

轻量级变邻域搜索的多无人机巡逻路径规划

徐若易¹ 陈 实¹ 王麒向¹
XU Ruoyi CHEN Shi WANG Qixiang

摘 要

多无人机协同巡逻路径规划可建模为最小最大多仓库多旅行商问题。现有最优模因算法采用六邻域完整变邻域下降进行局部搜索，单次迭代耗时约 1 s，紧时间预算下迭代次数有限。基于此，文章提出一种轻量级变邻域搜索策略，仅保留路径内 2-opt、单点移动与单点交换三种算子并以两遍固定序列执行，单次迭代耗时降至约 0.02 s，吞吐量提升约 50 倍。将该策略嵌入模因算法框架，在两组基准集共 66 个实例上的实验表明，所提方法在 15 s 和 60 s 时间预算下均显著优于现有最优方法。

关键词

变邻域搜索；模因算法；路径规划；多旅行商问题；迭代优化

doi: 10.3969/j.issn.1672-9528.2026.07.017

0 引言

多无人机协同巡逻在城市安防、应急搜救与环境监测等场景中有广泛需求。在多基站部署条件下，需在秒级时间内为各无人机规划均衡路径，使最长路径尽可能短，该问题可建模为最小最大多仓库多旅行商问题。目前该问题的最优求解方法采用包含六个邻域的完整变邻域下降进行局部搜索，按序遍历并在任一邻域获得改进后从头循环至收敛，单次迭代约耗时 1 s，60 s 预算下仅能执行约 60 次迭代，搜索范围受到制约。本文提出轻量级变邻域搜索（L-VND），以三种算子的两遍固定序列替代六邻域循环下降，将单次迭代耗时降低约 50 倍，以单次搜索深度的适度让步换取迭代吞吐量的大幅提升，使算法在相同时间内覆盖更多候选解。

1 多仓库多旅行商问题的研究背景

1.1 问题背景与应用场景

多无人机协同巡逻可形式化为最小最大多仓库多旅行商问题（min-max multi-depot multiple traveling salesman problem, Min-Max MDMTSP）：给定若干仓库与一组待访问城市，多个旅行商分别从各自仓库出发，协同访问所有城市后各自返回出发仓库，优化目标是最小化所有路径中的最长路径^[1]。该问题在无人机编队任务规划、多机器人协同探索和分布式物流配送等领域有直接的应用背景^[2]。相比经典旅行商问题仅需为单一主体求解一条最短回路，Min-Max MDMTSP 的求解难度体现在三个方面：一是需要将全部城市分配给不同

旅行商，任务分配与路径规划两个子问题相互耦合；二是最小最大目标关注最长路径而非总路径长度，要求各旅行商之间的负载尽可能均衡；三是多仓库设置使得不同旅行商的起止位置各异，进一步增大了空间的规模^[3]。上述特征使得精确算法仅适用于小规模实例，面对实际应用中的大规模场景，启发式与元启发式方法成为主要的求解手段。

1.2 求解方法研究现状

Min-Max MDMTSP 的求解方法可分为传统启发式和基于深度学习两类。Wang 等^[4]提出分配—路由—改进三阶段启发式算法，并构建了基准集 Set M。He 等^[5]将模因算法引入该问题，设计了广义边汇编交叉与六邻域变邻域下降局部搜索，在 Set M 上取得了最优结果。其后续工作^[6]进一步引入强化学习指导邻域选择，但需额外离线训练。Zheng 等^[7]采用聚类构造初始解并结合变邻域搜索优化。深度学习方面，Park 等^[8]提出 ScheduleNet 模型算法，Son 等^[9]设计了 Equity-Transformer 模型算法，Zuo 等^[10]提出 CMIP 框架并将规模扩展至万级节点，但上述方法均需数小时离线训练且不支持多仓库异构配置。综上所述，He 等^[5]的模型算法是目前在线求解性能最优的方法，但其完整 VND 包含路径内插入、路径内交换、跨路径子串交换及路径内迭代局部搜索等六个邻域，按序执行并在任一邻域改进后从头循环至收敛，单次调用约耗时 1 s。在 60 s 预算下仅约 60 次迭代，搜索覆盖的解空间有限。本文的出发点是：在紧时间预算下，适度降低单次搜索深度以换取更高的迭代吞吐量，通过覆盖更多候选解来提升整体求解质量。

