

本文引用格式: 耿永忠,黄秋原,余太松,等.面向物流车辆路径规划的改进粒子群优化算法[J].自动化与信息工程,2025,46(2):25-31;62.

GENG Yongzhong, HUANG Qiuyuan, YU Taisong, et al. Improved particle swarm optimization algorithm for logistics vehicle routing planning[J]. Automation & Information Engineering, 2025,46(2):25-31;62.

面向物流车辆路径规划的改进粒子群优化算法

耿永忠 黄秋原 余太松 刘家鹏

(国药控股上海生物医药有限公司, 上海 200000)

摘要: 为优化物流车辆的配送路径, 提高配送效率, 降低运营成本, 提出一种面向物流车辆路径规划的改进粒子群优化 (PSO) 算法。该算法采用贪婪算法初始化解, 并利用基于迭代过程的粒子交叉更新算法实现搜索范围和搜索速度的平衡, 使求解方案的总路径长度更短。相较于遗传算法和经典的 PSO 算法, 该算法规划的总路径长度、适应度值、计算耗时均更小。在实际规划任务中, 该算法比人工调度规划更合理。

关键词: 物流车辆路径规划; 改进的粒子群优化算法; 贪婪算法; 粒子交叉更新算法

中图分类号: TP301.6

文献标志码: A

文章编号: 1674-2605(2025)02-0004-08

DOI: 10.12475/aie.20250204

开放获取

Improved Particle Swarm Optimization Algorithm for Logistics Vehicle Routing Planning

GENG Yongzhong HUANG Qiuyuan YU Taisong LIU Jiapeng

(Sinopharm Group Shanghai Biopharmaceutical Co., Ltd., Shanghai 200000, China)

Abstract: To optimize the logistics vehicle routing planning, improve delivery efficiency, and reduce operational costs, an improved particle swarm optimization (PSO) algorithm for logistics vehicle routing planning is proposed. This algorithm employs a greedy algorithm to initialize the solutions and utilizes a particle crossover update algorithm based on the iterative process to balance the search scope and search speed, thereby achieving shorter total path lengths for the solutions. Compared to genetic algorithms and classical PSO algorithms, this algorithm exhibits smaller total path lengths, lower fitness values, and reduced computational time. In practical routing tasks, the algorithm generates more rational planning results than manual scheduling.

Keywords: logistics vehicle routing planning; improved particle swarm optimization algorithm; greedy algorithm; particle crossover update algorithm

0 引言

车辆路径问题 (vehicle routing problem, VRP) 是运筹学和物流管理中的经典组合优化问题, 旨在有限资源 (如车辆数量、载重、时间窗) 约束下, 设计最优的车辆调度方案, 以最小化成本 (如总行驶距离、时间), 同时满足客户需求, 具有搜索规模大、非线性计算等特点。因此, 研究 VRP 对提高物流效率、降低配送成本和提供更优质的服务至关重要。

目前, VRP 的求解方法主要分为 3 类。第一类为

精确算法, 主要包含分支定界法^[1]、穷举法、动态规划算法^[2]和近似算法等。文献[3]提出分支代价和消减算法, 用于求解带时间窗的 VRP, 减少了计算复杂度, 但求解速度较慢, 适用性有限。文献[4]结合遗传算法和动态规划算法求解 VRP, 利用动态规划算法确定车辆的最优时刻表, 减少了计算时间, 提高了运算效率, 但动态规划算法没有统一的标准模型, 且求解时存在维数灾难。精确算法过程点数量较多, 求解时间较长, 适用于小规模 VRP, 应用范围受限。第二类为深度

