

改进分散搜索算法求解包装废弃物回收路径规划问题

张琦琪, 陈群*

(上海出版印刷高等专科学校, 上海 200093)

摘要: **目的** 将包装废弃物回收路径规划归纳为一个带回路和时间窗的逆向物流车辆路径问题 (RL-VRPBTW), 以最小化回收成本、发车成本和时间窗惩罚为联合优化目标进行建模。**方法** 引入“车辆剩余空间回收能力”因素, 改进经典节约里程算法, 求得较好的初始解; 基于分散搜索框架, 设计基于初始解改进的分散搜索算法 (ISISS), 根据问题模型, 采用含0的编码方式, 通过多样性产生、参考集更新、子集产生、子集合并、解改进等5个步骤实现算法功能。**结果** 在“部分回收点分布较密集”的城市型地理场景下, 针对快消企业的低值固废包装, 生成回收点数量分别为50、100、200的3种规模算例, 并考虑大小两种车型进行仿真实验。将ISISS算法与改进节约里程、遗传和分散搜索3种算法比较后可知, ISISS算法在大规模包装废弃物回收车辆路径问题上具有更优的求解性能。**结论** 仿真实验结果表明, ISISS是一种求解多目标大规模包装废弃物回收路径规划问题的较优算法。

关键词: 逆向物流; 带时间窗和回路的车辆路径问题; 分散搜索; 局部搜索

中图分类号: TP301; TB498 **文献标志码:** A **文章编号:** 1001-3563(2024)09-0193-08

DOI: 10.19554/j.cnki.1001-3563.2024.09.025

Improved Scatter Search Algorithm to Solve Packaging Waste Recovery Vehicle Routing Problem

ZHANG Qiqi, CHEN Qun*

(Shanghai Publishing and Printing College, Shanghai 200093, China)

ABSTRACT: The work aims to summarize the packaging waste recovery routing as a reverse logistics vehicle routing problem with back path and time window (RL-VRPBTW), so as to construct a model with the minimum recovery cost, departure cost and time window penalty as the joint optimization objective. A better initial solution was obtained by introducing the factor of "vehicle remaining space recovery ability" to improve the classical heuristic C-W algorithm. Based on the Scatter Search framework, a Scatter Search algorithm (ISISS) based on initial solution improvement was designed. According to the problem model, the algorithm functions were realized through five steps of diversity generation method, reference set update method, subset generation, merging method and solution improvement method. In the city-type geographical scenario of "dense distribution of some recovery points", three scale examples with 50, 100 and 200 recovery nodes were randomly generated, and two vehicle types were considered for simulation experiments. The ISISS algorithm was compared with C-W, GA and SS algorithms to verify that the algorithm proposed had better performance in solving the routing problem of large-scale packaging waste recovery vehicles. The simulation results indicate that ISISS is a better algorithm to solve the multi-objective large-scale packaging waste recovery routing problem.

KEY WORDS: reverse logistics; vehicle routing problem with time windows and back path; scatter search; local search

收稿日期: 2023-06-28

基金项目: 国家社科基金 (18BT058); 国家新闻出版署“智能与绿色柔版印刷”重点实验室项目 (KLIGFP-01); 上海市东方学者特聘教授基金 (TP2022126)

*通信作者

绿色包装产业发展是加快推进我国生态文明建设的重要一环。2020年11月,国家发展改革委、国家邮政局等八部门联合发布了《关于加快推进快递包装绿色转型的意见》,明确提出“规范包装废弃物回收和处置”及“提升快递包装可回收性能”等要求。废弃包装品回收路径规划属于逆向物流车辆路径问题,是对传统车辆路径问题的扩展^[1-3],具有重大的社会意义。快消连锁餐饮网点的食品级包装固体废弃物具有材料同质、回收量小、批次多且分布不均等特点,企业对包装固体废弃物的回收循环利用从一定程度上缩减了生产中的物料成本,在提升经济效益的同时实施绿色制造,实现了社会效益的增长。由此可见,基于逆向物流理论,研究包装废弃物回收路径规划、科学安排运输回收路线具有较大的现实意义。

带回路和时间窗的逆向物流车辆路径问题(RL-VRPBTW, Reverse Logistics Vehicle Routing Problem with Backhauls and Time Windows)属于大规模组合优化问题。Anily等^[4]证明逆向物流VRP问题属于NP难题,精确算法难以在有效时间内求解该类问题,而经典启发式算法也只能在数据规模较小时求得一个近优解。随着遗传算法、蚁群算法等进化计算方法的出现,逆向物流车辆路径问题的求解有了新方向。文献[5]基于贪婪算法,设计了初始解改进遗传算法,以求解包装废弃物回收车辆路径问题。文献[6]设计了改进蚁群算法,以求解逆向物流车辆路径问题。文献[7]考虑了CO₂排放量因素,构建了逆向物流车辆短路径优化模型,并结合局部搜索和果蝇算法求解。文献[8]构建了考虑货架寿命的配送车辆路径优化模型,通过改进遗传算法进行求解。以上研究将进化计算与逆向物流问题结合,从最小化运输成本角度出发建模求解,有效验证了进化算法在解决逆向物流VRP问题上的计算能力。以上研究未在回收路径的规划过程中考虑车载容量是否有利于回收旅程的问题,因此还需进一步研究。

