

基于K-Means聚类的混合粒子群算法求解旅行商问题

何宏瑞, 金 中

上海海事大学理学院, 上海

收稿日期: 2025年3月13日; 录用日期: 2025年4月18日; 发布日期: 2025年4月25日

摘 要

本文针对基本粒子群算法在求解TSP问题时容易陷入局部最优的问题, 提出了一种改进的混合粒子群算法。在利用粒子群算法的快速搜索能力引导全局优化的基础上, 引入遗传算法的交叉、变异等操作增强种群多样性与局部寻优能力。同时使用K-Means算法对混合粒子群算法进行改进, 提高算法迭代速度。通过TSPLIB中实例的求解与分析, 实验结果表明改进后的混合粒子群算法能更加有效解决TSP问题。

关键词

粒子群算法, 遗传算法, K-Means算法, 旅行商问题

Solving the Traveling Salesman Problem Using a Hybrid Particle Swarm Optimization Algorithm Based on K-Means Clustering

Hongrui He, Zhong Jin

School of Science, Shanghai Maritime University, Shanghai

Received: Mar. 13th, 2025; accepted: Apr. 18th, 2025; published: Apr. 25th, 2025

Abstract

This paper addresses the issue of the basic Particle Swarm Optimization (PSO) algorithm easily falling into local optima when solving the Traveling Salesman Problem (TSP). It proposes an improved hybrid PSO algorithm. Building on the rapid search capability of PSO to guide global optimization, the algorithm introduces crossover and mutation operations from the Genetic Algorithm (GA) to enhance population diversity and local search ability. Additionally, the K-Means algorithm is utilized

to refine the hybrid PSO algorithm, thereby increasing the iteration speed of the algorithm. Through the solution and analysis of instances from TSPLIB, experimental results demonstrate that the improved hybrid PSO algorithm can more effectively solve the TSP problem.

Keywords

Particle Swarm Optimization, Genetic Algorithm, K-Means Algorithm, Traveling Salesman Problem

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

旅行商问题(Travelling Salesman Problem, TSP)是计算机科学和运筹学领域中经典的组合优化问题之一,它在路线规划、电路板布线、生产调度、物流配送等许多实际问题中都有广泛的应用。TSP 问题可被描述为:求解一条一次性且不重复地通过 n 个城市且最终回到起点的最短路径。它是一个 NP-hard 问题,其复杂程度随着问题规模的增大呈指数增长[1]。目前求解 TSP 问题的方法主要分为两类,第一类是传统的近似算法,如线性规划法,动态规划法等,随着问题复杂度的增长,该类算法的效果显著下降。而随着计算机的发展,智能优化算法逐渐成为了解决 TSP 问题的主流方法,并取得了不错的效果。常见的优化算法主要有:遗传算法(GA)、粒子群算法(PSO)、蚁群算法(ACO)等等。其中,标准粒子群算法(PSO)基于群体和个体的适应度大小进行进化更新,通过随机初始化一组解,迭代寻找最优解,用于求解 TSP 问题时具有参数简单、容易实现以及快速收敛等特点[2]。但是在寻优过程中,最优解附近粒子容易产生“振荡”现象,陷入局部最优[3]。而且 PSO 最初是用于连续域上求解, TSP 问题却是个离散型组合优化问题[1]。基于这些缺陷,近年来不断有学者提出了改进的 PSO 算法。文献[1]提出结合贪心策略、动态学习概率和 2-opt 局部优化的改进粒子群算法。文献[2]提出了一种改进自适应杂交退火粒子群算法。文献[3]提出将遗传算法的交叉与变异操作加入粒子群算法,使粒子群具有变异能力[1]-[3]。同时引入模拟退火算法的 Metropolis 准则,使粒子能跳出局部最优。针对上述解决 TSP 问题的改进粒子群算法,本文提出了结合遗传算法、粒子群算法、K-Means 算法的改进混合粒子群算法。最后,利用 Matlab 将本文所提出的改进算法和其他智能算法进行仿真实验并进行比较。证明了本文改进算法在求解精度、求解速度上有更好的表现。同时,对于大规模的 TSP 问题也有较好的求解能力。

