计算机科学
COMPUTER SCIENCE

基于分层学习和差分进化的混合PSO算法求解车辆路径问题

陈莹，黄佩萱，陈锦萍，王祖怡，沈映珊，樊小毛

引用本文

陈莹，黄佩萱，陈锦萍，王祖怡，沈映珊，樊小毛基于分层学习和差分进化的混合PSO算法求解车辆路径问题[J]. 计算机科学, 2022, 49(11A): 210800271-7.

CHEN Ying, HUANG Pei-xuan, CHEN Jin-ping, WANG Zu-yi, SHEN Ying-shan, FAN Xiao-mao. Hybrid Particle Swarm Optimization Algorithm Based on Hierarchical Learning and Different Evolution for Solving Capacitated Vehicle Routing Problem [J]. Computer Science, 2022, 49(11A): 210800271-7.

相似文章推荐（请使用火狐或 IE 浏览器查看文章）

Similar articles recommended (Please use Firefox or IE to view the article)

基于FWA-PSO-MSVM的船舶区域配电电力系统故障诊断

Fault Diagnosis of Shipboard Zonal Distribution Power System Based on FWA-PSO-MSVM

计算机科学, 2022, 49(11A): 210800209-5. https://doi.org/10.11896/jsjkx.210800209

基于差分进化算法的字符对抗验证码生成方法

Adversarial Character CAPTCHA Generation Method Based on Differential Evolution Algorithm

计算机科学, 2022, 49(11A): 211100074-5. https://doi.org/10.11896/jsjkx.211100074

基于DE-lightGBM模型的上市公司高送转预测实证研究

Empirical Study on the Forecast of Large Stock Dividends of Listed Companies Based on DE-lightGBM

计算机科学, 2022, 49(11A): 211000017-7. https://doi.org/10.11896/jsjkx.211000017

基于局部梯度强度图的动态规划检测前跟踪算法

Dynamic Programming Track-Before-Detect Algorithm Based on Local Gradient and Intensity Map

计算机科学, 2022, 49(8): 150-156. https://doi.org/10.11896/jsjkx.210700135

多无人机使能移动边缘计算系统中的计算卸载与部署优化

Computation Offloading and Deployment Optimization in Multi-UAV-Enabled Mobile Edge Computing Systems

计算机科学, 2022, 49(6A): 619-627. https://doi.org/10.11896/jsjkx.210600165

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210800271

基于分层学习和差分进化的混合 PSO 算法求解车辆路径问题

陈 莹 黄佩萱 陈锦萍 王祖怡 沈映珊 樊小毛
华南师范大学计算机学院 广州 510631
(cying@m.scnu.edu.cn)

摘 要 车辆路径问题旨在求解每辆车的服务路线,使其在完成配送任务的情况下行驶距离之和最短,是运筹学中经典的组合优化问题,属于 NP 难问题,且具有较高的理论意义与实际应用价值。针对该问题,提出了一种基于分层学习和差分进化的混合粒子群优化算法(Hybrid Particle Swarm Optimization Algorithm Based on Hierarchical Learning and Different Evolution,DE-HSLPSO)。DE-HSLPSO 中引入了分层学习策略,以适应度值和迭代次数为依据将种群粒子动态划分为 3 层,在前两层粒子的进化过程中引入了社会学习机制,而第三层粒子进行差分进化,通过变异和交叉有效地增加粒子的多样性,从而开拓空间,有利于跳出局部最优。通过在经典的 CVRP 数据集上进行仿真实验,来探究 DE-HSLPSO 各部分对整体性能的影响,实验证明分层策略与差分进化均可提升算法的整体性能。另外,在 7 个基本算例上对 DE-HSLPSO 与其他优化算法进行了测试,综合时间与最优解进行对比,结果表明 DE-HSLPSO 的求解性能优于其他对比算法。

关键词:分层学习;社会学习;差分进化;粒子群优化算法;车辆路径问题
中图法分类号 TP301

Hybrid Particle Swarm Optimization Algorithm Based on Hierarchical Learning and Different Evolution for Solving Capacitated Vehicle Routing Problem

CHEN Ying, HUANG Pei-xuan, CHEN Jin-ping, WANG Zu-yi, SHEN Ying-shan and FAN Xiao-mao
School of Computer Science, South China Normal University, Guangzhou 510631, China

Abstract The purpose of the vehicle routing problem(VRP) is to search the service route of each vehicle, so as to minimize the sum of driving distances in the case of completing all of the distribution tasks. CVRP, a classical combinatorial optimization problem in operations research, belongs to NP-hard problem and has high theoretical significance and practical value. In order to solve this problem, a hybrid particle swarm optimization algorithm based on hierarchical learning and different evolution(DE-HSLPSO) is proposed. First, the hierarchical learning strategy is introduced and the population particles are divided into three layers according to their fitness values and number of iterations. Secondly, the social learning mechanism is introduced in the evolution of the first two layers of particles, while particles in the third layer carry out differential evolution which effectively increases the diversity of particles, thus expanding the space and jumping out of local optimal. Simulation experiment whose examples are taken from the classical CVRP data sets explores the impact of each part of DE-HSLPSO on the overall performance. It is found that both hierarchical strategy and differential evolution can improve the overall performance of the algorithm. In addition, DE-HSLPSO and other algorithms are tested on seven benchmark examples. With comprehensive comparison of time and optimal solution, the result shows that the solution performance of DE-HSLPSO is better than that of other algorithms.

