

基于粒子群算法的满载需求可拆分车辆路径规划

卿东升^{1,2}, 邓巧玲², 李建军^{1†}, 刘帅¹, 刘鑫², 曾素平¹

(1. 中南林业科技大学, 湖南长沙 410000; 2. 湖南应用技术学院, 湖南常德 415000)

摘要: 目的: 为了更加合理地规划车辆配送路径, 尽可能地使用最少的车辆数和最短路径长度来完成整个客户点的配送任务. 方法: 提出一种基于粒子群算法的满载需求可拆分车辆路径 (F-SDVRP) 规划策略, 在配送过程中通过确保任何一辆满载的配送车辆从配送点出发后均已“最优”的配送路径进行配送来达到配送的总路径“最优”要求, 并通过粒子群算法不断优化整个客户点的配送顺序. 结果: 仿真结果表明, 在求解相关客户点配送问题时, 文中的车辆规划策略得到的结果优于参与对比文献中的求解方法, 在配送车辆数相同的情况下, 最大的路径长度减少率达到 8.21%. 此外, 各算例的仿真结果也表明了文中策略的寻优结果稳定. 结论: 粒子群算法可以解决满载需求可拆分车辆路径规划问题, 如何更加合理地拆分客户点的需求及如何提高优化效率仍然是路径优化的难点.

关键词: 需求可拆分; 车辆路径规划; 粒子群算法; 智能优化算法; 配送路径; 满载; 需求可拆分车辆路径规划

中图分类号: TP391.9

文献标志码: A

DOI: 10.13195/j.kzyjc.2019.1323

Split Vehicle Route Planning with Full Load Demand Based on Particle Swarm Optimization

QING Dong-sheng^{1,2}, DENG Qiao-ling², LI Jian-jun^{1†}, LIU Shuai¹, LIU Xin², ZENG Su-ping¹

(1. Central South University of Forestry and Technology, Changsha 410000, China; 2. Hunan Applied Technology University, Changde 415000, China)

Abstract: Objective: In order to more reasonably plan the vehicle distribution path, use the minimum number of vehicles and the shortest path length as possible to complete the distribution task of the entire customer point. Method: Propose a particle swarm algorithm based on a full load demand split vehicle path (F-SDVRP) planning strategy, in the distribution process, by ensuring that any fully-loaded distribution vehicle has “optimal” distribution path for delivery after starting from the distribution point to achieve the “optimal” requirement of the total distribution path, and through The particle swarm optimization algorithm continuously optimizes the distribution order of the entire customer point. Results: The simulation results show that the vehicle planning strategy in this paper is better than the solution method in the participating comparative literature when solving the related customer point distribution problem. With the same number of delivery vehicles, the maximum path length reduction rate reaches 8.21%. In addition, the simulation results of the examples also show that the optimization results of the strategy in this paper are stable. Conclusion: Particle swarm algorithm can solve the problem of split-vehicle path planning with full load requirements How to more reasonably split the needs of customer points and how to improve the optimization efficiency is still the difficult point of path optimization.

Keywords: demand separable; vehicle routing planning; particle swarm optimization; intelligent optimization algorithm; distribution path; full load; demand separable vehicle routing planning

0 引言

车辆路径问题 (VRP) 自提出以来, 一直是路径规划领域的研究难点^[1-2]. 该问题早期的研究主要集中在每个客户的需求量小于车辆最大载重的情况下对车辆的路径进行规划^[3]. 然而, 在实际配送过程中经常会出现客户需求量大于车辆最大载重的情

况, 在大部分情况下需对客户的需求进行拆分, 特别是当前物流行业的高速发展, 对于车辆配送路径的优化也显得越来越迫切^[4-5]. 因而, 需求可拆分的车辆路径问题 (SDVRP) 自提出以来便得到了广泛的关注, 它的特点是允许客户被多次访问^[6]. 当前, 求解需求可拆分的车辆路径规划问题的策略主要集中

收稿日期: 2019-09-20; 修回日期: 2019-12-25.

基金项目: 国家自然科学基金项目 (31570627); 湖南省科技计划项目 (2015WK3017); 湘教通 (2019)90 号 (18B579).

†通讯作者. E-mail: 390033868@qq.com.

在智能优化算法和聚类算法上. 杨鹏等使用改进的蚁群算法解决了带时间窗集送货需求可分车辆路径问题^[7]; 胡钟骏等针对需求可拆分车辆路径优化问题, 并设计了一个改进遗传算法进行求解^[8]. 有相关研究者从聚类的角度上对可拆分车辆路径问题进行了规划^[3,9-10]; 李相勇等提出了一种混合蚁群优化算法来解决开放式车辆路径规划问题^[11]; 姜婷研究了需求可拆分的车辆路径问题 (SDVRP) 的基本数据模型, 分析了相关解的基本特点, 提出了一种改进的人工蜂群算法进行求解^[12]; 孟凡超等通过改进传统的数学模型, 建立改进的可拆分车辆路径问题 (Split Delivery VRP, SDVRP) 数学模型, 并利用禁忌搜索算法 (Taboo Search Algorithm, TSA) 进行求解^[13]. Dror 等提出了局部搜索算法来解决可拆分车辆路径规划问题^[14]; Gendreau M 等提出了一种带广义插入的禁忌搜索算法的解决方案^[15]; 从研究进展来看, 到目前为止, 并未有文献正式探究过车辆在“满载”需求可拆分的情况下最优路径规划问题. 文中的“满载”是指所有车辆在配送过程中最多只有一辆车没有达到车辆的最大承重量. 此外, 有相关文献表明, 当车辆路径规划问题规模逐渐变大时, 利用智能算法求解该问题变得越来越有效, 但如何提高智能算法的优化效率及如何寻找最优路径仍然是当前研究的难点问题^[11-13].

