

doi: 10.7690/bgzdh.2021.11.015

## 智能优化算法在车辆配送方面的研究

程翔, 李海平, 姜立伟, 谢明化

(湖南华南光电(集团)有限责任公司, 湖南 常德 415007)

**摘要:** 针对已有算法对高维多目标问题存在多样性差且计算复杂的缺陷, 提出利用冲突信息分区的高维多目标并行进化的智能优化算法。利用目标间的冲突信息将目标空间划分为若干子区间, 独立进化以降低求解问题的难度; 在每个子区间中加入其他子区间的聚合信息, 考虑全局信息避免局部收敛; 再根据子区间中目标数目的不同, 采用并行独立优化算法缩小搜索空间, 避免削弱进化算子的作用, 提高算法优化性能。选取 3 种经典算法与该算法作对比, 验证其优劣性。对比结果表明: 该算法在车辆路径中可减少污染, 并能在满足客户要求的情况下实现企业成本最小化。

**关键词:** 车辆路径规划; 高维多目标; 冲突信息; 聚合; 并行

**中图分类号:** TP273 **文献标志码:** A

## Research on Intelligent Optimization Algorithm in Vehicle Distribution

Cheng Xiang, Li Haiping, Jiang Liwei, Xie Minghua

(Hunan Huanan Optoelectronic (Group) Co., Ltd., Changde 415007, China)

**Abstract:** In view of the shortcomings of existing algorithms for high-dimensional multi-objective problems such as poor diversity and complex computation, an intelligent optimization algorithm based on conflict information partition and high-dimensional multi-objective parallel evolution is proposed. The objective space is divided into several subintervals by using the conflict information among the objectives, and the independent evolution is used to reduce the difficulty of solving the problem; the aggregation information of other subintervals is added to each subinterval, and the global information is considered to avoid local convergence; then according to the different number of objectives in the subinterval, the parallel independent optimization algorithm is used to reduce the search space, so as to avoid weakening the role of the evolution operator. The optimization performance of the algorithm is improved. Three classical algorithms are compared with this algorithm to verify its advantages and disadvantages. The comparison results show that the algorithm can reduce the pollution in the vehicle routing, and can realize the enterprise cost minimization while meeting the customer requirements.

**Keywords:** vehicle routing planning; high dimensional multi-objective; conflict information; aggregation; parallel

### 0 引言

现实世界中的许多问题都由多个目标组成, 且目标之间存在相互耦合又冲突的关系, 约定将目标个数大于 3 的优化问题称之为高维多目标优化问题 (many-objective optimization problem, MaOP)。目前, 高维多目标优化在物流配送<sup>[1]</sup>、车辆路径规划<sup>[2]</sup>和网络通信<sup>[3]</sup>等领域的需求越加突出; 因此, 开展高维多目标进化算法的研究具有重要的理论意义和实用价值。已有的一些经典多目标进化算法<sup>[4-6]</sup>对于 MaOP 几乎都难以平衡解集的收敛性与多样性, 并且计算复杂。

目前有 2 类可行方法解决上述问题, 一是目标维数约减的算法<sup>[7]</sup>, 通过目标间的冗余, 删除一部分目标来降低目标维数; 二是种群并行进化算法<sup>[8]</sup>, 通过将目标空间分割成多个子区间, 再并行求解;

两者原理都是通过降低目标维数来求取, 但第一类算法势必会丢失一部分信息, 导致最终优化结果不准确, 所以笔者针对第一类算法进行分析。R. C. PURSHOUSE 等<sup>[9]</sup>详细讲述了这 3 种关系对目标间的影响。若目标之间只考虑一种目标间的关系, 算法效果更好。在文献[10]与[11]中, 都采用 Spearman 相关系数进行相关关系分析, 对变量的形式及其满足的分布没有任何要求, 两者不足之处在于划分后的子区间都是根据 Pareto 支配进化的, 而 Pareto 支配在高维优化问题中会失效。此外, 由于高维 MaOP 的搜索空间非常大, 极大削弱了进化算子的作用, 如重组、交叉和变异, 所以缩小目标空间能有效提高算法性能<sup>[12]</sup>。SCHÜTZE 等<sup>[13]</sup>已经证明空间的划分可提高 MOEA 的性能。