学习方法, 其将强化学习与神经网络相结合^[5-8]。文献[9]提出一种基于深度强化学习和约束规划组合的模型, 用于求解带时间窗的旅行商问题, 通过强化学习训练问题实例, 利用约束规划评估问题实例; 经实验证明, 该模型的效果与主流求解器 OR-Tools 相当。深度学习方法的通用性、条件可扩展性较强, 但存在模型泛化能力差、训练时间长、收敛性能差等问题。第三类为启发式算法, 包括禁忌搜索算法^[10]、模拟退火算法^[11]、蚁群算法^[12]、粒子群优化 (particle swarm optimization, PSO) 算法^[13]和遗传算法^[14]等。文献[15]利用混合变邻域禁忌搜索算法求解带硬时间窗的 VRP, 具有良好的收敛性和稳定性, 获得了更短的车辆总行驶距离, 但求解过程较为复杂。文献[16]采用经典的 PSO 算法求解农产品运输中带时间窗的 VRP, 但该算法易陷入局部最优解, 且对参数敏感, 导致其求解性能有限。文献[17]提出基于混沌遗传模拟退火算法的配送路径优化模型, 综合考虑了客户需求拆分、多配送中心和车辆就近返回等因素; 实验结果表明, 该模型可有效求解不同规模和类型的 VPR, 降低了配送成本。文献[18]提出一种基于聚类算法和 PSO 算法的改进多配送中心车辆路径规划方法, 先利用聚类算法将多配送中心问题降维为单配送中心问题, 再利用 PSO 算法求解单配送中心多路径规划的最优解; 实验结果表明, 该方法的收敛速度比经典的 PSO 算法提升了 n (配送中心个数) 倍。启发式算法在大规模过程点条件下能够求解近似最优解, 但不同启发式算法的求解过程特点各异, 如模拟退火算法的参数选择对算法性能影响较大; 蚁群算法的参数调整较为复杂; 遗传算法每次迭代计算复杂且迭代步数较多, 导致计算时间较长; PSO 算法在搜索过程中依赖参数选择, 且对初始解的质量较为敏感。

在求解物流的 VRP 过程中, 针对经典的 PSO 算法易陷入局部最优解、收敛速度慢和对参数敏感性不足等问题, 本文提出一种改进的 PSO 算法。该算法融合了贪婪算法和 PSO 算法的优点, 并在迭代过程中增加了动态粒子交叉更新操作, 从而降低求解方案的总路径长度。

1 模型构建

1.1 问题描述

某些医疗药品需要冷链物流配送, 时效性要求较高, 且配送需求随机, 可以归类于 VRP。

1.2 数学模型

VRP 描述为从一个配送中心出发, 将货物运送到指定的配送点, 再返回配送中心, 且车队中每辆车都是相同的, 每个配送点只能被一辆车服务一次。VRP 的目标是找到车辆的最佳路线, 以最小化任务成本, 具有如下特征:

- 1) 方向, 单向配送;
- 2) 单配送中心, 只有一个配送中心;
- 3) 单车型, 只考虑一种车型;
- 4) 需求不可拆分, 只有一辆车满足配送需求;
- 5) 车辆封闭, 完成配送任务后, 车辆返回配送中心;
- 6) 车辆充足, 不限制车辆数量, 即配送车辆需求均能得到满足;
- 7) 不考虑装载量, 配送需求远小于车辆的最大载重。

VRP 数学模型的符号说明如表 1 所示。

表 1 VRP 数学模型的符号说明

符号	说明
C_0, C_1	车辆启用的固定成本、单位距离的行驶成本
N	配送点的数量, 索引为 i, j
n	配送集合 ($n = 0, 1, 2, \dots, N$), 其中 0 表示配送中心; $1, 2, \dots, N$ 表示配送点
D	车辆行驶的最大距离
K	需要的车辆数量
m	车辆集合 ($m = 1, 2, 3, \dots, K$)
d_{ij}	配送点 $i(x_i, y_i)$ 和 $j(x_j, y_j)$ 的欧氏距离

决策变量为

$$x_{ijk} = \begin{cases} 1, & \text{表示第 } k \text{ 辆车从 } i \text{ 行驶到 } j \\ 0, & \text{表示第 } k \text{ 辆车不从 } i \text{ 行驶到 } j \end{cases} \quad (1)$$

式中： x_{ijk} 为第 k 辆车从 i 到 j 的行驶距离。

目标函数为

$$\min Z = C_0 K + C_1 \sum_{j=0}^N \sum_{i=0}^N \sum_{k=0}^K d_{ij} x_{ijk} \quad (2)$$

式中： Z 为车辆行驶总成本。

约束条件为

1) 配送中心约束, 所有车辆均从配送中心出发, 完成配送任务后返回配送中心, 即满足公式(3):

$$\sum_{k=1}^K \sum_{j=1}^N x_{0,jk} = \sum_{k=1}^K \sum_{j=1}^N x_{j,0k} = K \quad (3)$$