粒子群算法最早是由美国电气工程师 Eberhart 和社会心理学家 Kennedy 在 1995 年根据鸟类觅食的原理提出的一种智能优化算法, 目前该算法已经成功应用于各个领域^[16]. 相比其它智能算法, 粒子群算法容易理解、实现简单, 参数较少^[17-19]. 特别是对于解决多目标规划问题, 粒子群算法有一定的优势^[20-21]. 需求可拆分的车辆路径问题 (SDVRP) 可以看成多目标规划问题中的一种. 为了解决车辆路径规划问题, 文中提出一种基于粒子群算法的满载需求可拆分车辆路径问题 (Full-Split Delivery VRP, F-SDVRP) 规划策略, 探索在整个配送过程中使用最少的车辆数和最短的总路径长度来完成各客户点的配送任务. 该配送策略无需像文献 [3]、文献 [12] 等要先对客户点进行聚类、分组等预处理, 然后再确定要拆分的客户点, 预处理的结果将直接影响最终车辆的配送路径, 特别是当客户数量增多时, 预处理也将变得复杂, 不利于寻找需拆分的客户点. 文中车辆配送策略所有需拆分的客户点均由配送车辆在配送过程中根据粒子群算法自动决定, 无需人工干预, 并最终输出“最优”路径.

1 F-SDVRP 的数学模型

为了简化研究过程, 文中按文献 [3] 和文献 [12] 中问题的设计方法, 研究的是单车场、单车型、无时间窗要求、纯装货 (或者纯卸货) 并且客户需求量可拆分的车辆路径规划问题. F-SDVRP 的数学模型就是在 SDVRP 数学模型的基础上添加“满载”约束条件, 此外, 根据实际客户点之间的距离关系修正了任意三个客户点之间的距离关系不等式. 模型假设:

- (1) 任意两个点之间的距离对称, 即 $d_{ij} = d_{ji}$.
- (2) 任意三个客户点之间的距离满足 $d_{kj} + d_{ik} \geq d_{ji}$.
- (3) 所有车辆必须从原点车场出发, 完成配送任务后, 必须返回车场.
- (4) 每个客户的所有需要必须全部配送完, 可由 1 辆车或者多辆车来完成配送.
- (5) 所有配送车辆中最多只有一辆车没有满载, 其余车辆均满载.

文中研究的目的是合理规划车辆的配送路线, 使得客户配送的总成本尽可能的少. 文中配送的总成本使用路径总长度和车辆数量来表示. 假定 C 为客户点集合, $C = 1, 2, 3, \dots, n$; $q_i (1, 2, 3, \dots, n)$ 为客户 i 的需求量; C_0 为配送原点; ω 为车辆的最大运载量; R 为完成所有客户任务所需要的最小车辆数 (或者总趟数). 得到的数学模型如下:

$$\min \sum_{r=1}^R \sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ij}^r \quad (1)$$

$$R = \left\lceil \frac{\sum_{i=1}^n q_i}{\omega} \right\rceil \quad (2)$$

S.T.

$$\sum_{i=0}^n x_{ik}^r = \sum_{j=0}^n x_{kj}^r, \quad k = 0, 1, \dots, n, \quad (3)$$

$$r = 1, 2, \dots, R$$

$$\sum_{r=1}^R \sum_{i=0}^n x_{ij}^r \geq 1, \quad j = 0, 1, \dots, n \quad (4)$$

$$\sum_{r=1}^R y_i^r = q_i, \quad i = 1, 2, \dots, n \quad (5)$$

$$\sum_{i \in s^r} \sum_{j \in s^r} x_{ij}^r = |s^r| - 1, \quad r = 1, 2, \dots, R, \quad (6)$$

$$s^r \subseteq C - C_0$$

$$\sum_{i=1}^n y_r^i \leq \omega, \quad r = 1, 2, \dots, R \quad (7)$$

$$\sum_{j=0}^n x_{ij}^r q_i \geq y_{ri}, \quad r = 1, 2, \dots, R, i = 1, 2, \dots, n \quad (8)$$

$$T_z = \omega, T_f \leq \omega, z = 1, 2, \dots, R-1, f = R \quad (9)$$

$$q_i \geq y_i^r \geq 0, i = 1, 2, \dots, n, r = 1, 2, \dots, R \quad (10)$$

$$x_{ij}^r \in \{0, 1\}, i, j = 1, 2, \dots, n, r = 1, 2, \dots, R \quad (11)$$

(1) 和 (2) 分别代表车辆配送过程中最短路径及最少车辆数; 约束 (3) 表示流量守恒, 即进入某点的车辆数与离开该点的车辆数相等; 约束 (4) 和 (5) 确保每个点至少被访问一次且需求均得到满足; 约束 (6) 表示每条线路中被服务客户之间的弧边数等于被服务客户点的个数减 1; 约束 (7) 为车辆运载能力限制; 约束 (8) 表示当且仅当车辆路过客户 i 时, 该客户才能得到服务; 约束 (9) 表示在所有配送车辆中最多有一辆车的载重量小于车辆最大载重量; 约束 (10) 表示每条线路中配送给某客户的需求量不会超过该客户的最大需求量^[12], 约束 (11) 为决策变量^[3], 当且仅当第 r 条路线中车辆通过弧 (i, j) 时, $x_{ij}^r = 1$, 否则 $x_{ij}^r = 0$.

2 算法求解过程

2.1 粒子位置和速度更新策略

粒子群算法 (PSO) 由 Kennedy 和 Eberhart 在 1995 年提出. 该算法对于 Hepper 模拟鸟群 (鱼群) 的模型进行了修正, 使粒子能够“飞向”解空间, 并在最好解处“降落”. 该算法能“智能”地解决一些复杂问题的核心在于其独特的粒子位置及速度更新策略, 基本粒子群算法的粒子速度与位置更新公式如 (12) 和 (13) 所示.

$$v_{id}^k = wv_{id}^{k-1} + c_1r_1(pb_{est_{id}} - x_{id}^{k-1}) + c_2r_2(gb_{est_d} - x_{id}^{k-1}) \quad (12)$$

$$x_{id}^k = x_{id}^{k-1} + v_{id}^k \quad (13)$$

其中, v_{id}^k 表示第 k 次迭代粒子 i 飞行速度, x_{id}^k 表示第 k 次迭代粒子 i 的位置, c_1, c_2 表示加速度常数, 调节学习最大步长, r_1, r_2 表示两个随机数, w 表示惯性权重, d 代表解空间的维数, $pb_{est_{id}}$ 为个体最优, gb_{est_d} 为全局最优.