对于上述文献的分析, 笔者从目标维数、目标

收稿日期: 2021-07-11; 修回日期: 2021-08-20

作者简介: 程翔(1980—), 男, 湖南人, 工程师, 从事图像处理、人工智能、多目标优化、机器人控制研究。E-mail: 48851000@qq.com。

间的关系及缩小搜索空间 3 方面考虑,并在文献[11]思想框架的基础上(该算法简称 CP-MOEA),提出了一种利用冲突信息分区的高维多目标并行进化算法,简称 CP-MOPEA。

将该算法应用到车辆路径规划问题(vehicle routing problem, VRP)中,来验证其实用价值。目前,对于城市配送路径问题,国内外已有较多研究。起初,研究范围仅限于减少运输费用、优化配送车辆数量与行驶距离等单个方面<sup>[14]</sup>。之后,R. NECULA 等<sup>[15]</sup>使用改进的蚁群算法求解有时间窗约束的 VRP,不仅达到用户要求的目的,而且减少了车辆配送数目。我国对 VRP 的研究较晚,取得了显著成果。廖良才等<sup>[16]</sup>将遗传算法与 C-W 节约算法相结合,得到的混合算法收敛性能更优袁雨果等<sup>[17]</sup>创立的 4 步骤优化算法,可实现有效的货物

分配及路径优化。

大部分研究者是对一个配送中心进行路径规划<sup>[14]</sup>,而如今物流配送网络发展迅速,许多企业都建立了多个配送中心,且需在满足时间窗约束的条件下送达货物;因此,对于多车场车辆路径问题(multiple depot vehicle routing problem, MDVRP)的研究很有必要。为响应国家节能减排的号召,也必须降低车辆运输中所产生的 CO<sub>2</sub> 排量。

综上所述,根据实际需求,笔者重点分析了碳排放的 MDVRPTW 问题。

## 1 智能优化算法

### 1.1 算法总体框架

利用冲突信息分区的高维多目标并行进化算法,其总体如图 1 所示。

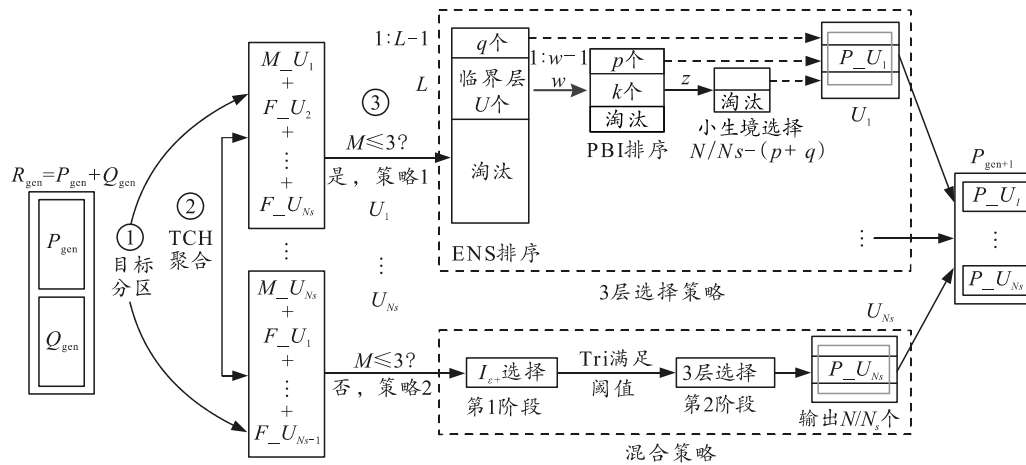


图 1 CP-MOPEA 总体

首先,初始种群经差分进化、高斯变异得到子代种群,合并子父代种群后,对其目标域,根据 Spearman 相关系数进行目标分区;其次,对每个分区目标进行 TCH 聚合,再在每个分区中加入其他分区聚合的信息,前期考虑种群收敛性,后期考虑种群多样性,该步骤仅在进化后半周期进行,不仅可以提高运算效率,而且考虑全局信息,避免种群陷入局部最优;最后,根据分区目标数目的大小,采用并行独立优化算法,从而达到更好的优化效果,若目标数不大于 3 个,则选用策略 1,否则选用策略 2,最终选出下一代优秀个体,依次循环,直到满足终止条件为止。

