

doi:10.11835/j.issn.1000-582X.2021.09.010

基于聚类非支配排序的电物流动车路径规划及充电策略

徐婷婷¹, 胡晓锐¹, 胡文¹, 李双庆², 池磊¹

(1. 国网重庆市电力公司营销服务中心, 重庆 401123; 2. 重庆大学计算机学院, 重庆 400044)

摘要: 电物流动车电池容量有限、充电时间长以及配套设施不健全等问题制约着其在物流配送领域中有有效推广。为此, 提出基于聚类非支配排序算法(AP-NSGA-II)来解决电物流动车的多目标路径优化问题, 建立了一种充电策略, 通过设计加权 AP 聚类划分配送簇, 避免初始种群的随机性和盲目性, 簇内配送点规模降低了非支配排序算法的运行时间和复杂度, 根据充电站的分布和距离关系, 电物流动车执行部分充电策略。最后, 通过仿真实验证明该算法的有效性, 比较了电物流动车满充和部分充电条件的差异。

关键词: 电物流动车; 物流配送; 充电策略; 车辆路径优化; 非支配排序算法

中图分类号: TP 39

文献标志码: A

文章编号: 1000-582X(2021)09-098-11

Path planning and charging strategy for electric logistics vehicles with clustering non-dominated sorting

XU Tingting¹, HU Xiaorui¹, HU Wen¹, LI Shuangqing², CHI Lei¹

(1. Marketing Service Center, State Grid Chongqing Electric Power Co., Chongqing 401123, P. R. China;

2. College of Computer Science, Chongqing University, Chongqing 400044, P. R. China)

Abstract: The limited battery capacity, long charging time and inadequate supporting facilities of electric logistics vehicles restrict their effective promotion in the logistics and distribution field. In this paper, an improved cluster non-dominated sorting genetic algorithm (AP-NSGA-II) is proposed to solve the multi-objective route optimization problem of electric logistics vehicles. A charging strategy is established: dividing the distribution clusters by designing weighted AP clusters to avoid the randomness and blindness of the initial population, and reducing the running time and complexity of the non-dominated sorting algorithm by the scale of distribution points within the clusters. According to the distribution and distance relationship of charging stations, the electric logistics vehicles execute partial charging strategy. Finally, the effectiveness of the algorithm is demonstrated by simulation experiments, and the differences between full charging and partial charging conditions for electric logistics vehicles are compared.

Keywords: electric logistics vehicle; logistics distribution; charging strategy; vehicle routing optimization; non-dominated sorting genetic algorithm

收稿日期: 2020-11-12

基金项目: 国家电网公司科技资助项目(1400-202057220A-0-0-00)。

Supported by Science and Technology Project of SGCC(1400-202057220A-0-0-00).

作者简介: 徐婷婷(1990—), 女, 主要从事电力系统分析和电动汽车大数据分析方向研究, (Tel)18723396127, (E-mail)825328370@qq.com。

物流配送对规模和时效性要求越来越高,物流车辆路径优化问题显得极其重要,合理的车辆路径规划方案可以为企业节省大量配送成本。另一方面,传统物流车辆已造成严重的污染物排放问题,而物流汽车电动化是解决排放治理的重要手段,但电动物流车最大的问题在于配送里程焦虑。考虑物流总路程、用户时间窗口、总时间的条件下,提出一种加权 AP 聚类的非支配排序遗传算法(AP-NSGA-II)来解决电动物流车的路径优化配送问题。

电动物流车物流配送大部分关注的是路径优化。葛显龙等^[1]运用混合模拟退火算法研究了灵活充电策略带时间窗口的电动物流车的配送问题,Schneider 和 Stenger 等^[2]在 VRP 条件下,通过车载电池容量的增加,搭建了电动物流车的车辆路径优化问题,并建立电动汽车数量最小和路径最短的多目标,寻求最优行驶路径。Lim 和 Kuby^[3]基于公共设施的选址问题,提出电动汽车的能源补充设施选址模型。冯智泉等^[4]应用蚁群算法对分段路线进行优化,在全局充电方案中加入行驶路径的约束,将两部分结合形成全局的最优行驶路径。文展等^[5]提出一种改进的粒子群优化算法解决离散域的组合优化问题及车辆路径规划。但这些方法普遍存在以下问题:1)随着配送点的增加,车辆路径可行解呈阶乘增长,使用启发式或者亚启发式算法的求解效率低,且容易陷入局部最优,贪婪随机自适应搜索算法等^[6-10];2)大多数车辆路径问题的优化目标函数较少,不切合现实场景需求,未考虑相关变量的相互关系,导致实际应用的局限性^[11-14]。3)电动物流车的里程焦虑问题,限制了每辆电动物流车路径规划的最长距离,导致电动物流车的用车数量增加,综合用车成本增加^[15-17]。

研究提出一种改进的非支配遗传算法(AP-NSGA-II)解决大规模物流电动车的路径规划问题,根据配送点的分布,采用先聚类再并行优化的方式解决多目标问题,聚类采用带权 AP 算法,并行多目标优化采用改进的非支配排序算法得到初始配送方案。避免了初始配送方案的随机性和盲目性,同时也降低了算法的执行复杂度。在初始路径规划的基础上执行邻近配送点搜索进行局部重构,以获得最终配送方案。最后根据配送点的权值大小,优选配送路途充电策略。