Keywords Hierarchical learning, Social learning, Differential evolution, Particle swarm optimization algorithm, Capacitated vehicle routing problem

1 引言

车辆路径问题(Vehicle Routing Problem,VRP)由 DANTZING 与 RAMSER 于 1959 年首次提出[1],用于优化汽油运输卡车的运输路线。基本的 VRP 问题又称为带容量约束的车辆路径问题(Capacitated VRP,CVRP)[2],即为一组起点、运载能力等均相同的车辆规划路线,使得该组车辆可以满足客户点需求且总的行驶路程最短的问题。CVRP 是现今货物配送领域的重要问题,探索其优质的解对促进电子商务发展、提高物流服务质量及提升人民生活水平具有重要指导意义。

CVRP 是运筹学中经典的组合优化问题,也是一个 NP 难问题[3]。目前 CVRP 问题的求解方法主要分为精确解法和启发式算法两类。其中,精确解法包括分支定界法、动态规划法等;启发式算法则包括模拟退火算法、蚁群算法、遗传算法、粒子群算法、蜂群算法等及其改进算法。随着 CVRP 的规模越来越大,精确解法求解效率将越来越低,难以满足实际应用需要;而启发式算法因具有收敛快、效率高等特点,较

基金项目:国家重点研发计划(2018YFB1800705)
This work was supported by the National Key R&D Program of China(2018YFB1800705).
通信作者:沈映珊(shenys@m.scnu.edu.cn)

适合用于求解 CVRP 问题。Ilhan 等人提出了一种带交叉算子的改进模拟退火算法，所得结果优于其他改进的 SA 算法，但计算消耗较大[4]；Ines 等[5]使用自适应遗传算法求解 CVRP 问题，与先进算法相比取得了更好的解，但收敛速度较慢，需要较多的进化代数；Song 等[6]提出的改进的带局部搜索的差分进化算法，加快了收敛速度并增强了鲁棒性，其中的变异方式和局部搜索策略十分重要，选择不恰当会对结果产生较大的影响。综合来看，利用启发式搜索算法求解 CVRP 问题在求解结果和收敛性能等方面还有探索空间。

粒子群算法（Particle Swarm Optimization，PSO）属于常用启发式算法的一种，它从随机解出发，通过迭代寻找最优解，具有实现容易、求解结果质量高、收敛快等优点，在解决优化问题中展示了其优越性。但 PSO 自身也存在易陷入局部收敛等缺陷。为了进一步提升 PSO 的性能，国内外学者在改进 PSO 方面进行了许多研究。Kohler 等人在标准 PSO 算法中添加了 6 个新模块，使用了修复算子，在空间探索、收敛性能等方面都表现良好[7]；Yu 等[8]提出了一种多策略混合的粒子群优化算法，以应对复杂优化问题的挑战；Jin 等[9]于 2015 年将社会学习机制引入 PSO 中，提出了社会学习粒子群算法（Social Learning PSO，SLPSO）；Li 等[10]提出了一种基于异维变异的差分混合粒子群算法，引入了熵度量方式筛选初始粒子，并通过异维变异学习策略引导粒子在陷入局部极值时跳出循环；Ge 等人于 2017 年对粒子群设计了层次结构，采用随机算子等更新粒子速度与位置，充分地利用了每个粒子的特性[11]；Yuan 等[12]于 2019 年提出一种对种群进行动态分层的方法，增加了粒子多样性和自主学习能力，使算法更容易跳出局部最优位置。采用分层结构的粒子群算法由于充分考虑到粒子所处层次的特性，对不同层次的粒子采取不同的对待方法，一定程度上解决了标准 PSO 算法容易陷入局部最优、收敛性能较差的问题。

本文结合各粒子群改进算法及 CVRP 问题的特性，提出一种基于分层学习和差分进化的混合 PSO 算法 DE-HSLPSO，用于解决 CVRP 问题。DE-HSLPSO 算法根据粒子的适应度值和迭代次数，使用动态分层方式将种群分为 3 层，前两层对粒子采用 PSO 社会学习方式，使种群向优良区域移动，第三层采用差分进化，以增加粒子的多样性，提高全局搜索能力。

2 带容量约束的车辆路径问题的数学模型

2.1 问题描述

本文在只有一个配送中心，且已知客户的数量、位置、货物需求量以及车辆最大载重量等信息的情况下，对若干辆车进行路线规划，求得每辆车完成配送任务并返回配送中心的行驶距离之和最短的配送方案。问题约束是每辆车装载货物量不得超过其最大容量，每个客户有且仅有一辆车为其送货。

假设配送中心的货物充足，存储量能满足所有客户的总需求量；客户所在位置固定且已知；车辆从 A 地行驶到 B 地的距离等于两地之间的欧氏距离；每辆车只完成一次配送；每辆车的最大载重量相同。