2. 旅行商问题

旅行商问题(Travel Salesman Problem, TSP)是组合优化领域中最具代表性的难题之一,最早可追溯至 19 世纪的数学文献。其核心任务是:给定一组城市及它们之间的距离(或成本),旅行商需要找到一条访问每个城市恰好一次并最终返回出发点的最短闭合路径。TSP 因其看似简单的描述与极高的计算复杂度之间的反差而闻名,被称为 NP-hard 问题的典型代表。尽管问题直观,但城市数量 n 增大时,可能的路径组合数高达 $\frac{(n-1)!}{2}$ (对称 TSP),导致穷举法在现实中不可行。这一特性促使研究者开发启发式算法(如遗传算法、蚁群优化)、动态规划及近似解法,同时推动了计算复杂性理论的发展。

其可以描述为:设 $G=(V,E)$ 为一个完全加权无向图(对称 TSP)或有向图(非对称 TSP),其中:
 $V=\{v_1,v_2,\dots,v_n\}$ 表示城市集合;
 $E=\{(v_i,v_j)|i\neq j\}$ 表示边集合;每条边 (v_i,v_j) 的权重 d_{ij} 表示城市 v_i 到 v_j

的距离(或成本)。找到一个哈密顿回路(Hamiltonian cycle) $\pi = (v_{\sigma(1)}, v_{\sigma(2)}, \dots, v_{\sigma(n)}, v_{\sigma(1)})$, 其中 σ 是 $\{1, 2, \dots, n\}$ 的排列, 使得总路径长度最小化:

$$D(\pi) = \sum_{k=1}^{n-1} d_{\sigma(k), \sigma(k+1)} + d_{\sigma(n), \sigma(1)} \quad (1)$$

3. 算法实现

3.1. 基本粒子群算法

3.1.1. 算法描述

粒子群算法(Particle Swarm Optimization, PSO)是一种基于群体智能理论的仿生优化算法, 其灵感来源于鸟群或鱼群的觅食行为。算法通过模拟群体中个体的简单协作机制, 将每个粒子视为解空间中的一个候选解, 并通过速度与位置的动态更新机制迭代搜索最优解。在 PSO 中, 粒子的速度由三部分决定: 惯性项(保留当前运动趋势)、个体认知项(个体历史最优位置 P_{best} 的引导)和社会认知项(群体历史最优位置 G_{best} 的引导)。位置更新则基于速度向量调整, 逐步逼近全局最优解。基本 PSO 的速度与位置的更新公式如下:

$$v_i^{t+1} = \omega \cdot v_i^t + c_1 \cdot r_1 \cdot (pbest_i - x_i^t) + c_2 \cdot r_2 \cdot (gbest - x_i^t) \quad (2)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

其中, v_i^t 、 x_i^t 分别表示经过第 t 次迭代后, 第 i 个粒子的速度和位置; $pbest_i$ 表示第 i 个粒子的个体历史最优位置; $gbest$ 表示种群的群体历史最优位置; ω 表示惯性权重, 用于控制历史速度的影响, 平衡全局与局部搜索; c_1, c_2 表示学习因子, 分别控制粒子对自身最佳位置和全体粒子最佳位置的吸引程度; r_1, r_2 表示从 $[0, 1]$ 中随机选取的随机数[1]。

3.1.2. 粒子群算法的步骤

其步骤如下:

- (1) 对群体进行初始化, 设粒子总数为 N , 对所有参数赋初值。
- (2) 随机初始化每个粒子的速度与位置。
- (3) 计算群体中每个粒子的适应度值。先比较当前位置与局部最优, 选择最优位置作为新的局部最优; 再比较局部最优和全局最优, 选择最优位置作为全局最优。
- (4) 使用粒子速度和位置更新公式, 根据当前的速度、历史最佳位置和全局最佳位置来更新粒子的速度和位置。
- (5) 判断是否满足终止条件, 如果满足则停止, 反之继续从第(2)步开始[4]。