1 问题描述

在已知区域内有一定数量的配送点和充电站,从物流中心出发,若干辆电动物流车对所有配送点进行双向货物配送,在配送过程中,要综合考虑配送点的分布和充电站的地理位置,当电动物流车的电池余量不足完成剩余配送,则寻求最适合的充电站进行充电,以满足配送任务,考虑总路程、时间窗口和总时间的多目标优化问题。图1展示了典型电动物流车的配送场景,其中0为配送中心, $N_1 \sim N_7$ 为配送点, C_1, C_2 为充电站。

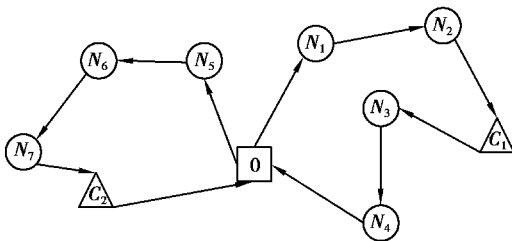


图1 途中充电策略示意图

Fig. 1 An illustration of en-route charging strategy

2 数学模型

2.1 假设条件

实际场景中大规模电动物流车配送问题较为复杂,比如路况拥堵,电池衰减、货物体积容量,人力成本计算等,这些都是影响配送结果的因素。为了简化模型,做如下假设:

- 1) 每辆电动物流车从配送中心出发最终回到配送中心,形成环路;
- 2) 每个配送点的送货量和取货量不能大于电动物流车的最大载货量;

- 3) 仅考虑货物的重量, 不考虑货物的体积容量;
- 4) 每辆电物流车的电池容量、行驶速度、充电系数等参数配置一致;
- 5) 电物流车的行驶距离、充电时长与电池容量呈线性关系;
- 6) 每个充电站能多次被访问, 且无等待时间
- 7) 不考虑交通拥堵、车辆故障等特殊因素, 且配送过程的车速是均匀的;
- 8) 每个配送点的装卸货时长为 0, 仅考虑路途时间和充电时间。

AP-NSGA-II 算法的思想: 以充电桩为重要潜在聚类中心, 根据各配送点的取送货量为权值, 执行加权 AP 算法以获得配送点的聚类中心, 各簇执行 NSGA-II 多目标算法, 并考虑电池笑话以获得最终配送方案。数学符号如表 1 所示。

表 1 数学符号及含义
Table 1 Mathematical symbols and meaning

数学符号	含义
N	配送点集合 $N = \{1, 2, 3, \dots, n\}$
C	充电站集合 $C = \{1, 2, 3, \dots, c\}$
V	所有节点结合 $V = 0 \cup N \cup C$, 其中 0 为配送中心
K	电物流车编号, $K = \{1, 2, 3, \dots, k\}$
x_i	所有节点的 x 坐标, $i \in V$
y_i	所有节点的 y 坐标, $i \in V$
d_{ij}	节点 i 到节点 j 的欧式距离, $j \in V$
D_i	配送点 i 的送货量, $i \in N$
R_i	配送点 i 的取货量, $i \in N$
L	电物流车的最大载货量
L_i	电物流车在节点 i 装卸完成货物的载货量, L_0 表示电物流车从配送中心出来的初始载货量, $i \in N$
Q	电物流车的最大电池容量
Q_i	电物流车在节点 i 剩余电池容量, $i \in V$
ca	单位距离电池能量消耗系数
tf_i	电物流车在充电站的充电时间 $i \in C$
e_i	配送点最早服务时间 $i \in N$
l_i	配送点最晚服务时间 $i \in N$
t_i	电物流车到达配送点的时间 $i \in N$
e_p	电物流车提前到达的惩罚成本
l_p	电物流车晚到惩罚成本
s	电物流车的行驶速度
S	加权 AP 算法中的相似度矩阵
R	加权 AP 算法中的吸引度矩阵
A	加权 AP 算法中的归属度矩阵
M	加权 AP 算法中最终簇的个数, $M = \{1, 2, 3, \dots, m\}$
x_{ijk}	当电物流车 k 由节点 i 到节点 j 时, $x_{ijk} = 1$, 否则 $x_{ijk} = 0$
y_{ik}	当电物流车 k 在充电站 i 处充电时, $y_{ik} = 1$, 否则 $y_{ik} = 0$

2.2 加权 AP 聚类算法

AP(affinity propagation)算法称为邻近传播算法或者亲和度传播算法,其基本思想是将所有数据点看成潜在的聚类中心,根据点与点之间的距离自主划分聚类区域,算法中每一个点的权重等值。作为大规模的电动物流车的配送问题,区域划分可以明显降低计算复杂度。由于配送点和充电桩的权重不等价,在考虑配送和充电方案时,区域的划分不能简单自主聚类,通过设置每个点的权重来控制 AP 聚类结果。配送点 i 的权重设置为送货量和取货量之和,即 $D_i + R_i$,充电桩的权值设置为所有配送点权值的最大值 $\max\{D_i + R_i\}, i \in N \cup C$ 。加权 AP 算法的初始化相似度矩阵 S 的各元素为

$$s(i, k) = -\frac{\sqrt{(x_i - x_k)^2 + (y_i - y_k)^2}}{(D_i + R_i) \times (D_k + R_k)}, i \neq k, \quad (1)$$

