

11-30-2023

Multi-depot Half-open Vehicle Routing Problem with Simultaneous Delivery-pickup and Time Windows

Yingyu Zhang

School of Business Administration, Henan Polytechnic University, Jiaozuo 454003, China,
934985982@qq.com

Liyun Wu

School of Business Administration, Henan Polytechnic University, Jiaozuo 454003, China,
jitwly@hpu.edu.cn

Shengtai Jia

School of Energy Science and Engineering, Henan Polytechnic University, Jiaozuo 454003, China

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation. For more information, please contact xtfzxb@126.com.

Multi-depot Half-open Vehicle Routing Problem with Simultaneous Delivery-pickup and Time Windows

Abstract

Abstract: To solve the multi-depot half-open vehicle routing problem with simultaneous delivery-pickup and time windows, this paper builds a mathematical model of a multi-depot half-open vehicle routing problem with simultaneous delivery-pickup and time windows by balancing the vehicle in and out of the distribution center and minimizing vehicle delivery distance as the goal. According to the characteristics of the problem, a brain storm algorithm based on chaotic mutation is designed to solve this problem, and the sequential crossover strategy is adopted to increase the population diversity. Meanwhile, the algorithm selects two chaotic maps for chaotic mutation operation, which employs the diversity, ergodicity, and randomness of chaotic mutation to enhance the overall search capability of the algorithm. Multiple numerical example comparison not only verifies the effectiveness and stability of the proposed algorithm for solving various vehicle routing problems but also indicates the distribution mode of multi-depot half-open simultaneous delivery-pickup and time windows is superior to that of multi-depot simultaneous delivery-pickup and time windows. The research results expand the vehicle routing problem and provide a decision-making reference for related logistics enterprises.

Keywords

vehicle routing problem, multi-depot, simultaneous delivery-pickup, time windows, brain storm optimization algorithm based on chaotic mutation

Recommended Citation

Zhang Yingyu, Wu Liyun, Jia Shengtai. Multi-depot Half-open Vehicle Routing Problem with Simultaneous Delivery-pickup and Time Windows[J]. Journal of System Simulation, 2023, 35(11): 2464-2475.

带时间窗的多中心半开放式 VRPSDP 问题研究

张颖钰¹, 吴立云^{1*}, 贾胜钰²

(1. 河南理工大学 工商管理学院, 河南 焦作 454003; 2. 河南理工大学 能源科学与工程学院, 河南 焦作 454003)

摘要: 针对带时间窗的多中心半开放式同时送取货车车辆路径问题, 构建了配送中心车辆进出平衡且以车辆配送距离最小化为目标的带时间窗的多中心半开放式同时送取货车车辆路径问题的数学模型。设计了混沌变异头脑风暴算法求解该问题, 采用顺序交叉策略增加种群多样性, 设置 2 种混沌映射进行混沌变异操作, 利用混沌变异的多样性、遍历性和随机性, 增强算法全局搜索能力。通过多组算例对比, 不仅验证所提算法求解多种车辆路径问题的有效性与稳定性, 还验证了带时间窗下的多中心半开放同时送取货配送模式优于多中心闭合式同时送取货配送模式。研究成果不仅拓展了车辆路径类的模型, 还为相关物流企业提供一种决策参考。

关键词: 车辆路径问题; 多中心; 同时送取货; 时间窗; 混沌变异头脑风暴算法

中图分类号: TP391.9; F252 文献标志码: A 文章编号: 1004-731X(2023)11-2464-12

DOI: 10.16182/j.issn1004731x.joss.22-0727

引用格式: 张颖钰, 吴立云, 贾胜钰. 带时间窗的多中心半开放式 VRPSDP 问题研究[J]. 系统仿真学报, 2023, 35(11): 2464-2475.

Reference format: Zhang Yingyu, Wu Liyun, Jia Shengtai. Multi-depot Half-open Vehicle Routing Problem with Simultaneous Delivery-pickup and Time Windows[J]. Journal of System Simulation, 2023, 35(11): 2464-2475.

Multi-depot Half-open Vehicle Routing Problem with Simultaneous Delivery-pickup and Time Windows

Zhang Yingyu¹, Wu Liyun^{1*}, Jia Shengtai²

(1. School of Business Administration, Henan Polytechnic University, Jiaozuo 454003, China;

2. School of Energy Science and Engineering, Henan Polytechnic University, Jiaozuo 454003, China)

Abstract: To solve the multi-depot half-open vehicle routing problem with simultaneous delivery-pickup and time windows, *this paper builds a mathematical model of a multi-depot half-open vehicle routing problem with simultaneous delivery-pickup and time windows* by balancing the vehicle in and out of the distribution center and minimizing vehicle delivery distance as the goal. According to the characteristics of the problem, *a brain storm algorithm based on chaotic mutation is designed to solve this problem*, and the sequential crossover strategy is adopted to increase the population diversity. Meanwhile, the algorithm selects two chaotic maps for chaotic mutation operation, which employs the diversity, ergodicity, and randomness of chaotic mutation to enhance the overall search capability of the algorithm. Multiple numerical example comparison not only verifies the effectiveness and stability of the proposed algorithm for solving various vehicle routing problems but also indicates the distribution mode of multi-depot half-open simultaneous delivery-pickup and time windows is superior to that of multi-depot simultaneous delivery-pickup and time windows. The research results expand the vehicle routing problem and provide a decision-making reference for related logistics enterprises.