2) 配送点流量平衡约束, 进出车辆数量相等, 即满足公式(4):

$$\sum_{i=1}^N x_{ijk} = \sum_{i=1}^N x_{jik} \quad (k \in m, \forall j = 1, 2, \dots, N) \quad (4)$$

3) 配送点服务约束, 每个配送点被服务 1 次, 即满足公式(5):

$$\sum_{k=1}^K \sum_{i=0}^N x_{ijk} = 1 \quad (j = 1, 2, \dots, N) \quad (5)$$

4) 车辆行驶距离约束, 每辆车配送距离不超过最大配送距离, 即满足公式(6):

$$\sum_{i=0}^N \sum_{j=0}^N x_{ijk} d_{ij} \leq D \quad (k \in m) \quad (6)$$

5) 所需车辆数量约束, 即满足公式(7):

$$K \geq \left\lceil \frac{\sum_{j=1}^N m_j}{M} \right\rceil \quad (k \in N^*) \quad (7)$$

2 PSO 算法

2.1 经典的 PSO 算法

经典的 PSO 算法是一种模拟鸟群觅食行为的优化算法, 其将潜在解视为搜索空间中的点, 称为粒子。每个粒子都有一个由目标函数确定的适应度值。另外, 每个粒子还有速度, 其决定了粒子在搜索空间中的移动方向和距离。在经典的 PSO 算法中, 粒子通过追随当前最优的粒子, 在解空间中寻找最优解。经典的 PSO 算法流程如图 1 所示。

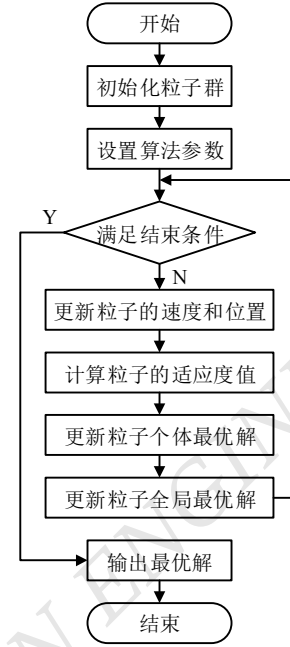


图 1 经典的 PSO 算法流程图

如图 1 所示, 经典的 PSO 算法核心为粒子的速度和位置更新, 其中, 位置更新由速度更新决定; 速度更新由上一时刻的速度惯性、个体认知部分和全局认知部分决定, 速度更新公式为

$$v_i(d) = w \cdot v_i(d-1) + s_1 \cdot r_1 \cdot (pbest_i(d) - x_i(d)) + s_2 \cdot r_2 \cdot (gbest_i(d) - x_i(d)) \quad (8)$$

式中： v_i 为第 i 个粒子的速度； x_i 为第 i 个粒子的位置； w 为粒子速度的惯性权重； d 为迭代次数； r_1 、 r_2 为随机系数； s_1 为粒子的个体学习因子，也称个体加速因子； s_2 为粒子的全局学习因子，也称全局加速因子； $pbest_i(d)$ 为到第 d 次迭代为止，第 i 个粒子经过的个体历史最优位置； $gbest_i(d)$ 为到第 d 次迭代为止，第 i 个粒子经过的全局历史最优位置。

经典的 PSO 算法位置更新公式为

$$x_i(d) = x_i(d-1) + v_i(d-1) \cdot t \quad (9)$$

经典的 PSO 算法在求解 VRP 过程中, 因初始解的质量不确定, 导致收敛过程不稳定; 当解空间较大时, 存在收敛速度慢、搜索效率低、对参数敏感等问题。

2.2 PSO 算法的改进策略

在经典的 PSO 算法中, 初始解的质量对最终解的影响较大。而贪婪算法虽然计算速度较快, 但容易达到局部最优解。将贪婪算法与 PSO 算法结合使用, 通过贪婪算法提供快速且相对优质的初始解, 使 PSO 算法更有效地搜索全局最优解, 实现解的质量和算法运行效率之间的良好平衡。