2.2 数学模型

设某配送中心车场有 m 辆车，每辆配送车的最大载重量为 C，需要对 n 个客户进行运输配送，每辆车从中心车场出发给若干个客户送货，最终回到中心车场。记中心车场编号为 0，客户点编号为 $1,2,\cdots,n$。客户点 i 的货物需求量为 c_i（$i=1,2,\cdots,n$ 且 $c_i\leqslant C$）。s_{ij}（$i,j=1,2,\cdots,n$）表示地点 i 到地点 j 的距离。x_{ijk}（$i,j=1,2,\cdots,n;k=1,2,\cdots,m$）表示车辆 k 是否从地点 i 到地点 j，当车辆 k 从地点 i 到地点 j 时 $x_{ijk}=1$，否则 $x_{ijk}=0$。y_{ki}（$i=1,2,\cdots,n$ 且 $k=1,2,\cdots,m$）表示车辆 k 是否访问地点 i，当车辆 k 访问地点 i 时 $y_{ki}=1$，否则 $y_{ki}=0$。满足车辆行驶总路程最短的配送方案可表述如下：

$$\min Z=\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=1}^{m}s_{ij}x_{ijk} \tag{1}$$

$$\text{s. t.}\begin{cases}\sum_{k=1}^{m}y_{ki}=\begin{cases}1, & \text{if } i\neq 0\\ m, & \text{otherwise}\end{cases} & (2)\\ \sum_{i=0}^{n}x_{ijk}=y_{kj} & (3)\\ \sum_{j=0}^{n}x_{ijk}=y_{ki} & (4)\\ \sum_{i=0}^{n}c_iy_{ki}\leqslant C & (5)\\ i\in\{0,1,2,\cdots,n\}\\ j\in\{0,1,2,\cdots n\}\\ k\in\{1,2,\cdots m\}\end{cases}$$

式(1)为目标函数，表示 m 辆车的行驶总路程最短；式(2)－式(4)表示每个客户点应恰好被访问一次；式(5)表示每辆车的载重量约束。

3 标准 PSO 算法

标准 PSO 算法由 Eberhart 和 Kennedy 提出[13]，是一个基于群体智能的启发式算法，实现简单，参数数量较少，同时能获得较好的解。

标准 PSO 算法设置了由 N 个粒子组成的粒子群 $P=\{p_1,p_2,\cdots,p_N\}$，每个粒子 p_i 都具有动态变化的速度 V_i 和位置 Pos_i，速度 V_i 即为粒子在解空间中移动的速度，位置 Pos_i 即为问题的候选解。在解空间为 D 维的问题中，粒子的速度和位置均可以为 D 维向量，即 $V_i=(v_i^1,v_i^2,\cdots,v_i^D)$，$Pos_i=(pos_i^1,pos_i^2,\cdots,pos_i^D)$。在每一次迭代中，PSO 算法还记录下每个粒子的历史最优解 $PBest_i$，以及粒子群体的历史最优解 $GBest$。迭代直至满足终止条件，PSO 算法输出群体最优解 $GBest$。

在标准 PSO 算法中，首先，随机初始化每一个粒子的位置和速度，并将位置代入到适应度函数中，计算粒子的适应度值，以此值来选出个体最优值 $PBest_i$（仅有一个值时，该值即为最优值）和群体最优解 $GBest$。之后的每一次迭代，粒子不断向自身的最优解 $PBest_i$ 和群体的最优解 $GBest$“学习”，更新自己的飞行速度和位置，再重新选出自身的最优解 $PBest_i$ 和群体最优解 $GBest$。粒子速度和位置的更新公式如下：

$$V_i=wV_i+s_1r_1(PBest_i-Pos_i)+s_2r_2(GBest-Pos_i) \tag{6}$$

$$Pos_i=Pos_i+V_i \tag{7}$$

其中，式(6)是粒子的速度更新公式，由惯性、自身认知、社会学习三大部分组成。w 是惯性权重，表示粒子对原来运动趋势的保持情况；s_1 和 s_2 是学习因子，表示粒子向自身最优解和群体最优解学习的程度；r_1 和 r_2 是随机数，取值于[0.0，1.0]

的范围中且服从均匀分布。式(7)是粒子的位置更新公式，表示粒子在当前位置以速度V_i飞行。在有边界的解空间中，粒子的飞行速度需限制在一定的范围$[-V_{\max}, V_{\max}]$内，避免粒子飞行过快跳出边界，$V_{\max}$表示粒子的最大飞行速度。

4　基于分层学习和差分进化的混合 PSO 算法求解 CVRP 问题

在标准 PSO 算法中，由粒子本身的历史最优解和群体全局最优解影响粒子速度和位置的更新。在种群进化过程中，粒子的多样性会降低，从而导致结果容易陷入局部最优的困局。为了更好地解决 CVRP 问题，本文提出了一种基于分层学习和差分进化的混合 PSO 算法(DE-HSLPSO)，引入了分层学习策略，差别对待不同状态的粒子，并为了减小陷入局部最优的风险，引入了差分进化算法，借助差分进化中的交叉变异等操作增加粒子的多样性。同时由于 CVRP 问题为离散域问题，每个客户点应恰好被访问一次，需在标准 PSO 算法以及差分进化算法的具体操作中进行一些改变，以避免粒子中出现重复的元素。

4.1　CVRP 问题初始解的构造

对于 CVRP 的求解，本文采用客户编号进行粒子的编码。首先随机生成一个长度为 n(n 为客户点数量)的 TSP 序列，且序列中不会有重复的数字出现，然后从头开始遍历序列，根据车辆载重约束对序列进行截取，每截取得到的一个子序列对应一辆车的路径方案，如随机序列‘1－2－3－5－4－6－8－7’被分为‘1－2’‘3－5－4－6’‘7－8’，表示共需要 3 辆车参与配送，第 1 辆车配送路线为‘0－1－2－0’，第 2 辆车配送路线为‘0－3－5－4－6－0’，第 3 辆车配送路线为‘0－7－8－0’，其中 0 为配送中心的编号。