3.2. 基于遗传算法的混合粒子群算法

3.2.1. 遗传算法

遗传算法(Genetic Algorithm, GA)是一种基于自然选择和遗传机制的全局优化算法, 灵感来源于生物进化过程。是一种基于自然选择和遗传机制的随机搜索方法。该算法模拟生物进化中的繁殖、交叉和基因突变, 通过对候选解的群体进行持续筛选和进化, 以寻找问题的最佳解决方案。

在遗传算法中, 解决方案被编码为类似生物染色体的个体。每个个体都有一个相应的适应度值, 用于评价其在解决特定问题时的优劣程度。算法从随机生成的初始种群开始, 在每一代中通过适应度值进行选择操作, 选取表现较好的个体, 使其有更大的机会参与繁殖。交叉操作模拟了生物的交配过程, 通

过交换和组合两个或多个个体的基因, 产生新的个体, 从而可能将优良基因传递给后代。而变异操作则以一定概率随机改变个体中的某些基因, 为种群引入新的遗传变异, 从而增加种群的多样性, 以防止算法过早陷入局部最优解。通过不断实施选择、交叉和变异等操作, 种群中的个体逐渐适应环境, 适应度值随之提高。经过多代进化, 最终能够获得符合终止条件的最优解或近似最优解。遗传算法在函数优化、组合优化以及机器学习等多个领域展现出强大的搜索能力和良好的应用效果。

3.2.2. 基于遗传算法的特性对粒子群算法进行优化

基于遗传算法的混合粒子群算法(Hybrid PSO-GA, HPSO)其核心思想是利用 PSO 的快速搜索能力引导全局优化, 同时引入 GA 的交叉、变异等操作增强种群多样性与局部寻优能力。将 $\omega \cdot v_i^t$ 部分视作遗传算法中的变异, $c_1 \cdot r_1 \cdot (pbest_i - x_i^t) + c_2 \cdot r_2 \cdot (gbest - x_i^t)$ 部分视作交叉, 将当前解与两部分交叉, 作为下一次迭代的解[4]。通过交叉操作和学习因子来产生和选取每代粒子个体的优秀子代; 再通过变异操作来实现粒子的位置更新操作。最终通过多次迭代找到最优解[1]。近年来, 许多学者研究了 HPSO 算法并验证了其有效性和优异的表现。

3.3. 基于 K-Means 算法的改进粒子群算法

K-Means 算法是一种常用的聚类方法, 其基本思想是通过以 k 为参数, 将 n 个对象划分成 M 个簇。这些同簇内对象具有较高的相似度, 而不同簇间的相似度较低[5]。每个簇由一组相似的数据点构成, 簇的中心(质心)是簇内所有数据点的平均值。K-Means 算法的步骤如下:

- (1) 随机选择 k 个数据点作为初始质心。
- (2) 对于每个数据点, 计算其与所有质心的距离(通常使用欧氏距离)。
- (3) 将数据点分配到距离最近的质心所在的簇。
- (4) 对于每个簇, 重新计算质心, 即簇内所有点的平均值。
- (5) 重复步骤(2)到步骤(4), 直到质心不再显著变化。