分散搜索^[9](Scatter Search, SS)是一类新型优化算法框架,它基于种群的全局搜索策略,利用一系列子方法构建新解,既可以提高搜索的集中性,又可以保持种群的多样性。近年来,将SS与其他搜索策略相结合,被广泛应用于求解复杂的生产计划调度问题^[10-12]。在VRP问题的研究中,文献[13]利用分散搜索算法,求解同时送取货的随机旅行时间车辆路径问题。文献[14]设计了分散搜索算法,以求解多配送中心车辆路径问题。文献[15]提出一种基于分散搜索的自适应记忆编程技术,以求解多商品多变体的取货车辆路径问题。文中针对连锁餐饮快消企业对低值固体废弃包装品的回收需求,以最小化车辆运输成本、回收中心发车成本及时间窗提前或拖期惩罚为联合优化目标,建立整数规划模型,基于运输与回收协同的理念,引入“车辆剩余空间回收能力”改进节约里程算

法(Clarke-Wright, C-W)^[16]求得初始解,然后利用局部搜索策略对初始解进行多样化处理,最后根据分散搜索框架设计基于初始解改进的分散搜索算法进行求解。

1 问题描述与模型建立

1.1 问题描述

可以将连锁餐饮企业低值废弃包装回收问题抽象为一个考虑时间窗约束的带回路的逆向物流车辆路径问题(RL-VRPBTW)。具体描述如下:完全连接网络 $G=(J_0, E)$,其中节点集合为 J_0 ,节点0表示回收中心,集合 $J=J_0-\{0\}$ 表示回收节点集合, J 中各节点 j 均有回收需求 P_j ,且每个节点的需求服务受到时间窗 $[E_j, L_j]$ 的约束;有向边集合 E 是 J_0 的二元子集, $e \in E$, $e=(v_i, v_j)$ 。假设车辆载重在回收过程中不变,车辆在有向边 (v_i, v_j) 上匀速行驶,且所有车辆的载重上限相同,每辆车的行驶总时间不超过事先确定的最大值;车辆从回收中心空车出发,从回收点取货,每个回收点有且仅有1辆车服务1次;不考虑车辆在回收点的服务时间,车辆如提前于时间窗到达回收点,需等待;当超过车载上限或最大行驶时间限制时返回回收中心,直到服务完所有回收点。废弃包装回收路径规划需要解决的问题是在满足所有节点回收需求、时间窗约束、车辆载重限制及车辆行驶时间限制等前提下,如何安排车辆的行驶路径,使得回收中心的回收成本最小。

1.2 符号说明

符号说明: J 表示所有回收点集合; J_0 表示区域内所有节点集合,节点包括回收中心和回收点, $J_0=J \cup \{0\}$; V 表示所有车辆集合; D_{ij} 表示2个节点 i 与 j 之间的欧几里得距离,其中 $i, j \in J_0$; v_{ijk} 表示车辆 k 在节点 i 与 j 之间的行驶速度; t_{ijk} 表示车辆 k 在节点 i 与 j 之间的行驶时间, $t_{ijk}=D_{ij}/v_{ijk}$; C_{ijk} 表示车辆 k 在节点 i 与 j 之间的行驶成本,假设车辆的单位行驶成本为1,则有 $C_{ijk}=t_{ijk}$; C_k 表示车辆 k 的发车成本; P_j 表示回收点 j 的回收量; $[E_j, L_j]$ 表示节点 j 的时间窗,其中 $[E_0, L_0]$ 为回收中心节点0开启服务的时间窗; q_{ijk} 表示车辆 k 访问完节点 i 后在访问节点 j 之前的载重量; Q_k 表示车辆 k 的载重上限; B_k 表示车辆 k 行驶的总时间上限; μ 表示回收中心允许选择的最小车辆数量; α 表示提前到达的惩罚系数; β 表示延期到达的惩罚系数。

定义决策变量: x_{ijk} 表示0-1变量,车辆 k 从节点 i 行驶到节点 j 时为1,否则为0; z_{jk} 表示0-1变量,节点 j 由车辆 k 服务为1,否则为0; t_{jk} 表示车辆 k 到达节点 j 的时间。

以上描述中, $i, j \in J_0, k \in V$, 且 $n=|J_0|, m=|V|$ 。

1.3 数学模型

考虑上述问题, 建立模型, 见式 (1) ~ (12)。

$$\min f = \sum_{k=1}^m \sum_{j=0}^n C_k z_{jk} + \sum_{k=1}^m \sum_{i=0}^n \sum_{j=0}^n C_{ijk} x_{ijk} + \left(\alpha \sum_{k=1}^m \sum_{j=1}^n \max \{E_j - t_{jk}, 0\} + \beta \sum_{k=1}^m \sum_{j=1}^n \max \{t_{jk} - L_j, 0\} \right) \quad (1)$$

s. t.