在粒子群寻优迭代过程中, 所有粒子追随个体最优位置和群体最优位置飞行, 保证群体始终朝着最优目标前进, 相对于传统寻优方法, 粒子群算法对于目标解的寻找更具有“针对性”, 更容易找到目标解.

文中在解决车辆满载需求可拆分的 (F-SDVRP) 问题中, 以粒子群算法作为其基础的搜索算法, 各参数的设定如下.

粒子的维数: 每个粒子代表一组客户的访问顺

序, 假设配送客户的数量为 n 个, 粒子的维数为 n 维, 其位置记为 x_{in} .

粒子个数设定: 粒子群算法中粒子的个数也是影响算法运算效率的关键因素, 粒子个数设定太多会导致搜索速率减慢, 过少可能会陷入局部最优, 文中结合配送的客户点个数按以下公式设定粒子的个数, 记为 E .

$$E = \begin{cases} 10 \times n, & n \geq 6 \\ 2 \times n, & n < 6 \end{cases} \quad (14)$$

n 为客户点个数.

初始粒子位置: 粒子群算法在解决车辆配送路径之前各粒子的初始位置按如下公式产生.

$$\begin{aligned} X &= [x_{in}, x_{2n}, \dots, x_{En}]^T \\ x_{in} &= [C_i, C_k, \dots, C_g]^T \\ i &\in (1, E), g, k \in (1, n) \end{aligned} \quad (15)$$

n 为客户数量, E 为粒子个数, $[C_i, C_k, \dots, C_g]^T$ 为各配送客户点的随机排列顺序, X 为各粒子初始位置向量集合.

粒子初始飞行速度: 各粒子的初始速度为 $(0, 1)$ 之间的随机数, 按下面公式产生.

$$\begin{aligned} V &= [v_{1n}, v_{2n}, \dots, v_{En}]^T \\ v_{in} &= [r_1, r_2, \dots, r_j, \dots, r_n]^T \\ j &\in (1, n) \end{aligned} \quad (16)$$

n 为客户数量, E 为粒子个数, $[r_1, r_2, \dots, r_j, \dots, r_n]^T$ 为 n 个 $(0, 1)$ 之间的随机数, V 为各粒子初始速度向量集合.

目标函数: 文中车辆路径规划问题是寻找在某个客户配送顺序下使得总路径长度最短, 故该目标函数可以记为 $F(x_{in})$.

$$F(x_{in}) = F([C_i, C_k, \dots, C_g]) \quad (17)$$

粒子个体最优解表示为:

$$Pbest_{in} = [pb_{est_{1n}}, pb_{est_{2n}}, \dots, pb_{est_{sn}}]^T \quad (18)$$

s 为算法循环次数.

粒子全局最优表示为:

$$gb_{est_n} = \min(F(Pbest_{in})), i \in (1, E) \quad (19)$$

E 为粒子个数.

根据粒子群算法速度和位置更新的基本原理, 在车辆路径优化中按以下公式对粒子进行更新.

$$\begin{aligned} v_{in}^k &= wv_{in}^{k-1} + c_1r_1(pb_{est_{in}} - x_{in}^{k-1}) + \\ & c_2r_2(gb_{est_n} - x_{in}^{k-1}) \end{aligned} \quad (20)$$

$$x_{in}^k = x_{in}^{k-1} + v_{in}^{k-1} + d(x, y) \quad (21)$$

其中, n 为解空间的维数即客户的个数, v_{in}^k 表示第 k 次迭代粒子 i 飞行速度, x_{in}^k 表示第 k 次迭代粒子 i 的位置, c_1, c_2 表示加速度常数, 调节学习最大步长, r_1, r_2 表示两个 (0,1) 之间的随机数, w 表示惯性权重, $pbest_{in}$ 为个体最优, $gbest_n$ 为全局最优, $d(x, y)$ 为欧式距离, 确保 x_{in}^k 更新后恰好落在某客户点上.

2.2 总体思路

文中按客户点的坐标将所有需要配送的客户映射至一个二维平面中, 将客户需求量看成每个点的权重, 并把该二维平面作为粒子群算法的搜索空间, 将车辆配送路径问题转换成粒子群算法寻优问题, 每一个粒子代表一组总客户点配送顺序. 每一辆“满载”的车辆均以“最短”路径进行配送, 即任何一辆“满载”车从配送原点随机出发后需满足以下要求: (1) 该车辆的货物必须全部配送完; (2) 该车必须回到配送原点, 且按“最短”路径配送. 由于配送过程中客户的需求可拆分, 文中每一辆车的“最短”路径均为在某种条件下的最短. 通过每一辆“满载”车辆配送路径的“最短”来保证整体配送的总路径“最短”. 数学函数关系式如下.

$$L = \min(l_1) + \min(l_2) + \min(l_3) + \dots + \min(l_R) \quad (22)$$

L 为所有配送车辆的总路径; R 为配送车辆数 (或者总趟数); $l_z, z \in (1, 2, \dots, R)$ 为配送过程中的每一辆车所走的路径长度.

2.3 “满载”配送及拆分方法

根据 F-SDVRP 的数学模型, 所有配送车辆中最多只有一辆车没有满载, 其余车辆均满载, 且任何车辆从配送原点出发后均需回到配送原点. 在配送过程中任何一辆满载车辆配送路线中客户的需求均有被拆分的可能, 每辆车从配送原点出发到返回配送原点均以“最优”的路线进行配送 (由于车辆的配送路径与总客户点的配送顺序有关, 因此, 该“最优”路线仅仅是在某个总客户点配送顺序条件下的最优). 为了使得每辆车均已“最优”路线的进行配送, 文中按“最邻近配送”和“返回路径最短拆分”策略.

最邻近配送策略是指在车辆配送完任意客户点 C_i 后总是从未送客户点集合中选择离客户点 C_i 最近的客户点 C_j 进行配送.

最短返回路径拆分策略是指在车辆配送完任意

客户点 C_i 后, 利用 Dijkstra 算法寻找从客户点 C_i 返回配送原点的最短路径 $C_i \rightarrow C_k \rightarrow \dots \rightarrow C_0$ (已配送的客户点会删除), 若返回路径中客户的总需求量大于等于当前车辆剩余载重量, 则按该路径进行配送, 需拆分的客户为从 C_i 返回配送原点线路中的客户.