目标分区、信息交换、策略 1 及策略 2 等将在下文详细介绍。

图中:  $P_{gen}$  为父代种群;  $Q_{gen}$  为子代种群;  $R_{gen}$  为合并种群;  $P_{gen+1}$  为下一代精英种群;  $M_{U_i}$  为每

个子区间的目标集合;  $F_{U_i}$  为其他子区间聚合函数值;  $U_i$  为所分子区间;  $P_{U_i}$  为所选精英个体,  $i=1 \dots N_s$ ,  $N_s$  为所分子区间个数;  $Tri$  为极值点的变化幅度;  $M$  为子区间目标个数。

### 1.2 目标分区策略

首先介绍目标分区中用到的一个重要参数,是反映 2 变量间相似程度的大小,即皮尔森相关系数,用  $\rho$  表示,其数学表达如式(1)所示。所建“冲突矩阵”值为[-1,1]的范围内,“-1”为两目标完全冲突,则“1”为完全支持。

$$\rho_{xy} = \frac{\sum_i (x_i - \bar{X})(y_i - \bar{Y})}{\sqrt{\sum_i (x_i - \bar{X})^2 \sum_i (y_i - \bar{Y})^2}}, \quad -1 \leq \rho_{xy} \leq 1. \quad (1)$$

其次以 3—8 目标 WFG2 测试函数为例说明算法分区思想,根据上式计算出 3—8 目标的冲突矩阵如表 1 所示。其特点是对角线元素均为 1,左下三

角元素与右上三角元素对称；因此，只需求取任意 三角元素即可。

表 1 WFG2 的冲突矩阵

| 目标数 | 3        | 4        | 5        | 6        | 7        | 8        |
|-----|----------|----------|----------|----------|----------|----------|
| 3   | 1.000 0  | 0.041 6  | 0.019 1  | -0.001 7 | -0.024 1 | -0.054 5 |
| 4   | 0.041 6  | 1.000 0  | 0.026 8  | -0.002 3 | -0.033 8 | -0.076 4 |
| 5   | 0.019 1  | 0.026 8  | 1.000 0  | -0.003 3 | -0.047 8 | -0.107 9 |
| 6   | -0.001 7 | -0.002 3 | -0.003 3 | 1.000 0  | -0.068 1 | -0.153 9 |
| 7   | -0.024 1 | -0.033 8 | -0.047 8 | -0.068 1 | 1.000 0  | -0.224 7 |
| 8   | -0.054 5 | -0.076 4 | -0.107 9 | -0.153 9 | -0.224 7 | 1.000 0  |

注：表中数据是以 WFG2 的真实 Pareto 前沿解为例计算出的冲突矩阵。

最后，判断  $M/N_s$  的余数是否为 0，若是，则所有子区间具有相同维度为  $M/N_s$ ；否则，一个子区间具有维度为  $M/N_s$ ，其余的子区间维度为  $(M/N_s)+1$ ，其中  $M$  为目标数， $N_s$  为子区间个数。划分原理是目标之间的冲突度由大到小依次划分，若只划分 2 个子区间，即  $N_s=2$ ，得出目标数较小的子区间目标为 6, 7, 8，则剩余目标在另一个子区间里。

### 1.3 子区间信息交换策略

子区间内聚合采用 Tchebycheff 法，其计算公式如式(2)所示。原因如下：一是算法前期考虑收敛性，后期考虑多样性，而 Tchebycheff 聚合方法具有分布性好的特性，正好适用于后期；二是与性能较好的 PBI 聚合相比，Tchebycheff 聚合的计算复杂度远小于 PBI 聚合。

$$\min g^{te}(x|\lambda, z^*) = \max_{1 \leq i \leq m} \{\lambda_i |f_i(x) - z_i^*|\} \quad (2)$$