分别计算每个子序列对应的路径长度，子序列数量等于车辆需求数，各子序列路径长度之和便为一个粒子的适应度值，路径长度总和越小，表示粒子适应度越好。

4.2　分层策略

种群中不同粒子在精确搜索能力和空间探索能力上的表现不同。适应度较好的粒子有更大概率靠近最优位置，精确搜索能力表现较好，并可以引导种群向优良方向进化；而适应度较差的粒子有更大概率偏离最优位置，空间探索能力表现较好，有助于开拓空间，跳出局部最优。

因此，在本文的算法中，以粒子的适应度值为依据，把粒子进行升序排序并分为 3 层。每一层用N_i($1\leqslant i\leqslant 3$)表示，即N_1，N_2，N_3分别代表第一、二、三层，每次迭代更新粒子后都要根据粒子的适应度重新划分粒子层，具体的划分方式参考文献[12]，如图 1 所示。

$$low=\frac{1}{8}size+size\left(0.5-\frac{t}{T}\right)^3 \tag{8}$$

$$high=\frac{7}{8}size+size\left(0.5-\frac{t}{T}\right)^3 \tag{9}$$

其中，low 为第一、二层粒子的分界，$high$ 为第二、三层粒子的分界，$size$ 表示粒子总数，t 和 T 分别表示当前迭代次数和最大迭代次数。记排序完成的 N 个粒子为 $X=\{X_1, X_2, \cdots, X_N\}$，则第一层粒子 $N_1=\{X_1, X_2, \cdots, X_{low}\}$，第二层粒子 $N_2=\{X_{low+1}, X_{low+2}, \cdots, X_{high}\}$，第三层粒子 $N_3=\{X_{high+1}, X_{high+2}, \cdots, X_N\}$。

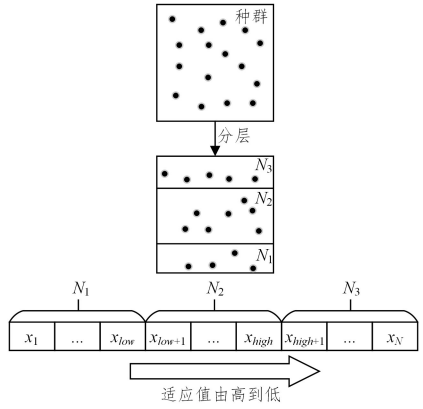

图 1　粒子分层示意图

Fig. 1　Hierarchical particles

4.3　各层次粒子学习策略

由于每层粒子所展现的特性不同，各层粒子的学习策略也互不相同。第一层粒子利用 PSO 算法向当前层进行学习，第二层粒子利用 PSO 算法向当前层以及第一层学习，前两层粒子的学习使得种群向优良区域移动，保证粒子的优越性；第三层粒子则进行差分进化，通过变异交叉等进一步提高种群的多样性，扩大种群的搜索空间，提高种群的全局搜索能力。

4.3.1　社会学习思想

第一、二层粒子的学习方式引入了社会学习[9]的思想，更改了向粒子本身的历史最优解和群体全局最优解学习的基本学习模式，在目标学习层中随机选取比当前粒子(以适应度为标准)更好的粒子进行学习。即第一层的粒子随机在当前层选择更好的粒子进行学习，第二层的粒子在第一、二层各随机选择一个粒子进行学习，以更新速度和位置。

因为 CVRP 问题是离散型问题，在位置更新和速度更新中引入了交换子和交换序[1-15]的概念。交换序由交换子构成，表示粒子的速度，具体的更新公式如下：

$$v_{N_1,j}=(r_1\times v_{N_1,j})\oplus[r_2\times(x_{N_1,rand}-x_{N_1,j})] \tag{10}$$

$$x_{N_1,j}=x_{N_1,j}+v_{N_1,j} \tag{11}$$

$$v_{N_2,j}=(r_3\times v_{N_2,j})\oplus[c\times r_4\times(x_{N_1,rand}-x_{N_2,j})]\oplus[r_5\times(x_{N_2,rand}-x_{N_2,j})] \tag{12}$$

$$x_{N_2,j}=x_{N_2,j}+v_{N_2,j} \tag{13}$$

其中，$x_{N_i,j}$表示第N_i层第 j 个粒子的位置，$x_{N_i,rand}$表示第N_i层随机的一个粒子的位置，$v_{N_i,j}$是第N_i层第 j 个粒子的速度，$r_1, r_2, \cdots, r_5$均为 0 到 1 之间的随机数，c 为控制参数。各运算符号的含义如表 1 所列。

表 1　运算符号释义表

Table 1　Definition of operation symbols

符号	释义
⊕	合并两个交换序
＋	执行交换操作
－	求解基本交换序
×	保留交换序中前整数个交换子

以下介绍 CVRP 中粒子群算法的一些基本操作。

小数×速度：速度以交换序的形式表示，该运算结果为速度中的交换子依次保留整数个交换子，由保留下来的交换子重新组成的一个新的速度。

速度⊕速度：结果为两个速度中的交换子的并集，并集构成了一个新的速度。

位置－位置：位置本身表示了 CVRP 的一个解，位置之差表示一个交换序。

位置＋速度：速度中的交换子依次作用在粒子位置上，得到一个新的位置。

求解 CVRP 问题过程中表示粒子速度大小的交换序中交换子的次序是有意义的，交换子必须依次作用于粒子。例如一个由 5 个客户点构成的粒子序列为 $X=(3,4,2,5,1)$，产生的速度 $V=((1,5),(4,5),(2,3))$，则新的解 $X'=X+V=[(X+(1,5))+(4,5)]+(2,3)=(1,2,4,3,5)$。