2 问题建模与算法框架

2.1 最小最大多仓库多旅行商问题模型

给定完全图 $G=(V, E)$ ，节点集 $V = \{0, 1, \cdots, n+n_d-1\}$ 中前

1. 浙江警察学院信息网络安全学院 浙江杭州 310053

[基金项目] 浙江省科技计划项目“尖兵领雁+X”科技计划(2025C01030)

n 个节点为巡逻点, 后 n_d 个节点为基站, 边 $(i, j) \in E$ 上的权重 c_{ij} 为两节点之间的欧氏飞行距离。共 m 架无人机分布于各基站, 基站 k 配备 $m_k \geq 1$ 架, $\sum_{k=1}^{n_d} m_k = m$ 。每架无人机从所属基站出发, 依次访问若干巡逻点后返回同一基站, 所有巡逻点恰好被一架无人机访问一次, 不同无人机的巡逻点集合互不相交。第 j 架无人机的路径长度为:

$$L_j = c(\sigma_j, v_{j1}) + \sum_{i=1}^{n_j-1} c(v_{ji}, v_{j,i+1}) + c(v_{jn_j}, \sigma_j) \quad (1)$$

式中: σ_j 为所属基站, v_{ji} 为其第 i 个巡逻点, n_j 为该路径包含的巡逻点数。优化目标为:

$$f(S) = \min \max_{j=1,2,\dots,m} \{L_j\} \quad (2)$$

即最小化所有路径中最长者的长度, 迫使各无人机之间的飞行负载趋于均衡。当 $n_d=1$ 、 $m=1$ 时退化为经典 TSP^[1]。

一个可行解 S 包含 m 条路径, 每条路径表示为巡逻点的有序序列。解的编码采用整数排列: 将 n 个巡逻点排成一维序列, 再用 $m-1$ 个分隔符将其划分为 m 段, 每段对应一架无人机的访问顺序, 段的两端隐含所属基站。该编码同时决定了任务分配和访问路径, 后续的交叉与局部搜索算子均直接操作这一排列表示。问题的约束条件包括: 每个巡逻点恰好出现一次, 每架无人机的路径以所属基站为起止点, 各基站的无人机数量不少于其配额 m_k 。

2.2 算法整体框架

本文方法分为种群初始化和快速迭代搜索两个阶段, 如图 1 所示。给定总时间预算 T (如 60 s), 前 20% 分配给种群初始化, 后 80% 用于快速迭代搜索。两个阶段共用同一种群结构, 种群大小固定为 $\mu=30$ 。

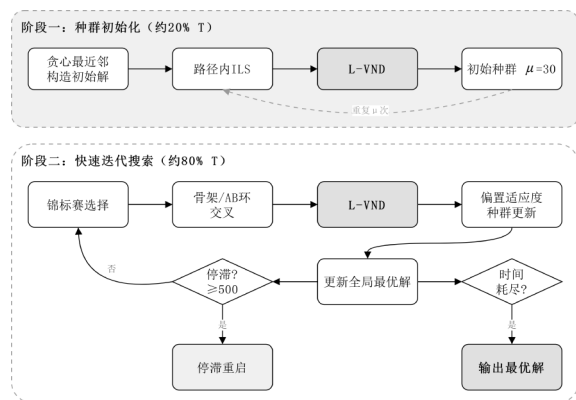


图1 算法整体框架

种群初始化阶段构建一组质量较高且彼此不同的初始个体。对每个个体, 首先采用贪心最近邻方法构造初始解, 依次为每架无人机选取距当前位置最近的未访问巡逻点。随后对每条路径执行路径内迭代局部搜索 (ILS), 交替使用 2-opt 和 Or-opt 改善路径顺序, 并以 double bridge 扰动跳出局部最优。ILS 完成后, 对整个解执行一次 L-VND, 通过跨路径的