$$\sum_{k=1}^m \sum_{i=0}^n x_{ijk} = 1 \quad j=1, 2, \dots, n \quad (2)$$

$$\sum_{i=0}^n x_{isk} = \sum_{j=0}^n x_{sjk} \quad s=1, 2, \dots, n, k=1, 2, \dots, m. \quad (3)$$

$$\sum_{j=1}^n x_{0jk} + \sum_{j'=1}^n x_{j'0k} = 2z_{0k} \quad k=1, 2, \dots, m \quad (4)$$

$$\sum_{i=0}^n \sum_{j=1}^n P_j x_{ijk} \leq Q_k \quad k=1, 2, \dots, m \quad (5)$$

$$\sum_{i=0}^n \sum_{j=0}^n t_{ijk} x_{ijk} + \max \{E_j - t_{jk}, 0\} \leq B_k \quad k=1, 2, \dots, m \quad (6)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq |S| - 1 \quad S \subset J, k=1, 2, \dots, m \quad (7)$$

$$\sum_{i=0}^n \sum_{j=0}^n x_{ijk} P_j = \sum_{j=1}^n q_{j0k} \quad k=1, 2, \dots, m \quad (8)$$

$$\sum_{i=0}^n q_{ijk} + P_j = \sum_{i'=0}^n q_{ji'k} \quad j=1, 2, \dots, n, k=1, 2, \dots, m \quad (9)$$

$$0 \leq P_j \leq Q_k, j=1, 2, \dots, n, k=1, 2, \dots, m \quad (10)$$

$$x_{ijk} \in \{0, 1\} \quad (11)$$

$$z_{jk} \in \{0, 1\} \quad (12)$$

其中, 式 (1) 为目标函数, 由 3 个部分构成, 第 1 部分为回收中心的发车成本; 第 2 部分为车辆的回收成本; 第 3 部分为软时间窗惩罚, 对不能按要求到达回收点 (提前或拖期) 的情况给予惩罚。式 (2) 确保每个回收节点都会被服务 1 次且仅有 1 次。式 (3) 确保到达每个节点并离开的是同一辆车。式 (4) 确保车辆从回收中心出发, 最终返回回收中心。式 (5) 表示车辆上的回收总数量不超过载重限额的约束。式 (6) 确保车辆的总行驶时间 (包括提前到达的等待时间) 不超过行驶时间上限。式 (7) 表示奇异性回路排除约束。式 (8) 确保车辆的最终回收载重量等于它访问的所有回收点回收数量之和。式 (9) 表示车辆在访问节点前后的载重量约束。式 (10) 确保每个节点的回收量不大于车载上限。式 (11) ~ (12) 表示决策变量取值范围约束。

2 算法设计

在分散搜索框架下, 基于初始解改进的分散搜索

算法 (Initial Solution Improvement Scatter Search algorithm, ISSS) 由 5 个计算步骤组成, 分别是多样性产生、参考集更新、子集产生、子集合并和解改进等。SS 只是一个柔性框架, 其中每个步骤都可以根据实际问题利用多种方法实现, 这就使 SS 在拥有进化算法交叉和变异等功能的同时, 还可以通过控制分散和收敛聚集, 提升算法的收敛性和全局覆盖能力。

2.1 多样化初始解

采用含 0 的整数编码方式, 0 表示回收中心, 自然数为路径上回收点的编号。比如编码 0102030405060708090, 表示解中有 9 条路径, 每条路径上只有 1 个回收点。

首先利用改进的节约里程算法生成模型的一个初始解, 记作 CW 解。节约里程算法的基本思想: 首先, 使每个回收点与回收中心单独连接, 构造出 $(n-1)$ 条路径; 然后, 基于三角不等式关系, 对初始解进行缩小总距离处理, 基于路径间合并可以节约距离, 计算任意 2 个回收点组合对应的节约值 $\text{Save}(i, j)$, 将所有节约值从大到小排序, 依序构造路线, 对满足条件的回收点进行连接, 直到满足约束条件为止, 得到问题的一个 CW 解。

文中主要研究废弃包装回收问题, 将节约值统一在时间尺度上进行考量, 同时将车辆行驶过程中的“剩余空间回收能力”引入节约值的计算, 即越有利于接下来回收装载的回收点, 应该尽早插入当前回收路径中, 根据行车运输时间和回收点的回收量计算回收点两两之间的节约值, 见式 (13)。

$$\text{Save}(i, j) = t(0, i) + t(0, j) - t(i, j) - \lambda P_j 2t(0, j) \quad (13)$$

式中: $\text{Save}(i, j)$ 为路线上插入回收点 j 后的节约值; $t(i, j)$ 为 2 个回收点间的行驶时间; λP_j 表示从回收中心出发, 仅访问回收点 j 的车载空间损失 (或收益); λ 为惩罚因子。

必须指出, 以上节约里程算法只能生成一个较好解, 且不能作为进化算法的初始解集合, 需设计多样化策略, 形成初始解集合。这里通过随机交换 CW 解中任意 2 个回收点的方法, 产生 N 个初始解。由于采用含 0 的编码方式, 在 CW 解中任意交换回收节点可能导致生成的新解不满足模型的约束条件, 必须重新根据车载上限约束式 (5) 和行驶时间最大上限约束式 (6) 等, 对产生的新解进行修复。由于回收点的位置选择具有随机性, 所以在保证初始解是较好解的同时, 随着回收点规模的增大, 还可以保证初始解的多样性。