式中： $x \in \Omega$ ， $z^* = (z_1^*, \dots, z_m^*)$  为参考点， $\lambda = (\lambda_1, \dots, \lambda_m)$  为权重向量； $f(x)$  为目标函数； $m$  为目标个数。

在每个子区间内加入其他子区间的聚合函数值，融入全局信息，不仅有利于寻找全局最优解，而且也避免种群陷入局部最优。

由 1.2 节中 3—8 目标 WFG2 为例，所分的 2 个目标空间分别为  $M_{-U_1} = \{f_3, f_4, f_5\}$  和  $M_{-U_2} = \{f_6, f_7, f_8\}$ ，且由式(2)计算所得的 2 空间聚合函数值分别为  $F_{-U_1}$  与  $F_{-U_2}$ ，融入全局信息表达：

$$\begin{cases} \min P_{-U_1} = (M_{-U_1}, F_{-U_2}) \\ \min P_{-U_2} = (M_{-U_2}, F_{-U_1}) \end{cases} \quad (3)$$

### 1.4 算法流程

主要算法流程如下。

#### 1) CP-MOPEA 算法流程：

Algorithm: CP-MOPEA

N: Population size,  $gen$ : Current evolutionary algebra of population,  $M$ : Target number,  $\max gen$ : Maximum evolutionary algebra Tri: Extreme point

change amplitude, Ref: Penalty vector,  $N_s$ : Number of sub intervals,  $V$ : Number of decision variables.

- 1: Set  $Tri = \inf$ ,  $N_s = 2$  and  $Ref_i = 0, i = 1, 2, \dots, N$ ;
- 2: Get the initial population  $P_0$ ;
- 3: Calculating the value of the objective function  $fit(P_0)$ ;
- 4: for  $gen = 1$ : max  $gen$ ;
- 5: Generation of offspring population  $Q_{gen}$  by recombination variation of  $P_{gen}$ ;
- 6: Merge  $P_{gen}$  and  $Q_{gen}$  into  $R_{gen} (2N)$ ;
- 7: Calculating the value of the objective function  $fit1 (R_{gen})$ ;
- 8: Finding conflict matrix correlation of target from  $fit1$ , Sub interval  $Feng(l)$  is divided by information of conflict matrix,  $l = 1 \dots N_s$ ;
- 9: if  $gen = 1$ ;
- 10: Set the values of  $P(l)$ , calculating the number of reference points  $H(l)$  according to the formula  $H = C_{M+P-1}^P$ , generating reference point sets  $r$  and  $s, l = 1 \dots N_s$ ;
- 11: end;
- 12: if  $gen > (\max gen / 2)$ ;
- 13: for  $h = 1 : 2 * N$ ;
- 14: Renewal of Ideal Points  $Z(l), l = 1 \dots N_s$ ;
- 15: Get aggregate function  $scalarfit(l)$  from TCH,  $l = 1 \dots N_s$ ;
- 16: end;
- 17:  $Feng(l) = [R_l(:, 1:V) \quad Feng(l) \quad scalarfit(l)]$ ,  $l = 1 \dots N_s$ ;
- 18: else;
- 19:  $Feng(l) = [R_l(:, 1:V) \quad Feng(l)]$ ,  $l = 1 \dots N_s$ ;
- 20: end;
- 21: if  $M(l) \leq 3, l = 1 \dots N_s$ ;
- 22:  $Feng(l)$  effective non dominated  $Feng\_1(l)$ ;
- 23: For  $Feng\_1(l)$ , select  $N/2$  individuals from strategy 1 and put them into chromosome 1;
- 24: else;
- 25: if  $gen = 1$ ;
- 26: Set the values of  $P\_1(l)$ , calculating the number of reference points  $H\_1(l)$  according to the

formula  $H = C_{M+P-1}^P$ , generating reference point sets  $r1$  and  $s1$ ,  $l=1 \dots N_S$ ;

27: end;

28: For  $Feng\_1(l)$ , select  $N/2$  individuals from strategy 2 and put them into chromosome2;

29: end;

30:  $P_{gen}=[\text{chromosome1 chromosome2}]$ ;

31: Calculating the value of the objective function  $fit2(P_{gen})$ ;

32:  $P_{gen}=[P_{gen} \text{ fit2}]$ ;

33: end.