式中: $s(i, k)$ 表示第 i 个配送点到第 k 个配送点的带权欧式距离的负值,当 $i = k$ 时,更新相似度矩阵 S 对角线的值为相似度矩阵中所有值的最小值;

初始化吸引度矩阵 R 为 0 矩阵,其中 $r(i, k)$ 描述了配送点 k 作为配送点对象 i 的聚类中心的程度,初始化归属度矩阵 A 为 0 矩阵,其中 $a(i, k)$ 描述了配送点 i 选择配送点 k 作为聚类中心的合适程度。

更新吸引度矩阵 R

$$r_{t+1}(i, k) = \begin{cases} s(i, k) - \max_{j \neq k} \{a_t(i, j) + r_t(i, j)\}, i \neq k \\ s(i, k) - \max_{j \neq k} \{s(i, j)\}, i = k \end{cases}, \quad (2)$$

式中: t 为迭代序号; j 表示除配送点 k 的任意一个配送点; $r(i, k)$ 表示配送点 k 成为配送点 i 的聚类中心的证明,如果 $r(i, k) > 0$,越大表示配送点 k 成为聚类中心的能力越强。

更新归属度矩阵 A ,

$$a_{t+1}(i, k) = \begin{cases} \min\{0, r_{t+1}(k, k) + \sum_{j \neq i, k} \max\{r_{t+1}(j, k), 0\}\}, i \neq k \\ \sum_{j \neq k} \max\{r_{t+1}(j, k), 0\}, i = k \end{cases}, \quad (3)$$

式中: $a(i, k)$ 分为两种情况:当 $i \neq k$ 时,取所有大于 0 的吸引度累加,加配送点 k 作为聚类中心的可能性与 0 之间的小值;当 $i = k$,表示所有其他配送点传递给配送中心 k 的正吸引度的累加。

根据衰减系数 λ 对吸引度矩阵和归属度矩阵进行更新,所述衰减系数 λ 为 $[0, 1]$ 之间的正数;

$$\begin{aligned} r_{t+1}(i, k) &= \lambda \times r_t(i, k) + (1 - \lambda) \times r_{t+1}(i, k) \\ a_{t+1}(i, k) &= \lambda \times a_t(i, k) + (1 - \lambda) \times a_{t+1}(i, k) \end{aligned}, \quad (4)$$

重复执行式(2)~式(4),直到吸引度矩阵 R 和归属度矩阵 A 稳定并不再更新,当 $r(k, k) + a(k, k) > 0$ 就认为是配送点 k 是一个聚类中心,再根据其他配送点到聚类中心的最短欧式距离划分各簇,最终划分为 M 个簇。

2.3 电动物流车路径规划

利用加权 AP 算法把大规模的配送点和充电桩进行划分后,在簇内执行考虑充电策略的改进 NSGA-II 算法。建立以簇内电动物流车的总路程、簇内电动物流车行驶总时间、簇内时间惩罚成本为最小的多目标优化函数。

$$f_1 = \sum_{k \in K, i \in N, j \in N} x_{ijk}, \quad (5)$$

$$f_2 = \sum_{k \in K, i \in N, j \in N} \frac{d_{ij}}{s} x_{ijk} + \sum_{i \in C} t f_i, \quad (6)$$

$$f_3 = e_p \sum_{i \in N} \max(t_i - e_i, 0) + l_p \sum_{i \in N} \max(l_i - t_i, 0), \quad (7)$$

$$\text{s.t.} \sum_{j \in N, i \neq j} x_{ijk} = 1, \forall i \in N, k \in K, \quad (8)$$

$$\sum_{j \in V, i \neq j} x_{0jk} = 1, k \in K, \quad (9)$$

$$\sum_{i \in V, i \neq j} x_{jik} - \sum_{i \in V, i \neq j} x_{ijk} = 0, \forall j \in V, k \in K, \quad (10)$$

$$L_{i-1} + (R_i - D_i)L, \forall i \in N, \quad (11)$$

$$Q_i - \sum_{j \in V} ca \cdot d_{ij} \cdot x_{ijk}, \forall i \in V, k \in K, \quad (12)$$

$$Q_i + g \cdot tf_i Q, \forall i \in C, \quad (13)$$

$$x_{ijk} \in \{0; 1\}, \forall i \in V, \forall j \in V, k \in K, \quad (14)$$

$$y_{ik} \in \{0; 1\}, \forall i \in V, k \in K, \quad (15)$$

式(5)~式(7)分别为目标函数 f_1 电动物流车的总路程;目标函数 f_2 电动物流车配送过程花费的时间;目标函数 f_3 时间惩罚成本的计算;式(8)表示每个配送点仅被电动物流车 k 服务一次,式(9)表示电动物流车 k 从配送中心出发只服务一条配送线路,式(10)表示电动物流车 k 进入某个节点后必须离开。式(11)表示电动物流车在进入配送点 i 之前的装车容量要满足配送点 j 的货送货之差。式(12)表示配送车辆 k 从节点 i 出发的剩余电池容量 Q_i 要满足到达节点 i ;式(13)表示电动物流车在充电站 i 充电后的电池容量不能大于电池最大容量;式(14)和(15)是决策比变量约束。