2.2 参考解更新

参考集 b 由高质量的解子集 b_1 和多样性解子集 b_2 两部分构成。 b_1 由整个种群的解目标值从小到

大排序决定。 b_2 根据种群中所有解与高质量解子集的距离决定, 距离越大则多样性越好。参考集更新步骤如下。

1) 根据目标函数, 计算整个种群所有解的目标值, 并从小到大排序, 取前 $|b_1|$ 个解, 加入参考集 b 中。

2) 计算种群中剩余解 ($N-b_1$) 与子集 b_1 中解的距离。遍历种群的所有剩余解, 对于任意一个剩余解 i , 计算解中所有有向边的数量, 记作 s_i 。计算高质量子集 b_1 中任意一个解 j 所有有向边的数量, 记作 s_j , 并计算解 i 与解 j 中相同的边数量, 记作 s_{ij} 。根据式 (14) 计算剩余解 i 对于高质量解集合 b_1 的距离。取剩余解中距离最大的 $|b_2|$ 个解加入参考集 b 中。

$$\text{distance}(i) = \min_{j \in b_1} \{ \max(s_i, s_j) - s_{ij} \} \quad (14)$$

2.3 子集产生与子集合并

利用参考集 b 中所有不同的 2 个解两两组合的方式, 产生 $C_{|b|}^2$ 个子集; 随机选择 N 个子集组合, 成为子集合并操作的子种群。子集合并主要是为了将种群中优良的解保留到下一代, 同时增加种群的多样性, 以便跳出局部最优。文中设计的子集合并方法类似于遗传算法中的单点交叉算子, 具体步骤如下。

1) 首先去掉每个子集 2 个解中的所有 0, 在回收节点长度范围内随机产生一个交换基因位。

2) 保留其中一个解的前段, 并在另一解中去除已保留的回收节点, 将剩余的回收节点按原顺序排于保留路径之后。具体合并过程如图 1 所示。

3) 根据车载上限, 车辆最大行驶时间上限等约束在新解中重新插入 0。

4) 比较原解 1、原解 2、新解 1、新解 2 的目标值, 保留目标值最小的那个解到下一代种群。

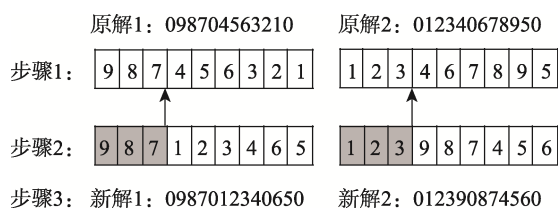


图 1 子集合并过程
Fig.1 Subset merging procedure

2.4 解改进方法

由于文中采用含 0 的编码设置, 因此在解的改进时可以考虑“路径内部”的随机交换回收点 (局部搜索策略 1), 以及“路径之间”的随机交换回收点 (局部搜索策略 2) 这 2 种局部搜索改进策略。

2.4.1 局部搜索策略 1

遍历整个种群, 针对种群中的任意解进行如下步骤。

1) 随机选择一条回收路径上的 2 个不同的点,

如果 2 个回收点之间含 0, 或者回收点本身为 0, 则继续随机选择。

2) 将行驶路径上这 2 个回收点之间所有回收点的车辆行驶方向逆转, 得到一个新解。

3) 计算新解的目标值, 如果目标值优于原解, 则保留新解; 否则不保留此新解。

局部搜索策略 1 只在解的一条路径内部进行, 因此不需要考虑车辆的最大负载上限和车辆最大行驶时间上限等约束, 相当于在原解的邻域进行搜索, 在解改进时不考虑会产生非可行解的情况。

2.4.2 局部搜索策略 2

遍历整个种群, 针对种群中的任意解进行如下步骤。

1) 随机选择 2 条回收路径上的 2 个非 0 回收点, 交换回收点在解中的基因位置。

2) 去掉解中所有的 0, 采用穷尽车载上限及最大车辆行驶时间的方式, 即能够纳入前一车辆的回收点尽量纳入前一条行驶路线, 重新插入 0, 得到一个新解。

3) 计算新解的目标值, 如果目标值优于原解, 则保留新解; 否则不保留此新解。

局部搜索策略 2 在一个解的多条路径之间随机选择交换点, 扩大了搜索空间中这个解的邻域范围。由于交换路径间的回收点可能导致车载上限或车辆最大行驶时间上限等约束不满足的情况出现, 所以需要重新判断解的可行性, 对非可行情况进行修复。

3 仿真实验

为了验证文中提出的 ISSS 算法在包装废弃物回收路径规划问题中的有效性, 以“部分回收点分布较密集”的类似市区分布为地理场景, 针对连锁餐饮快消企业对低值固废包装品的回收需求量级, 利用 Dethloff^[17]和 Solomon^[18]提出的测试数据和时间窗生成方法, 随机生成分别包含 50、100、200 个回收点的 3 种规模, 且最小车辆数 μ 为 3 或 8 的各 10 组样本数据进行仿真实验。

3.1 算例生成方法

以 100 个回收点测试规模为例, 在“部分回收点分布较密集”的地理场景下, 利用 Dethloff 随机测试算例生成方法, 首先在区间 $[0, 100]$ 的二维正方形区域内, 以均匀分布方式随机生成 50 个回收点的坐标; 50 个回收点坐标在 $[100/3, 200/3]$ 的二维正方形区域内, 以均匀分布方式随机生成, 从而在 $1/9$ 的区域内生成密集的市区配置地理场景。由于这里只考虑了回收问题, 每个回收点的废弃包装回收量 $P_j = (0.5 + r_j) R_j$, 其中 R_j 为均匀分布在 $[0, 100]$ 之间的整数, r_j 为 $[0, 1]$ 之间的随机数。由此, 确定每个算例的车载限额