节点移动和交换均衡各路径长度。改善后的个体加入种群, 重复至规模达到 μ 。

快速迭代搜索阶段执行进化主循环: 通过锦标赛选择两个父代, 执行交叉算子生成后代, 对后代执行 L-VND 改善, 再按偏置适应度策略决定是否替换种群中的最差个体。该循环持续至时间预算耗尽。L-VND 单次耗时约 0.02 s, 60 s 预算下可执行约 3 000 次迭代。

3 基于轻量级变邻域搜索的快速求解策略

3.1 轻量级变邻域搜索设计

完整变邻域下降算法 (VND) 嵌入 MA, 共设置六种邻域结构, 包含路径内插入、路径内交换、跨路径子串交换与路径内 ILS 等操作。按序执行并在任一邻域获得改进后返回首个邻域重新循环, 最多循环 1 000 轮直至所有邻域均无法改进。这一设计追求单次调用的搜索深度, 但代价是每次调用耗时约 1 s, 60 s 预算下仅能执行约 60 次迭代。L-VND 采用相反的设计思路: 仅保留三种算子并以固定顺序执行两遍, 不循环, 牺牲单次搜索深度换取迭代吞吐量。两者结构对比如图 2 所示。

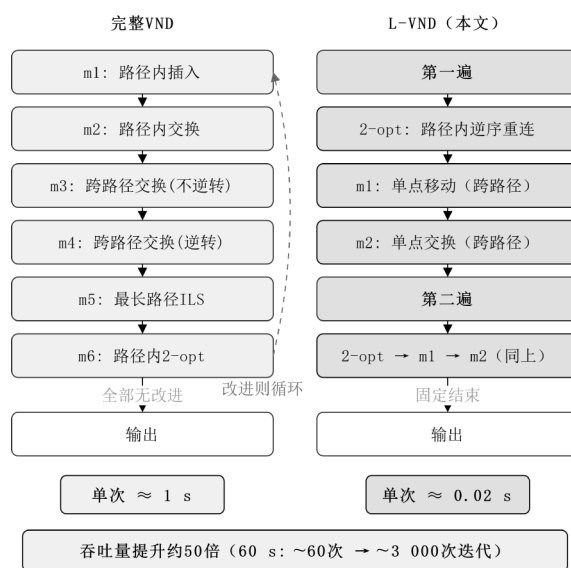


图2 完整 VND 与 L-VND 搜索结构对比

L-VND 包含以下三种算子:

(1) 路径内 2-opt。对当前最长路径执行 2-opt 改善, 选取路径中两条边将中间段逆序重连, 路径长度变化为:

$$\Delta = c(v_i, v_j) + c(v_{i+1}, v_{j+1}) - c(v_i, v_{i+1}) - c(v_j, v_{j+1}) \quad (3)$$

若 $\Delta < 0$ 则接受该改进, 则反复迭代直至路径内无法继续改进。2-opt 消除路径内的边交叉, 计算代价低且收益稳定, 因此作为每遍的第一步执行。

(2) 单点移动 (m1)。从最长路径中取出一个巡逻点, 插入其他路径的最优位置, 选择使全局目标 $f(S)$ 降低最多的

方案。利用预计算的最近邻列表 ($k=10$) 快速定位候选路径, 直接将负载从最长路径转移至较短路径。

(3) 单点交换 (m2)。在最长路径与其他路径之间交换一对巡逻点, 选择使 $f(S)$ 降低最多的方案。m2 在转移负载的同时保持各路径节点数量不变, 适用于 m1 无法改进的情形。

三种算子按 “2-opt $\rightarrow$ m1 $\rightarrow$ m2” 顺序执行两遍后结束。第一遍消除主要的路径交叉和负载失衡, m1 和 m2 的移动改变路径结构后, 第二遍的 2-opt 可对调整后的路径重新消除交叉, m1 和 m2 处理残余改进机会; 实测表明第三遍及之后边际收益极小。两遍设定使 L-VND 单次耗时约 0.02 s, 60 s 预算下可执行约 3 000 次迭代, 相比完整 VND 的约 60 次提升约 50 倍。