收稿日期: 2022-06-24 修回日期: 2022-08-22

基金项目: 国家自然科学基金(51874121); NSFC-河南联合基金重点项目(U1904210); 河南省高校基本科研业务费专项资金(NSFRF180104)

第一作者: 张颖钰(1997-), 女, 硕士生, 研究方向为现代物流与供应链管理。E-mail: 934985982@qq.com

通讯作者: 吴立云(1973-), 女, 教授, 博士, 研究方向为现代物流与供应链管理、质量与安全管理等。E-mail: jitwly@hpu.edu.cn

Keywords: vehicle routing problem; multi-depot; simultaneous delivery-pickup; time windows; brain storm optimization algorithm based on chaotic mutation

0 引言

近年来电商联盟发展迅速,人们对物流的服务要求逐渐提高,针对电商退换货的逆向物流也随之备受关注。因此国内外众多学者对车辆路径问题展开广泛研究,衍生出了多中心半开放式车辆路径问题(multi-depot half open vehicle routing problem, MDHOVRP)以及同时送取货车辆路径问题(vehicle routing problem with simultaneous delivery and pickup, VRPSDP)等众多VRP的拓展问题。为了提升服务质量和客户满意度,物流企业不仅要满足客户需求,还要考虑客户的服务时间窗。因此,本文研究了带时间窗的多中心半开放式同时送取货车辆路径问题(multi-depot half open vehicle routing problem with simultaneous delivery-pickup and time windows, MDHOVRPSDPTW)。该问题描述的配送网络包括多个配送中心和客户点,每个客户点同时具有送取货需求,所有配送车辆从配送中心出发对客户点执行送取货服务,且必须在客户点要求的访问时间内完成服务。相对于闭合式的配送中心,半开放式配送中心允许车辆完成配送任务后在保持配送中心车辆均衡的原则下可就近返回任意配送中心。MDHOVRPSDPTW问题更加符合现实物流配送的复杂情况,如在一个区域内含有多家配送公司负责多个小区的快递配送及揽收服务,通过配送中心与车辆之间的共享,缩短配送距离、减少用车数量,进而降低配送成本。MDHOVRPSDPTW问题是由带时间窗的同时送取货车辆路径问题(vehicle routing problem with simultaneous pickup-delivery and time windows, VRPSDPTW)和MDHOVRP结合而来。目前,针对该问题的研究涉及较少。但其包含的VRPSDPTW, MDHOVRP等问题仍是当前的研究热点。

关于VRPSDPTW, MDHOVRP国内外学者都

对其模型和求解方法进行了广泛和深入的研究。VRPSDPTW问题由Angelelli等提出,针对该问题设计精确算法进行算例规模为20的小型算例求解^[1]。但是精确算法会随着客户规模增大导致求解效率变低,故更多学者采用启发式算法对其进行求解。文献[2]对于VRPSDPTW问题采用协同进化遗传算法求解,并在经典算例Solomon的基础上设计了测试算例。文献[3]设计离散布谷鸟搜索算法求解VRPSDPTW问题,并与文献[2]做对比验证且更新了5个国际已知最优解。于次年又设计回溯搜索优化算法有效求解VRPSDPTW问题^[4]。文献[5]为解决VRPSDPTW问题,提出基于模拟退火与自适应大规模邻域搜索结合的混合算法,对56个大规模算例进行验证。关于MDHOVRP问题的研究,文献[6]提出一种自适应局部搜索的混合遗传算法来求解带时间窗的多中心半开放式车辆路径问题(multi-depot half open vehicle routing problem with time windows, MDHOVRPTW)。文献[7]针对MDHOVRPTW问题建立混合整数规划模型,提出一种基于3种有效局部搜索的构造型启发式算法并求得满意解。文献[8]将粒子群算法与遗传算法结合求解MDHOVRP问题。文献[9]基于生鲜品的物流配送提出MDHOVRPTW配送模式,构建多目标优化模型并设计蚁群算法求解该问题。文献[10]将多中心联合配送与电动车辆结合,构建带时间窗的多中心半开放式纯电动车辆路径优化模型,并改进蚁群算法对其进行求解。文献[11]针对MDHOVRPTW问题,设计了一种三阶段求解算法,并改进多蚁群算法求解该问题。文献[12]考虑软时间窗约束和车辆速度变化情况,针对MDHOVRP问题,设计了改进粒子群算法和变邻域搜索算法的两阶段求解算法。

综上所述,国内外文献主要集中在对

VRPSDPTW 问题及 MDHOVRP 问题分开的研究, 将 VRPSDPTW 与 MDHOVRP 问题结合考虑的文献较少。此外在模型构建方面, 多中心车辆路径文献中车辆大多数按照就近原则返回配送中心, 容易造成配送中心车辆不均衡, 甚至导致配送中心无车可用。本文在模型构建中加入车辆进出数量守恒约束, 即车辆返回配送中心时应先满足配送中心车辆进出平衡, 再实行就近原则。由于 MDHOVRPSDPTW 问题约束条件复杂, 使求解难度进一步增加, 为了提升求解效率和解的质量, 本文设计了一种针对 MDHOVRPSDPTW 问题特点的混沌变异头脑风暴算法, 采用 k-means 聚类操作, 按照适应度值将相似的个体聚成 k 类, 并引入顺序交叉策略, 增强个体之间交流, 增加种群的多样性。当算法陷入局部最优时, 随机选取混沌映射, 利用混沌变异的多样性、遍历性和随机性生成新的解集合, 平衡搜索时的收敛和发散操作, 进而提高算法跳出局部最优的能力。

1 MDHOVRPSDPTW 数学模型

1.1 问题描述

本文研究的 MDHOVRPSDPTW 问题可以描述如下: 在配送网络中存在多个配送中心和客户点, 客户点同时具有送取需求, 同一类型的车辆从任意配送中心以负载状态出发, 对客户点执行送取任务, 在一次服务中完成客户点的所有需求, 车辆在配送路径中均满足自身载重约束、客户点与配送中心服务时间窗要求, 当车辆完成配送任务后在保证配送中心车辆均衡的情况下就近返回任意配送中心。目标是找到一条满足所有约束条件的配送路径, 并使配送距离最小化。

模型假设如下:

- (1) 配送中心、客户点的地理信息已知;
- (2) 配送中心、客户点的服务时间窗信息已知;
- (3) 客户点的送取需求已知;

- (4) 同一类型车辆的最大装载已知;
- (5) 车辆对客户点无服务顺序要求;
- (6) 车辆匀速行驶。

1.2 模型构建

以配送距离最小化为优化目标, 建立如下 MDHOVRPSDPTW 模型。

最小化配送距离:

$$Z_{\min} = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij} x_{ijk} \quad (1)$$

式中: c_{ij} 为节点 i 到节点 j 的距离, $i, j \in V$; x_{ijk} 为车辆 k 是否从节点 i 到节点 j , 如果是, 则 $x_{ijk} = 1$, 否则, $x_{ijk} = 0$ 。

客户点需求不能被拆分且只能被一辆车服务:

s.t.