4.3.2 差分进化

差分进化算法（Differential Evolution Algorithm，DE）由 Storn 和 Price 于 1997 年提出，其思想源于遗传算法，是一类基于群体差异的全局优化方法。与许多其他效果较好的全局优化方法相比，DE 收敛速度更快，控制变量数量较少，使用方便[16]。

根据向量之间的差异，DE 在解空间中以向量的形式进行搜索，随机寻找问题最优解。DE 是在遗传算法（Genetic Algorithm，GA）的基础上改进的，进化流程与 GA 类似，核心部分都是变异、交叉和选择算子，但在这些算子的具体定义上两者并不相同。本文 DE 算法的变异、交叉和选择算子具体定义如下。

(1)变异

DE 算法根据差分策略来实现个体变异。一个常见的差分策略是从父代个体中选择两个不同的个体 $x_{r_1}(g)$ 和 $x_{r_2}(g)$ 进行向量差操作，生成差分矢量，再选择另一个体 $x_{r_3}(g)$ 与差分矢量进行向量求和，生成变异个体 $m_i(g+1)$。

$$m_i(g+1)=x_{r_1}(g)+F(x_{r_2}(g)-x_{r_3}(g)) \tag{14}$$

其中，$m_i(g+1)$ 表示第 $g+1$ 代变异种群的第 i 个个体；$x_i(g)$ 表示第 g 代种群的第 i 个个体；r_1,r_2,r_3 是 3 个互不相同的随机数，且 $r_1,r_2,r_3\in\{1,2,\cdots,N\}$，$N$ 表示种群大小；F 是一个实数，且 $F\in[0,2]$，控制差分矢量的缩放，称为缩放因子。

变异操作的编码方式参考了文献[17]的编码设计，用于解决离散域中的优化问题。在求解 CVRP 问题时，原来粒子经过 DE 变异后，整数运算不再满足封闭性，但可对粒子变异后的实数按从小到大排序，接着找出原来的数在新的序列中的位置。例如 CVRP 问题中产生的一个变异个体为 $X=(-2.1,4.3,3.5,1.6,2.5)$，则升序排序后的变异个体为 $X'=(-2.1,1.6,2.5,3.5,4.3)$，则对应的编码为 $(1,5,4,2,3)$。

(2)交叉

由于在传统 DE 算法的交叉操作中，编码有可能产生相同的元素，不适用于 CVRP 问题的求解。针对上述问题，本文的交叉操作使用局部交叉映射（PMX）方法[18]，对第 g 代种群及其变异的中间体 $m_i(g+1)$ 进行个体间的交叉操作，产生试验向量 $u_i(g+1)$。首先在两个粒子中选择交叉的位置，然后交换这两个粒子中被选位置的元素，接着利用两组交换元素构建映射关系，根据映射关系在产生的子代中消除冲突，即在子代中不存在相同的元素。交叉示意图如图 2 所示。

(3)选择

DE 算法根据适应度值的大小，采用贪婪准则来选择进入下一代种群的个体。

$$x_i(g+1)=\begin{cases}u_i(g+1), & \text{if } f(u_i(g+1))\geqslant f(x_i(g))\\ x_i(g), & \text{otherwise}\end{cases} \tag{15}$$

其中，f 为适应度函数。

DE 算法在随机初始化种群后，进行变异、交叉操作，随后根据当前个体和新个体的适应度值大小选择一个优秀个体进入下一代种群，淘汰劣质个体。随后的每次迭代，均进行变异、交叉和选择操作，更新种群，直至满足终止条件。从最终种群中选择适应度最优的个体，即为 DE 算法获得的最优解。

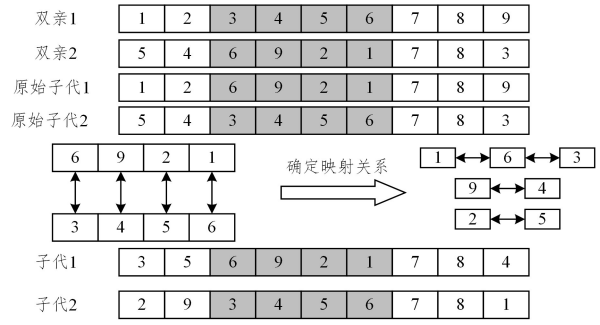

图 2 PMX 交叉示意图

Fig. 2 PMX cross diagram

4.4 算法流程

在 DE-HSLPSO 算法中，对粒子位置与速度等进行初始化后，计算粒子的适应度，根据粒子适应度值使用动态分层方法将种群分为 3 层 N_1,N_2,N_3。前两层粒子以 PSO 社会学习方式更新速度和位置；第三层粒子进行差分进化，通过差分变异、交叉、选择，使位置发生变化而速度保持不变。每经过一次迭代更新，都要重新计算粒子适应度，进行层次划分以及记录全局最优解，直到满足最大迭代次数要求。