$Q=\sum P_j/\mu$ 。其中, $\sum P_j$ 表示所有回收点的回收总量, μ 为允许选择的最小车辆数量, 这里选择 μ 为 3 或 8 生成算例。本算例的数据无实际物理意义, 是在 $[0, 100]$ 区间生成的, 因此相应的目标值也无量纲。

关于时间窗的生成, 参照 Solomon 提出的 VRPTW 基准数据生成方法, 时间窗的中间值在区间 $(E_0+t_{0,i}, L_0-t_{i,0})$ 中随机生成, 且时间窗宽度为标准正态分布随机数, 其中 $[E_0, L_0]$ 为回收中心进行回收服务的时间窗, $t_{0,i}$ 和 $t_{i,0}$ 分别表示车辆从回收中心到回收点 i 的行驶时间和该回收点回到回收中心的行驶时间。在模型中假设回收中心时间窗 $[E_0, L_0]$ 为 $[0, 24]$, 所有车辆的发车成本 C_k 为 50, 所有车辆的行驶速度 v_{ijk} 为 60, 提前到达或延期到达回收点的惩罚系数 α 、 β 均为 1。

3.2 算法比较

为了验证 ISIS 算法的求解性能, 与改进的节约里程 (C-W)、分散搜索 (SS) 和遗传 (Genetic Algorithm, GA) 等 3 种算法进行对比。其中, SS 和 GA 的初始解均采用随机初始解, GA 的交叉率 p_c 为 0.6, 变异率 p_m 为 0.1, 种群规模为 100, 运行代数 20 000; SS 和 ISIS 算法的参考集 $|b|=20$, 其中较好解集 $|b_1|=10$, 多样解集 $|b_2|=10$, 种群规模为 100, 运行代数为 100。由于 GA 与 SS 算法在迭代方式上存在不同, 2 种算法无法基于相同的迭代次数进行比较, 因此将 2 种算法在种群规模相同且运算时间大致相同的前提下对不同规模的算例进行比较。对比了 3 种规模下, 最小车辆数量 μ 分别为 3 或 8 的各 10 组数据目标值平均计算结果, 如图 2 所示。