总配送策略步骤如下, 假设某时刻客户点的配送顺序为 $\{C_1, C_2, \dots, C_n\}$, 令其为未送客户点集合.

I 从未送客户点集合中选择需先配送客户点 C_1 , 若 C_1 的需求量 q_1 大于车辆最大载重, 则车辆返回配送原点, 更新客户点需求量 q_1 和配送车辆数.

II 若 C_1 的需求量 q_1 小于车辆最大载重, 则配送完 C_1 后执行最短返回路径拆分策略, 判断其是否成立. 如果成立则执行该策略, 否则, 执行最邻近配送策略. 更新已送配送客户点集合、未配送配送客户点集合及配送车辆数.

III 按 I 和 II 配送原则继续从未送客户点集合中选择客户进行配送, 直到所有的客户都配送完成为止.

配送线路及拆分客户点如图 1 所示.

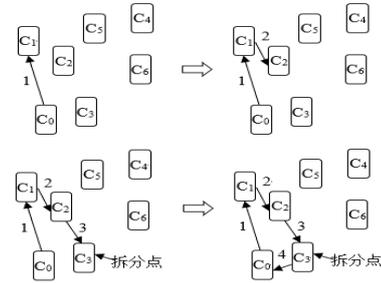


图 1 配送线路及拆分客户点

图 1 中表示某趟满载车辆的配送路径和客户点需求拆分图, 其中该线路中所有客户点的总需求量大于等于配送车辆的最大载重. 过程 1 表示某满载配送车辆从配送原点 C_0 按照某客户配送顺序先选择客户 C_1 进行配送; 过程 2 表示按照最邻近配送策略选择距离客户点 C_1 最近的客户点 C_2 进行配送; 过程 3 表示执行最短返回路径拆分策略, 选择客户点 C_3 而不是选择距离客户点 C_2 最近的客户点 C_5 进行配送, 其中, 客户点 C_3 为可能需拆分的客户点, 即当车辆配送到该客户点时, 若配送车辆剩余载重小于该客户点需配送的需求量时就拆分; 过程 4 表示车辆从客户点 C_3 返回配送原点 C_0 .

由于任意满载车辆在各轮配送过程中均已“最短”路径访问各客户点并以“最短”的路径返回配送原点, 故能保证在某个总客户点配送顺序条件下

的车辆配送总路径的最优。

鉴于客户需求可拆分原则, 通过算例仿真测试发现, 车辆在满载的情况下, 客户的配送顺序是影响配送总路径的关键因素, 文中通过粒子群算法不断调整客户配送顺序来优化车辆配送的总路径。例如, 在处理 7 个客户点的配送任务时, 当配送客户顺序为 $0 \rightarrow 5 \rightarrow 6 \rightarrow 3 \rightarrow 7 \rightarrow 2 \rightarrow 1 \rightarrow 4$ 时, 满载的车辆配送策略如下。

(1) 初始化参数: 未送客户点 $ws=[5\ 6\ 3\ 7\ 2\ 1\ 4]$; 已配送客户点 $ys=[]$; 总路径 $L=0$; 总趟数 $T=0$;

(2) 满载车辆按配送顺序先配送 5 号客户点。

配送原则如下:

若 5 号客户需求大于车辆满载量 ($q_5 > \omega$), 更新客户点需求量即 $q_5 = q_5 - \omega$; 车辆总路径 $L = L + l_1$; 总趟数 $T = T + 1$ 。

若 5 号客户需求等于车辆满载量 ($q_5 = \omega$), 车辆总路径 $L = L + l_1$; 总趟数 $T = T + 1$; 更新未送客户点 $ws=[6\ 3\ 7\ 2\ 1\ 4]$, 更新已送客户点 $ys=[5]$ 。

若 5 号客户需求小于车辆满载量 ($q_5 < \omega$), 更新该车辆载重量 ($\omega = \omega - q_5$), 执行最短返回路径拆分策略, 判断其是否成立。如果成立, 则访问该路线中的客户点, 否则, 执行最邻近配送策略, 在 ws 客户点集合中选择离客户点 5 最近的客户点 M , 更新车辆载重 $\omega = \omega - q_m$ 。若 $\omega > 0$, 更新已送客户点 $ys=[5\ w]$, 未送客户点数量减一, 此时在 ws 客户点集合中选择离客户点 M 最近的客户点 N , 更新车辆载重、未配送客户点及已配送客户点, 以此类推, 直到找到客户点 U , 在该客户点时车辆载重 $\omega \leq 0$, 更新该客户点需求量 $q_u = q_u - \omega$, 车辆返回配送原点, 该趟车辆总路径 $L = L + l_1$, 总趟数 $T = T + 1$ 。

(3) 按照配送顺序从未送客户点 ws 中选择客户点按 (2) 中的配送原则进行配送。若 $ws=[]$, $ys=[1\ 2\ 3\ 4\ 5\ 6\ 7]$, 则配送完成, 输出车辆总路径 L 和总趟数 T 。否则重复执行 (3)。

(4) 利用粒子群算法更新客户点配送顺序, 重复执行 (1)(2)(3), 如达到算法最大循环次数, 则输出车辆总配送最优路径。

算法输出的最优总路径大小数学表达式如下。

$$L_j = \sum_{k=1}^R l_{kj}, j = 1, 2, \dots, N \quad (23)$$

$$F(L) = \min(L_j) \quad (24)$$

N 为粒子群算法循环次数, l_{kj} 为算法在第 j 轮循环中车辆 K 配送的“最短”路径, $K = (1, 2, \dots, R)$, $F(L)$ 为最优的车辆配送总路径长度。

2.4 算法流程

算法流程如 (1)-(6) 所示。

(1)Initial: 对粒子群进行初始化, 包括初始化粒子 x_{in} 的位置 (客户点随机配送顺序), 粒子更新速度 v_{in} 及群体规模 E , 最大循环次数 $\max num$ 等。