DE-HSLPSO 算法描述如算法 1 所示。

算法 1 DE-HSLPSO 算法

输入：客户的数量、位置、货物需求量以及车辆最大载重量

输出：满足车辆行驶总路程最短的配送方案随机初始化 n 个粒子，初始化粒子速度及位置

```
for i=1 to n do
    计算每个粒子的适应度值大小
end for
初始化全局最优解
while 当前迭代次数不大于最大迭代次数 do
    根据适应度值大小将粒子进行升序排列
    根据式(8)和式(9)确定分界 low、high，将粒子分为 3 层
for i=1 to n do
if 粒子处于第一层 then
        粒子向当前层学习，根据式(10)和式(11)更新速度和位置
else if 粒子处于第二层 then
        粒子向当前层和第一层学习，根据式(12)和式(13)更新速度和位置
else 粒子处于第三层
        粒子进行差分进化，对位置进行更新
        end if
        计算粒子适应度值大小
    end for
    记录当前最优解
if 当前最优解<历史最优解 then
    历史最优解=当前最优解
    end if
end while
```

DE-HSLPSO 算法流程图如图 3 所示。

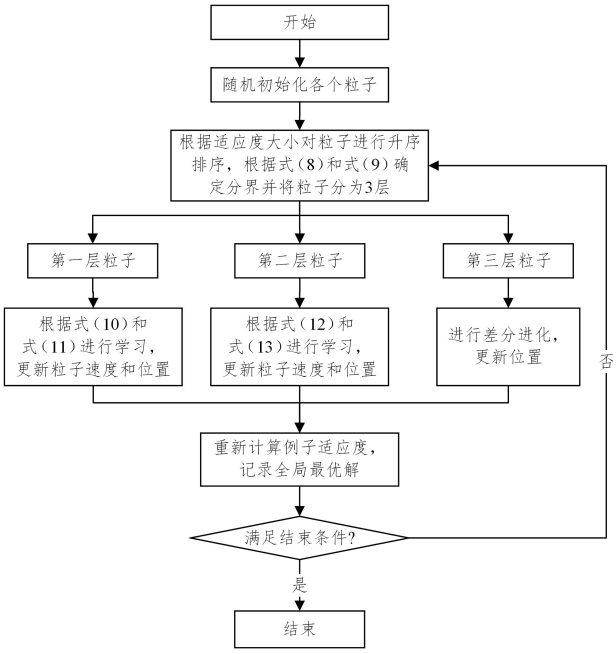

图 3　DE-HSLPSO 算法示意图

Fig. 3　Flow chart of DE-HSLPSO algorithm

5　实验与分析

5.1　实验设置

本文实验算例来自于经典的 CVRP 算例的 SetP。实验软件为 PyCharm，实验环境为 Win10 操作系统，2.60 GHz 主频 Intel 处理器和 8 GB 内存。实验重要参数分别设置为：粒子总数 $size=400$，控制参数 $c=1.5$，缩放因子 $F=0.5$。

5.2　算法测试与分析

5.2.1　改进算法测试与结果

本文首先使用经典算例中的 P-n19-k2 算例[1)] 进行测试，该算例中有 18 个客户点和 1 个配送中心，已知配送总距离最优解为 212。通过 DE-HSLPSO 算法的计算，可以求出车辆配送路径的总最短距离为 212，共需要 2 辆车，并获得路径规划方案，具体如表 2 与图 4 所示。路径中的数字‘0’表示配送中心，其余数字代表问题描述中的客户点序号，如‘1’代表客户点 1。

使用 DE-HSLPSO 求解 P-n19-k2 算例的结果与该算例的已知最优值 212 相同，表明了该算法具有优秀的求解能力，求解的收敛情况如图 5 所示。算法在进化前期，尤其是在 200 代以内易于寻找更优解，可快速逼近算例已知最优解；在进化后期仍然可以跳出局部最优求得更优解，直到获得最优解，表现了改进算法良好的寻优能力和收敛性能。

表 2　车辆路径规划方案

Table 2　Planning scheme of vehicles

Vehicle number	The path of the vehicle	Distance
1	0－18－5－13－15－9－7－2－10－1－0	97
2	0－6－8－16－17－3－12－14－11－4－0	115
Total		212

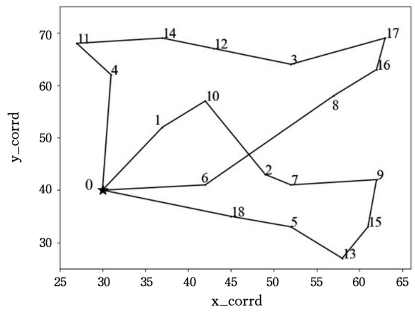

图 4　P-n19-k2 算例路径规划

Fig. 4　Route plan of P-n19-k2

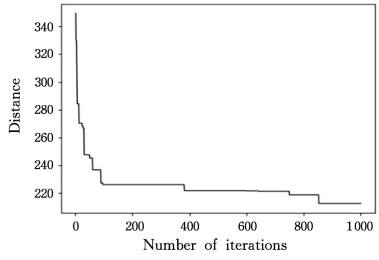

图 5　P-n19-k2 算例收敛曲线图

Fig. 5　Convergence curve of P-n19-k2

5.2.2　对改进算法各部分的探究

为了探究改进算法各部分的影响，即分层策略与差分进化对算法整体性能的提升是否有意义，本文在 P-n19-k2 算例上对改进算法整体与算法各部分进行了比较实验。对比方法如表 3 所列，其中方法 *M*4 即为本文的改进算法 DE-HSLPSO，在每个方法上均测试 10 次，测试结果如表 4 所列，分别从结果的最优值 *Best*、最差值 *Worst*、平均值 *Average*、最优值偏差 *Best-Dev* 进行了比较，其中最优值偏差＝(最小值－已知最优解)/已知最优解。