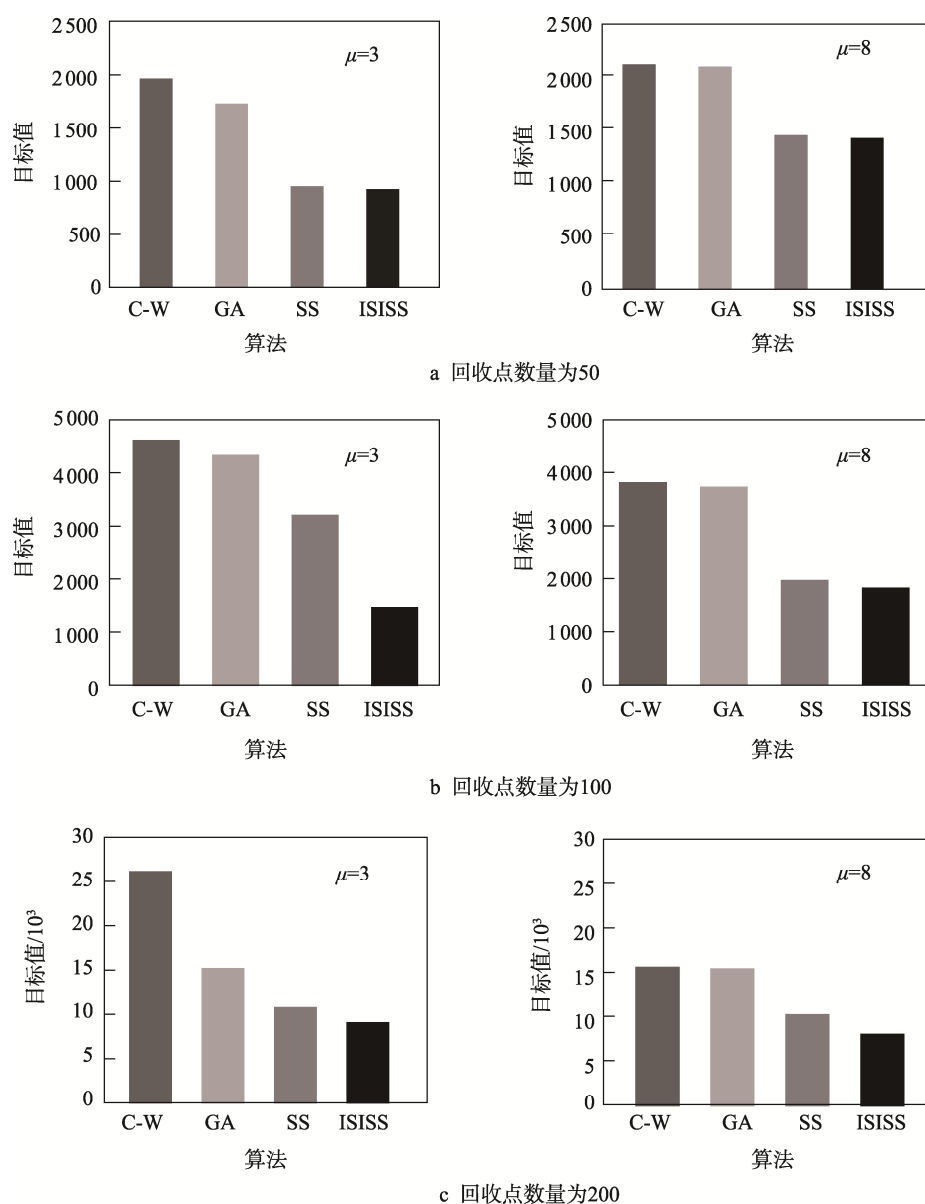


图2 3种规模下目标值的比较

Fig.2 Comparison of the objective values of algorithms under three scales

由图 2 可知, 相对于改进 C-W 算法生成的较好解, 其他 3 种进化算法的求解优势明显。当回收点数量为 50 时, GA 求得的目标值仅略优于 C-W 算法, 而 SS 和 ISISS 算法求得的目标值明显优于 C-W, 说明在相同的数据规模和计算复杂度下, ISISS 算法能够充分发挥参考集更新和解改进等策略的优势, 但 ISISS 相对于随机初始解 SS 的改进效果并不显著。当回收点数量扩大为 100 时, 选择较大的车型 ($\mu=3$) 时, ISISS 相对于 GA 的改进幅度达到了 65.8%, 相对于 SS 的改进率也很明显; 选择较小车型 ($\mu=8$) 时, ISISS 相对于 SS 的改进效果仍不够显著。随着数据规模的进一步扩大, 当回收点为 200 时, ISISS 算法在 2 种车型选择下的改进效果才显示出来。由于 ISISS 在初始解生成时就采用 C-W 启发式算法生成了较好的初始解, 并在此基础上进行了多样化处理, 因此 ISISS 具有更优的多样性解和较好解, 从而增强了算法的寻优能力, 更加适合于求解大规模回收车辆路径问题。

另一方面, 从 4 种算法得出的发车数量 (表 1) 比较结果可以看出, 在不同规模下, C-W 算法都可以得到较少的发车数量, 但这是以高额的时间窗惩罚和

回收成本为代价, ISISS 算法可以在同时考虑回收成本和时间窗惩罚的基础上, 将回收中心总发车数量控制在一个较小的数量上。当回收点为 50、100 时, ISISS 得到的发车数量 (4、9) 仅比对应的最小车辆数 (3、8) 多 1 辆车, 而随机初始解的 GA 和 SS 都无法达到这个发车数量, 说明对于同一个回收路径规划问题, ISISS 可以更好地发挥已派出车辆的装载率, 从而降低总用车数量。

文中将时间窗惩罚引入优化目标, 而非单纯采用运输成本。表 2 列出了 GA、SS、ISSS 等 3 种算法在不同数据规模下 3 个具体算例 C50-3、C100-3、C200-3 的目标值分项。数据表明, 相对于随机初始解的 GA、SS 算法, ISISS 算法能够得到更优的可行解, 找到了满足约束且在不同的目标上表现更好的解。当回收点数量较小时, ISISS 在运输成本和提前拖期惩罚 2 项目标值上比随机初始解 SS 略有改进, 但并不明显; 当回收点数量增至 200 时, ISISS 除了能达到更好的运输成本, 同时还能尽可能地满足不同回收点服务时间窗的要求, 大幅降低了时间窗提前拖期惩罚目标, 并减少了发车数量, 降低了发车成本。

表 1 发车数量比较
Tab.1 Comparison of the number of departures

回收点 数量	最小 车辆数量	C-W		GA		SS		ISSS	
		目标值	车辆数量	目标值	车辆数量	目标值	车辆数量	目标值	车辆数量
50	3	1 956.78	4.0	1 721.50	4.0	943.63	4.0	918.25	4.0
	8	2 055.10	10.0	2 034.15	9.2	1 406.42	9.0	1 382.04	9.0
100	3	4 727.86	4.0	4 430.29	5.0	3 285.76	5.0	1 513.10	4.0
	8	3 902.36	9.4	3 813.38	9.0	2 016.42	9.0	1 866.99	9.0
200	3	27 076.12	4.0	15 823.90	16.0	11 235.01	12.9	9 471.33	10.9
	8	16 153.20	9.2	15 971.10	16.0	10 655.58	13.3	8 420.66	11.0

表 2 基于算例 C50-3、C100-3、C200-3 的 3 种算法目标值分项比较
Tab.2 Itemized comparison of the objective values based on C50-3, C100-3 and C200-3

客户点	算法	总目标值	运输成本	提前拖期惩罚	发车成本	改进幅度/%
50	GA	1 806.62	1 298.65	307.98	200.00	
	SS	1 015.87	672.52	143.35	200.00	43.77
	ISSS	961.45	636.13	125.33	200.00	46.78
100	GA	4 576.08	3 255.88	1 070.20	250.00	
	SS	3 464.23	2 532.19	682.04	250.00	24.30
	ISSS	1 569.71	1 010.88	358.83	200.00	65.70
200	GA	15 525.69	12 271.27	2 454.42	800.00	
	SS	10 792.97	8 269.71	1 898.26	650.00	30.48
	ISSS	9 518.19	7 178.93	1 772.60	625.00	38.69

