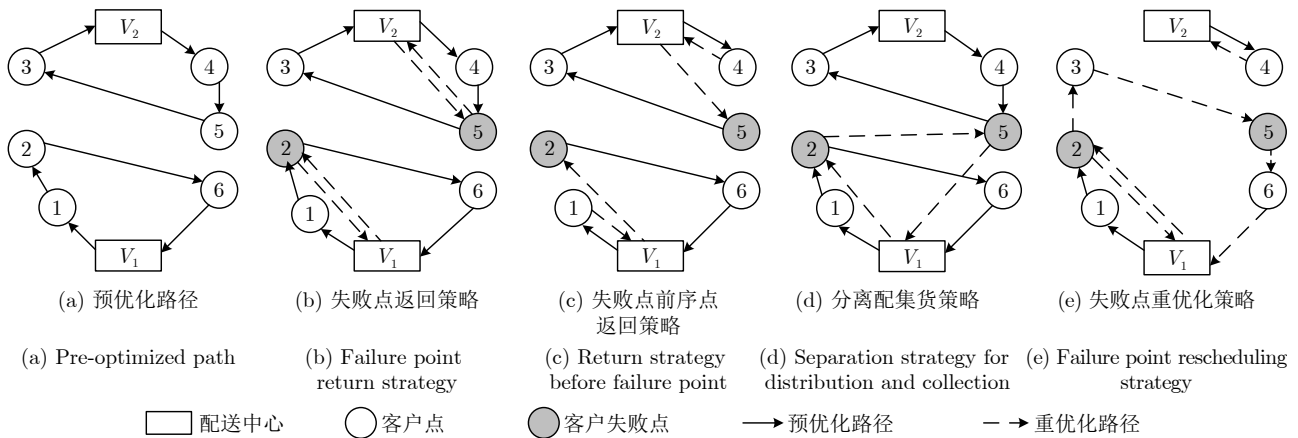


图 2 不同失败点重优化策略对比图

Fig. 2 Different failure points re-optimized strategies

失败原因) 时, 车辆返回配送中心, 统计所有路径的失败点及失败点后续客户 (客户点 2、3、5、6), 根据最近邻法求解原理, 重新设计配集货路径, 生成第二阶段重优化路径. 在重优化路径中, 客户点的集货量仍然是随机变量, 因此重优化路径仍可能出现路径失败, 对该情况执行失败点返回策略. 由于在重优化阶段, 失败点及其后续客户点的规模相对较小, 而车辆的载重量较大, 因此重优化路径中再次出现失败点的概率极小.

2 自适应邻域文化基因算法

MDVRPSDDSPJD 是一类 NP-hard (Non-deterministic polynomial-hard) 问题, 在以往对于 VRP 及其拓展问题的求解算法中, 元启发式算法具备较强的寻优能力, 能在合理的时间内给出问题的近似最优解, 广泛应用于各种 VRP 拓展问题中, 表现出了较强的竞争优势. 其中, 文化基因算法是基于文化进化理论中的隐喻机制而提出的, 具有较强的全局搜索性能和局部搜索性能. 变邻域搜索算

法采用多个不同的邻域结构进行系统搜索, 具有较强的局部搜索能力. 因此, 本文结合文化基因算法和变邻域搜索算法设计了自适应变邻域文化基因算法 (Adaptive memetic algorithm and variable neighborhood search, AMAVNS), 将变邻域搜索算法应用到文化基因算法的局部搜索策略中, 加强种群的寻优能力; 引入顺序交叉算子, 加强种群个体间的交流; 设计自适应邻域搜索次数策略和自适应劣解接受机制平衡进化所需的广度和深度, 加快算法的收敛以及加强算法跳出局部最优的能力. 本文的算法框架如图 3 所示.

2.1 编解码方式及初始种群的生成

本文算法采用整数编码的方式, 将客户的全排列存储到矩阵 *pop_route* 中, 记录客户的服务顺序. 解码方式分为两个阶段, 第一阶段通过随机机会约束机制以及车辆载重约束检验, 将客户按照初始排列顺序划分给车辆, 当对下一个客户进行检验发现当前车辆不能满足要求时, 派出新车对该客户进行服务; 第二阶段根据每辆车首尾客户到配送中心的