(2)Evaluation: 根据满载车辆配送策略, 计算各粒子的适应度值 (公式 (17) 和公式 (22))。

(3)Find the Pbest: 将当前各个粒子计算得到的适应度值与其历史最优适应度值做比较, 如果当前粒子的适应度值比历史最优的适应度值要好, 则使用当前粒子对应的客户点配送顺序替换为该粒子个体的历史最佳客户点配送顺序, 即更新 $Pbest_{in}$ 。

(4)Find the Gbest: 将当前各个粒子计算得到的粒子个体最优适应度值与全局最优适应度值作比较, 如果当前粒子的适应度值比全局最优适应度值要好, 则使用当前粒子对应的客户点配送顺序替换为全局最佳客户点配送顺序, 即更新 $gbest_n$ 。

(5)Update the Velocity: 根据公式 (20) 与公式 (21) 依次更新各个粒子对应的客户点配送顺序。

(6) 如果没有达到算法终止条件, 则返回步骤 (2), 否则, 跳出循环, 输出“最优”路径 (公式 (23) 和公式 (24)), 算法结束。

3 仿真结果分析

为测试算法的有效性, 本文利用以往文献中的数据进行仿真测试, 并与相关文献中的求解结果做了比较, 所有的测试均在 Matlab 中编码运行。

仿真算例 1

相关文献了求解一个包含 15 个客户点、车辆的最大运载量为 500 的需求可拆分车辆路径规划问题, 15 个客户点的基本信息如表 1 所示, 客户点 1 为配送原点。

表 1 15 个客户点的基本信息

客户点	横坐标	纵坐标	需求量
1	0	0	0
2	32	41	468
3	96	9	335
4	7	58	1
5	97	87	170
6	26	21	225
7	23	100	479
8	52	31	359
9	76	43	463
10	74	17	465
11	72	104	206
12	40	99	146
13	8	16	282
14	27	38	328
15	78	69	462
16	46	16	492

表2 不同文献计算改组数据最优总路径长度比较

文献	主要方法	最优路径长度	车辆数(最大趟数)
文献 [3]	聚类算法	1764.4	10
文献 [24]	蚁群算法	1836	10
文献 [12]	蜂群算法	1757.6	10
文献 [25]	蜂群算法	1762.7	10
文献 [26]	分段求解	1827.7	10
本文算法	粒子群算法	1722.1	10

仿真结果表明,文中算法的求得的客户点最优配送顺序为 [5,14,6,13,2,3,10,8,4,7,12,11,15,9,16], 最优路径长度为 1722.1, 该路径长度比文献 [3] 中最优路线总长度减少了 42.2; 比文献 [24] 中最优路线总长度减少了 113.8; 比文献 [12] 中最优路线总长度减少了 34.8; 比文献 [25] 中最优路线总长度减少了 40.5; 比文献 [26] 中最优路线总长度减少了 105.5.

表3 本文算法求得的最优配送路径

编号	回路路径及客户配送量	路径长度	装载率 (%)
1	1-5 (170) -15 (330) -1	260.61	100
2	1-14 (328) -2 (172) -1	104.46	100
3	1-6(225)-13(275)-1	69.99	100
4	1-13(7)-2(296)-8(197)-1	135.44	100
5	1-3(335)-10(165)-1	195.76	100
6	1-10(300)-8(162)-16(38)-1	166.86	100
7	1-4(1)-7(479)-12(20)-1	227.17	100
8	1-12(126)-11(206)-15 (132) -9(36)-1	288.07	100
9	1-9(427)-16(73)-1	176.37	100
10	1-16(381)-1	97.4	76.2
	总计	1722.13	

仿真算例 2

相关文献求解了一个包含 20 个客户点、车辆的最大运载量为 5 的需求可拆分车辆路径规划问题,20 个客户点的基本信息如表 4 所示,客户点 1 为配送原点.

表4 20 个客户点的基本信息

客户点	横坐标	纵坐标	需求量
1	14.5	13	0
2	12.8	8.5	2
3	18.4	3.4	2
4	15.4	16.6	2
5	18.9	15.2	1
6	15.5	11.6	3
7	3.9	10.6	2
8	10.6	7.6	1
9	8.6	8.4	3
10	12.5	2.1	1
11	13.8	5.2	4
12	6.7	16.9	2
13	14.8	2.6	2
14	1.8	8.7	3
15	17.1	11	1
16	7.4	1	1
17	0.2	2.8	3
18	11.9	19.8	1
19	13.2	15.1	4
20	6.4	5.6	1
21	9.6	14.8	1

通过仿真测试,文中算法求得客户点最优配送顺序为 [9,8,3,13,10,20,16,17,4,21,12,18,19,2,11,7,14,5,15,6], 最优路径长度为 171.09. 比文献 [23] 中的禁忌算法求得的最优总路径长度少 6.1, 比遗传算法求得的最优总路径长度少 2.9, 比文献 [3] 中的聚类算

法求得的最优路径少 8.4, 比文献 [22] 中的最优总路径长度少 1.5.

表5 与不同文献最优总路径长度比较

文献	主要方法	最优路径长度	车辆数(最大趟数)
文献 [3]	聚类算法	179.5	8
文献 [23]	禁忌搜索算法	186.4	8
文献 [23]	遗传算法	183.2	8
文献 [22]	聚类算法	172.6	8
本文算法	粒子群算法	171.1	8

表6 本文算法求解最优路径线路安排

编号	回路路径及客户配送量	路径长度	装载率 (%)
1	1-9(3) -8(1) -2(1) -1	16.82	100
2	1-3(2) -13(2) -10(1) -11-1	27.48	100
3	1-20(1) -16(1) -17(3) -14-1	40.66	100
4	1-4(2) -19(3) -1	8.84	100
5	1-21(1) -12(2) -18(1) -19(1) -1	22.1	100
6	1-2(1) -11(4) -1	16.08	100
7	1-7(2) -14(3) -1	26.2	100
8	1-5(1) -15(1) -6(3) -1	12.91	100
	总计	171.09	

仿真算例 3

文献 [27] 利用蚁群算法求解了一个车辆载重量为 8t、客户数量为 20 个的车辆路径规划问题,客户点 1 为配送原点,利用本文算法求得最优配送序列为 [20,16,17,14,7,9,2,8,11,13,10,3,15,6,5,4,19,21,12,18]. 文献中的运行结果与本文算法运行结果如表 7 所示.