3.3 路线明细

在回收点为 100、最小车辆数量 μ 为 3 时, 算例 C100-3 的一次路径规划明细如表 3 所示, 对应的回收路径规划如图 3 所示。

表 3 算例 C100-3 路径排序
Tab.3 Route sorting of C100-3

编号	客户 点数	装载率/%	路线规划
1	36	98.68	0—66—74—61—63—78—92—5—
			4—84—69—77—80—86—96—
			99—1—82—57—52—39—32—4—
			7—14—22—41—38—43—17—
2	30	99.06	27—25—45—31—48—34—50—
			29—53—0
			0—64—71—83—76—59—75—1—
			2—56—55—65—10—60—89—
3	33	99.61	6—11—19—46—40—36—7—3—
			7—23—21—9—28—33—72—1—
			6—81—98—0
			0—73—91—67—68—100—70—
4	1	2.70	85—58—93—88—62—20—44—
			4—26—3—2—5—24—42—49—
			15—8—13—35—30—95—51—9—
			4—97—87—18—79—0

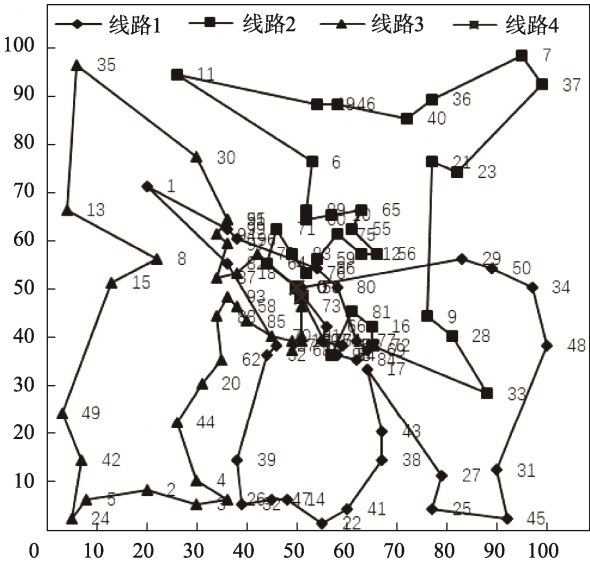


图 3 ISISS 求得算例 C100-3 的一个较好解
Fig.3 A better solution of C100-3 by ISISS

随着回收点数量的增多, ISISS 算法仍然可以将规划车辆数控制在 4 辆车, 前 3 辆车的车辆负载都尽可能充分利用, 基本接近车载上限, 且从路径规划(图 3)可以看出, 在“部分回收点分布较密集”的地理场景下, ISISS 算法能够在一辆车的路径规划上兼顾市区密集部分点和郊区零散分布点, 各条路径彼此重叠覆盖的区域较少, 且各条路线上的交叉回路较少。在

相同的回收点分布下, 选择较小的车型 ($\mu=8$) 时, 算例 C100-8 的路径规划如图 4 所示, 对应的路径排序如表 4 所示。

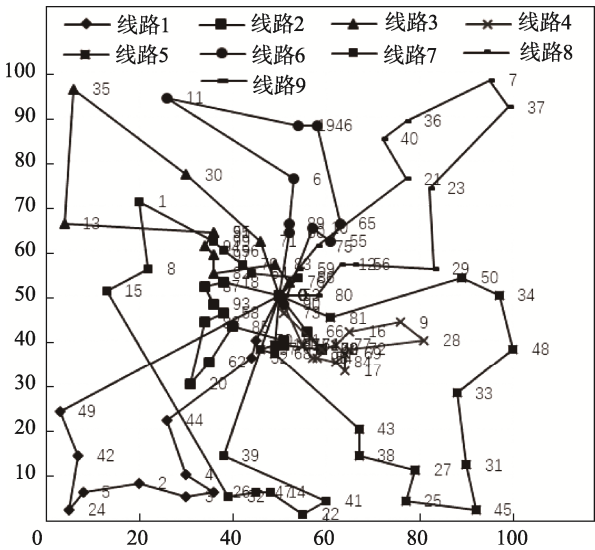


图 4 ISISS 求得算例 C100-8 的一个较好解
Fig.4 A better solution of C100-8 by ISISS

表 4 算例 C100-8 路径排序
Tab.4 Route sorting of C100-8

编号	客户 点数	装载率/%	路线
1	11	99.85	0—70—52—44—4—26—3—
			2—5—24—42—49—0
2	13	98.53	0—66—92—78—67—91—
			85—62—20—88—58—93—
			87—18—0
3	11	99.56	0—76—83—82—97—94—9—
			5—51—13—35—30—71—0
4	14	96.04	0—74—61—98—54—84—
			17—69—72—28—9—16—
			77—63—73—0
5	14	98.68	0—100—57—68—43—38—
			27—25—45—31—33—48—
			34—50—81—0
6	10	94.57	0—90—10—55—65—46—
			19—11—6—89—60—0
7	14	99.41	0—86—64—79—96—99—
			1—8—15—32—47—14—2—
			2—41—39—0
8	12	96.92	0—80—12—56—29—23—
			37—7—36—40—21—75—
9	1	16.59	59—0
			0—53—0

4 结语

从废弃包装品回收的实际问题出发,研究了逆向物流车辆路径规划问题,将软时间窗约束转化为逆向物流车辆路径问题的优化目标,以最小化运输成本、发车成本和时间窗惩罚成本为目标,建立了带回路的车辆路径问题联合优化数学模型。采用含 0 的整数编码,同时考虑车辆旅行时间节约值和车辆剩余回收空间收益两方面因素,确定节约值,利用改进的节约里程算法得到一个初始解,并设计初始解多样化策略,生成初始解集合。基于改进的优质多样性初始解,利用分散搜索进化算法求解。针对“部分回收点分布较密集”的类似市区型地理场景生成多组算例,在不同的数据规模下验证算法的性能。下一步将对 ISSS 算法在同时送取货的回收场景及考虑回收服务时间的路径规划问题进行研究。