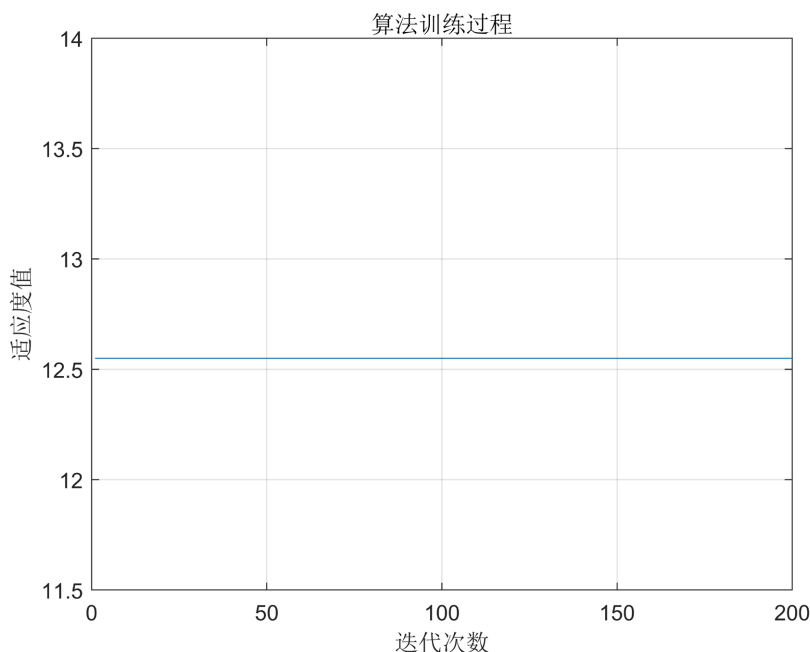


Figure 1. Training process of the improved hybrid particle swarm optimization algorithm
图 1. 改进混合粒子群算法训练过程

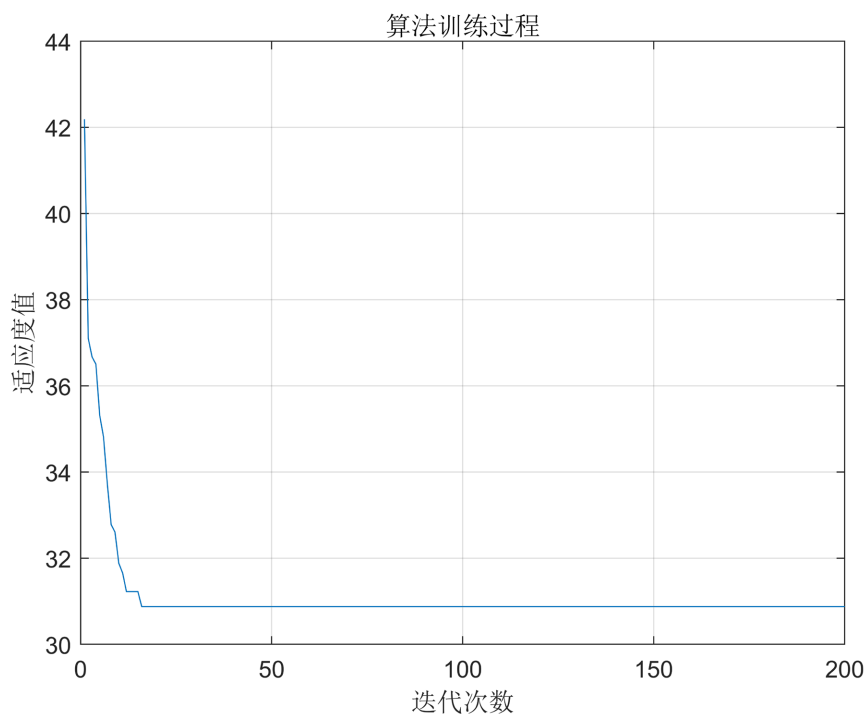


Figure 2. Training process of the hybrid particle swarm optimization algorithm
图 2. 混合粒子群算法训练过程

笔者考虑到在混合粒子群算法中加入 K-Means 算法的聚类思想, 有助于寻找种群中的最优值。先依据城市坐标, 将实验对象进行聚类分析, 再对每个簇分别使用 HPSO 算法。为了验证此想法是否正确, 笔者挑选了一个规模较小的 TSP 问题, 并利用 Matlab 软件进行了仿真实验。数据集选自 TSPLIB 中的 burma14, 对两种算法测试 30 次, 其实验结果见表 1。两种算法的训练过程分别见图 1 和图 2。