2.4 非支配排序遗传算法

2.4.1 编码方式

本算法采用的智能编码方式为自然数编码,所有节点的编码方式为自然数,配送中心为 0,对配送点和充电站依次进行编码:1,2,3,4,5,..., m 。在某个簇的配送网络中,如有 10 个配送点编号依次为 1 到 10,2 个充电站编号为 11 和 12,若使用 2 辆电动物流车进行配送活动,一条完整个体编码形式为:0→8→3→12→4→0→1→10→11→5→7→9→6→2→0,其中,0→8→3→12→4→0 和 0→1→10→11→5→7→9→6→2→0 分别表示两条完整的配送环路,前一条环路展示了电动物流车从配送中心 0 出发,依次经过了配送点 8 和 3,然后在充电站 12 进行充电后,再经配送点 4,最后返回配送中心。

NSGA-II 的工作原理的伪代码所示,算法如下

输入:初始化种群

输出:新种群的第一层支配集

- 1: 输入初始化种群;
- 2: 执行非支配快速排序;
- 3: while(true)
- 4: 遗传操作生成子代;
- 5: 父代与子代个体合并;
- 6: if(未生成新父种群个体)
- 7: 执行快速非支配排序;
- 8: 拥挤距离计算
- 9: 选择合适个体组成新父种群;
- 10: if(进化代数 > Maxgen)
- 11: 选出新种群的第一层非支配集;
- 11: break;
- 12: 输出新种群的第一层支配集。

2.4.2 初始化种群

初始化个体编码的基本思想为:以电动物流车所在位置的作为参照中心,随机选取离车辆最近的 h 个配送点的其中一个进行配送,按照同一规则依次把配送点加入个体序列,再根据电动物流车的载货量和电池容量,分配车辆和插入充电站,具体步骤如下:

步骤 1: 获取配送中心和簇内各配送点的坐标信息, 计算簇内各配送点之间的欧式距离 $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$;

步骤 2: 车辆从配送中心出发, 从最近的 h 个配送点随机选取 1 个作为下一个配送点, 即 $v_m^1 = \text{rand}(\min_{\text{toph}} \{d_{0i}\})$, $m \in M, i \in V$, 其中函数 $\text{rand}()$ 表示随机选取一个配送点, \min_{toph} 表示选取最近的 h 个配送点。

步骤 3: 然后再从 v_m^1 出发, 按照步骤 2 规则, 选取下一个配送点 v_m^2 , 直到遍历完簇 m 内所有的配送点, 最后回到配送中心, 形成一条环路;

步骤 4: 根据电动物流车的载货量和电池容量约束以及充电站的位置信息, 进行车辆分配和插入充电站, 生成一条符合配送要求的个体;

步骤 5: 重复操作步骤 2~4, 生成满足约束条件的 N 条个体, 形成初始化种群。

2.4.3 快速非支配排序

根据式(5)~式(7)的目标函数, 比较任意 2 个体之间的支配和非支配关系, 找到种群中所有支配个体数为 0 的个体, 并保存在当前集合 F_1 中, 对于 F_1 中的每一个个体 i 其所支配的个体集合为 S_i , 遍历 S_i 中的每一个个体 b , 执行递减步骤, 如果执行完毕, 将个体 b 保存在集合中 H ; 在 F_1 中得到的个体为第一个非支配层个体, 并以 H 作为当前集合, 重复操作得到 $F_2, F_3 \dots$, 直到初始种群的个体关系被全部分层, 示意图如 2 所示。

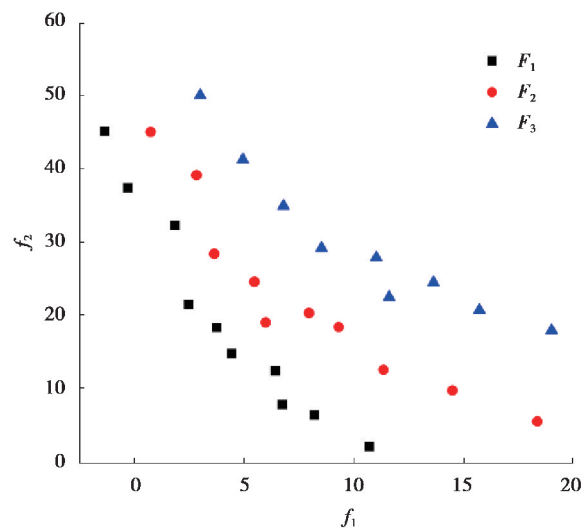


图 2 初始种群的个体关系

Fig. 2 Individual relations of the initial population

2.4.4 遗传操作

遗传操作包括: 个体的选择、交叉、变异, 从而产生新种群。由于考虑充电策略和充电站的分布, 遗传操作首先是删除充电站进行操作, 最后再根据耗电量插入序列。

a) 选择操作: 尽量选择质量较优的父代个体, 以 p_s 的概率选择父代, 根据各层的非支配关系, 选择概率为 $pF_{1s} > pF_{2s} > pF_{3s}$, 其中 pF_{1s} 表示的集合 F_1 选择父代个体的概;