3 层选择策略流程:

Strategy 1

Input:  $N$ ,  $gen$ ,  $\max gen$ ,  $M(l)$ ,  $N_S$ ,  $V$ ,  $Feng\_1(l)$

Output: chromosome1.

1: if  $gen > \max gen/2$ ;

2: Set a larger  $P$  value and calculate the number of reference points  $H$  from  $H = C_{M+P-1}^P$ ;

3: Generating reference point set  $Z$ ;

4: else;

5: Set a small  $P$  value and calculate the number of reference points  $H$  from  $H = C_{M+P-1}^P$ ;

6: Generating reference point set  $Z$ ;

7: end;

8:  $pop=N/N_S$ ;

9: Producing  $S_{gen}$ ,  $Feng\_1(l)$  anterior L-layer individuals greater than or equal to  $pop$  through ENS domination;

10: Renewal of ideal points  $Z^*(S_{gen})$ ;

11: Decomposition  $Feng\_1(l)$  into the front L-1 layer and L layer into S-L-1 (q) and S-L (critical layer) of different archives sets, respectively;

12: Constructing Hyperplanes<sup>[23]</sup> for S-L;

13: Normalization(S-L);

14: PBI-2 clustering (S-L);

15: Hierarchical  $\{F_1, F_2, \dots, F_N\}$  in C according to aggregate function value;

16: Individuals selected in turn by PBI were placed in Population  $Y_{gen}$ ;

17: Decomposition  $Y_{gen}$  into the front  $w-1$  layer and  $w$  layer into  $Y-w-1$  and  $Y-w$ (critical layer) of different archives sets, respectively;

18: For  $Y-w$ , select  $pop-(p+q)$  individual from niche and put it into  $Y-z$ . Combine L-1layer,  $w-1$ layer and  $z$  layer to get chromosome1.

2) 混合策略流程:

Strategy 2

Input:  $gen$ ,  $M(l)$ ,  $Tri$ ,  $Feng\_1(l)$ ,  $H\_1(l)$ ,  $r1$ ,  $s1$ ,

Ref, Tri.

Output: chromosome2.

1: if  $gen < 2$ ;

2: Operation  $Feng\_1(l)$  finds the boundary solution set  $BS_i, i=1, 2, \dots, M$  and finds the extreme point  $B$ ;

3: end;

4: if  $gen > 50$ ;

5: Update Tri according to formula (4);

6: end;

7: if  $Tri \geq 0.6$ ;

8: phase=1;

9: Phase 1 algorithm;

10: else;

11: phase=2;

12: Phase 2 algorithm;

13: end;

14: To update Ref and  $BS$  (See literature 8);

15: Finding New Extremum Points  $B$  ;

16: if  $\max gen - gen = 50$ ;

17: To update  $s$  (See literature 8);

18: end;

19: Output chromosome2.

第 1 阶段:

Phase 1

Input:  $Feng\_1(l)$ ,  $H\_1(l)$ ,  $r1$ .

Output:  $P_{gen}=P_{one}$ .

1: Using PBI-1 and  $r$  as reference line,  $Feng\_1(l)$  is divided into  $H\_1(l)$  sub-regions  $R(i)$ ;

2: for  $i=1:H1$ ;

3:  $R(i)$  selects  $I_{\epsilon+}$  independently;

4:  $P_{one}=[P_{one}; R(i)]$ ;

5: end;

6:  $P_{gen}=P_{one}$ .

第 2 阶段:

Phase 2

Input:  $Feng\_1(l)$ ,  $H\_1(l)$ ,  $r1$ ,  $gen$ ,  $B$ .

Output:  $P_{gen}$ .