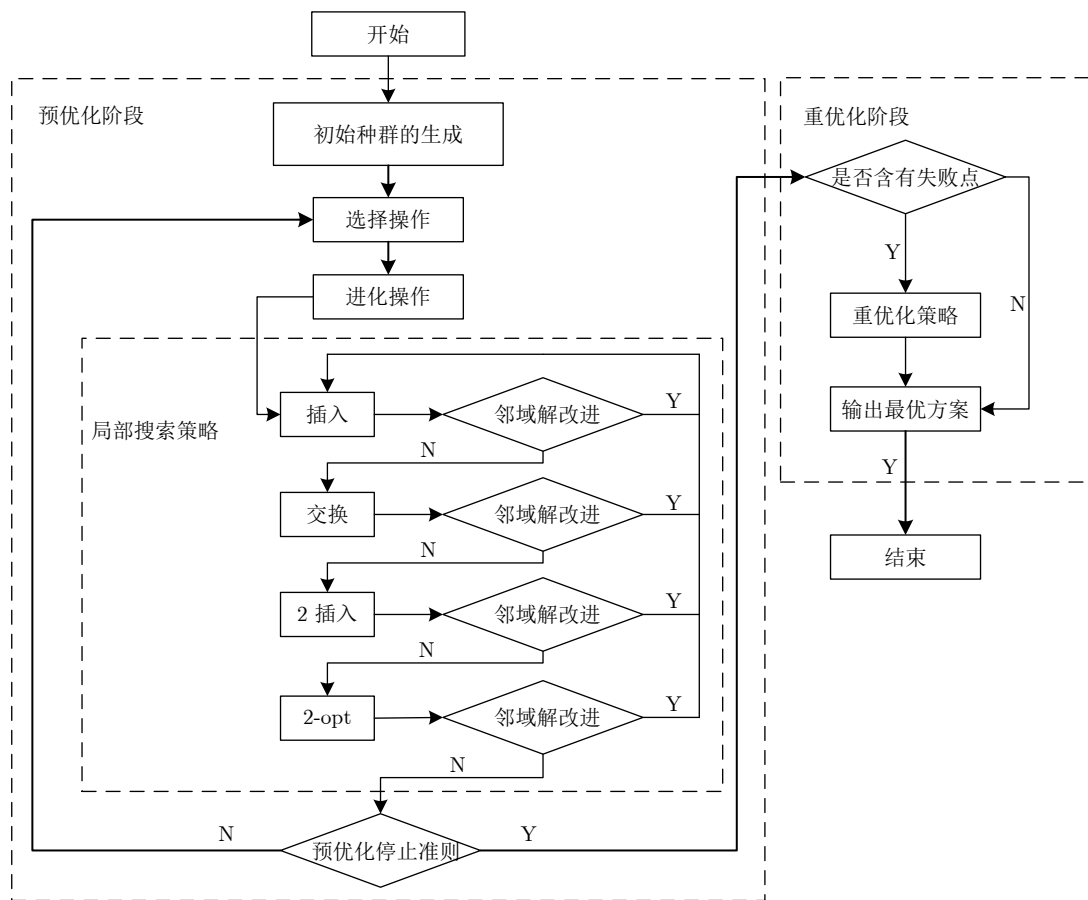


图 3 自适应邻域文化基因算法框架图

Fig.3 The basic flow of adaptive memetic algorithm and variable neighborhood search

距离以及配送中心车辆进出平衡约束确定车辆的始末配送中心. 本文给出一个简单例子来进行说明, 如图 4 所示, 假设客户矩阵 pop_route 中客户排列为 135674829, 派出第一辆车按照客户排列顺序对第一个客户进行服务, 在车辆派遣矩阵 Veh_K 中记录 $Veh_K(1) = 1$, 即第一辆车的初始服务客户为 pop_route 矩阵中的第一个客户; 按照 pop_route 中客户排列顺序对下一个客户进行随机机会约束机制以及车辆载重约束检验, 假设当对第四个客户 6 进行检验时, 不能满足检验要求, 则派出第二辆车对客户 6 进行服务, 记录 $Veh_K(2) = 4$; 以此类推, 直到最后一个客户检验完成. 最后根据每辆车首尾客户到配送中心的最小距离确定始末配送中心, 根据车辆进出平衡原则进行检查, 对于不符合车辆进出平衡约束的配送中心进行调整, 确定最终的始末配送中心. 即车辆始发的配送中心矩阵 Veh_K_sta 以及车辆返回的配送中心矩阵 Veh_K_end . 客户排列最终解码路径如图 5 所示.

pop_route	1	3	5	6	7	4	8	2	9
Veh_K	1	4	8						
Veh_K_sta	1	1	2						
Veh_K_end	1	1	2						

图 4 编码方式示意图

Fig. 4 Encoding mode diagram

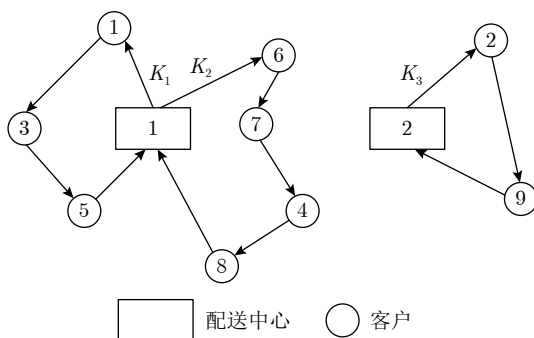


图 5 解码路径图

Fig. 5 Decoding routing diagram

可以看出, 仅需记录每辆车服务的首个客户的位置, 就可以结合 pop_route 矩阵求得该车辆服务的客户顺序, 在后续的进化操作以及局部搜索策略中也仅对编码长度固定的 pop_route 矩阵进行操作. 采用客户排列顺序、车辆服务的客户以及车辆往返的配送中心分离编码的方式, 可以使得在后续的迭代扰动时, 总能生成可行解, 避免了编码长度不固定以及对不可行解的修复困难等问题.

在生成初始种群方面, 采用随机生成和最近邻法相结合的方式, 既能够保证种群的多样性, 又能通过最近邻解提高初始种群的质量, 加快种群的收敛速度.

2.2 适应度函数

染色体的适应度函数根据目标函数式 (1) 进行构建. 显然, 目标函数值越大, 染色体的适应度值越小, 染色体 S 的适应度函数 f_S 可以表示如下:

$$f_S = \frac{1}{Z_S} \quad (23)$$

其中, Z_S 为染色体 S 的目标函数值.

2.3 选择操作

选择操作采用轮盘赌和精英保留相结合的策略. 采用轮盘赌的方式, 每个个体被选中的概率与其适应度函数值的大小成正比, 即适应度函数值越大的个体被选中的概率就越高; 反之, 被选中的概率越低. 采用精英保留策略, 将每一代的最优个体进行保留, 即用父代适应度值最优的个体替换掉子代适应度值最差的个体. 采用轮盘赌和精英保留相结合的策略, 可以在进化的过程中形成稳定的下一代, 使得算法快速收敛.

2.4 进化操作

本文的进化操作选用顺序交叉算子. 如图 6 所示, 对个体 $pop1$ 进行顺序交叉时, 从种群中随机选取个体 $pop2$, 分别随机产生点位 i_{11} 、 i_{12} , i_{21} 、 i_{22} . $pop1$ 随机点位 i_{11} 、 i_{12} 之间的部分作为新个体 new_pop1 的第一段; new_pop1 的后续点位与 $pop2$ 有关, 即先消除 $pop2$ 中包含 $pop1$ 随机点位 i_{11} 、 i_{12} 之间的客户点, 在消除过程中, $pop2$ 中客户点的位置顺序不改变, 再将消除后的客户点排列作为 new_pop1 的第二段, 此时 new_pop1 个体生成完毕. new_pop2 同理.