表7 文献 [27] 与本文算法计算结果

文献	主要方法	最优路径长度/km	车辆数(最大趟数)
文献 [27]	蚁群算法	1078.4	4
本文算法	粒子群算法	1155.7	3

表8 本文算法求得车辆最优路径安排

编号	回路路径及客户配送量	路径长度	装载率 (%)
1	1-20(1.7) -16(1.7) -17(1.1) -14(1.3) -7(1.3) -9(0.6) -8(0.3) -1	453.5	100
2	1-2(0.1) -8(1.4) -11(0.4) -13(1.3) -10(1.2) -3(0.4) -15(1.9) -6(0.8) -19(0.5) -1	384.45	100
3	1-5(1.5) -4(1.2) -19(1.1) -21(1.5) -12(0.9) -18(1.5) -1	317.78	96.25
	总计	1155.73	

文献 [27] 中的最优路径长度为 1078.4, 比文中算法少 77.7, 路径趟数为 4 趟, 文中的算法虽然最优路径长度没有文献 [27] 中的好, 但是路径趟数比文献 [17] 要少 1. 仿真结果表明, 需求可拆分为车辆规划路径问题是个多目标规划问题, 受车辆数 (最大趟数) 和路径长度共同的影响, 在车辆路径规划问题中, 总路径长度和车辆数互相制约.

仿真算例 4

文中设计了一个包含 35 个客户点、车辆的最大运载量为 8t 的需求可拆分车辆路径规划问题,35 个客户点的基本信息如表 9 所示, 客户点 1 为配送原点, 坐标为 (15.31,12.61).

表 9 35 个客户点基本信息

客户点	横坐标/km	纵坐标/km	需求量/t
1	15.31	12.61	0
2	12.9	8.5	0.43
3	18.43	2.77	1.44
4	16.11	16.48	0.65
5	2.51	6.77	1.59
6	15.9	11.13	0.69
7	4.38	10.07	1.26
8	10.79	7.17	0.35
9	8.63	8.3	0.68
10	13.23	1.6	2.4
11	14.18	4.4	1.02
12	6.75	16.8	0.45
13	15.17	1.64	2.71
14	2.16	8.27	1.99
15	12.19	2.63	2.22
16	7.86	0.72	2.16
17	0.92	2.59	2.36
18	12.28	19.7	2.63
19	13.87	14.59	2.42
20	7.28	4.6	1.33
21	10.01	14.37	2.42
22	6.88	8.84	0.98
23	4.1	14.64	1.94
24	5.53	4.6	2.43
25	0.94	12.93	0.48
26	12.94	7.58	1.01
27	21.8	6.05	1.3
28	8.46	10.51	1.36
29	19.74	14.3	2.41
30	10.61	17.66	0.72
31	11.8	6.67	1.69
32	13.07	2.46	0.9
33	5.22	9.27	2.29
34	10.01	11.05	1.45
35	4.24	15.8	1.47
36	17.2	10.6	1.19

表 10 文中算法求解该问题的 10 次计算结果

计算次数	总路径长度/km	车辆数 (最大趟数)
1	211.47	7
2	214.04	7
3	214.45	7
4	208.24	7
5	217.77	7
6	208.24	7
7	218.62	7
8	214.35	7
9	208.24	7
10	216.52	7

表 11 车辆最优路径安排

编号	回路路径及客户配送量	路径长度/km	装载率 (%)
1	1-12(0.45)-35(1.47)-23(1.94)-25(0.48)-7(1.26)-33(2.29)-22(0.11)-1	33.57	100
2	1-28(1.36)-34(1.45)-9(0.68)-22(0.87)-8(0.35)-31(1.69)-26(1.01)-2(0.43)-6(0.16)-1	27.05	100
3	1-17(2.36)-5(1.59)-14(1.99)-24(2.06)-1	41.17	100
4	1-13(2.71)-10(2.40)-32(0.90)-15(1.99)-1	25.13	100
5	1-18(2.63)-30(0.72)-21(2.42)-19(2.23)-1	20	100
6	1-24(0.37)-20(1.33)-16(2.16)-15(0.23)-11(1.02)-3(1.44)-27(1.30)-36(0.15)-1	44.19	100
7	1-19(0.19)-4(0.65)-29(2.41)-36(1.04)-6(0.53)-1	17.09	60.25
总计		208.24	

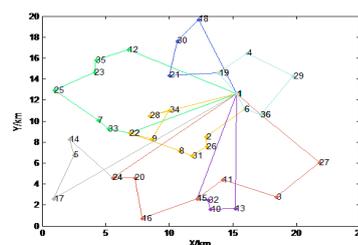


图 2 35 个客户点配送线路途经安排图

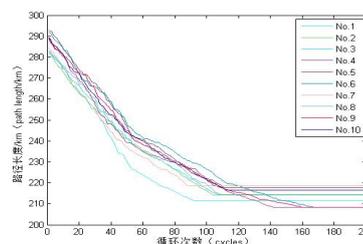


图 3 10 次寻优过程中算法收敛情况图

由表 10 和图 3 可知, 该算法求解性能稳定, 10 次仿真求得回路数均为 7, 算法均能在较少循环次数的情况下收敛, 最优路线总长度为 208.24, 最优配送顺序为 [12,35,23,25,7,33,28,34,9,22,8,31,26,2,17,5,14,13,10,32,18,30,21,24,20,16,15,11,3,27,19,4,29,36,6]. 根据仿真运算结果, 该算法对 22、6、24、19、36 及 15 号客户的需求进行了拆分.