$$\sum_{i \in V} \sum_{k \in K} x_{ijk} = 1, \forall j \in N \quad (2)$$

式中: N 为客户点集合, $N \in \{1, 2, \dots, n\}$ 。

车辆在路径上货物进出平衡:

$$\sum_{j \in V} x_{ijk} = 1, \forall k \in K \quad (3)$$

式中: k 为车辆标号 (车辆型号统一), $k \in \{1, 2, \dots, K\}$; K 为车辆总数。

$$\sum_{i \in V} x_{ijk} = \sum_{i \in V} x_{jik}, \forall j \in N, \forall k \in K \quad (4)$$

车辆 k 从客户 i 到客户 j 的行驶时间等于客户 i 到客户 j 之间的距离与车辆行驶速度比值:

$$t_{ij} = \frac{c_{ij}}{G}, \forall i, j \in V \quad (5)$$

式中: c_{ij} 为节点 i 到节点 j 的距离, $i, j \in V$; G 为车辆行驶速度; V 为所有节点集合, $V = N \cup M$ 。

车辆行驶时间的连续性:

$$w_{ik} + s_i + t_{ij} - w_{jk} \leq B(1 - x_{ijk}) \quad (6)$$

式中: w_{ik} 为车辆 k 对节点 i 开始服务时间, $i \in V$; s_i 为客户 i 的服务时间, $i \in N$; t_{ij} 为从节点 i 到节点 j 的行驶时间, $i, j \in V$; B 为足够大的正数。

保证车辆满足客户点和配送中心的时间窗要求:

$$a_i \left(\sum_{j \in V} x_{ijk} \right) \leq w_{ik} \leq b_i \left(\sum_{j \in V} x_{jik} \right), \forall i \in V, \forall k \in K \quad (7)$$

式中: a_i 为节点 i 的左时间窗, $i \in V$; b 为节点 i 的右时间窗, $i \in V$ 。

车辆初始装载:

$$L_{0k} = \sum_{i \in N} d_i \sum_{j \in N} x_{ijk}, \forall k \in K \quad (8)$$

式中: d_i 为客户 i 的送货需求, $i \in N$ 。

车辆完成任意客户点后的车辆载重及完成后不超过自身载重:

$$L_j \geq L_i - d_i + p_j - B \left(1 - \sum_{k \in K} x_{ijk} \right), \forall i \in N, \forall j \in N \quad (9)$$

式中: L_j 为车辆对客户 j 服务后的装载量, $j \in N$; p_j 为客户 i 的取货求量, $j \in N$ 。

$$L_j \leq Q + B \left(1 - \sum_{i \in V} x_{ijk} \right), \forall j \in N, \forall k \in K \quad (10)$$

式中: Q 为车辆最大载重;

车辆从任意配送中心出发且仅离开一次, 并最终返回任意配送中心:

$$\sum_{i \in M} \sum_{j \in N} x_{ijk} = \sum_{i \in N} \sum_{j \in M} x_{ijk} = 1, \forall k \in K \quad (11)$$

从配送中心出发的车辆数等于返回该配送中心的车辆数:

$$\sum_{j \in N} \sum_{k \in K} x_{ijk} = \sum_{j \in N} \sum_{k \in K} x_{jik}, \forall i \in M \quad (12)$$

决策变量取值:

$$x_{ijk} \in \{0, 1\}, \forall i, j \in V, \forall k \in K \quad (13)$$

2 混沌变异头脑风暴算法

MDHOVRPSDPTW 问题是 NP-Hard 问题, 其多中心、半开放式以及时间窗等特点增加了求解难度。为了求解该问题, 本文将头脑风暴优化 (brain storm optimization, BSO)^[13] 算法和混沌变异策略^[14] 相结合, 设计了混沌变异头脑风暴 (brain storm optimization based on chaotic mutation, CMBSO) 算法。传统头脑风暴算法是模拟人类求解问题集思广益的行为, 由量变到质变的过程, 其基本流程包含解码、聚类及迭代搜索。本文设计的 CMBSO 算法采用非 0 自然数编码, 两段式解码, 提高可行解的产生。在迭代搜索中采用顺序

交叉策略与混沌变异策略, 不仅扩大了算法的搜索空间, 还增强了算法全局搜索能力, 旨在获得高质量的满意解。

2.1 编码与解码

本文采用非 0 自然数编码, 将客户点随机全排列, 所有客户点储存在 C_Num 中, 1~8 为客户点, 记录服务顺序如图 1 所示。

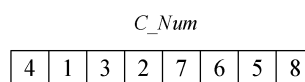


图 1 编码示意图
Fig. 1 Encoding diagram

解码过程分为 2 个阶段。第 1 阶段, 车辆依照从左到右顺序依次服务客户点, 并进行载重、时间窗约束检验。若满足约束, 则对客户点分配路径; 否则, 结束分配, 由新派车辆进行服务。第 2 阶段, 对每辆车服务首尾客户点随机分配配送中心, 并进行车辆进出平衡约束检验, 满足约束再按照就近原则, 确定每辆车的始末配送中心。

举例对图 1 进行求解。假设车辆 k_1 在满足载重和时间窗约束下服务客户点 4、客户点 1, 当对客户点 3 服务时不满足载重约束, 则退出路径分配, 由车辆 k_2 继续服务客户点 3、客户点 2、客户点 7 和客户点 6, 车辆 k_2 不满足客户点 5 的时间窗约束退出路径分配, 由车辆 k_3 进行余下客户点的服务。当路径分配完毕后, 先对车辆服务的初始客户点按照就近原则分配配送中心, 满足车辆进出平衡情况下, 再对末尾客户点按照就近原则分配配送中心, 9~11 为配送中心。求解过程如图 2 所示。