表 3　对比方法构成

Table 3　Composition of comparison methods

Method	PSO	Hierarchical Learning	DE
*M*1	√		
*M*2			√
*M*3	√	√	
*M*4	√	√	√

表 4　各方法测试结果

Table 4　Test results of each method

Method	*Best*	*Worst*	*Average*	*Best-Dev*/%
*M*1	250	295	269	17.92
*M*2	275	284	279	29.72
*M*3	215	221	219	1.41
*M*4	212	221	217	0

从实验结果中我们可以获知，单独的粒子群算法 *M*1 最优值偏差较大，求解问题能力较差，最优值与最差值相差 45，结果浮动性大，算法稳定性较差。单独的差分进化算法 *M*2 求解 CVRP 得到的最优值偏差为 29.72%，偏差最大，平均值与已知最优值差值也较大。*M*3 在粒子群的基础上引入了分层策略后，性能获得较大的提升，最优值偏差从 17.29%下降至 1.41%，平均值从 269 下降至 219。*M*4 进一步结合差分进化方法后，粒子多样性增加，全局搜索能力增强，算法获得了

1) http://vrp.atd-lab.inf.puc－rio.br/index.php/en/

更好的效果，求解到的最优值偏差可达到0，在最优值、最差值、平均值、最优值偏差方面表现均为最佳。以上说明，对于改进算法DE-HSLPSO而言，分层策略和差分进化方法的引入均是有意义的。

5.3 算法对比与分析

为了更好地评价DE-HSLPSO算法在解决CVRP问题上的性能，本文将DE-HSLPSO算法分别与用于求解CVRP问题的其他算法（Adaptive Sweep＋VTPSO[19]，HSSA[20]，LNS-ACO[21]）进行对比，算例为经典CVRP数据集中SetP的部分算例。考虑到启发式算法具有一定的波动性，为了减小结果误差，每个算例计算20次，记录结果最优值*Best*、最优值偏差*Worst*、平均值偏差*Avg-Dev*。其中，平均值偏差＝(平均值－已知最优解)/已知最优解，已知最优解为*BSK*。其对比结果如表5所列。

表5 算法对比结果

Table 5 Comparison results of different algorithms

Example	BSK	Adaptive Sweep＋VTPSO			HSSA			LNS-ACO			DE-HSLPSO		
		Best	*Best-Dev*/%	*Avg-Dev*/%	*Best*	*Best-Dev*/%	*Avg-Dev*/%	*Best*	*Best-Dev*/%	*Avg-Dev*/%	*Best*	*Best-Dev*/%	*Avg-Dev*/%
P-n16-k8	450	549	22.00	—	451	0.22	0.22	450	0	0	450	0	0
P-n19-k2	212	246	16.04	-	212	0	0	212	0	0	212	0	0
P-n20-k2	216	249	15.28	—	217	0.46	0.46	216	0	0	216	0	0
P-n21-k2	211	211	0	—	212	0.47	0.47	211	0	0	211	0	0
P-n22-k2	216	216	0	—	—	—	—	216	0	0	216	0	0
P-n22-k8	603	633	4.98	—	—	—	—	590	−2.16	−2.16	589	−2.32	−2.16
P-n23-k8	529	634	19.85	—	536	1.32	1.32	529	0	0	530	0.19	0.19

在对比实验中可以观察到，Adaptive Sweep＋VTPSO的寻优能力最差，仅在P-n21-k2和P-n22-k2上结果最优值与已知最优值的偏差为0，且在4个算例中出现最优值偏差大于10%；HSSA寻优能力比Adaptive Sweep＋VTPSO更理想，总体上结果最优值偏差较小，均低于1.5%；LNS-ACO的寻优性能则比HSSA更胜一筹，在P-n22-k8算例中找到了比已知最优解更优秀的结果，在其他算例上结果最优值偏差、平均值偏差均为0；本文算法DE-HSLPSO只有在P-n23-k8算例的结果比已知最优值差，但偏差微小，最优值偏差、平均值偏差均为0.19%。因此，在对比算法中，HSSA，LNS-ACO和本文算法DE-HSLPSO相对获得更优秀的结果。但为了对比的公平性，需从时间方面进行进一步探究，比较指标为单次运行平均寻优时间。HSSA，LNS-ACO和DE-HSLPSO在各个算例上的寻优时间如表6所列。

表6 各算法运行时间

Table 6 Running time of different algorithms

(单位：s)

Example	HSSA	LNS-ACO	DE-HSLPSO
P-n16-k8	22	737.30	312.01
P-n19-k2	14	363.90	257.71
P-n20-k2	14	353.04	217.41
P-n21-k2	15	399.88	217.83
P-n22-k2	—	413.12	222.33
P-n22-k8	—	562.84	428.51
P-n23-k8	41	615.53	531.17

在耗费时间方面，在各个算例上HSSA耗费时间最短，且比其他两个算法节省较多时间；而LNS-ACO耗费时间最长；DE-HSLPSO耗费时间次之。但结合时间与结果两方面，DE-HSLPSO的效果应为最佳，多次最优结果与已知最优值一致，且运行时间不会耗费过多。