1:  $Feng\_1(l)$  is divided into inner and outer intervals ( $P_{in}$  and  $P_{out}$  respectively, and  $K$  is the number of individuals in the inner interval) by using extreme point  $B$ ;

2:  $pop=N/N_S$ ;

2: if  $K < pop$ ;

3:  $P_{two}=P_{in}$ ;

4: According to the minimum distance from the ideal point; select  $(pop-K)$  individuals from  $P_{out}$  and put them into  $P_{two}$ ;

- 5: else;
- 6: Strategy 1;
- 7: end;
- 8:  $P_{gen}=P_{two}$ .

## 2 碳排放的 MDVRPTW 模型建立

### 2.1 相关规定及定义

#### 2.1.1 涉及目标

- 1) 最小化路径总长度(所有路径长度之和);
- 2) 最小化成本(车辆成本、时间窗惩罚费用、燃油成本及碳交易费用);
- 3) 最小化运输时间(所有客户等待时间之和为所有车辆行驶时间和所有客户等待时间);
- 4) 最小化最大容量差(运输车辆两两之间最小之差)。

#### 2.1.2 问题总假设

- 1) 某企业有 1 个或 3 个配送中心;
- 2) 每个配送中心有多辆类型相同的车;
- 3) 车辆在一次分配任务中仅配送一次;
- 4) 车辆不超载;
- 5) 车辆配送完成后返回到原车场;
- 6) 每个客户仅且服务一次;
- 7) 每个客户都接受服务的时间窗;
- 8) 配送中不存在其他突发情况。

满足以上期望条件后, 设每辆车载重量为 5 t, 且送货过程中匀速行驶车速为  $v=60$  km/h, 迟到惩罚  $ad$  为 60 元/h, 提前惩罚  $de$  为 30 元/h, 碳交易价格  $C_e$  为 173 元/t, 燃油费用为 5.6 元/L, 车辆固定成本平均为 250 元。

#### 2.1.3 建立模型所需参数符号及变量定义

- $M$ : 车场集合,  $M=\{1,2,\dots,m\}$ , 共有  $m$  个车场;
- $K$ : 车辆集合,  $K=\{1,2,\dots,k\}$ , 每个车场有  $k$  辆车;
- $N$ : 客户集合,  $N=\{1,2,\dots,n\}$ , 共有  $n$  个客户;
- $q_i$ : 第  $i$  个客户的货物需求量;
- $s_i$ : 第  $i$  个客户的服务时间;
- $Q$ : 车辆最大容量;
- $Q_1$ : 碳配额, 即碳交易量=企业碳排放量-碳排放权限额;
- $a_i$ : 第  $i$  个客户要求最早送达的时间, 即时间窗约束上限值;
- $b_i$ : 第  $i$  个客户要求最晚送达的时间, 即时间窗约束下限值;

- $ad$ : 早到惩罚系数;
- $de$ : 晚到惩罚系数;
- $D$ : 最大行驶距离;
- $C_e$ : 单位碳交易价格;
- $d_{ij}$ : 从客户  $i$  到客户  $j$  的距离;
- $x_{ijmk}$ : 是否存在第  $m$  个配送中心出发的第  $k$  辆车从客户  $i$  到客户  $j$ , 是则变量为“1”, 否则为“0”;
- $veh\_cost$ : 每辆汽车的固定成本费用;
- $veh\_num$ : 完成送货时的总车辆数目;
- $\lambda, \beta, \gamma$ : 影响燃油的重要指标, 其表达式分别为:  $\lambda = \varepsilon / (\delta\varphi)$ ,  $\beta=0.5C_dAP$  与  $\gamma=1/(1\ 000*n_{ij}*\eta)$ ;
- $\Phi$ : 一升燃油转换为  $CO_2$ (单位: kg) 的换算系数。