2.5 局部搜索策略

文化基因算法的局部搜索策略是算法的核心功能之一, 决定着算法的局部搜索能力, 本文将变邻域搜索算法应用到局部搜索策略中, 加强对种群的深度搜索. 设定 l 个邻域结构集合 $N_k = \{N_1, N_2, \dots, N_l\}$, 对种群中个体 x 从第一个邻域结构 N_1 开始扰动. 若在预设的邻域搜索次数 S_n 内未找到改进解, 则执行下一个邻域结构. 若在某一邻域结构内获得改进解 x' , 则令 $x = x'$, 并返回第一个邻域结构重新开始迭代, 直到循环到最后一个邻域

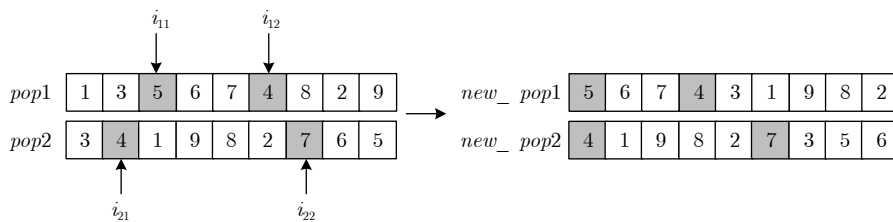


图 6 顺序交叉算子示意图

Fig.6 The diagram of ordered crossover operator

结构仍未找到改进解时, 搜索终止; 或当变邻域搜索循环次数达到预设值 Max_Sn 时, 搜索终止, 算法进入下一阶段. 本文采用如下 4 种邻域结构, 增强算法的局部搜索能力.

1) 插入: 随机选择一个客户点 i , 将其从原方案中移除, 随机插入到某个客户点 j 的后面. 如图 7 (a) 所示, 客户 3 插入到客户 6 的后面.

2) 交换: 随机选中两个客户点 i 和 j , 交换两客户点的位置. 如图 7 (b) 所示, 客户 3 和客户 6 交换位置.

3) 2-插入: 在原方案中随机选择连续的两个客户点, 将其插入到随机选择的客户点 j 的后面. 如图 7 (c) 所示, 客户 3、4 插入到客户 6 后面.

4) 2-opt: 随机选择两客户点 i 和 j , 并将客户点间的顺序进行交换. 如图 7 (d) 所示, 保持客户 3 的位置不变, 客户 4、5、7、6 逆序.

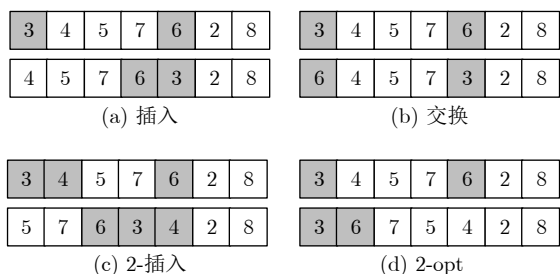


图 7 邻域结构示意图

Fig.7 Neighborhood structures

2.6 自适应机制

本文设计自适应邻域搜索次数策略和自适应劣解接受机制平衡进化所需的广度和深度, 使算法跳出局部最优.

1) 在算法迭代的不同时期, 种群所需的扰动强度不同. 在迭代初期, 采用较低的邻域搜索次数, 加快种群的收敛; 随着最优解连续未改变次数的增大, 增加邻域搜索次数, 加强算法的搜索能力, 并结合自适应劣解接受机制使算法跳出局部最优. 本文的自适应邻域搜索次数策略如下.

a) 设置初始邻域搜索次数为 $Sn = 1$, 最优解连续未改变的次数为 Con_num ;

b) 若本次迭代后的种群最优解未改进, 则令 $Con_num = Con_num + 1, Sn = Sn + 1$; 若迭代后的最优解存在改进, 则令 $Con_num = 0, Sn = 1$;

c) 当最优解连续未改变的次数 Con_num 达到预设值 $Stop_num$ 时, 算法终止, 输出最优解.

2) 设计自适应劣解接受机制对是否接受扰动后的解进行判断, 本文的自适应劣解接受机制如下.

a) 设置阈值参数 $t, t = 1 + Con_num/gen$, gen 为迭代的代数;

b) 当扰动产生的新解 x' 满足 $f(x') \leq tf(x)$, 则令 $x = x'$.

设置最小可接受劣解的代数 Min_gen , 即算法的前 Min_gen 代扰动操作禁止接受劣解, 当算法迭代到中后期时, 算法逐步陷入局部最优, 劣解接受机制被激活. 随着最优解连续未改变的次数 Con_num 的增加, t 值逐渐增大, 可接受劣解的范围也逐渐增大, 增大种群的多样性, 使得算法跳出局部最优.

3 算例验证及结果分析

本文选取了三类算例验证本文算法的有效性, 包括 MDVRP 算例、VRPSDP 算例和 MDVRPSD-DSPJD 算例, 由于 MDVRP 和 VRPSDP 是本文研究的基础问题, 因此首先选取 MDVRP 算例和 VRPSDP 算例对本文算法进行验证. 本文算法编程工具采用 MATLAB R2017b, 操作系统为 Windows10, 电脑内存为 8 GB, CPU 为 Intel i7-7700, 主频 3.60 GHz. 经过反复测试, 本文算法参数设置如下: 最大迭代次数 $Max_gen = 800$, 种群规模 $Pop_size = 30 \sim 150$, 初始变邻域搜索次数 $Sn = 1$, 最大邻域循环次数 $Max_Sn = 1000$, 最优解连续未改变的最大次数 $Stop_num = 20 \sim 50$, 最小可接受劣解的代数 $Min_gen = 30 \sim 100$. 参数的设置值同对应算例中客户点规模 n 相关, 当 $n \leq 30$ 时, Pop_size 设置为 30, $Stop_num$ 设置为 20, 最小可

接受劣解的代数 $Min_gen = 30$; 当 $30 < n \leq 75$ 时, Pop_size 设置为 $80 \sim 100$, $Stop_num$ 设置为 30 , 最小可接受劣解的代数 $Min_gen = 80 \sim 100$; 当 $75 < n \leq 100$ 时, Pop_size 设置为 150 , $Stop_num$ 设置为 50 , 最小可接受劣解的代数 $Min_gen = 100$.