2.2 适应度函数

解码过程第 2 阶段确定始末配送中心时不一定满足时间窗需求, 为保证解码的配送路径都是可行解, 加入违反时间窗惩罚条件, 以达到配送路线满足约束, 惩罚函数为

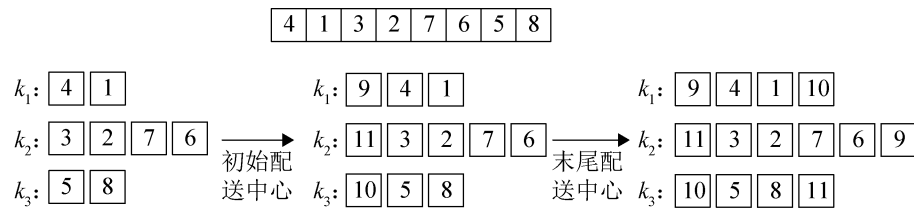


图2 解码示意图
Fig. 2 Decoding diagram

$$t(s) = \sum_{i=1}^n \max\{a_i - b_i, 0\} \quad (14)$$

式中： a_i 为车辆到达客户点 $i(i \in N)$ 的时间； b_i 为配送中心或客户点 i 的右时间窗，计算出违反时间窗约束之和的惩罚函数 $t(s)$ 后，由目标函数确定适应度函数，个体 S 的适应度函数为

$$f_s = \frac{1}{Z_s} + t(s) \quad (15)$$

式中： Z_s 为个体 s 的目标函数值。

2.3 聚类操作

CMBSO 算法采用 k-means 聚类操作，具体步骤如下：

step 1: 随机选择 m 个个体作为聚类中心；

step 2: 依次计算每个个体的适应度值，与 m 个聚类中心适应度的差值的绝对值，将其差值的绝对值最小的聚类中心归到一个类中；

step 3: 求出每个类中的平均适应度值，将类中最接近平均适应度值的个体作为新类中心；

step 4: 判断是否达到终止条件(循环是否达到 20 次)，若是，则终止循环；否则转到 step 2。

2.4 更新操作

本文采用交换、交叉操作增加种群多样性，加强个体间交流，快速形成稳定的下一代。交换规则如图 3(a)所示，将客户点 3 和客户点 6 交换产生新个体。

交叉操作的对象为 2 个个体，采用传统遗传算法中的顺序交叉策略，规则如图 3(b)所示。随机选取客户点 3 和客户点 6 之间的客户点作为交叉片段，将 g_2 交叉片段移动到 g_1 前面， g_1 交叉片段移动到 g_2 前面，然后从左到右将重复客户点删除，形成新个体 g_{11} 和 g_{22} 。

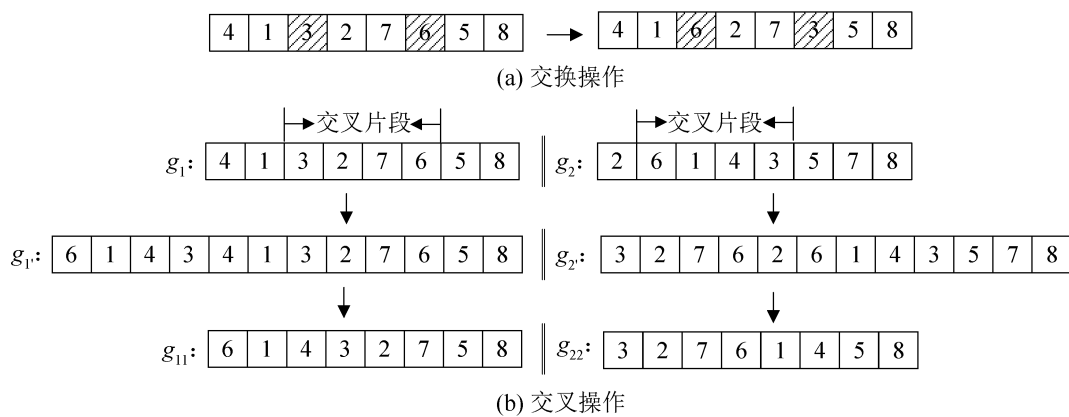


图3 交换和交叉操作
Fig. 3 Swap and cross operation

2.5 局部搜索操作

本文设置更新操作完最优个体适应度值最大循环次数 D_{max} 来判断个体是否陷入局部最优。当判断为否时, 执行高斯变异操作, 新个体为

$$X_{new}^d = X_{old}^d + \varepsilon \times N(\mu, \sigma) \tag{16}$$

$$\varepsilon = \text{logsig}((D_{max}/2 - D)/k) \times \text{rand}(0, 1) \tag{17}$$

式中: X_{old}^d 为选中个体; $N(\mu, \sigma)$ 为均值是 μ 方差是 σ 的高斯随机函数; ε 为高斯随机函数的系数; $\text{logsig}()$ 是 S 形变换函数; D_{max} 为最大迭代次数; D 为当前迭代次数; k 为改变 $\text{logsig}()$ 函数的斜率; rand 为 (0,1) 之间的随机数。高斯变异算子局部搜索能力较强, 侧重于搜索局部区域, 但是容易导致种群搜索停滞。

当判断为是时, 执行混沌变异操作, 混沌解具有随机性和遍历性。本文选取 2 个混沌映射, 分别为 Sine 混沌映射与 Logistic 混沌映射^[15], 选取公式为

$$x(n)_i = \text{rand}\{x(1)_i, x(2)_i\} \tag{18}$$

$$X_{new}^d = X_{old}^d + R \times (X(n)_i^d - 0.2) \tag{19}$$

$$R = X_{old}^d - X_{center}^d \tag{20}$$

式中: $n \in [1, 2]$ 是选取混沌序号; i 为混沌迭代次数; R 为搜索半径; X_{center}^d 为原个体所在类的类中心的 d 维值。

Sine 混沌映射为正弦映射:

$$x(1)_{i+1} = \mu \sin(\pi x(1)_i) \tag{21}$$

式中: $\mu \in [0, 1]$ 。当 μ 越趋近于 1 时, 混沌性越强, $x(1)_i \in (-1, 1)$, 取 $x(1)_i$ 的绝对值。