Table 1. Results of solving burma14 by using two algorithms

表 1. 两种算法求解 burma14 结果

算法名称	最优值	最差值	平均值
混合粒子群算法	30.88	30.88	30.88
改进混合粒子群算法	30.2694	30.2694	30.2694

可以看出, 两种算法求解结果都比较精准、稳定, 但是混合粒子群算法需要经过迭代才能找到最优解, 而改进的混合粒子群算法的训练过程是一根直线。这表明在解决 burma14 问题时, 改进后的算法在初始聚类的时候就可以找到最优解, 体现了改进算法的优异性能。

4. 仿真实验及分析

为了验证算法的有效性, 本文采用 MATLAB 2023a 对选自 TSPLIB 中的 Oliver30、bayg29、ch130 三组数据集进行仿真实验。采用混合粒子群算法、改进混合粒子群算法分别求解, 并比较结果。仿真环境为 12th Gen Intel (R) Core (TM) i9-12900H 2.50 GHz CPU, 16G RAM, Windows 11 系统, 仿真实验的算法参数为: 种群个数 N 为 1000, 最大迭代次数 n 为 200 次, K-Means 聚类数为 3。对两种算法测试 30 次。

分别使用两种算法求解 Oliver30, 实验结果见表 2, 两种算法的训练过程分别见图 3 和图 4。

Table 2. Results of solving Oliver30 by using two algorithms
表 2. 两种算法求解 Oliver30 结果

算法名称	最优值	最差值	平均值
混合粒子群算法	423.74	427.83	427.336
改进混合粒子群算法	421.0166	447.4668	425.5361

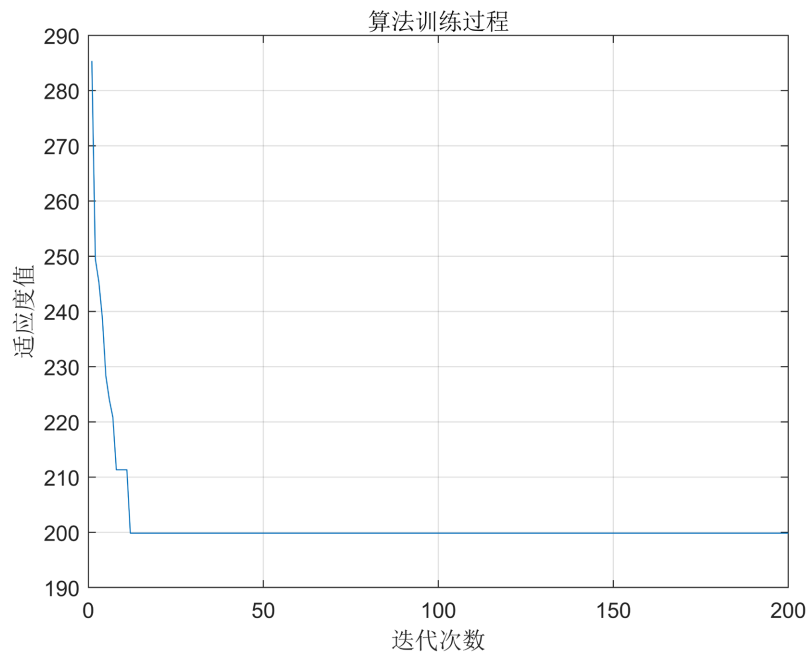


Figure 3. Training process of the improved hybrid particle swarm optimization algorithm
图 3. 改进混合粒子群算法训练过程