### 2.2 数学模型

假设以某 3 省的省会为配送中心。3 省内的其他城市视为客户。

考虑碳排放的 MDVRPTW 的数学模型如下:

$$\min f_1 = \sum_{m \in M} \sum_{k \in K} \sum_{i, j \in N} d_{ij} x_{ijmk}; \quad (4)$$

$$\min f_2 = F_1 + F_2 + C_r \times F_3 + F_4; \quad (5)$$

$$\min f_3 = \sum_{m \in M} \sum_{k \in K} \sum_{i, j \in N} W_{ijmk}; \quad (6)$$

$$\min f_4 = \max_{k, k'} \left( \sum_{i \in N} q_i x_{imk} - \sum_{i \in N} q_i x_{imk'} \right). \quad (7)$$

其中:

$$F_1 = veh\_cost * veh\_num; \quad (8)$$

$$F_2 = ad \sum_{i=1}^n \max[a_i - t_i, 0] + de \sum_{i=1}^n \max[t_i - b_i, 0]; \quad (9)$$

$$F_3 = \sum_{m \in M} \sum_{k \in K} \sum_{i, j \in N} \left( \delta s V \lambda d_{ij} x_{mkij} / v_{ij} + \omega \gamma \lambda d_{ij} x_{mkij} \right. \\ \left. + \lambda \gamma d_{ij} q_{ij} + \beta \lambda \gamma d_{ij} x_{mkij} (v_{ij})^2 \right); \quad (10)$$

$$F_4 = C_e * (\Phi * F_3 - Q_1); \quad (11)$$

$$\text{s.t.} \quad \sum_{i=1}^N x_{ijmk} = 1 \quad j=1, 2, \dots, N, \quad k=1, 2, \dots, K, \\ m=1, 2, \dots, M; \quad (12)$$

$$\sum_{j=1}^n x_{ijmk} = 1 \quad i=1, 2, \dots, N, \quad k=1, 2, \dots, K, \\ m=1, 2, \dots, M; \quad (13)$$

$$\sum_{i=0}^N \sum_{j=0}^N q_i x_{ijmk} \leq Q \quad k=1, 2, \dots, K, \\ m=1, 2, \dots, M; \quad (14)$$

$$\sum_{i=0}^N \sum_{j=0}^N d_{ij} x_{ijmk} \leq D \quad k=1, 2, \dots, K, \quad m=1, 2, \dots, M; \quad (15)$$

$$x_{ijmk} \in \{0,1\} \quad i, j = 1, 2, \dots, N, \quad k = 1, 2, \dots, K, \\ m = 1, 2, \dots, M; \quad (16)$$

$$t_i \in [a_i, b_i] \quad \forall i \in \{2, 3, \dots, N\}. \quad (17)$$

式(4)为车辆运输货物行驶路径总和。式(5)为花费成本总和，由 4 部分组成：1) 式(6)为车辆车本费用；2) 式(7)为表示违反时间窗约束产生的费用；3) 式(8)为车辆行驶产生的燃油费用；4) 式(9)为碳交易费用。式(10)为所有客户等待时间之和。式(11)为两两运输车辆实际所装货物重量之差。式(12)为车辆  $k$  仅有一次从客户  $i$  出发的机会。式(13)为车辆  $k$  仅有一次到达客户  $j$  的机会。式(14)为配送车辆不能超载。式(15)为配送车辆不超过最大行驶距离的限制。式(16)为变量在  $0 \sim 1$  的范围。式(17)为第  $i$  个客户的时间窗约束。

### 2.3 参数设置

3 车场路径规划中，所设置种群个数为 220，最大进化代数为 200，交叉概率  $P_c=0.8$ ，变异概率  $P_m=0.1$ ，客户数量  $N=32$ ，其余参数与数据在文献[18]中有详细介绍。

### 2.4 优化结果

为了体现实验数据的可比性，使用 2 种表示方式：1) 不做数据处理的单车场实验数据，如表 2 所示；2) 归一化的单车场实验数据，如表 3 所示。为体现算法对 VRP 的优劣性，选取 3 种算法与之作对比，选取经典的 MOEA/D<sup>[19]</sup>与改进 Two-Arch2 算法<sup>[20]</sup>体现算法的有效性，选取最新的基于正态分布和自调节算子及自适应  $\varepsilon$  截断算法<sup>[21]</sup>(STOAE)，是一种约束进化算法，因 VRP 是一类含有约束的实际问题；因此，该算法与之相比更具先进性。