综合上述实验，可以获知本文的改进算法是有意义的，分层策略使得粒子群算法寻优能力得到较大的提升，差分进化的引入使原本适应度较差的粒子发生变异交叉，增加了粒子的多样性，从而增加了获取到空间中解的可能性，因而获得了更强的全局搜索能力，同时结合分层策略和差分进化可使算法获得更强的寻优能力。在对比实验中，从结果来看，DE-HSLPSO比Adaptive Sweep＋VTPSO和HSSA更优秀；从时间方面来看，DE-HSLPSO比LNS-ACO节省时间。结合时间和结果两方面，DE-HSLPSO在4个算法中的效果最好。

结束语 本文提出的基于分层学习和差分进化的混合PSO算法(DE-HSLPSO)，将粒子以动态划分方式分成3层，其中前两层粒子进行社会学习，使得种群向全局最优区域移动，保证了粒子的优越性；第三层粒子则进行差分进化，提高了种群的多样性和种群的全局搜索能力。使用经典算例对DE-HSLPSO进行测试，表明了改进算法是可行且有效的；分层策略与差分进化均提高了算法的整体性能，其中分层策略的影响相对更大。综合时间与结果来看，DE-HSLPSO比对比的优化方法更为优越。将DE-HSLPSO应用于车辆路径问题的求解，为物流行业的车辆路径规划提供了参考。在未来的研究中，将尝试使用本文的方法求解变形的车辆路径问题，例如带时间窗的车辆路径问题，并尝试优化差分进化部分的变异、交叉操作，以获得更优解。

参 考 文 献

[1] DANTZIG G B, RAMSER J H. The truck dispatching problem[J]. Management Science, 1959, 6(1): 80-91.

[2] YANG G, CAI Y. A review of hybrid particle swarm optimization algorithms for vehicle routing problems and its variants[J]. Automation and Information Engineering, 2021, 42(2): 7-13.

[3] LENSTRA J K, KAN A. Complexity of vehicle routing and scheduling problems[J]. Networks, 2010, 11(2): 221-227.

[4] LHAN L. An Improved Simulated Annealing Algorithm with Crossover Operator for Capacitated Vehicle Routing Problem[J]. Swarm and Evolutionary Computation, 2021: doi: 10.1016/J.SWEVO.2021.100911.

[5] KRICHEN S, SBAI I, LIMAM O. An effective genetic algorithm for solving the capacitated vehicle routing problem with two-dimensional loading constraint[J]. International Journal of Computational Intelligence Studies, 2020, 9(1/2): 85-106.

[6] SONG L, YUN D. An improved differential evolution algorithm with local search for capacitated vehicle routing problem[C]//

2018 Tenth International Conference on Advanced Computational Intelligence(ICACI). 2018.

[7] KOHLER M, VELLASCO M, TANSCHEIT R. PSO+: A new particle swarm optimization algorithm for constrained problems [J]. Applied Soft Computing, 2019, 85(5): 437-442.

[8] YU W W, XIE C W. A multi-strategy hybrid particle swarm optimization algorithm[J]. Computer Science, 2018, 45(S1): 133-136.

[9] CHENG R, JIN Y. A social learning particle swarm optimization algorithm for scalable optimization[J]. Information Sciences An International Journal, 2015, 291(5): 43-60.

[10] LI J, LUO Y K, LI B, et al. Differential hybrid particle swarm optimization algorithm based on different-dimensional mutation [J]. Computer Science, 2018, 45(5): 208-214.

[11] GE H, SUN L, TAN G, et al. Cooperative Hierarchical PSO With Two Stage Variable Interaction Reconstruction for Large Scale Optimization [J]. IEEE Transactions on Cybernetics, 2017, 47(9): 2809-2823.

[12] YUAN X, JIANG S. Improved particle swarm optimization algorithm based on hierarchical autonomous learning[J]. Computer Applications, 2019, 39(1): 148-153.

[13] KENNEDY J, EBERHART R. Particle Swarm Optimization [C] // Icnn95-international Conference on Neural Networks. IEEE, 2002.

[14] WU G. Research on path planning based on particle swarm algorithm[D]. Qinhuangdao: Yanshan University, 2016.

[15] WU B. Research and application of particle swarm optimization algorithm for vehicle routing problem[D]. Hangzhou: Zhejiang University of Technology, 2008.

[16] STORN R, PRICE K. Differential Evolution-A Simple and Efficient Heuristic for global Optimization over Continuous Spaces [J]. Journal of Global Optimization, 1997, 11(4): 341-359.

[17] ANONYMOUS. Discrete differential evolution algorithm for solving problems [J]. Computer and Digital Engineering, 2017, 45(337): 58-64.

[18] SHEN J. Application Research of Hybrid Discrete Differential Evolution Algorithm in Green Logistics Vehicle Path Planning [D]. Taiyuan: Taiyuan University of Science and Technology, 2020.

[19] AKHAND M, PEYA Z J, MURASEK. Capacitated vehicle routing problem solving using adaptive sweep and velocity tentative PSO[J]. International Journal of Advanced Computer Science and Applications, 2017, 8(12): 288-295.

[20] ZHANG X N, FAN H M. Hybrid decentralized search algorithm for vehicle routing problem with capacity constraints [J]. Control and Decision Making, 2015, 30(11): 1937-1944.

[21] AKPINAR S. Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem[J]. Expert Systems with Applications, 2016, 61: 28-38.

CHEN Ying, born in 2001, undergraduate. Her main research interests include deep learning and computational intelligence.

SHEN Ying-shan, born in 1975, Ph. D, associated professor. Her main research interest include model recognition and educational technology.