4 结论

为了更加合理的规划车辆配送路径问题, 文中提出了一种基于粒子群算法的满载需求可拆分车辆路径规划策略, 该策略的核心思想是通过保证任何一辆满载的配送车辆从配送点出发后均已“最优”的配送路径进行配送来确保配送的总路径“最优”, 并通过粒子群算法不断调整整个客户点的配送顺序. 由于配送车辆均在满载的情况下进行配, 故能确保配送过程中的总车辆数 (总趟数) 最少, 车辆的装载率也高. 文中的仿真算例 1 和仿真算例 2 表明, 基于粒子群算法的“满载”策略比相关文献中的方法要好, 在相同车辆数的情况下减少了客户点配送总路径, 提升了客户点配送效率, 可以作为车辆路径规划问题的一种参考思路. 仿真算例 3 表明需求可拆分为车辆规划路径问题是个多目标规划问题, 受车辆数 (最大趟数) 和路径长度共同的影响, 在车辆路径规划问题中总路径长度和车辆数互相制约, 在有的车辆路径规划问题中如果要减少总路径长度可能需要增加必要的车辆数量. 文中虽然优化了相关文献中解决车辆路径规划问题的配送策略, 但算法也还可以进一步完善. 在未来的车辆路径规划研究中, 如何使用最少的车辆数和最短的总路径来完成配送任

务仍然是车辆路径规划问题的重要探究方向。

粒子群算法可以很大程度上提升了问题的解决效率,可以在相对较少的循环次数中找到问题的解.相比传统方法,智能算法在解决非线性问题时确实有优势.但智能算法也不是完美的,容易陷入局部最优是智能算法存在的主要问题之一,当智能算法的参数设定不合理时,其就可能陷入局部最优^[28-31].通过仿真发现,文中在设定的参数 $w = 0.5, c_1 = 0.8, c_2 = 1.2$ 时算法的收敛陷入局部最优的概率为 4.06%.一般来说,粒子群算法中各参数的设定根据问题的实际处理情况来决定,但大部分问题中在 $w = [0.25, 1.75], c_1 = c_2 = [0.75, 3]$ 的范围之内效果会比较好.相关研究表明,种群全局搜索能力和局部搜索能力并不是依靠某个参数决定,而是依靠 w 、 c_1 和 c_2 的相互配合来调节的^[32].此外,仿真发现,粒子群算法初始粒子位置也对算法的收敛性有一定的影响,比较“优”的粒子初始位置算法陷入局部最优的概率较小,算法收敛的速度较快。

参考文献 (References)

- [1] Dantzing G, Ramser J. The truck dispatching problem[J]. *Management Science*, 1959, 10(6): 80-91.
- [2] Schouwenaars T, De Moor B, Feron E, et al. Mixed integer programming for multi-vehicle path planning[C]//2001 European control conference (ECC). IEEE, 2001: 2603-2608.
- [3] 刘旺盛, 杨帆, 李茂青, 陈培芝. 需求可拆分车辆路径问题的聚类求解算法 [J]. *控制与决策*, 2012, 27(04): 535-541.
(Liu W S, Yang F, Li M Q, Chen P Z. Cluster Solution Algorithms for Demand Separable Vehicle Routing Problem [J]. *Control and Decision*, 2012, 27 (04): 535-541.)
- [4] 鲍春玲, 张世斌. 考虑碳排放的冷链物流联合配送路径优化 [J]. *工业工程与管理*, 2018, 23(05): 95-100+107.
(Bao C L, Zhang S B. Cold chain logistics joint distribution path optimization considering carbon emissions [J]. *Industrial Engineering and Management*, 2018, 23 (05): 95-100+107.)
- [5] 杨柳. 基于混合遗传算法的物流配送路径优化研究 [J]. *电子设计工程*, 2018, 26(17): 70-74.
(Yang L. Research on Logistics Distribution Route Optimization Based on Hybrid Genetic Algorithms [J]. *Electronic Design Engineering*, 2018, 26 (17): 70-74.)
- [6] Dror M, Trudeau P. Savings by split delivery routing[J]. *Transportation Science*, 1989, 23(2): 141-145.
- [7] 杨鹏, 邹浩, 徐贤浩. 带时间窗集送货需求可分车辆路径问题的改进蚁群算法 [J]. *系统工程*, 2015, 33(09): 58-62.
(Yang P, Zou H, Xu X H. An improved ant colony algorithm for vehicle routing problem with time-window demand separable [J]. *Systems Engineering*, 2015, 33 (09): 58-62.)
- [8] 胡钟骏, 周泓. 改进遗传算法的需求可拆分车辆路径优化研究 [J]. *计算机仿真*, 2018, 35(03): 80-83.
(Hu Z J, Zhou H. Research on Demand Separable Vehicle Route Optimization Based on Improved Genetic Algorithms [J]. *Computer Simulation*, 2018, 35(03): 80-83.)
- [9] 闵嘉宁, 金成, 陆俐君. 解决需求可拆分车辆路径问题的先聚类后路径方法 [J]. *制造业自动化*, 2018, 40(11): 142-147.
(Min J N, Jin C, Lu L J. Clustering before routing method for resolving demand-splittable vehicle routing problem [J]. *Manufacturing Automation*, 2018, 40 (11): 142-147.)
- [10] 向婷, 潘大志. 求解需求可拆分车辆路径问题的聚类算法 [J]. *计算机应用*, 2016, 36(11): 3141-3145.
(Xiang T, Pan D Z. Clustering algorithm for solving demand-splittable vehicle routing problem [J]. *Journal of Computer Applications*, 2016, 36(11): 3141-3145.)
- [11] 李相勇, 田澎. 开放式车辆路径问题的蚁群优化算法 [J]. *系统工程理论与实践*, 2008(06): 81-93.
(Li X Y, Tian P. Ant Colony Optimization for Open Vehicle Routing Problem [J]. *Systems Engineering-Theory & Practice*, 2008 (06): 81-93.)
- [12] 姜婷. 求解需求可拆分车辆路径问题的人工蜂群算法 [J]. *四川理工学院学报 (自然科学版)*, 2017, 30(03): 6-9.
(Jiang T. Artificial Bee Colony Algorithms for Demand Separable Vehicle Routing Problem [J]. *Journal of Sichuan University of Science & Engineering (Natural Science Edition)*, 2017, 30 (03): 6-9.)
- [13] 孟凡超, 陆志强, 孙小明. 需求可拆分车辆路径问题的禁忌搜索算法 [J]. *计算机辅助工程*, 2010, 19(01): 78-83.
(Meng F C, Lu Z Q, Sun X M. Tabu Search Algorithms for Demand Separable Vehicle Routing Problem [J]. *Computer Aided Engineering*, 2010, 19 (01): 78-83.)
- [14] Dror M, Trudeau P. Savings by split delivery routing[J]. *Transportation Science*, 1989, 23(2): 141-145.
- [15] Gendreau M, Hertz A, Laporte G. A tabu search heuristic for the vehicle routing problem[J]. *Management science*, 1994, 40(10): 1276-1290.
- [16] Kennedy J, Eberhart R. Particle swarm optimization (PSO)[C]//Proc. IEEE International Conference on Neural Networks, Perth, Australia. 1995: 1942-1948.
- [17] 谭阳, 唐德权, 曹守富. 基于超球形模糊支配的高维多目标粒子群算法 [J/OL]. *计算机应用*: 1-13 [2019-09-07].
(Tan Y, Tang D Q, Cao S F. High-dimensional multi-objective particle swarm optimization algorithm based on hyperspherical fuzzy domination [J/OL]. *Journal of Computer Applications*: 1-13 [2019-09-07].)
- [18] 赵乃刚, 邓景顺. 粒子群优化算法综述 [J]. *科技创新导报*, 2015, 12(26): 216-217.
(Zhao N G, Deng J S. Summary of Particle Swarm Optimization Algorithms [J]. *Science and Technology*