为了使粒子在搜索过程中能够有效迭代, 不受超参数的影响, 本文提出一种基于迭代过程的粒子交叉更新算法。该算法对粒子位置进行编码, 并根据迭代次数动态更新编码方式, 实现了搜索范围和搜索速度的平衡。

在基于迭代过程的粒子交叉更新算法中, 粒子位置编码更新分为前半部分和后半部分。

1) 前半部分更新, 由上一代粒子位置编码的前半部分决定, 记为操作 A。该操作可继承上一代的解决方案, 避免在搜索过程中出现较大的突变。

2) 后半部分更新, 根据迭代轮数与最大迭代次数的比率, 动态选择以下 3 种操作之一:

① 当迭代轮数与最大迭代次数的比率 < 0.1 时, 后半部分编码为前半部分编码的逆序, 记为操作 B; 该操作结合贪婪算法, 在搜索初期可获得更广阔的搜索范围;

② 当 $0.1 \leq$ 迭代轮数与最大迭代次数的比率 < 0.3 时, 后半部分编码为历史个体最优解, 记为操作 C; 该操作可继承个体最优解, 加快搜索的收敛速度;

③ 当迭代轮数与最大迭代次数的比率 ≥ 0.3 时, 后半部分编码为历史全局最优解, 记为操作 D; 该操作可使不同粒子的个体最优解影响全局解, 提升了整体的搜索效果。

基于迭代过程的粒子交叉更新算法的伪代码如算法 1 所示。

算法 1: 基于迭代过程的粒子交叉更新算法

输入: bird, pbest, gbest, iterI, iterMax

输出: nextbird

```
1. nextbird = []
2. parent1 = bird 的前部分 # 选择 parent1
3. 如果 iterI / iterMax  $\leq 0.1$ :
4.     parent2 = parent1 的逆序
5. 如果 iterI / iterMax  $\leq 0.3$ :
6.     parent2 = pbest
7. 否则:
8.     parent2 = gbest
9. nextbird = parent1 + parent2
10. 返回 nextbird
```

2.3 算法设计

面向物流车辆路径规划的改进 PSO 算法主要利用贪婪算法生成初始解, 通过基于迭代过程的粒子交叉更新算法实现粒子的有效迭代, 流程如图 2 所示。

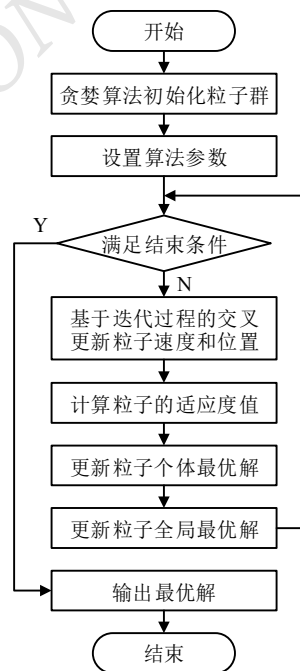


图 2 面向物流车辆路径规划的改进 PSO 算法流程图

3 实验

3.1 仿真对比实验

为验证本文提出的面向物流车辆路径规划的改进 PSO 算法的有效性, 将本文算法与遗传算法、经典的 PSO 算法进行仿真对比实验。实验采用的计算机配置为: Windows 10 操作系统, Intel Core i7-12700-CPU, 英伟达 RTX3060 显卡, 16 GB 内存, Python 编

程语言。实验场景包含 31 个配送点和 1 个中心点，31 个配送点分布在 $100\text{ km} \times 100\text{ km}$ 的区域内，中心点坐标为 $(50\text{ km}, 50\text{ km})$ ，配送车辆为 3 辆。配送车辆从中心点出发，根据规划好的路径依次前往各个配送点，最终返回中心点，各配送点的坐标已知。

本文算法的性能评估指标为总路径长度、适应度值和计算耗时。其中，适应度值=单位距离行驶成本 $\times$ 路线行驶总距离+启动成本 $\times$ 路线总数量。适应度值越低，说明规划的路径越短，算法性能更优。算法迭代次数为 1500 次，根据遗传算法和 PSO 算法的收敛区间选定。