3.2 交叉算子与种群管理

主循环每次迭代通过二元锦标赛选择两个父代个体: 从种群中随机抽取两个个体, 选取目标值较优者作为一个父代, 独立重复一次选取另一个父代。选择后以 70% 概率执行骨架交叉, 30% 概率执行 AB 环引导交叉。

骨架交叉保留两个父代中一致的分配结构: 首先找出在两个父代中被分配给同一架无人机的巡逻点, 其归属在后代中保持不变; 对分配不一致的差异节点, 随机继承某一父代的方案并按最小增量准则插入对应路径。AB 环引导交叉基于两个父代的对称差分图, 将边集取对称差形成若干交替环, 随机选择部分环将环内节点按另一父代的分配方案重新归属, 能够在较大范围内重组节点分配。

后代经 L-VND 改进后参与种群更新。种群更新采用偏置适应度策略:

$$bf_i = 0.7 \cdot r_c(i) + 0.3 \cdot r_d(i) \quad (4)$$

式中: $r_c(i)$ 为目标值排名 (归一化至 [0,1], 值越大质量越差), $r_d(i)$ 为多样性排名 (与最相似个体的汉明距离越小排名越高)。偏置适应度最高的个体被优先替换, 使种群同时维持解的质量和结构多样性。

3.3 停滞重启机制

L-VND 的搜索深度较浅, 单次调用仅执行两遍固定序列, 对局部最优的逃离能力有限。当种群中的个体结构趋于相似时, 交叉算子难以产生具有实质差异的后代, 搜索陷入停滞。停滞重启机制通过定期注入全新个体来打破这一困境, 与 L-VND 的浅搜索策略形成互补。

停滞检测采用计数器机制: 每次迭代后检查全局最优解是否更新, 若更新则计数器归零, 否则加一。当计数器达到 500 时判定搜索停滞, 触发重启过程。阈值 500 的设定综合考虑了迭代速度和收敛特征: L-VND 每秒约执行 50 次迭代,

500 次对应约 10 s 的无改进时段, 足以确认搜索确实停滞而非正常的短期波动。

重启过程分为四个步骤: 第一步从种群中按偏置适应度排名选取 $\mu/2$ 个最差个体标记为待替换; 第二步对每个待替换位置重新执行贪心最近邻构造, 由于贪心过程包含随机化的候选选择, 新个体的结构与现有种群存在较大差异; 第三步对新构造的个体依次执行路径内 ILS 和 L-VND 改善; 第四步将改善后的新个体替换原有个体, 计数器归零, 恢复正常主循环。

该机制与 L-VND 形成互补: L-VND 以浅搜索换取高吞吐量, 密集探索当前种群附近的解空间; 当该区域被充分搜索后, 停滞重启引入结构差异大的新个体将搜索重心转移至其他区域。

4 实验验证与分析

4.1 实验环境与参数设置

实验采用两个经典基准集验证本文方法。Set M 由 Wang 等^[4]构建, 含 43 个实例, 城市数 $n \in [10, 500]$, 仓库数 $n_d \in [0, 20]$, 按规模分为小规模 (5 个实例)、中规模 (19 个实例) 和大规模 (19 个实例) 三类, 覆盖了从简单到复杂的多种问题配置。Set P 源自 Cordeau 等^[11]的多仓库车辆路径问题基准, 经适配后用于 MDMTSP 研究, 含 23 个实例, 城市数 $n \in [50, 360]$, 仓库数 $n_d \in [2, 9]$, 规模集中在中等范围。两个基准集共 66 个实例, 对比方法为 He 等^[5]的 MA 算法 (使用原作者官方程序, 参数与原文一致)。