Logistic 映射为

$$x(2)_{i+1} = \mu x(2)_i (1 - x(2)_i) \tag{22}$$

式中: $\mu \in (1, 4]$ 。当 $\mu = 4$ 时, 系统则处于完全混沌状态, $x(2)_i \in (0, 1)$ 。

混沌映射的选取采用随机策略, 既保留了个体多样性, 也避免了算法复杂性的增加。算法流程图如图 4 所示。

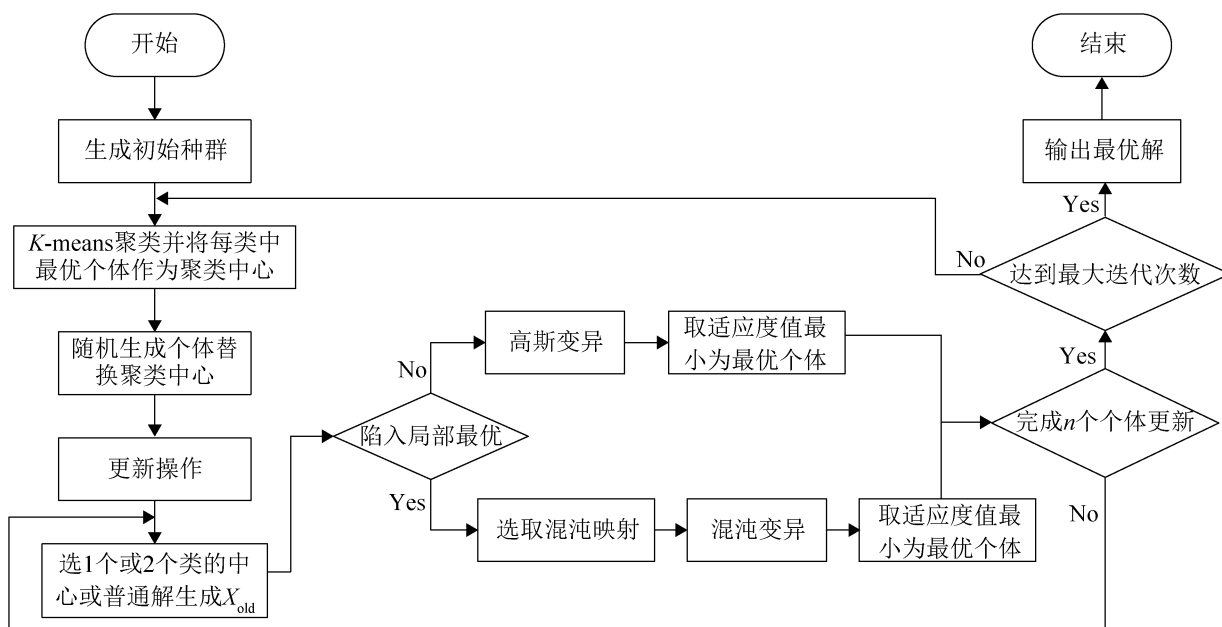


图 4 CMBSO 算法流程图

Fig. 4 CMBSO algorithm flow

3 实验与结果分析

MDHOVRPSDPTW 问题是由 MDVRP、VRPSDPTW 等问题衍生而来, 本文所提算法释放

部分约束可对 MDVRP、VRPSDPTW 问题求解。为验证本文算法的有效性, 选用 MDVRP 算例、VRPSDPTW 算例以及 MDHOVRPSDPTW 算例对其验证。本文在 Matlab2016B 中编程, 并使用

Inter®Core™i5-9400F CPU 2.94GHz 计算机系统在 Windows 10 上执行, 该系统具有 16 GB 内存。实验基本参数见表 1, n 为客户规模; N 为种群数量; T 为迭代数; J 为聚类数目; J_{\max} 为聚类最大循环次数; D_{\max} 为局部最优解次数、 P_o 为选取 1 个聚类的概率; P_t 为选取 2 个聚类的概率。

3.1 MDVRP 算例验证

实验 1: 为验证 CMBSO 算法有效性, 选取标准算例库中 MDVRP 算例中的 p01-p06 算例进行验证 (算例集来源: <http://neumann.hec.ca/>

chairedistributive/ data/mdvrp/old/), 并将本文算法运行 10 次的最优解与文献[16]的迭代局部搜索 (iterated local search, ILS) 算法、文献[17]的协同进化 (co-evolutionary algorithm based on evolution strategies, CoES) 算法以及文献[18]的自适应邻域文化基因 (adaptive memetic algorithm and variable neighborhood search, AMAVNS) 算法作对比, 结果见表 2。表中 BKS 为算例已知最优解, $Best$ 为算法运行 10 次的最优解, Dev 为最优解偏差。

表 1 参数设置
Table 1 Parameter setting

n	N	T	J	J_{\max}	D_{\max}	P_o	P_t
$0 < n \leq 25$	50	100~300	5~10	20	20~50	0.5~0.8	$1-P_o$
$25 < n \leq 50$	100	100~300	5~10	20	20~50	0.5~0.8	$1-P_o$
$50 < n < 100$	100	100~500	5~10	20	20~50	0.5~0.8	$1-P_o$

表 2 实验 1 结果对比
Table 2 Result comparison of experiment 1

算例	客户规模	BKS	ILS		CoES		AMAVNS		CMBSO	
			$Best$	$Dev/\%$	$Best$	$Dev/\%$	$Best$	$Dev/\%$	$Best$	$Dev/\%$
P01	50	576.87	606.11	5.07	576.87	0.00	576.87	0.00	576.87	0.00
P02	50	473.53	496.45	4.84	475.06	0.32	473.53	0.00	474.53	0.21
P03	75	641.19	675.32	5.32	643.57	0.37	646.33	0.80	642.27	0.17
P04	100	1 001.59	1 062.60	6.09	1 011.42	0.98	1 039.69	3.80	1 001.59	0.00
P05	100	750.03	782.34	4.31	752.39	0.31	765.98	2.13	877.86	0.16
P06	100	876.50	910.13	3.84	877.86	0.16	902.27	2.94	882.88	0.72
<i>Ave</i>		719.95	755.49	4.91	722.86	0.36	734.11	1.61	742.67	0.21

由表 2 得出, 粗略比较 4 种算法获得最优解个数, 可得算法有效性 $CMBSO(2)=AMAVNS(2)>CoES(1)>ILS(0)$ 。比较 4 种算法的平均路程可得算法有效性 $CMBSO(742.67) < CoES(722.86) < AMAVNS(732.61) < ILS(755.49)$, 比较 4 种算法最大 $Dev\%$ 可得算法稳定性 $CMBSO(0.72) < CoES(0.98) < AMAVNS(3.80) < ILS(6.09)$ 。比较算法的平均 $Dev\%$ 可得算法的稳定性 $CMBSO(0.21) < CoES(0.36) < AMAVNS(1.61) < ILS(4.91)$, 通过比较验证了本文算法求解多中心车辆路径问题的优势与稳定性。