3.1 MDVRP 算例分析

实验 1. 选取文献 [21] 中提出的 MDVRP 算例, 客户规模为 30 , 含有 3 个配送中心 (A、B、C) 且每个配送中心含有 4 辆车, 目前已有文献对该算例求得的最优解为 116.01 . 表 1 给出聚类分层法^[22], 狼群算法^[23], 文献 [24] 中提出的禁忌搜索算法、量子遗传算法、云量子遗传算法以及本文的自适应邻域文化基因算法求解 10 次的计算结果. 其中文献 [22] 中聚类分层法采用分区域配送的方式, 文献 [23–24] 以及本文算法均采用联合配送的方式进行配送. BKS 为目前已有文献求得的最优解, $Best$ 、 $Worst$ 、 Avg 分别为算法运行 10 次所求得的最优解、最劣解和平均解, $\%Dev$ 为算法求得的解同目前已知最优解的偏差, CPU 为算法运行时间.

由表 1 可以看出, 在求解质量方面, 文献 [22] 通过分区域配送方式将多中心问题转化为单中心问题进行求解, 这种求解思想从本质上来看是局部优化的思想, 求解效果最差. 本文算法对现有最优解进行了改进, 求得最优解为 113.62 , 较原有文献求得的最优解改进了 2.06% ; 在求解最劣解以及平均解方面, 也均优于其他算法. 在计算时间方面, 本文算法运行时间仅为 4.78 s, 相较于其他算法, 具有较强的竞争优势. 表 2 给出本文求解的车辆行驶路径. 图 8 给出了最短路径变化趋势图, 可以看出本文算法可以在较少的迭代次数内快速收敛, 当算法陷入局部最优解时, 可快速跳出局部最优, 具有较好的寻优性能.

实验 2. 选择由标准算例库提供的 MDVRP 算例集对本文算法进行验证 (算例集来源: <http://neo.lcc.uma.es/vrp/vrp-instances/multiple-depot-vrp-instances/>). 表 3 给出了贪婪随机自适应搜索

算法 (GRASP/VND)^[25]、遗传算法 (GA)^[26]、迭代局部搜索算法 (ILS)^[27] 以及本文的 AMAVNS 算法运行 10 次的最优解. BKS 为算例库中给出的已知最优解, $Best$ 为算法运行 10 次所求得的最优解, $\%Dev$ 为算法求得的最优解同已知最优解的偏差, CPU 为算法平均运行时间.

由表 3 可以看出, 在对 7 个标准算例的求解中, GRASP/VND 算法、GA 算法以及 ILS 算法均未找到最优解. GRASP/VND 算法求解质量最差, 求解最大偏差为 11.85% , 平均偏差 4.81% ; GA 算法求解质量一般, 最大偏差 9.05% , 平均偏差 3.12% ; ILS 算法求解最大偏差 6.09% , 平均偏差 4.51% . 本文 AMAVNS 算法求解速度略差于其他三种算法, 但求解质量最优, 求得其中两个最优解, 最大偏差 3.80% , 平均偏差 1.76% . 验证了本文算法的有效性.

3.2 VRPSDP 算例分析

实验 3. 选择由 Dethloff^[28] 提出的客户规模为 50 的测试算例, 算例根据两种不同的地理情况分为 SCA (Scattered) 和 CON (Concentrated) 两大类. 表 4 给出了禁忌搜索算法 (TS)^[29]、并行节约算法 (EPSA3)^[30]、节约蚁群算法 (SavAnt)^[31]、分散搜索算法 (SS)^[32] 以及本文的 AMAVNS 算法在 SCA8 和 CON8 的 20 个算例中运行 10 次的最优解. $Best$ 表示算法求解该算例得到的最优值, $\%Dev$ 表示每个算例中各算法的最优解同 5 个算法中最优解的偏差, CPU 为运行时间. 图 9 给出了本文算法在算例 SCA8-0、SCA8-1 的求解路径图.

由表 4 可以看出, 在 SCA8、CON8 共 20 个算例中, TS 算法求得了其中 6 个最优解, 最大偏差 2.65% , 平均偏差 0.83% , 算法耗时较短; EPSA3 算法求解运行时间最短, 但求解质量最差, 仅求得其中 1 个最优解, 最大偏差 5.57% , 平均偏差 2.52% ; SavAnt 算法求解时间最长, 求得 4 个最优解, 最大偏差 2.69% , 平均偏差 0.61% ; SS 算法求得其中 6 个最优解, 最大偏差 2.65% , 平均偏差 0.83% ; 本文 AMAVNS 算法求解速度良好, 求解质量优于

表 1 算法性能比较
Table 1 Performance comparison of different types of metaheuristics

算法	BKS	$Best$	$\%Dev$ (%)	$Worst$	$\%Dev$ (%)	Avg	$\%Dev$ (%)	CPU (s)
聚类分层法 ^[22]		123.33	6.31					
狼群算法 ^[23]		122.42	5.53					
禁忌搜索算法 ^[24]	116.01	116.01	0.00	343.31	195.93	187.34	61.49	243.00
量子遗传算法 ^[24]		116.01	0.00	326.48	181.42	176.37	52.03	216.00
云量子遗传算法 ^[24]		116.01	0.00	162.38	39.97	156.32	34.75	168.00
自适应邻域文化基因算法		113.62	-2.06	115.39	-0.53	114.33	-1.45	4.78

表 2 本文算法求解路径
Table 2 The algorithm solution path of this paper

算法	车辆行驶路径	总路程
本文	A-22-30-14-10-7-4-A	113.62
	B-11-29-28-13-8-15-1-B	
	C-16-25-5-12-26-18-3-6-C	
	C-17-21-19-20-23-24-2-9-27-C	

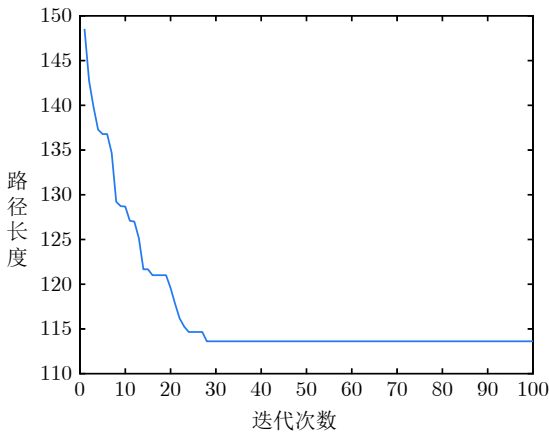


图 8 最短路径变化趋势图

Fig. 8 The iterative trend of optimal solution

其他 4 种算法, 求得其中 13 个最优解, 最大偏差 0.83 %, 平均偏差 0.12 %, 在未取得最优解的算例中, 偏差均保持在 1 % 以内. 再次验证了本文算法的有效性及其算法的稳定性能.