遗传算法、经典的 PSO 算法、本文算法的实验结果分别如图 3、4、5 所示，不同线段表示不同车辆的规划路径。

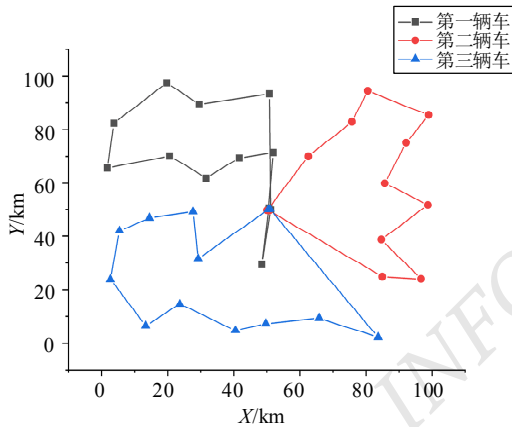


图 3 遗传算法路径规划图

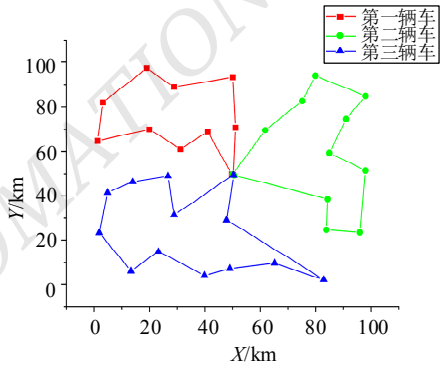


图 4 经典的 PSO 算法路径规划图

对比图 3~5 可知，本文算法相较于 PSO 算法、遗传算法，规划的路径点分布更均匀。

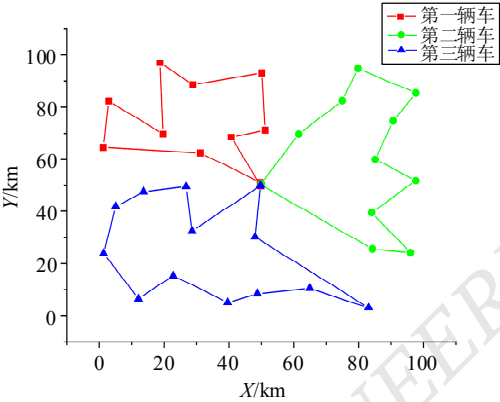


图 5 本文算法路径规划图

为了验证 3 种算法的性能，绘制其路径规划过程中的适应度值变化曲线，如图 6 所示。

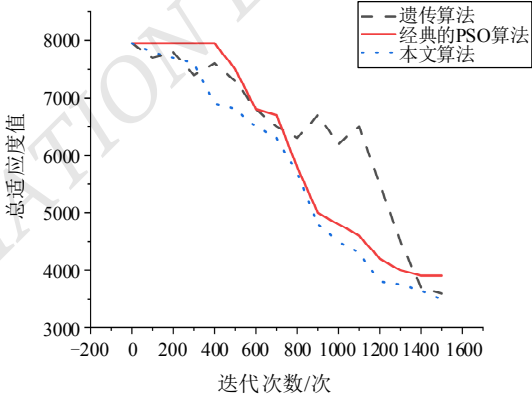


图 6 3 种算法的适应度值变化曲线对比图

由图 6 可知：遗传算法的适应度值呈现波动，最终适应度值高于本文算法；本文算法比经典的 PSO 算法收敛速度更快，且最终适应度值最低，表明本文算法的性能更优。

将 3 种算法的仿真对比实验重复 10 次，性能评估指标（总路径长度、适应度值和计算耗时）的平均值如表 2 所示。

表 2 3 种算法性能评估指标对比表

性能评估指标	遗传算法	经典的 PSO 算法	本文算法
总路径长度/m	958	975	946
适应度值	781	785	768
计算耗时/s	45.8	32.3	24.2