3.2 VRPSDPTW 算例验证

VRPSDPTW 算例选用 Wang 等在 2012 年公布的标准算例 (算例集来源: <http://oz.nthu.edu.tw/~d933810/test.htm>)。该算例集共包含 65 个算例, 分为 9 个中小规模算例和 56 个大规模算例。

实验 2: 中小规模算例的验证中, 选取 9 个算例, 将本文算法与文献[3]提出的离散布谷鸟 (discrete cuckoo search, DCS) 算法与文献[19]提出的模因 (memetic algorithm, MA) 算法作对比, 结果见表 3。

表3 实验二结果对比
Table 3 Result comparison of experiment 2

算例	客户规模	BKS	Best		
			DCS	MA	CMBSO
Rcdp1001	10	349.98	349.98	349.98	349.98
Rcdp1004	10	216.69	216.69	216.69	216.69
Rcdp1007	10	310.81	310.81	310.81	310.81
Rcdp2501	25	551.05	551.05	551.05	551.05
Rcdp2504	25	473.46	473.46	473.46	473.46
Rcdp2507	25	540.87	540.87	540.87	540.87
Rcdp5001	50	994.18	994.18	994.18	994.18
Rcdp5004	50	725.59	725.59	725.59	725.59
Rcdp5007	50	809.72	809.72	809.72	809.72

由表4可知, 本文算法求解VRPSDPTW问题中小规模算例均取得了最优解, 验证了本文算法求解带时间窗的同时送取货车辆路径问题中小规模客户问题的有效性。

实验3: 为进一步测试本文算法性能, 选取6个大规模算例进行验证, 选取算例特点见表4。将本文算法与文献[3]提出的DCS算法、文献[4]提出的回溯搜索优化(bachtracking search optimization algorithm, BSA)算法以及文献[20]提出的变邻域双结构禁忌搜索(variable neighborhood search and bi-structure based tatu seach, VNS-BSTS)算法作对比, 结果见表5。

由表5可知, 对Cdp201算例, 4种算法均取

得了最优解; 对Rcdp101算例的求解, 本文算法弱于DCS算法与BSA算法, 强于VNS-BSTS算法; 对Rdp201算例的求解, 本文算法弱于VNS-BSTS算法, 强于DCS算法与BSA算法; 对Rdp101算例、Cdp101算例, 本文算法与VNS-BSTS算法求解能力相同, 均强于DCS算法与BSA算法; 对于Rcdp201算例, 本文算法求解结果均优于其他算法。另外, 通过最大Dev与平均Dev对比可知, CMBSO(7.34)<VNS-BSTS(8.39)<BSA(12.94) <DCS(14.66), CMBSO(2.59) <VNS-BSTS(3.20)<BSA(4.26)<DCS(4.54), 比较平均路程可得 CMBSO(1 259.99) <VNS-BSTS(1 269.78) <BSA(1 280.72)<DCS(1 284.17), 从而验证了本文算法求解带时间窗的同时送取货车辆路径问题中大规模客户问题的有效性与稳定性。

表4 大规模算例特点
Table 4 Characteristics of large-scale examples

算例	客户规模	客户位置分布	时间窗宽紧
Rdp101	100	分散	较紧
Cdp101	100	聚集	较紧
Rcdp101	100	分散与聚集混合	较紧
Rdp201	100	分散	较宽
Cdp201	100	聚集	较宽
Rcdp201	100	分散与聚集混合	较宽

表5 实验3结果对比
Table 5 Result comparison of experiment 3

算例	BKS	DCS		BSA		VNS-BSTS		CMBSO	
		Best	Dev/%	Best	Dev/%	Best	Dev/%	Best	Dev/%
Rdp101	1 650.20	1 658.65	0.51	1 659.77	0.58	1 650.80	0.04	1 650.80	0.04
Cdp101	963.96	998.29	3.56	992.88	3.00	976.04	1.25	976.04	1.25
Rcdp101	1 652.90	1 654.32	0.08	1 655.77	0.17	1 708.21	3.35	1 666.12	0.80
Rdp201	1 181.73	1 281.63	8.45	1 286.55	8.87	1 254.57	6.16	1 268.52	7.34
Cdp201	591.56	591.56	0.005	591.56	0.00	591.56	0.00	591.56	0.00
Rcdp201	1 326.19	1 520.56	14.66	1 497.80	12.94	1 437.48	8.39	1 406.94	6.09
Ave	1 227.81	1 284.17	4.54	1 280.72	4.26	1 269.78	3.20	1 259.99	2.59

3.3 MDHOVRPSDPTW 算例验证

实验4: MDHOVRPSDPTW 问题相关约束较多, 目前没有通用算例, 本文选用标准算例库中 MDVRPTW 算例 pr01-pr02(算例集来源: <http://neumann.hec.ca/chairedistributique/data/>), 并按 MDHOVRPSDPTW 问题的特点修改后进行验证。pr01 算例包括48个客户点(编号为1~48), 4个配送中心(编号为49~52), 车辆容量为200。pr02 算例包括96个客户点(编号为1~96), 4个配送中心(编号为97~100), 车辆容量为195。客户点的送取需求由文献[21]提出的需求分离规则产生: x_i, y_i 为客户点的坐标, d_i, p_i 为客户点送货、取货需求, q_i 为客户点原始需求, $r_i = \min(x_i/y_i, y_i/x_i)$ 计算每个客户点比率, 由 $d_i = q_i r_i$ 和 $p_i = q_i(1 - r_i)$, 计算得出各个客户点的送取需求。