3.3 MDVRPSDDSPJD 算例分析

实验 4. 由于现有研究没有关于 MDVRPSD-DSPJD 问题的标准算例, 通过对标准 MDVRP 算例 p01 进行修改以适应本文的模型. p01 算例含有 4 个配送中心 (配送中心编号 51 ~ 54), 每个配送中心含有 4 辆同质车辆, 车辆容量为 80, 客户规模为 50 (客户编号 1 ~ 50). 客户的配、集货量由 Salhi 等^[33]提出的需求分离规则产生: x_i 和 y_i 是客户的坐标,

q_i 为客户 i 的原始需求, 计算每个客户的比率 $r_i = \min(x_i/y_i; y_i/x_i)$, 由 $d_i = r_i q_i$ 和 $\xi_i = q_i d_i$ 计算得出该客户的配、集货量. 在对客户进行配送路径优化时, 客户的集货量是未知的, 仅能知道客户需求服从 $\lambda = 6.34$ 的泊松分布. 车辆从配送中心出发, 服务完客户后返回配送中心. 目标是规划车辆行驶路径, 使得总路径成本最低. 表 5 给出传统的车辆返回原配送中心规则 (规则 1) 和本文提出的车辆返回规则 (规则 2) 在不同风险偏好值决策下分别求解 20 次的计算结果. $Best$ 为该规则下求解的最优解, Avg 为算法运行 20 次的平均解, $\%Dev$ 为规则 2 在最优解和平均解方面相较于规则 1 的偏差, CPU 为算法运行时间. 表 6 给出算法所求最优解对应的车辆行驶路径.

由表 5 可以看出, 在不同风险偏好值决策下, 求解结果具有明显的区别, 规则 1 在 $\alpha = 0.7$ 时求得最优解 533.85, 平均值为 624.97; 规则 2 在 $\alpha = 0.7$ 时求得最优解 499.96, 平均值为 610.54. 当风险偏好值 α 增大时, 决策者偏向于保守型, 希望以较大的概率成功服务客户. 对于服务成功概率较小的客户点, 选择不去配送, 提前返回配送中心, 导致在重优化阶段需要为其重新规划路径, 造成了路径成本的增加. 当风险偏好值 α 减小时, 决策者偏向于冒险型, 希望一辆车能够服务更多的客户. 然而由于随机因素的影响, 导致车辆剩余空间无法满足客户的集货需求, 造成服务客户失败, 需要折返配送中心后, 重新为其服务. 因此, 冒险型决策的路径失败造成的额外路径成本要大于保守型决策的额外路径成本. 基于此, 本文建议决策者风险偏好值 α 取 0.7 左右, 既能防止由于 α 值偏大, 一辆车服务客户数较少的弊端, 又能避免由于 α 值过小, 频繁折返配送中心, 造成额外成本的增加.

此外, 通过规则 1 和规则 2 的对比, 可以看出, 本文提出的车辆返回规则较传统的车辆返回原配送中心规则在最优解方面改进了 6.35 %, 使用车辆数

表 3 MDVRP 算例结果比较
Table 3 The results comparison of MDVRP instances

算例	客户规模	BKS	GRASP/VND ^[25]		GA ^[26]		ILS ^[27]		AMAVNS		
			Best	%Dev (%)	Best	%Dev (%)	Best	%Dev (%)	Best	%Dev (%)	CPU (s)
p01	50	576.87	592.21	2.66	598.45	3.74	606.11	5.07	576.87	0.00	32.60
p02	50	473.53	529.64	11.85	478.65	1.08	496.45	4.84	473.53	0.00	39.22
p03	75	641.19	648.68	1.17	699.23	9.05	675.32	5.32	646.33	0.80	84.06
p04	100	1 001.59	1 055.26	5.36	1 011.36	0.98	1 062.60	6.09	1 039.69	3.80	155.23
p05	100	750.03	769.37	2.58	—	—	782.34	4.31	765.98	2.13	143.73
p06	100	876.50	924.68	5.50	882.88	0.72	910.13	3.84	902.27	2.94	164.82
p07	100	885.80	925.80	4.52	—	—	904.44	2.10	909.22	2.64	159.01