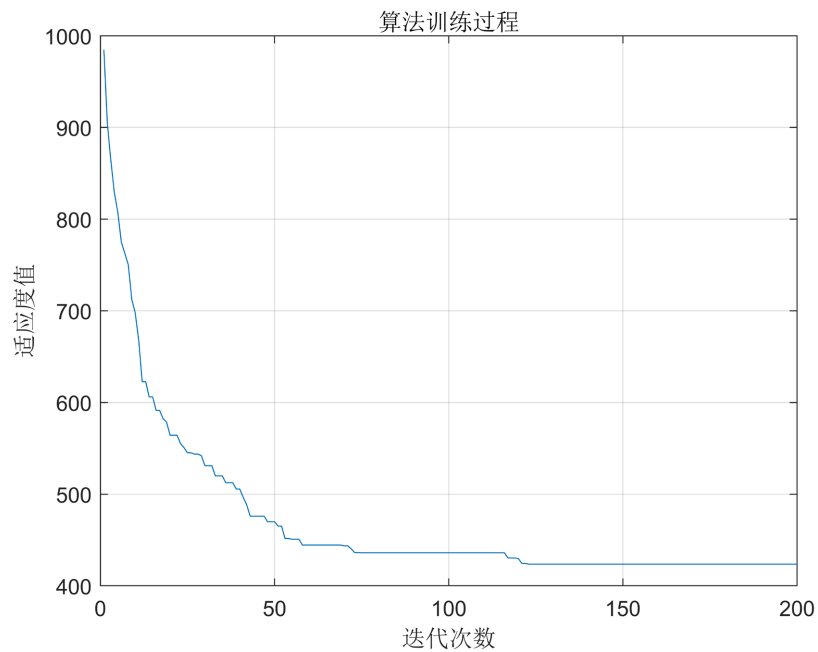


Figure 4. Training process of the hybrid particle swarm optimization algorithm
图 4. 混合粒子群算法训练过程

分别使用两种算法求解 bayg29, 实验结果见表 3, 两种算法的训练过程分别见图 5 和图 6。

Table 3. Results of solving bayg29 by using two algorithms

表 3. 两种算法求解 bayg29 结果

算法名称	最优值	最差值	平均值
混合粒子群算法	9197.05	9675.70	9343.379
改进混合粒子群算法	9167.6654	9463.4618	9217.7816

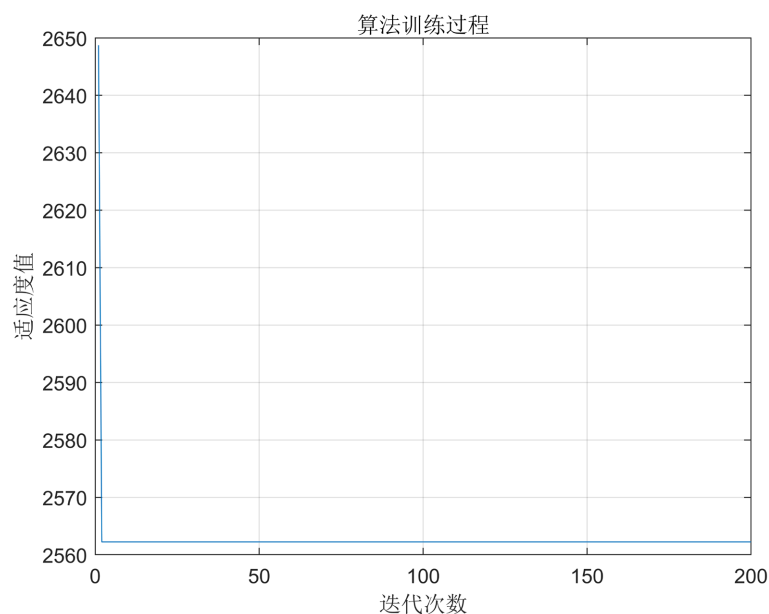


Figure 5. Training process of the improved hybrid particle swarm optimization algorithm