b) 交叉操作: 从父代中选择差异性最大 2 个的父代交叉产生下一代个体, 交叉的原则以一个父代中节点 i 和 j 为相邻最近的配送点, 另外一个父代中节点 i 和 j 为相邻最远的 2 个配送点, 如图 3 所示, 2 个父代的区域进行交换, 把序列最长的子代随机删除重复配送点后得到新的子代;

c) 变异操作: 父代的序列按照概率 p_m 随机生成 2 不相等的自然数, 进行变异操作, 把父代中的两个配送点的位置调换, 得到新的子代, 如图 4 所示;

如果遗传操作后新子代的目标函数小于之前的子代则保留, 否则丢弃, 依次生成 N 个, 得到新一代种群, 把父代和新一代种群合并形成 $2N$ 的候选种群;

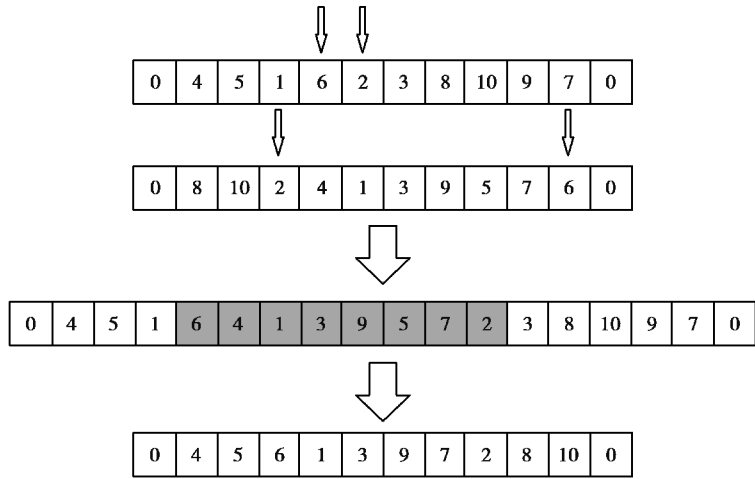


图 3 交叉操作

Fig. 3 Cross-over operation

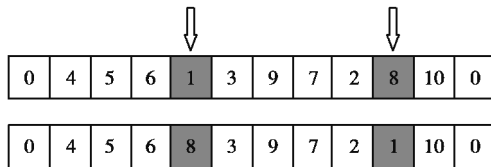


图 4 变异操作

Fig. 4 Mutation operation

2.4.5 拥挤距离计算

拥挤距离的计算保证了种群多样性,同时也是为了使得多目标的解在目标中间更加均匀。首先分别在 3 个目标函数下计算个体 i 在同一层中与相邻 2 个个体的平均距离,即以个体 $i+1$ 和 $i-1$ 为对角线形成的长度。如果长度越大则表示个体 i 与相邻点的拥挤距离越大,即多样性越好,就越应该保留,以防止局部最优的出现。通过遗传操作生成新的子代也使得 pareto 前沿更均匀。

3 仿真分析与验证

3.1 实验方法及算例场景设计

为验证本算法对路径规划和充电策略的有效性,在考虑实际场景容量和规模的情况下,指定 1 个配送中心、随机生成 90 个配送点和 10 个充电站,以及配送点对应的装卸货物量、服务的起止时间进行仿真分析。主要条件包含:电动物流车的最大载货量为 4 t,车载电池的最大容量 Q 为 40 kWh;每辆电动物流车的最大行驶距离 $Dis=150$ km,而配送车辆的电能消耗与行驶距离成正比,消耗速率 ca 为 3.75 km/KWh;电动物流车的行驶速度 s 为 40 km/h。时间窗惩罚成本系数 e_p 为 10 元/h; l_p 为 20 元/h,时间窗口小于 0.5 h 忽略惩罚,大于 0.5 h 记为 1 h。部分节点如表 2 所示:其中 0 代表配送中心,序号表示配送点。

表 2 部分节点的详细信息

Table 2 Details of some nodes

节点编号	x_i /km	x_i /km	D_i /t	R_i /t	e_i	l_i
0	50.00	50.00	0	0	0	100
1	67.89	56.67	0.2	0.1	0	1
2	36.17	51.77	0.3	0.3	1	2
3	89.48	99.54	0.3	0.3	2	4