表 4 VRPSDP 算例结果比较
Table 4 The results comparison of VRPSDP instances

算例	TS ^[29]			EPSA3 ^[30]			SavAnt ^[31]			SS ^[32]			AMAVNS		
	Best	%Dev (%)	CPU (s)	Best	%Dev (%)	CPU (s)	Best	%Dev (%)	CPU (s)	Best	%Dev (%)	Best	%Dev (%)	CPU (s)	
SCA8-0	981.47	2.07	4.14	1 015.06	5.57	—	961.6	0.01	—	981.17	2.05	961.50	0.00	35.31	
SCA8-11	077.44	2.65	4.27	1 098.91	4.69	—	1 063.0	1.27	—	1 077.44	2.65	1 049.65	0.00	42.05	
SCA8-21	050.98	1.00	4.20	1 064.54	2.30	—	1 040.6	0.00	—	1 050.98	1.00	1 049.22	0.83	36.63	
SCA8-3	983.34	0.00	4.17	1 021.61	3.89	—	985.9	0.26	—	983.34	0.00	983.34	0.00	31.59	
SCA8-41	073.46	0.55	4.13	1 114.50	4.40	—	1 071.0	0.32	—	1 073.46	0.55	1 065.49	0.00	42.87	
SCA8-51	047.24	1.68	4.02	1 060.43	2.96	—	1 054.3	2.36	—	1 047.24	1.68	1 029.95	0.00	38.19	
SCA8-6	995.59	2.37	3.85	1 004.66	3.31	—	972.5	0.00	—	995.59	2.37	973.27	0.08	34.88	
SCA8-71	068.56	0.84	4.22	1 080.17	1.93	—	1 059.7	0.00	—	1 068.56	0.84	1 063.22	0.33	29.01	
SCA8-81	080.58	0.88	3.85	1 098.02	2.51	—	1 082.7	1.08	—	1 080.58	0.88	1 071.18	0.00	33.54	
SCA8-91	084.80	1.32	4.20	1 123.55	4.94	—	1 081.4	1.00	—	1 084.80	1.32	1 070.71	0.00	46.03	
CON8-0	860.48	0.39	4.19	857.17	0.00	—	858.9	0.20	—	860.48	0.39	857.40	0.03	37.86	
CON8-1	740.85	0.00	3.89	742.41	0.21	—	740.9	0.01	—	740.85	0.00	740.85	0.00	40.75	
CON8-2	723.32	1.38	3.76	715.17	0.24	—	714.3	0.12	—	723.32	1.38	713.44	0.00	35.99	
CON8-3	811.23	0.00	4.12	815.59	0.54	—	812.3	0.13	—	811.23	0.00	811.23	0.00	39.37	
CON8-4	772.25	0.28	3.75	789.13	2.47	—	770.1	0.00	—	772.25	0.28	772.25	0.28	41.22	
CON8-5	756.91	0.23	3.99	759.09	0.52	—	766.6	1.51	—	756.91	0.23	755.16	0.00	40.84	
CON8-6	678.92	0.00	4.04	707.83	4.26	—	697.2	2.69	—	678.92	0.00	684.11	0.76	53.95	
CON8-7	814.50	0.00	4.00	834.64	2.47	—	814.8	0.04	—	814.50	0.00	814.77	0.03	49.99	
CON8-8	775.59	1.05	3.74	787.43	2.59	—	-771.3	0.49	—	775.59	1.05	767.53	0.00	33.40	
CON8-9	809.00	0.00	4.13	813.84	0.60	—	815.1	0.75	—	809.00	0.00	809.00	0.00	34.21	
平均值	909.33	0.83	4.03	925.19	2.52	2.98	906.71	0.61	441	909.31	0.83	902.27	0.12	39.38	

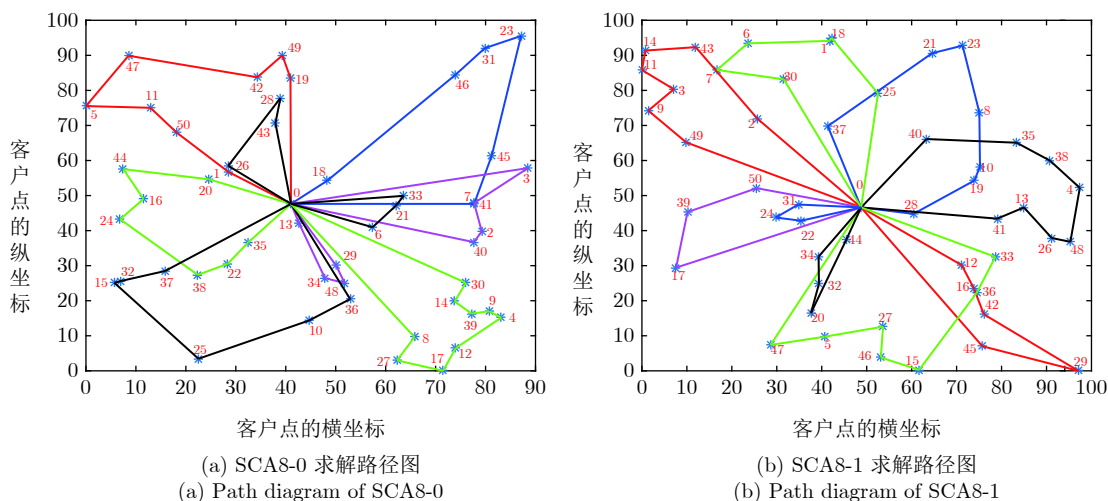


图 9 实验 3 部分算例的求解路径图

Fig.9 The specific path diagrams of some cases of experiment 3

减少了一辆, 在总的求解平均值方面改进了 4.45 %。这是由于传统的车辆返回原配送中心规则仅考虑了配送中心之间客户资源的共享, 缺乏对于车辆资源的统一调配, 导致车辆返回配送中心受到限制, 造成路径成本的增加。实验结果表明, 本文提出的车

辆返回规则能够有效地降低路径成本, 为多个配送中心之间车辆联合调度提供参考。

4 结论

本文针对多中心联合配送模式下集货需求随机

表 5 MDVRPSDDSPJD 算例结果比较
Table 5 The results comparison of MDVRPSDDSPJD

α	规则1			规则2				
	<i>Best</i>	<i>Avg</i>	<i>CPU</i> (s)	<i>Best</i>	<i>%Dev</i> (%)	<i>Avg</i>	<i>%Dev</i> (%)	<i>CPU</i> (s)
0.1	558.98	732.61	98.85	548.86	-1.81	674.40	-7.95	101.57
0.2	565.26	726.67	92.75	549.51	-2.79	673.54	-7.31	110.48
0.3	545.30	713.16	97.43	542.30	-0.55	676.67	-5.12	108.43
0.4	568.42	713.75	108.26	550.14	-3.22	678.61	-4.92	102.49
0.5	549.96	709.65	105.43	543.59	-1.16	650.79	-8.29	109.47
0.6	548.04	651.97	105.67	537.83	-1.86	640.07	-1.83	114.64
0.7	533.85	624.97	102.20	499.96	-6.35	610.54	-2.31	101.35
0.8	545.30	668.99	110.46	537.83	-1.37	641.70	-4.08	148.57
0.9	546.83	657.50	90.26	537.44	-1.72	653.35	-0.63	96.84
1.0	564.64	655.98	95.33	563.24	-0.25	650.85	-0.78	110.28
平均值	552.66	685.53	100.66	541.07	-2.10	655.05	-4.45	110.41