实验在 12 核 Intel Xeon Platinum 8260 CPU (2.40 GHz)、62 GB 内存的 Linux 服务器上以单线程运行, g++ -O2 -std=c++17 编译。种群规模 $\mu=30$, 该值参照 MA 算法的设置以保证对比公平。停滞重启阈值设为 500 次迭代: 由于 L-VND 单次迭代仅约 0.02 s, 500 次对应约 10 s 搜索时间, 既给予种群充分的进化空间, 又能在 15 s 短预算内至少触发一次重启。每个实例在 15 s 和 60 s 两个时间预算下各独立运行 10 次, 记录平均目标值。采用配对 Wilcoxon 符号秩检验评估差异显著性 ($\alpha=0.05$), 该检验不要求数据服从正态分布, 适用于启发式算法的成对比较。

4.2 算法性能对比分析

表 1 为两种方法在两个基准集、两个时间预算下的对比结果。在 Set M 的 43 个实例上, 本文方法 15 s 预算下均值 231.55, 优于 MA 的 235.58 (改进 1.71%), 胜负比 33:10; 60 s 预算下均值 226.03, 优于 MA 的 230.21 (改进 1.82%), 胜负比扩大至 36:7。少数不如 MA 的实例主要出现在小规模实例中, 此时问题较简单, 完整 VND 的深度搜索足以在有限迭代内逼近最优。

表 1 本文方法与 MA 在 66 个基准实例上的对比结果

基准集	时间 /s	MA 均值	本文方法均值	改进 /%	胜 / 负 / 平
Set M	15	235.58	231.55	-1.71	33/10/0
Set M	60	230.21	226.03	-1.82	36/7/0
Set P	15	536.30	513.96	-4.17	23/0/0
Set P	60	522.13	511.97	-1.95	23/0/0

在 Set P 的 23 个实例上，本文方法在两个时间预算下均实现 23:0 全胜。15 s 预算下均值 513.96，优于 MA 的 536.30（改进 4.17%）；60 s 预算下均值 511.97，优于 MA 的 522.13（改进 1.95%）。Set P 的实例规模集中在中等范围，L-VND 的吞吐量优势在该规模下得到充分发挥。本文方法在 Set P 上 15 s 的均值 513.96 已优于 MA 60 s 的 522.13，即仅需四分之一时间即可获得更好结果。

全部四组对比的 p 值均小于 0.000 1，差异高度显著。当时间延长至 180 s 时，两种方法在 Set M 上趋于一致（本文 225.29 vs MA 225.31， $p=0.862$ ），说明 L-VND 的优势集中在有限时间预算内。

4.3 迭代效率与搜索策略分析

表 2 为迭代效率角度对比两种搜索策略。本文提出的 L-VND 将邻域数量从 6 个减少到 3 个，取消循环至收敛机制改为固定执行两遍，单次迭代耗时从约 1 s 降至约 0.02 s。在 60 s 预算下，MA 中的完整 VND 仅能完成约 60 次迭代，而本文方法可执行约 3 000 次迭代。尽管每次迭代的搜索深度较浅，但大量迭代配合交叉算子的全局重组能力，使算法在相同时间内探索了远多于 MA 的候选解区域。

表 2 搜索策略效率对比

搜索策略	邻域数量	单次迭代耗时	60 s 内迭代次数	Set M 60 s 均值	去除初始化 60 s 均值
MA（完整 VND）	6 个，循环至收敛	≈1 s	≈60	230.21	—
本文方法	3 个，固定两遍	≈0.02 s	≈3 000	226.03	226.37
对比优点	简化	50 倍加速	50 倍吞吐量	L-VND 更优	本文更优

为区分性能提升来自搜索策略还是初始化质量，设计了消融实验：将本文初始化替换为与 MA 相同的贪心构造，使两种方法的初始目标值均为 272.31，仅凭 L-VND 搜索 60 s。结果目标值仍降至 226.37，优于 MA 的 230.21（改进 1.66%），说明 L-VND 搜索策略本身是性能提升的主要原因。

图 3 为两种方法在 Set M 上的收敛过程。初始化阶段（0~3 s），本文方法通过路径内 ILS 与 L-VND 预处理种群，建立起始优势。快速下降阶段（3~15 s），L-VND 凭借高吞吐量快速积累改进。稳定改进阶段（15~60 s），本文方法始终保持领先，停滞重启在此阶段帮助跳出局部最优。趋同阶段（60~180 s），L-VND 边际改进减小，MA 凭借完整 VND 的搜索深度缓慢追赶，180 s 后两者差距不再显著。