续表2

节点编号	x_i/km	x_i/km	D_i/t	R_i/t	e_i	l_i
4	6.12	1.27	0.3	0	7	8
5	96.93	76.61	0.3	0.3	5	6
6	95.41	69.46	0.5	0.3	3	5
7	4.02	45.68	0.8	0.1	0	2
8	84.71	10.80	0.4	0.4	7	8
9	38.91	4.51	0.5	0.1	1	3
10	58.66	26.20	0.7	0.4	4	5
11	1.11	33.03	0.7	0.5	1	2
12	37.07	45.47	0.6	0.2	3	4
13	36.11	72.54	0.2	0.2	0	1
14	92.88	19.81	0.2	0.2	1	3
15	29.29	5.72	0.3	0.3	3	4
16	54.99	64.32	0.5	0.3	1	3
17	10.32	21.63	0.3	0.4	0	1
18	99.18	5.21	0.1	0.2	1	2
19	30.03	20.63	0.4	0.5	0	2
20	47.32	90.91	0.3	0.4	0	2
91	8.73	29.93	0	0	0	100
92	44.94	45.68	0	0	0	100
93	65.25	94.45	0	0	0	100
94	88.55	40.23	0	0	0	100
95	35.19	78.66	0	0	0	100

注:最早时间 e_i ,最晚时间 l_i ,是基于配送中心为起点的偏移量

3.2 实验参数设定

为了验证实验的有效性,并定性分析与其他方法的优势,加权 AP 聚类算法与 NSGA-II 的参数设置如下:衰减系数设置为 0.5,聚类算法最大迭代次数为 200 次,聚类中心无变化,迭代为 15 次。最近距离的配送点 h 设置为 5,每个簇初始化种群数为 100 个,各层选择操作的 p_s 值为 0.8,每次递减 0.1,交叉系数设置为 0.8,变异系数设置为 0.5,与 NSGA-II 比较的其他优化算法的参数设置为:MOEA/D^[18],COEA^[19],AMOSA^[20],MOEA/D 的初始种群数大小为 100,每一个子问题对应的领域大小为 20,COEA 外部档案大小为 100,AMOSA 外部档案大小也为 100,初始温度为 100,冷却系数为 0.95,结束温度为 2.5×10^{-6} 。

3.3 加权 AP 聚类算法分析

聚类是为了降低多目标优化的计算复杂度。在不考虑充电站的情况下,分析带聚类算法的多目标优化与不带聚类算法的差异。验证聚类方法对路径规划算法的优越性。实验分别以配送点数量为 200 以下规模来测试带聚类算法的 NSGA-II 与不带聚类算法的 NSGA-II 的时间复杂度的区别。NSGA-II 分别独立执行 50 次的平均时间为基准,多目标优化的运行时间分别如图 5 所示:可以明显看出,不带聚类的 NSGA-II 运行时间与带聚类的算法,当配送点在 40 个以下差异小,当配送点 40 数量在以上时,NSGA-II 的计算复杂度为 $O(N^2)$ 呈指增长,增长速度,带聚类算法呈线性增长。

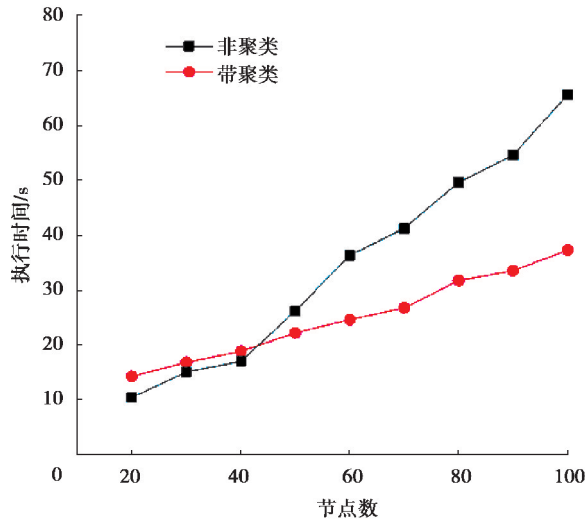


图 5 聚类与非聚类的执行时间比较

Fig. 5 Comparison of execution time between clustering and non-clustering

3.4 多目标优化算法比较

4 个多目标优化算法设置为电物流车到达充电站后,执行满充操作。图 6 给出了 4 种算法在 $f_1 \sim f_2$ 和 $f_1 \sim f_3$ 从三维映射到二维平面最终非支配解集前沿,在 $f_1 \sim f_2$ 总距离和总用时的关系图中可以看出,带 AP 聚类的 NSGA-II 算法求解路径规划时能在收敛性和多样性上取得较好的效果,该算法的前沿要明显优于其他 3 种算法,在前沿的两端,优化集中最短总路径能达到 1 277 km,总用时为 57.2 h,当优先考虑总用时的情况下,总用时为 48.26 h,总距离为 1 938.75 km,而其他 3 种算法 AMOSA 优化集中最短路径为 1 475.59 km,MODEA/D 的前沿的最短路径为 1 648.4 km,COEA 则为 1 490.2 km,由此可以分析得出该算法比其他 3 种算法更好平衡距离和总用时的关系。同样,图 7 是 $f_1 \sim f_3$ 距离和总惩罚金额的非支配解集前沿,AP-NSGA-II 大部分的前沿优于其他算法,虽然在总距离区间[1 800—2 000]的惩罚金额,AMOSA 算法稍优于 AP-NSGA-II 算法,但是总体来说,本算法在 3 个目标函数上表现优于其他算法。