表 6 MDVRPSDDSPJD 算例求解最优解
Table 6 The best solutions of MDVRPSDDSPJD

	车辆行驶路径	车辆数	总路程
	51-4-18-13-41-40-19-42-51		
	53-38-5-37-17-44-15-45-33-39-10-49-53		
	53-30-34-21-16-50-9-53		
规则 1	54-20-3-36-35-54	8	533.85
	52-11-32-27-48-23-7-43-24-52		
	54-29-2-1-8-26-31-28-22-54		
	52-47-12-46-52		
	52-6-14-25-52		
	54-20-2-16-21-29-54		
	54-35-36-3-28-31-26-8-22-54		
	53-38-5-37-17-44-15-45-33-39-10-49-53		
规则 2	52-46-11-32-1-27-48-23-7-43-24-6-52	7	499.96
	51-42-19-40-41-13-25-14-52		
	52-12-47-18-4-51		
	53-30-34-50-9-53		

的同时配集货车路径问题进行了研究, 主要结论如下。

1) 本文提出的 MDVRPSDDSPJD 问题不仅考虑了多中心联合配送、客户及车辆资源共享, 而且考虑到客户同时具有配集货需求且集货需求为随机变量的情况, 是对 MDVRP、VRPSDP 等研究的进一步深化和拓展。

2) 基于多中心联合配送的特点, 提出车辆返回规则。即车辆从配送中心出发, 可返回任意配送中心, 但需保持每个配送中心车辆进出平衡。经验证表明, 该规则较传统的车辆返回规则有明显的改进, 能够有效降低企业的路径成本。

3) 设计的自适应邻域文化基因算法, 将变邻域搜索算法引入文化基因算法的局部搜索策略中, 提高了算法的寻优性能; 设计的自适应邻域搜索次数策略和自适应劣解接受机制能够平衡算法进化所需的广度和深度, 提高了算法跳出局部最优的能力。

4) 通过多组算例对本文的模型及算法进行验证, 结果表明, 所建的模型及设计的算法能有效解决 MDVRPSDDSPJD 问题, 同其他类型的算法进行比较, 本文算法寻优能力较强、搜索稳定, 具有较强的优势。

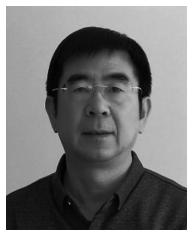
本文的研究能为具有多个配送中心的物流企业实施多中心联合配送提供较好的解决方案。但需要指出的是, 物流企业可能面临多重不确定环境且车辆调度会受到客户时间窗因素的影响, 如何在时间窗约束下刻画和解决多重不确定环境下的车辆调度问题是未来需要继续研究的方向。

References

- Dantzig G B, Ramser J H. The truck dispatching problem. *Management Science*, 1959, **6**(1): 80-91
- Ge Xian-Long, Wang Xu, Deng Lei. Research on open and dynamic vehicle routing problems based on joint distribution. *Journal of Industrial Engineering and Engineering Management*, 2013, **27**(3): 60-68
(葛显龙, 王旭, 邓蕾. 基于联合配送的开放式动态车辆路径问题及算法研究. 管理工程学报, 2013, **27**(3): 60-68)
- Yang Xiang, Fan Hou-Ming, Zhang Xiao-Nan, Li Yang. Optimization of multi-deport open vehicle routing problem with fuzzy time window. *Computer Integrated Manufacturing Systems*, 2016, **22**(7): 1768-1778
(杨翔, 范厚明, 张晓楠, 李阳. 基于模糊时间窗的多中心开放式车辆路径问题. 计算机集成制造系统, 2016, **22**(7): 1768-1778)
- Gruher A, Fikar C, Juan A A, Hirsch P, Contreras-Bolton C.