图 5. 改进混合粒子群算法训练过程

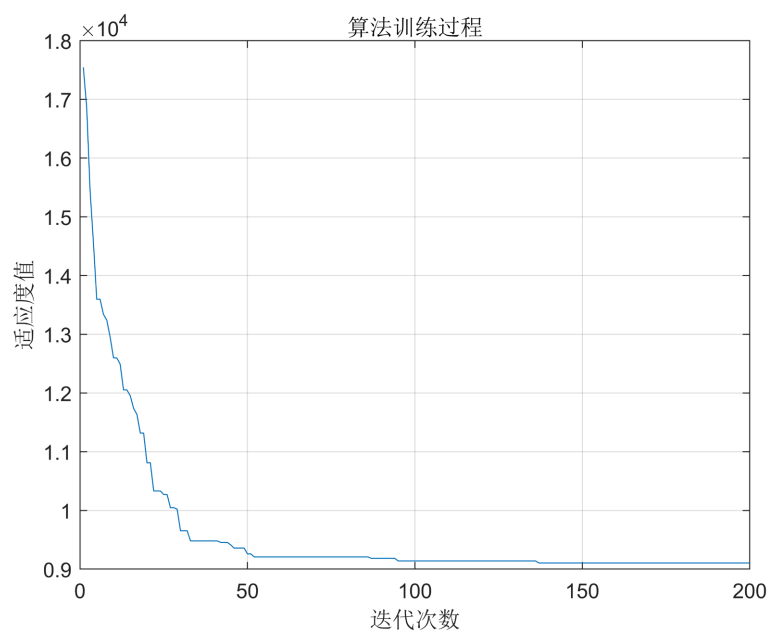


Figure 6. Training process of the hybrid particle swarm optimization algorithm

图 6. 混合粒子群算法训练过程

分别使用两种算法求解 ch130, 实验结果见表 4, 两种算法的训练过程分别见图 7 和图 8。

Table 4. Results of solving ch130 by using two algorithms

表 4. 两种算法求解 ch130 结果

算法名称	最优值	最差值	平均值
混合粒子群算法	6631.8901	13748.64	7145.8056
改进混合粒子群算法	6487.4878	6812.3394	6596.7171

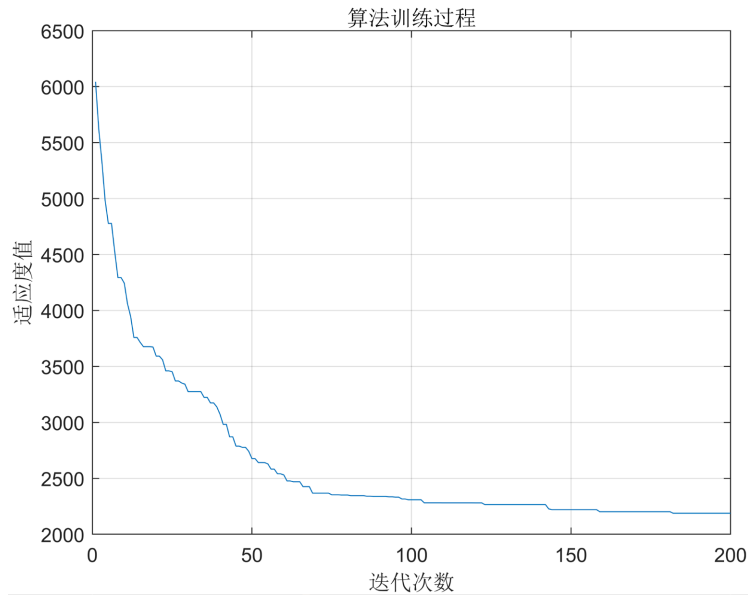


Figure 7. Training process of the improved hybrid particle swarm optimization algorithm