由表2可知,本文算法的总路径长度、适应度值、计算耗时均低于遗传算法和经典的 PSO 算法,验证了本文算法的有效性。

3.2 实际案例测试

为了进一步验证本文算法的实际应用效果,将其与人工调度规划进行比较,配送点位置如表 3 所示。

表 3 配送点位置

序号	收货单位	经度/(°)	纬度/(°)
1	上海市静安区疾病预防控制中心	121.615	31.031
2	上海市廊下社区卫生服务中心	121.194	30.787
3	上海市徐汇区中心医院	121.458	31.210
4	上海市静安寺社区卫生服务中心	121.444	31.228
5	上海市金山工业区社区卫生服务中心	121.345	30.851
6	上海市张堰社区卫生服务中心	121.420	30.792
7	上海市嘉定区外冈镇社区卫生服务中心	121.170	31.364
8	上海市亭林社区卫生服务中心	121.314	30.880
9	上海市枫泾社区卫生服务中心	121.021	30.891
10	上海市嘉定区安亭镇社区卫生服务中心 A 站	121.148	31.306
11	上海市嘉定区江桥镇金沙新城社区卫生服务中心	121.311	31.240
12	上海市嘉定区妇幼保健院	121.247	31.349
13	上海市嘉定区迎园医院	121.270	31.379
14	上海市嘉定区马陆镇社区卫生服务中心	121.299	31.376
15	上海市嘉定区南翔医院	121.322	31.291
16	上海市山阳社区卫生服务中心	121.371	30.766
17	上海市嘉定区安亭镇黄渡社区卫生服务中心	121.218	31.270
18	上海市石化社区卫生服务中心	121.350	30.711
19	上海市嘉定区江桥镇社区卫生服务中心	121.294	31.271
20	上海市嘉定中医医院	121.250	31.387
21	上海市嘉定区真新街道社区卫生服务中心	121.361	31.244
22	上海市金山卫社区卫生服务中心	121.313	30.725

续表

序号	收货单位	经度/(°)	纬度/(°)
23	上海市南翔镇社区卫生服务中心星城分中心	121.314	31.307
24	上海市嘉定区安亭镇社区卫生服务中心 B 站	121.213	31.322
25	上海市临汾路街道社区卫生服务中心	121.457	31.310
26	上海市朱泾社区卫生服务中心	121.169	30.891
27	上海市嘉定区南翔镇社区卫生服务中心	121.311	31.291
28	上海市嘉定区嘉定镇街道社区卫生服务中心	121.248	31.387
29	上海市漕泾社区卫生服务中心	121.423	30.793
30	上海市吕巷社区卫生服务中心	121.174	30.829
31	上海市嘉定区华亭镇社区卫生服务中心	121.238	31.465
32	上海市金山区漕廊公路	121.339	30.790

根据以往经验,人工调度规划偏向贪婪策略,难以进行长距离规划,容易导致车辆路径规划不平衡(部分车辆的路径较长,部分车辆的路径较短)或路径规划相对平衡,但总路径长度增加。以 2023 年的一次配送任务为例,人工调度规划路径图如图 7 所示,人工调度规划的总路径长度为 885.5 km。