参考文献:

- [1] GAO Z H, YE C Y. Reverse Logistics Vehicle Routing Optimization Problem Based on Multivehicle Recycling[J]. Mathematical Problems in Engineering, 2021, 2021: 5559684.
- [2] ALIZADEH FOROUTAN R, REZAEIAN J, MAHDAVI I. Green Vehicle Routing and Scheduling Problem with Heterogeneous Fleet Including Reverse Logistics in the Form of Collecting Returned Goods[J]. Applied Soft Computing, 2020, 94: 106462.
- [3] HAN Y. Evolutionary Algorithms for the Heterogeneous Fleet Green Vehicle Routing Problem with Reverse Logistics Characteristics[J]. Journal of the Korean Society of Supply Chain Management, 2017, 17(2): 133-143.
- [4] ANILY S, MOSHEIOV G. The Traveling Salesman Problem with Delivery and Backhauls[J]. Operations Research Letters, 1994, 16(1): 11-18.
- [5] 张异. 包装废弃物回收车辆路径问题的改进遗传算法[J]. 包装工程, 2018, 39(17): 147-152.
ZHANG Y. Improved Genetic Algorithm for Vehicle Routing Problem in Packaging Waste Recycling[J]. Packaging Engineering, 2018, 39(17): 147-152.
- [6] 陈秀娟, 高更君, 冯媛媛. 基于改进蚁群算法的逆向物流车辆路径优化[J]. 制造业自动化, 2019, 41(5): 46-49.
CHEN X J, GAO G J, FENG Y Y. Optimization of Reverse Logistics Vehicle Path Based on Improved Ant Colony Algorithm[J]. Manufacturing Automation, 2019, 41(5): 46-49.
- [7] 赵尧, 陈志华. 基于局部搜索的逆向物流车辆最短路优化[J]. 计算机仿真, 2022, 39(11): 215-219.
ZHAO R, CHEN Z H. Shortest Path Optimization of Reverse Logistics Vehicle Based on Local Search[J]. Computer Simulation, 2022, 39(11): 215-219.
- [8] 杨玮, 赵晶, 张堃, 等. 基于货架寿命的乳制品配送车辆路径优化研究[J]. 包装工程, 2019, 40(11): 72-79.
YANG W, ZHAO J, ZHANG K, et al. Path Optimization of Dairy Distribution Vehicle Based on Shelf Life[J]. Packaging Engineering, 2019, 40(11): 72-79.
- [9] GLOVER F. A Template for Scatter Search and Path Relinking[M]// Lecture Notes in Computer Science. Berlin: Springer Berlin Heidelberg, 1998: 1-51.
- [10] KALRA M, TYAGI S, KUMAR V, et al. A Comprehensive Review on Scatter Search: Techniques, Applications, and Challenges[J]. Mathematical Problems in Engineering, 2021, 2021: 5588486.
- [11] 毛丽丽, 张则强, 朱立夏. 混合模拟退火及分散搜索优化过道布置问题[J]. 计算机工程与应用, 2018, 54(3): 243-249.
MAO L L, ZHANG Z Q, ZHU L X. Hybrid Scatter Search Algorithm with Simulated Annealing for Corridor Allocation Problem[J]. Computer Engineering and Applications, 2018, 54(3): 243-249.
- [12] 范佳静, 曹玉华, 曹敏. 基于改进分散搜索算法的多资源跨单元调度问题研究[J]. 中国机械工程, 2017, 28(22): 2722-2731.
FAN J J, CAO Y H, CAO M. Study on Multi-Resource Intercellular Scheduling Problem Based on ASS Algorithm[J]. China Mechanical Engineering, 2017, 28(22): 2722-2731.
- [13] 张涛, 余焯娅, 刘岚, 等. 同时送取货的随机旅行时间车辆路径问题方法[J]. 系统工程理论与实践, 2011, 31(10): 1912-1920.
ZHANG T, YU C Y, LIU L, et al. Method for the Stochastic Traveling Time VRPSPD Problem[J]. Systems Engineering-Theory & Practice, 2011, 31(10): 1912-1920.
- [14] 张鹏威, 李英, 成琪. 有限充电设施下的多配送中心电动车辆路径问题研究[J]. 工业工程与管理, 2019, 24(5): 97-105.
ZHANG P W, LI Y, CHENG Q. Multi-Depot Electric Vehicle Routing Problem under Limited Recharging Infrastructures[J]. Industrial Engineering and Management, 2019, 24(5): 97-105.
- [15] EUCHI J. Hybrid Adaptive Memory Programming to Optimize the Multi-Commodity many to many Vehicle Routing Problem[J]. International Journal of Mathematics in Operational Research, 2020, 1(1): 1.
- [16] CLARKE G, WRIGHT J W. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points[J]. Operations Research, 1964, 12(4): 568-581.
- [17] DETHLOFF J. Vehicle Routing and Reverse Logistics: The Vehicle Routing Problem with Simultaneous Delivery and Pick-up[J]. OR-Spektrum, 2001, 23(1): 79-96.
- [18] SOLOMON M M. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints[J]. Operations Research, 1987, 35(2): 254-265.