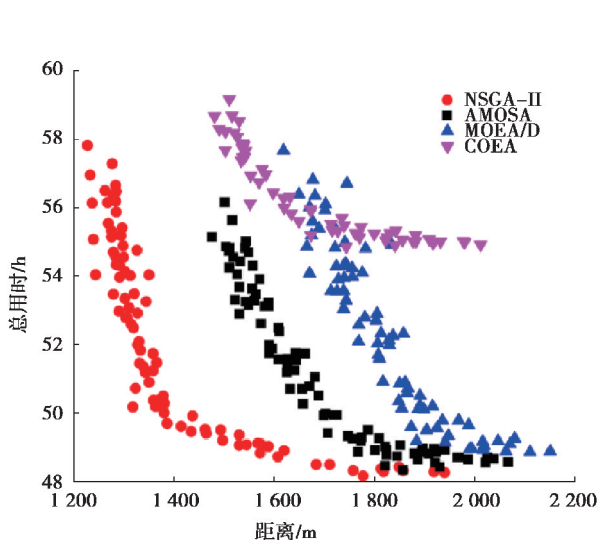


图 6 距离与总时间的关系

Fig. 6 The relationship between distance and total time

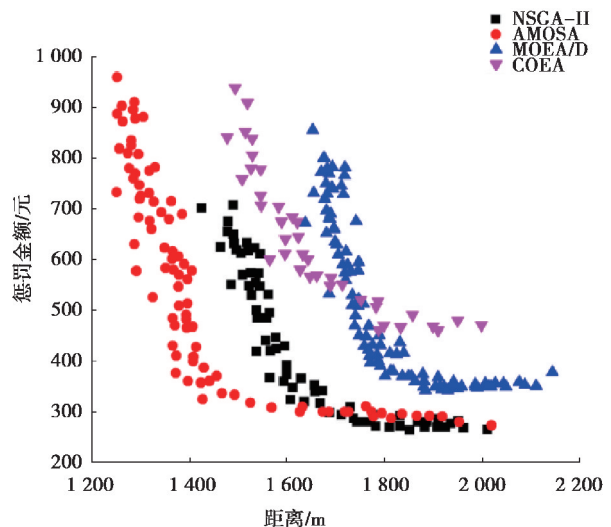


图 7 距离与惩罚金额的关系

Fig. 7 The relationship between distance and penalty amount

3.5 充电策略分析

电物流车在配送过程中,满充情况下势必增加单条路径时间和总时间,时间的增加会带来惩罚成本的

增加。考虑满充电情况的执行效果,根据充电站的位置和聚类程度,把它当成配送量为 0 的配送站进行访问,不单独考虑它为充电站,当经过充电站后进行满充操作以保证后续的行驶。为考察充电策略下的对目标函数的影响,将充电站节点不规划到多目标的路径中,以行驶距离消耗电量为条件,寻找最近的充电站,以满足后续配送的方式进行配送。

按照聚类结果选取 3 条典型的配送子线路,按照满充和部分充电来分析不同充电策略对配送的影响,结果如表 3 所示。

表 3 满充电和部分充电策略的比较
Table 3 Comparison of full charge and partial charge strategies

充电策略	车辆编号	车辆路径	充电位置	充电率 / %	行驶距离 / km	用时 / h
满充电	1	0→42→22→1→90→63→40→100→35→6→5→55→3→56→16→0	100	100	204.32	5.79
	2	0→48→95→79→29→53→81→0	95	100	184.98	5.20
	3	0→44→59→2→37→78→7→25→71→12→89→92→0	92	100	109.74	3.39
部分充电	1	0→42→22→1→90→63→35→6→5→55→3→56→100→40→16→0	100	18.56	177.83	4.81
	2	0→48→29→79→53→81→92→0	92	23.32	160.35	5.09
	3	0→44→59→2→37→78→7→25→71→12→89→0	—	—	104.74	2.62

对于车辆编号 1 的线路,在满充情况下,行驶距离为 204.32 km,总用时为 5.79 h,到达编号 100 的充电站后,执行 100% 充电后进行完成剩余线路;在部分充电情况下,编号 1 车辆的原始规划路线为 0→42→22→1→90→63→35→6→5→55→3→56→40→16→0,在到达编号 56 的配送点后,再去适配充电站,编号 100 的充电站加入节点 56 和 40 之间,电动物流车的充电率为 18.56%,行驶距离为 177.83 km,用时为 4.81 h。相同地,车辆编号 2 满充和部分充电率为分别是 100% 和 23.32%,距离分别为 184.98 km 和 160.35 km,用时分别为 5.20 h 和 5.09 h。车辆编号 3 在满充设置下编号 92 被规划到路径中,在部分充电情况下,由于行驶距离小于 150 km 而不需要充电,他们的行驶距离分别是 109.74 km 和 104.74 km,时间分别为 3.39 h 和 2.62 h。

4 结 论

基于聚类非支配排序算法(AP-NSGA-II)来解决电动物流车的多目标路径优化问题,建立了一种充电策略,通过设计加权 AP 聚类划分配送簇,避免了初始种群的随机性和盲目性,同时,聚类簇内配送点的规模降低了非支配排序算法的运行时间和复杂度,并根据充电站的分布和距离关系,电动物流车执行部分充电策略。最后,通过仿真实验证明该算法的有效性,比较了电动物流车满充和部分充电条件的差异性。