将带时间窗的多中心半开放同时送取货配送模式与带时间的多中心闭合式同时送取货配送模

式作对比, 计算结果稳定, 结果见表6, $Best, Avg, Num, Cpu$ 为平均解和平均车辆数及平均算法运行时间。

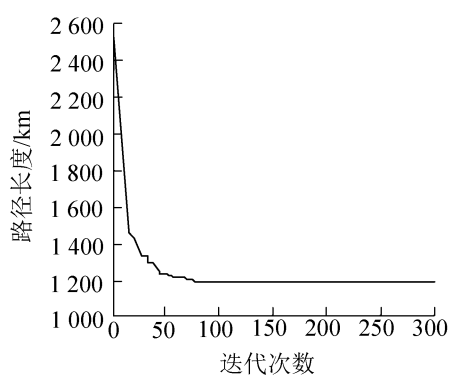
由表6可以得出, CMBSO 算法求解两类配送模式的最优解偏差最小值为1.01%, 最大值为6.54%, 能够有效的求出最优解且求解速度较快。在车辆总行驶距离方面, MDHOVRPSDPTW 配送模式比 MDVRPSDPTW 配送模式分别优化了8.3%与6.7%。

由图5最优解迭代图看出, CMBSO 算法能够以较少的迭代次数快速收敛, 具有较强的寻优能力。由表7可以看出, 从 MDVRPSDPTW 配送模式到 MDHOVRPSDPTW 配送模式, 配送中心由闭合式变为半开放式, 实现了各配送中心间车辆共享, 大幅度减少了车辆频繁往返同一配送中心产生的多余路径, 进而降低物流企业的配送成本。

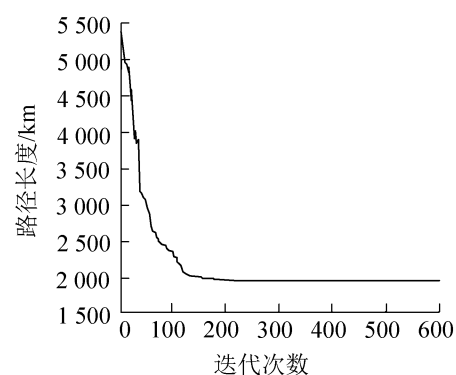
表6 实验4结果对比

Table 6 Result comparison of experiment 4

算例	客户规模	MDHOVRPSDPTW					MDVRPSDPTW				
		Num	Best	Avg	Cpu/s	Dev/%	Num	Best	Avg	Cpu/s	Dev/%
pr01	48	5	1 201.26	1 285.31	4.3	6.54	5	1 310.31	1 351.47	3.7	3.05
pr02	96	9	1 977.52	2 073.01	13.6	4.61	9	2 118.88	2 140.52	20.4	1.01



(a) pr01最优解



(b) pr02最优解

图5 MDHOVRPSDPTW 模式下最优解迭代图

Fig. 5 Optimal solution iteration under MDHOVRPSDPTW mode

表 7 实验 4 最优解路径
Table 7 Optimal solution paths of experiment 4

模式	算例	车辆行驶路线	总距离
MDHOVRPSDPTW	pr01	49-42-46-39-2-15-25-26-23-36-32-49	1 201.26
		51-33-13-8-29-20-4-19-22-50	
		50-9-47-24-12-30-40-38-21-43-52	
		52-44-31-41-7-37-28-14-1-5-17-18-16-51	
		49-35-34-10-45-6-27-3-48-11-50	
	pr02	100-95-65-42-85-66-80-48-6-10-24-18-14-22-1-47-98	1 977.52
		97-43-63-77-90-31-45-70-59-84-97	
		100-11-79-21-15-75-40-34-49-4-61-52-100	
		97-86-19-55-93-38-68-27-13-97	
		97-81-62-73-37-69-9-17-64-16-74-97	
MDVRPSDPTW	pr01	99-96-67-2-51-50-76-3-12-53-36-56-71-83-82-35-100	1 310.31
		97-41-89-54-29-57-46-30-23-91-26-28-58-99	
		97-8-92-20-39-7-5-44-97	
		98-25-60-72-33-88-78-94-32-87-97	
		49-42-46-12-24-15-25-23-30-40-38-21-43-49	
	pr02	51-33-13-8-29-20-4-19-22-51	2 118.88
		49-44-31-41-7-37-49	
		49-9-47-39-2-26-36-28-14-1-5-17-18-16-32-49	
		49-35-34-10-45-6-27-3-48-11-49	
		97-19-55-20-64-17-9-72-87-1-47-32-97	
pr02	97-62-73-37-69-65-33-88-78-22-46-91-28-52-58-97	2 118.88	
	97-25-96-67-51-76-3-12-50-6-24-15-18-10-94-97		
	100-63-8-13-100		
	97-43-41-70-59-84-97		
	97-86-92-93-38-68-39-27-74-97		
pr02	97-81-60-42-85-66-2-80-48-7-5-16-44-97	2 118.88	
	100-95-29-57-21-79-11-54-89-45-31-90-77-4-49-82-35-61-100		
	100-15-75-40-34-26-23-30-53-36-56-71-83-100		

4 结论

本文针对提出的 MDHOVRPSDPTW 问题主要进行了以下研究。①构建车辆进出平衡且以配送距离最小化的带时间窗的多中心半开放式同时送取货车辆路径数学模型。②设计混沌变异头脑风暴算法对该模型求解, 通过多组算例验证了算法的稳定性与有效性。③对 MDHOVRPSDPTW 配送模式与 MDVRPSDPTW 配送模式作对比, 验证了前者大幅度减少了车辆频繁往返配送中心产生的配送路径, 降低了配送成本。

随着生活质量的提高, 人们对于物流配送的

质量、时效等需求越来越高, MDHOVRPSDPTW 配送模式可更好地满足客户的配送需求, 在车辆路径中考考虑碳排放约束、客户满意度等多方面因素将是下一步研究的方向。

参考文献:

[1] Angelelli E, Mansini R. The Vehicle Routing Problem with Time Windows and Simultaneous Pick-up and Delivery[C]//Quantitative Approaches to Distribution Logistics and Supply Chain Management. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002: 249-267.
[2] Wang H F, Chen Y Y. A Genetic Algorithm for the Simultaneous Delivery and Pickup Problems with Time Window[J]. Computers & Industrial Engineering, 2012,