- Innovation Herald, 2015, 12 (26): 216-217.)
- [19] 薛洪波, 伦淑娴. 粒子群算法在多目标优化中的应用综述 [J]. 渤海大学学报 (自然科学版), 2009, 30(03): 265-269.
(Xue H B, Lun S X. Summary of the application of particle swarm optimization in multi-objective optimization [J]. Journal of Bohai University (Natural Science Edition), 2009, 30 (03): 265-269.)
- [20] 张进峰, 杨涛宁, 马伟皓. 基于多目标粒子群算法的船舶航速优化 [J]. 系统仿真学报, 2019, 31(04): 787-794.
(Zhang J F, Yang T N, Ma W H. Vessel speed optimization based on multi-objective particle swarm optimization [J]. Journal of Systems Simulation, 2019, 31 (04): 787-794.)
- [21] 郭鑫, 张炳义, 冯桂宏. 基于混合粒子群算法的低速大转矩直驱永磁同步电机多目标优化研究 [J]. 机电工程, 2018, 35(11): 1214-1219.
(Guo X, Zhang B Y, Feng G H. Multi-objective optimization of low-speed and high-torque direct-drive permanent magnet synchronous motor based on hybrid particle swarm optimization [J]. Journal of Mechanical & Electrical Engineering, 2018, 35(11): 1214-1219.)
- [22] 向婷, 潘大志. 求解需求可拆分车辆路径问题的聚类算法 [J]. 计算机应用, 2016, 36(11): 3141-3145.
(Xiang T, Pan D Z. Clustering algorithm for solving demand-splittable vehicle routing problem [J]. Journal of Computer Applications, 2016, 36(11): 3141-3145.)
- [23] 谢毅. 需求可拆分的物流车辆路线问题研究 [D]. 同济大学, 2006.
(Xie Y. Research on Demand Separable Logistics Vehicle Routing [D]. Tongji University, 2006.)
- [24] 隋露斯, 唐加福, 潘震东, 等. 用蚁群算法求解需求可拆分车辆路径问题 [C]. 中国控制与决策会议. 烟台, 2008: 997-1001.
(Sui L S, Tang J F, Pan Z D, et al. Solving demand-splittable vehicle routing problem with ant colony algorithm [C]. China Control and Decision Conference. Yantai, 2008: 997-1001.)
- [25] 汪婷婷, 倪郁东, 何文玲. 需求可拆分车辆路径问题的蜂群优化算法 [J]. 合肥工业大学学报 (自然科学版), 2014, (8): 1015-1018, 1024.
(Wang Tingting, Ni Yudong, He Wenling. Bee colony optimization algorithm for demand-splittable vehicle routing problem [J]. Journal of Hefei University of Technology (Natural Science Edition), 2014, (8): 1015-1018, 1024.)
- [26] 刘旺盛, 黄娟. 需求可拆分的车辆路径问题的分段求解 [J]. 集美大学学报 (自然科学版), 2011, 16(1): 38-44.
(Liu W S, Huang J. Segmental solution of vehicle routing problem with split demand [J]. Journal of Jimei University (Natural Science Edition), 2011, 16 (1): 38-44.)
- [27] 刘志硕, 柴跃廷, 申金升. 蚁群算法及其在有硬时间窗的车辆路径问题中的应用 [J]. 计算机集成制造系统, 2006(04): 596-602.
(Liu Z S, Chai Y T, Shen J S. Ant colony algorithm and its application in vehicle routing problem with hard time windows [J]. Computer Integrated Manufacturing Systems, 2006 (04): 596-602.)
- [28] Karaboga D, Akay B. A comparative study of artificial bee colony algorithm [J]. Applied mathematics and computation, 2009, 214(1): 108-132.
- [29] Reed M, Yiannakou A, Evering R. An ant colony algorithm for the multi-compartment vehicle routing problem [J]. Applied Soft Computing, 2014, 15: 169-176.
- [30] Moradi M H, Abedini M. A combination of genetic algorithm and particle swarm optimization for optimal DG location and sizing in distribution systems [J]. International Journal of Electrical Power & Energy Systems, 2012, 34(1): 66-74.
- [31] Dai M, Tang D, Giret A, et al. Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm [J]. Robotics and Computer-Integrated Manufacturing, 2013, 29(5): 418-429.
- [32] 王东风, 孟丽. 粒子群优化算法的性能分析和参数选择 [J]. 自动化学报, 2016, 42(10): 1552-1561.
(Wang D F, Meng L. Performance analysis and parameter selection of particle swarm optimization algorithm [J]. Acta Automatica Sinica, 2016, 42(10): 1552-1561.)

作者简介

卿东升 (1990—), 男, 博士生, 从事人工智能、生态规划等研究, E-mail: 673390302@qq.com.

邓巧玲 (1990—), 女, 讲师, 硕士, 从事基础数学、理论数学等研究, E-mail: 1006308937@qq.com.

李建军 (1970—), 男, 教授, 博士, 从事林业系统工程、人工智能等研究, E-mail: 390033868@qq.com.