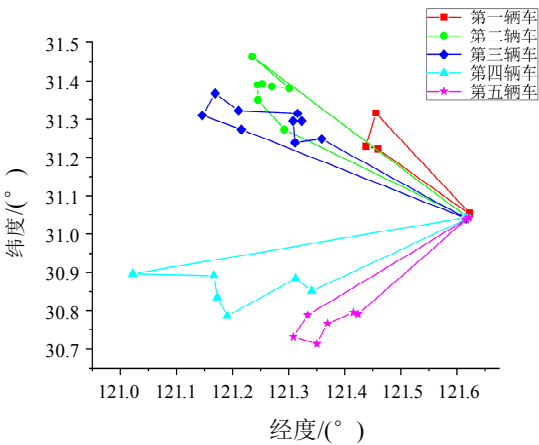


图 7 人工调度规划路径图

采用本文算法规划同样的任务,路径规划图如图 8 所示,本文算法规划的总路径长度为 843.8 km。

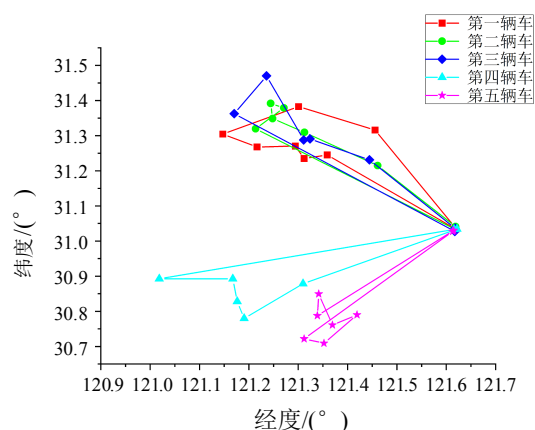


图8 本文算法路径规划图

人工调度与本文算法的路径规划对比效果如图9所示。

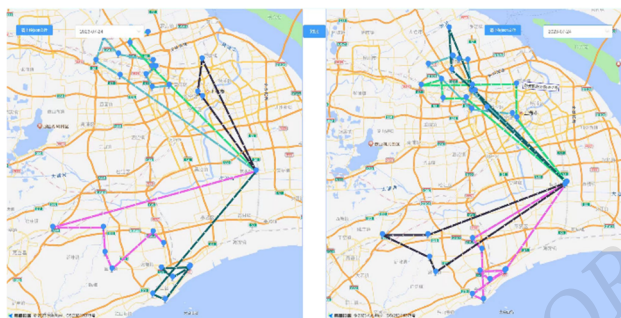


图9 人工调度(左)与本文算法(右)的路径规划对比效果图

由图9可知,本文算法规划的配送点为5个,人工调度规划的配送点为3个,表明本文算法规划的路径更加平均合理。

4 结论

为提高物流配送效率,降低运营成本,本文提出面向物流车辆路径规划的改进 PSO 算法,用于冷藏物流车辆的路径规划。该算法结合了贪婪算法、经典的 PSO 算法和基于迭代过程的粒子交叉更新算法,相较于遗传算法和经典的 PSO 算法,总路径长度更短且计算耗时更少;相较于人工调度规划,规划的路径分布更均衡,不仅减少了人员的工作量,还可作为下游任务安排的依据,使工作流程更高效。下一步将针对更大规模的配送点进行优化,并对算法和调度规划系统进行整合。

©The author(s) 2024. This is an open access article under the CC BY-NC-ND 4.0 License (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

参考文献

- [1] LAWLER E L, WOOD D E. Branch-and-bound methods: A survey[J]. Operations Research, 1966, 14(4): 699-719.
- [2] BELLMAN R. Dynamic programming[J]. Science, 1966, 153(3731): 34-37.
- [3] PECIN D, CONTARDO C, DESAULNIERS G, et al. New enhancements for the exact solution of the vehicle routing problem with time windows[J]. Informatics Journal on Computing, 2017, 29(3): 489-502.
- [4] XIAO Y, KONAK A. A genetic algorithm with exact dynamic programming for the green vehicle routing & scheduling problem[J]. Journal of Cleaner Production, 2016, 167: 1450-1463.
- [5] 牛鹏飞, 王晓峰, 芦磊, 等. 强化学习在车辆路径问题中的研究综述[J]. 计算机工程与应用, 2022, 58(1): 41-55.
- [6] 黄琰, 张锦. 基于深度强化学习的车辆路径问题求解方法[J]. 交通运输工程与信息学报, 2022, 20(3): 114-127.
- [7] 杨笑笑, 柯琳, 陈智斌. 深度强化学习求解车辆路径问题的研究综述[J]. 计算机工程与应用, 2023, 59(5): 1-13.
- [8] 王扬, 陈智斌, 吴兆蕊, 等. 强化学习求解组合最优化问题的研究综述[J]. 计算机科学与探索, 2022, 16(2): 261-279.
- [9] CAPPART Q, MOISAN T, ROUSSEAU L M, et al. Combining reinforcement learning and constraint programming for combinatorial optimization[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(5): 3677-3687.
- [10] 穆东, 王超, 王胜春, 等. 基于并行模拟退火算法求解时间依赖型车辆路径问题[J]. 计算机集成制造系统, 2015, 21(6): 1626-1636.
- [11] WANG Y, WU Q, GLOVER F. Effective metaheuristic algorithms for the minimum differential dispersion problem[J]. European Journal of Operational Research, 2017, 258(3): 829-843.
- [12] PINTEA C M, CHIRA C, DUMITRESCU D, et al. Sensitive ants in solving the generalized vehicle routing problem[J]. International Journal of Computers Communications and Control, 2011, 6(4): 734-741.
- [13] SUTTON R S. Learning to predict by the methods of temporal differences[J]. Machine Learning, 1988, 3(1): 9-44.

(下转第 62 页)