表 2 3 车厂实验对比

| 算法名称      | 总路径/km           | 成本/元             | 时间/h          | 容量差/kg          |
|-----------|------------------|------------------|---------------|-----------------|
| CP-MOPEA  | 11 560.71        | <b>12 794.71</b> | <b>271.91</b> | 3 060.22        |
| STOAT     | <b>10 810.61</b> | 15 923.23        | 290.45        | 2 951.13        |
| MOEA/D    | 19 036.65        | 21 147.85        | 485.25        | 4 461.53        |
| Two-Arch2 | 12 610.63        | 13 421.77        | 325.39        | <b>2 233.33</b> |

表 3 归一化后的 3 车场实验对比

| 算法名称      | 总路径            | 成本             | 时间             | 容量差            |
|-----------|----------------|----------------|----------------|----------------|
| CP-MOPEA  | <b>0.277 3</b> | <b>0.313 6</b> | <b>0.263 4</b> | 0.623 5        |
| STOAT     | 0.464 7        | 0.472 7        | 0.439 5        | <b>0.367 4</b> |
| MOEA/D    | 0.373 8        | 0.360 0        | 0.422 8        | 0.480 3        |
| Two-Arch2 | 0.399 5        | 0.392 0        | 0.412 2        | 0.399 1        |

注：使数据比较显示更加明确，加粗表示最优。

实验最终结果是将 220 个优化结果求其平均值，更能体现算法整体的优化性能，表 2 结果显示，单从优化结果不能直观看出哪种算法更优；因

此，采用归一化处理如表 3 所示，将不同量纲的数据处于同一级别，便于比较，对于实际问题，最优真实解集未知，无法用指标衡量，可行方法就是将处理数据累加求和转化为单目标进行比较，由表 4 可看出 CP-MOPEA 优化性能更好，Two-Arch2 算法次之，MOEA/D 位列第 3，STOAE 位列第 4，该算法对解决 VRP 取得较好效果。

表 4 目标总和

| 算法名称            | 总和      |
|-----------------|---------|
| CP-MOPEA-VRPTW  | 1.477 8 |
| STOAT-VRPTW     | 1.745 8 |
| MOEA/D-VRPTW    | 1.636 9 |
| Two-Arch2-VRPTW | 1.602 8 |

## 3 结论

笔者提出的智能优化算法在所建车辆路径模型中得到了验证，与对比算法相比有更大的实用价值。该算法利用目标间的冲突信息将目标空间进行划分，独立进化不仅能降低求解问题的难度，而且融入划分区域的聚合信息，能避免局部收敛。采用并行独立优化算法，在缩小的目标空间进行搜索，避免削弱进化算子的作用，从而提高算法优化性能。

## 参考文献：

- [1] ZHU C, CAI X, FAN Z, et al. A two-phase many-objective evolutionary algorithm with penalty based adjustment for reference lines[C]// Evolutionary Computation. 2016.
- [2] 张凯凯. 面向车辆路径问题的智能优化算法研究[D]. 泉州：华侨大学，2020.
- [3] 黄伟. 多目标进化算法在通信网络中的应用研究[J]. 湖北农机化，2017(6): 60-60.
- [4] DEB K, JAIN H. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving problems with box constraints[J]. IEEE Trans Evol Comput. 2014, 18(4): 577-601.
- [5] 刘建昌, 李飞, 王洪海, 等. 进化高维多目标优化算法研究综述[J]. 控制与决策, 2018, 33(5): 879-887.
- [6] 孙靖, 巩敦卫. 进化高维多目标优化研究进展[J]. 控制理论与应用, 2018, 35(7): 928-938.
- [7] 罗乃丽, 李霞, 王娜. 利用冲突信息降维的进化高维目标优化算法[J]. 信号处理, 2017, 33(9): 1169-1178.
- [8] CAO B, ZHAO J, LIU X, et al. Multi-objective recommendation optimization via utilizing distributed parallel algorithm[J]. Future Generation Computer Systems, 2017, 86: 1259-1268.