- 62(1): 84-95.
- [3] 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[J]. 计算机集成制造系统, 2018, 24(3): 570-582.
Wang Chao, Liu Chao, Mu Dong, et al. VRPSPDTW Problem Solving by Discrete Cuckoo Search[J]. Computer Integrated Manufacturing Systems, 2018, 24(3): 570-582.
- [4] 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送取货的车辆路径问题[J]. 计算机集成制造系统, 2019, 25(9): 2237-2247.
Wang Chao, Gao Yang, Liu Chao, et al. Vehicle Routing Problem with Simultaneous Delivery and Pickup Problem Solving by Backtracking Search Optimization Algorithm[J]. Computer Integrated Manufacturing Systems, 2019, 25(9): 2237-2247.
- [5] 李珺, 段钰蓉, 郝丽艳, 等. 混合优化算法求解同时送取货车车辆路径问题[J]. 计算机科学与探索, 2022, 16(7): 1623-1632.
Li Jun, Duan Yurong, Hao Liyan, et al. Hybrid Optimization Algorithm for Vehicle Routing Problem with Simultaneous Delivery-pickup[J]. Journal of Frontiers of Computer Science & Technology, 2022, 16(7): 1623-1632.
- [6] Li Jian, Li Yang, Pardalos P M. Multi-depot Vehicle Routing Problem with Time Windows Under Shared Depot Resources[J]. Journal of Combinatorial Optimization, 2016, 31(2): 515-532.
- [7] Bae H, Moon I. Multi-depot Vehicle Routing Problem with Time Windows Considering Delivery and Installation Vehicles[J]. Applied Mathematical Modelling, 2016, 40(13/14): 6536-6549.
- [8] 王勇, 任音吉, 刘永, 等. 基于多中心车辆路径问题的收益分配优化研究[J]. 交通运输系统工程与信息, 2018, 18(3): 210-217.
Wang Yong, Ren Yinji, Liu Yong, et al. Profit Allocation Optimization Based on Multi-center Vehicle Routing Problem[J]. Journal of Transportation Systems Engineering and Information Technology, 2018, 18(3): 210-217.
- [9] 范厚明, 杨翔, 李荡, 等. 基于生鲜品多中心联合配送的半开放式车辆路径问题[J]. 计算机集成制造系统, 2019, 25(1): 256-266.
Fan Houming, Yang Xiang, Li Dang, et al. Half-open Multi-depot Vehicle Routing Problem Based on Joint Distribution Mode of Fresh Food[J]. Computer Integrated Manufacturing Systems, 2019, 25(1): 256-266.
- [10] 马冰山, 胡大伟, 陈希琼, 等. 半开放式的多配送中心纯电动车辆路径优化问题[J]. 交通运输系统工程与信息, 2019, 19(6): 199-205.
Ma Bingshan, Hu Dawei, Chen Xiqiong, et al. An Optimization of Pure Electric Vehicle Routing Problem on Half-open Multi-distribution Center[J]. Journal of Transportation Systems Engineering and Information Technology, 2019, 19(6): 199-205.
- [11] 辜勇, 袁源乙, 张列, 等. 带时间窗的多中心半开放式车辆路径问题[J]. 中国机械工程, 2020, 31(14): 1733-1740.
Gu Yong, Yuan Yuanyi, Zhang Lie, et al. Multi-depot Half Open Vehicle Routing Problem with Time Windows [J]. China Mechanical Engineering, 2020, 31(14): 1733-1740.
- [12] 张凯庆, 嵇启春. 速度时变的多中心半开放式车辆路径问题研究[J]. 系统仿真学报, 2022, 34(4): 836-846.
Zhang Kaiqing, Ji Qichun. Research on Multi-depot Half-open Vehicle Routing Problem with Time-varying Speed [J]. Journal of System Simulation, 2022, 34(4): 836-846.
- [13] Yang Zhensu, Shi Yuhui. Brain Storm Optimization with Chaotic Operation[C]//2015 Seventh International Conference on Advanced Computational Intelligence (ICACI). Piscataway, NJ, USA: IEEE, 2015: 111-115.
- [14] Yu Yang, Gao Shangce, Cheng Shi, et al. CBSO: a Memetic Brain Storm Optimization with Chaotic Local Search[J]. Memetic Computing, 2018, 10(4): 353-367.
- [15] 衣俊艳, 施晓东, 杨刚. 多分支混沌变异的头脑风暴优化算法[J]. 计算机工程与应用, 2022, 58(16): 129-138.
Yi Junyan, Shi Xiaodong, Yang Gang. Brain Storm Optimization Based on Multi-branch Chaotic Mutation [J]. Computer Engineering and Applications, 2022, 58(16): 129-138.
- [16] Tlili T, Krichen S, Drira G, et al. On Solving the Multi-depot Vehicle Routing Problem[C]//Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics. New Delhi: Springer India, 2016: 103-108.
- [17] Fernando Bernardes de Oliveira, Enayatifar R, Sadaei H J, et al. A Cooperative Coevolutionary Algorithm for the Multi-depot Vehicle Routing Problem[J]. Expert Systems with Applications, 2016, 43: 117-130.
- [18] 范厚明, 刘鹏程, 刘浩, 等. 多中心联合配送模式下集货需求随机的VRPSDP问题[J]. 自动化学报, 2021, 47(7): 1646-1660.
Fan Houming, Liu Pengcheng, Liu Hao, et al. The Multi-depot Vehicle Routing Problem with Simultaneous Deterministic Delivery and Stochastic Pickup Based on Joint Distribution[J]. Acta Automatica Sinica, 2021, 47(7): 1646-1660.
- [19] 张庆华, 吴光谱. 带时间窗的同时取送货车车辆路径问题建模及模因求解算法[J]. 计算机应用, 2020, 40(4): 1097-1103.
Zhang Qinghua, Wu Guangpu. Modeling and Memetic

- Algorithm for Vehicle Routing Problem with Simultaneous Pickup-delivery and Time Windows[J]. Journal of Computer Applications, 2020, 40(4): 1097-1103.
- [20] Shi Yong, Zhou Yanjie, Boudouh T, et al. A Lexicographic-based Two-stage Algorithm for Vehicle Routing Problem with Simultaneous Pickup-delivery and Time Window[J]. Engineering Applications of Artificial Intelligence, 2020, 95: 103901.
- [21] Salhi S, Nagy G. A Cluster Insertion Heuristic for Single and Multiple Depot Vehicle Routing Problems with Backhauling[J]. Journal of the Operational Research Society, 1999, 50(10): 1034-1042.