- Supporting multi-depot and stochastic waste collection management in clustered urban areas via simulation-optimization. *Journal of Simulation*, 2017, **11**(1): 11–19
- 5 Alinaghian M, Shokouhi N. Multi-depot multi-compartment vehicle routing problem, solved by a hybrid adaptive large neighborhood search. *Omega*, 2018, **76**(3): 85–99
- 6 Ge Xian-Long, Zou Deng-Bo. Vehicle routing problem of multi-distribution centers with cross-docking in the supply chain. *Control and Decision*, 2018, **33**(12): 2169–2176
(葛显龙, 邹登波. 供应链环境下带越库配送的多配送中心车辆路径问题. 控制与决策, 2018, **33**(12): 2169–2176)
- 7 Chen Cheng-Pin, Han Sheng-Jun, Lu Jian-Sha, Chen Qing-Feng, Wang Cheng. A multi-chromosome genetic algorithm for multi-depot and multi-type vehicle routing problems. *China Mechanical Engineering*, 2018, **29**(2): 218–223
(陈呈频, 韩胜军, 鲁建厦, 陈青丰, 王成. 多车场与多车型车辆路径问题的多染色体遗传算法. 中国机械工程, 2018, **29**(2): 218–223)
- 8 Wang Y, Assogba K, Xu M Z, Liu Y, Wang H Z. Multi-depot green vehicle routing problem with shared transportation resource: Integration of time-dependent speed and piecewise penalty cost. *Journal of Cleaner Production*, 2019, **232**(27): 12–29
- 9 Li Y B, Soleimani H, Zohal M. An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *Journal of Cleaner Production*, 2019, **227**(22): 1161–1172
- 10 Li J, Pardalos P M, Sun H, Pei J, Zhang Y. Iterated local search embedded adaptive neighborhood selection approach for the multi-depot vehicle routing problem with simultaneous deliveries and pickups. *Expert Systems With Applications an International Journal*, 2015, **42**(7): 3551–3561
- 11 Kachitvichyanukul V, Sombuntham P, Kunnapadeelert S. Two solution representations for solving multi-depot vehicle routing problem with multiple delivery and pickup requests via PSO. *Computers & Industrial Engineering*, 2015, **89**(11): 125–136
- 12 Soriano A, Gansterer M, Hartl R F. The two-region multi-depot delivery and pickup problem. *OR Spectrum*, 2018, **40**(4): 1077–1108
- 13 Ben Alaia E, Harbaoui I, Borne P, Bouchriha H. A comparative study of the PSO and GA for the m-MDPDPTW. *International Journal of Computers Communications & Control*, 2018, **13**(1): 8–23
- 14 Bouanane K, Benadada Y, Bencheikh G. Multi-depots vehicle routing problem with simultaneous delivery and pickup and inventory restrictions: Formulation and resolution. *International Journal of Advanced Computer Science and Applications*, 2019, **10**(2): 110–120
- 15 Gauvin C, Desaulniers G, Gendreau M. A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 2014, **50**(10): 141–153
- 16 Yang W H, Mathur K, Ballou R H. Stochastic vehicle routing problem with restocking. *Transportation Science*, 2000, **34**(1): 99–112
- 17 Xie Bing-Lei, An Shi, Guo Yao-Huang. Multi-tour optimization policy for stochastic vehicle routing problem. *Systems Engineering-theory & Practice*, 2007, **27**(2): 167–171
(谢秉磊, 安实, 郭耀煌. 随机车辆路径问题的多回路优化策略. 系统工程理论与实践, 2007, **27**(2): 167–171)
- 18 Zhao Yan-Wei, Li Chuan, Zhang Jing-Ling, Lu You, Wang Wan-Liang. Novel algorithm for multi-objective vehicle routing problem with stochastic demand. *Computer Integrated Manufacturing Systems*, 2012, **18**(3): 523–530
(赵燕伟, 李川, 张景玲, 陆游, 王万良. 一种新的求解多目标随机需求车辆路径问题的算法. 计算机集成制造系统, 2012, **18**(3): 523–530)
- 19 Hu Z H, Sheu J B, Zhao L, Lu C C. A dynamic closed-loop vehicle routing problem with uncertainty and incompatible goods. *Transportation Research Part C*, 2015, **55**(6): 273–297
- 20 Li Yang, Fan Hou-Ming, Zhang Xiao-Nan, Yang Xiang. Two-phase variable neighborhood scatter search for the capacitated vehicle routing problem with stochastic demand. *Control Theory & Applications*, 2017, **34**(12): 1594–1604
(李阳, 范厚明, 张晓楠, 杨翔. 随机需求车辆路径问题及混合变邻域分散搜索算法求解. 控制理论与应用, 2017, **34**(12): 1594–1604)
- 21 Lang Mao-Xiang. Study on the model and algorithm for multi-depot vehicle scheduling problem. *Journal of Transportation Systems Engineering and Information Technology*, 2006, **6**(5): 65–69
(郎茂祥. 多配送中心车辆调度问题的模型与算法研究. 交通运输系统工程与信息, 2006, **6**(5): 65–69)
- 22 Yin Zhi, Ye Chun-Ming. Study on the hierarchical model for multi-depot logistic vehicle scheduling problem. *Journal of Systems & Management*, 2014, **6**(4): 602–606
(殷脂, 叶春明. 多配送中心物流配送车辆调度问题的分层算法模型. 系统管理学报, 2014, **6**(4): 602–606)
- 23 Ye Yong, Zhang Hui-Zhen. Wolf pack algorithm for multi-depot vehicle routing problem. *Application Research of Computers*, 2017, **34**(9): 36–39
(叶勇, 张惠珍. 多配送中心车辆路径问题的狼群算法. 计算机应用研究, 2017, **34**(9): 36–39)
- 24 Ge Xian-Long, Xu Mao-Zeng, Wang Wei-Xin. Route optimization of urban logistics in joint distribution. *Control and Decision*, 2016, **31**(3): 503–512
(葛显龙, 许茂增, 王伟鑫. 基于联合配送的城市物流配送路径优化. 控制与决策, 2016, **31**(3): 503–512)
- 25 Villegas J G, Prins C, Prodhon C, Medaglia A. GRASP/VND and multi-start evolutionary local search for the single truck and trailer routing problem with satellite depots. *Engineering Applications of Artificial Intelligence*, 2010, **23**(5): 780–794
- 26 Karakatic S, Podgorelec V. A survey of genetic algorithms for solving multi depot vehicle routing problem. *Applied Soft Computing Journal*, 2015, **27**(C): 519–532
- 27 Tlili T, Krichen S, Drira G, Faiz S. On solving the multi-depot vehicle routing problem. In: Proceedings of the 3rd International Conference on Advanced Computing, Networking and Informatics. New Delhi: Springer, 2016. 103–108
- 28 Dethloff J. Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum*, 2001, **23**(1): 79–96

- 29 Montané F A T, Galvao R D. A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service. *Science Technology & Engineering*, 2006, **33**(3): 595–619
- 30 Gajpal Y, Abad P. Saving-based algorithms for vehicle routing problem with simultaneous pickup and delivery. *Journal of the Operational Research Society*, 2010, **61**(10): 1498–1509
- 31 Catay B. A new saving-based ant algorithm for the vehicle routing problem with simultaneous delivery and pickup. *Expert Systems With Applications*, 2010, **37**(10): 6809–6817
- 32 Maquera G, Laguna M, Dan A G, Sant'Anna A P. Scatter search applied to the vehicle routing problem with simultaneous delivery and pickup. *International Journal of Applied Metaheuristic Computing*, 2011, **2**(2): 1–20
- 33 Salli S, Nagy G. A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling. *Journal of the Operational Research Society*, 1999, **50**(10): 1034–1042



范厚明 大连海事大学交通运输工程学院教授. 主要研究方向为交通运输系统规划与设计, 战略管理与系统规划. 本文通信作者.

E-mail: fhm468@163.com

(FAN Hou-Ming Professor at the College of Transportation Engineering, Dalian Maritime University. His research interest covers transportation system planning and design, strategic management and system planning. Corresponding author of this paper.)



刘鹏程 大连海事大学交通运输工程学院硕士研究生. 主要研究方向为交通运输规划与管理.

E-mail: lpc0369@163.com

(LIU Peng-Cheng Master student at the College of Transportation Engineering, Dalian Maritime University. His main research interest is transportation planning and management.)



刘浩 大连海事大学交通运输工程学院硕士研究生. 主要研究方向为物流工程与管理.

E-mail: lhao66@126.com

(LIU Hao Master student at the College of Transportation Engineering, Dalian Maritime University. His main research interest is logistics engineering and management.)



侯登凯 大连海事大学交通运输工程学院博士研究生. 主要研究方向为交通运输规划与管理.

E-mail: houdk@dlnusp.com

(HOU Deng-Kai Ph. D. candidate at the College of Transportation Engineering, Dalian Maritime University. His main research interest is transportation planning and management.)