参考文献:

[1] 葛显龙,李祖伟,葛小波. 考虑灵活充电策略的带时间窗物流配送路径优化研究[J]. 控制理论与应用, 2020, 37(6): 1293-1301.
Ge X L, Li Z W, Ge X B. Research on logistics distribution route optimization with time window considering flexible charging strategy[J]. Control Theory & Applications, 2020, 37(6): 1293-1301. (in Chinese)

[2] Keskin M, Akhavan-Tabatabaei R, Çatay B. Electric vehicle routing problem with time windows and stochastic waiting times at recharging stations[J]. 2019 Winter Simulation Conference (WSC), 2019: 1649-1659.

[3] Lim S, Kubay M. Heuristic algorithms for siting alternative-fuel stations using the Flow-Refueling Location Model[J]. European Journal of Operational Research, 2010, 204(1): 51-61.

[4] 冯智泉,黄亦翔,李杰,等. 蚁群算法求电动汽车最优行驶路径与充电方案[J]. 机电一体化, 2013, 19(8): 43-48.
Feng Z Q, Huang Y X, Li J, et al. Ant colony algorithm for EV path and charging optimization[J]. Mechatronics, 2013,

- 19(8): 43-48. (in Chinese)
- [5] 文展, 唐康健, 李文藻. 一种改进粒子群优化算法在车辆路径问题的应用研究[J]. 重庆邮电大学学报(自然科学版), 2020, 32(5): 891-897.
- Wen Z, Tang K J, Li W Z. Research on application of vehicle routing problem using an enhanced particle swarm optimization[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2020, 32(5): 891-897. (in Chinese)
- [6] Pillac V, Gendreau M, Guéret C, et al. A review of dynamic vehicle routing problems[J]. European Journal of Operational Research, 2013, 225(1): 1-11.
- [7] Mourão M C, Almeida M T. Lower-bounding and heuristic methods for a refuse collection vehicle routing problem [J]. European Journal of Operational Research, 2000, 121(2): 420-434.
- [8] Bräysy O, Gendreau M. Vehicle routing problem with time windows, part I: route construction and local search algorithms[J]. Transportation Science, 2005, 39(1): 104-118.
- [9] Kritikos M N, Lappas P Z. Computational intelligence and combinatorial optimization problems in transportation science [J]. Advances in Core Computer Science-Based Technologies, 2021: 325-367. DOI:10.1007/978-3-030-41196-1_15.
- [10] 钱小宇, 葛洪伟, 周竞, 等. 基于扩容和双距离决策的多目标粒子群优化算法[J]. 重庆邮电大学学报(自然科学版), 2020, 32(3): 368-376.
- Qian X Y, Ge H W, Zhou J, et al. Multi-objective particle swarm optimization algorithm based on expansion and dual distance[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2020, 32(3): 368-376. (in Chinese)
- [11] 潘斌斌. 多目标路径规划问题的算法综述[J]. 重庆工商大学学报(自然科学版), 2012, 29(5): 78-84.
- Pan B B. Review of the algorithms of multi-objective routing programming problems[J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2012, 29(5): 78-84. (in Chinese)
- [12] Rabani M, Manavizadeh N, Boostani A, et al. A multi-objective model for the residential waste collection location-routing problem with time windows[J]. Journal of Industrial and Systems Engineering, 2020, 12(4): 227-241.
- [13] Mirabi M, Shokri N, Sadeghieh A. Modeling and solving the multi-depot vehicle routing problem with time window by considering the flexible end depot in each route[J]. International Journal of Supply and Operations Management, 2016, 3(3): 1373-1390.
- [14] Miranda D M, Branke J, Conceição S V. Algorithms for the multi-objective vehicle routing problem with hard time windows and stochastic travel time and service time[J]. Applied Soft Computing, 2018, 70: 66-79.
- [15] Erdelić T, Carić T. A survey on the electric vehicle routing problem: variants and solution approaches[J]. Journal of Advanced Transportation, 2019, 2019: 1-48.
- [16] Raeesi R, Zografos K G. The electric vehicle routing problem with time windows and synchronised mobile battery swapping[J]. Transportation Research Part B: Methodological, 2020, 140: 101-129.
- [17] Breunig U, Baldacci R, Hartl R F, et al. The electric two-echelon vehicle routing problem[J]. Computers & Operations Research, 2019, 103: 198-210.
- [18] Zhang Q F, Li H. MOEA/D: a multiobjective evolutionary algorithm based on decomposition[J]. IEEE Transactions on Evolutionary Computation, 2007, 11(6): 712-731.
- [19] Goh C K, Tan K C. A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization[J]. IEEE Transactions on Evolutionary Computation, 2009, 13(1): 103-127.
- [20] Bandyopadhyay S, Saha S, Maulik U, et al. A simulated annealing-based multiobjective optimization algorithm: AMOSA [J]. IEEE Transactions on Evolutionary Computation, 2008, 12(3): 269-283.