图 7. 改进混合粒子群算法训练过程

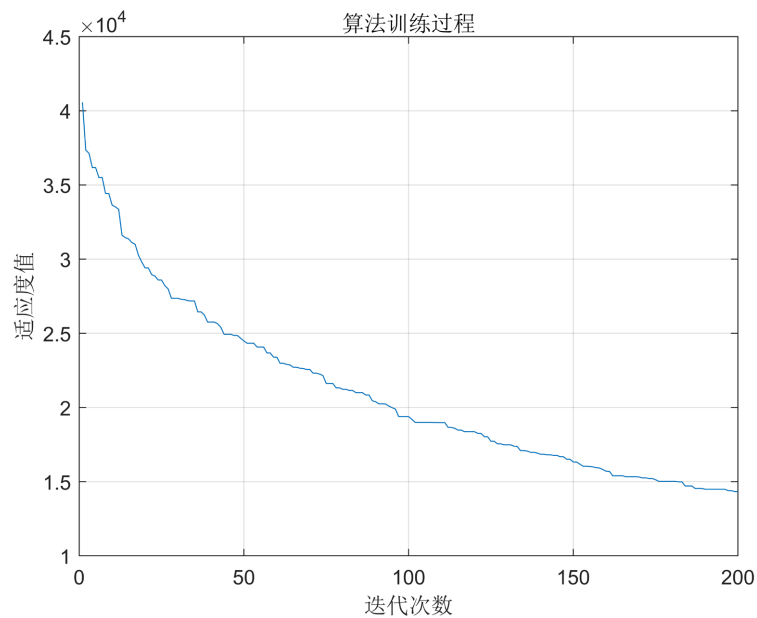


Figure 8. Training process of the hybrid particle swarm optimization algorithm

图 8. 混合粒子群算法训练过程

根据实验结果,可以看出本文所提出的基于 K-Means 算法的混合粒子群算法在解决 TSP 问题的质量上要优于 HPSO 算法,在精准度和迭代速度上都明显更优。两种算法在求解 burma14 这种规模较小的 TSP 问题上,两者相差不多,都相当稳定且精度较高。但是当规模变大时,从表中可以看出,优化混合粒子群算法解决 Oliver30、bayg29、ch130 时明显占优,不仅精度高,而且还具有更快的迭代速度,具有较强的全局搜索能力。当城市规模太大时,HPSO 算法容易陷入局部极值,容易得到异常值,而改进的算法有良好的效果。

5. 结语

本文提出了一种基于 K-Means 算法的改进混合粒子群算法来解决 TSP 问题,在种群初始化前使用了 K-Means 聚类,确定了每个粒子所处的类及其类的中心点,以加快 HPSO 算法的迭代速度。本文最后对选自 TSPLIB 中的 burma14、Oliver30、bayg29、ch130 四组数据进行仿真实验,实验结果证明了本文改进算法的优越性,求解精度更高、迭代速度更快,并且更加适用于大规模 TSP 问题的求解。未来可引入其他智能算法的策略,如变领域搜索,以望得出更有效的求解路径规划问题的方法。

基金项目

2024 年上海市级大学生创新创业训练项目“求解旅行商问题的智能算法研究与应用”(项目编号 S20240901)。

参考文献

- [1] 徐福强, 邹德旋, 章猛, 等. 改进混合粒子群算法求解旅行商问题[J]. 智能计算机与应用, 2022, 12(11): 229-235.
- [2] 董林威, 高宏力, 潘江. 基于改进粒子群算法的路径规划研究与应用[J]. 机械制造与自动化, 2023, 52(6): 81-84.
- [3] 周德荣. 一种求解 TSP 问题的改进粒子群优化算法[J]. 广西民族大学学报(自然科学版), 2015, 21(3): 66-69.
- [4] 陈琳. 基于遗传算法特性的混合粒子群算法求解 TSP 问题[J]. 白城师范学院学报, 2024, 38(5): 73-78.
- [5] 易云飞, 陈国鸿. 基于 K-Means 的改进粒子群算法求解 TSP 问题[J]. 微计算机信息, 2012, 28(9): 475-477.