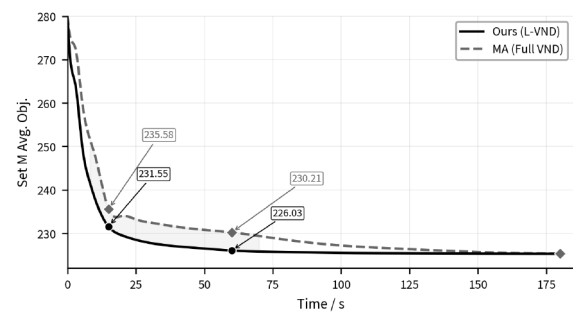


图 3 两种方法在 Set M 上的收敛曲线

5 结语

本文提出了基于轻量级变邻域搜索的快速路径规划方法，以三种算子的两遍固定序列替代六邻域循环下降，单次迭代耗时降低约 50 倍。66 个基准实例上的实验表明该方法在 15 s 和 60 s 预算下均显著优于现有最优方法，消融实验证实搜索策略本身是性能提升的主要原因。该方法适用于无人机编队实时任务分配、应急响应等时间敏感的在线规划场景。但当运行时间延长至 180 s 后两种方法性能趋同，浅层搜索在长时间运行下存在深度不足的局限。未来将探索自适应深度切换机制，搜索早期采用 L-VND 快速收敛，改进速率降低时切换至完整 VND 精细优化。

参考文献：

[1] Bektas T. The multiple traveling salesman problem: an overview of formulations and solution procedures[J]. Omega, 2006, 34(3): 209-219.

[2] Cheikhrouhou O, Khoufi I. A comprehensive survey on the multiple traveling salesman problem: applications, approaches and taxonomy[J]. Computer Science Review, 2021, 40: 100369.

[3] França P M, Gendreau M, Laporte G, et al. The m-traveling salesman problem with minmax objective[J]. Transportation Science, 1995, 29(3): 267-275.

[4] Wang Xingyin, Golden B, Wasil E. The min-max multi-depot vehicle routing problem: heuristics and computational results[J]. Journal of the Operational Research Society, 2015, 66(9): 1430-1441.

[5] He Pengfei, Hao J K. Memetic search for the minmax multiple traveling salesman problem with single and multiple depots[J]. European Journal of Operational Research, 2023, 307(3): 1055-1070.

采用融合 MM5 模式的神经网络预测风电功率研究

曾昌学¹
ZENG Changxue

摘 要

短期风速精准预测是风电场总发电量预估的核心环节，但单一模型难以兼顾气象场推演与场站精细化修正的需求。针对该难题，文章提出一种融合第五代中尺度气象模型（MM5）与神经网络的混合短期风速预测方法。风机小时平均风速作为发电量计算的关键输入参数，该方法以全球数值天气预报数据为基础，将多高度层实测观测数据作为 MM5 模式的初始场与边界条件完成气象场模拟；再通过神经网络对 MM5 模式输出的粗分辨率气象结果进行误差订正与降尺度处理，得到风电场各点位精细化小时风速预报。以西班牙东南部实际风电场开展实例验证，实验结果表明，所提 MM5 神经网络混合框架可有效实现风电场点位短期风速高精度预测，具备良好的工程应用效果。

关键词

短期风速预测；全球预测模型；降尺度；神经网络

doi: 10.3969/j.issn.1672-9528.2026.07.018

0 引言

当前全球面临持续的能源危机，开发利用替代能源，已成为发达国家缓解高油价冲击的重要途径。风能是目前应用最广泛的替代能源之一，在欧洲的能源消费占比约为 10%，在德国、西班牙、美国等国家更是超 15%^[1]。但风力发电存

在一个明显问题，风的波动具有随机性和间歇性，造成风电场的发电量很难准确预测。这种不确定性会给能源输送和电网的电力平衡带来挑战，所以精准的发电量预测，对风电场的日常管理和运行至关重要。

目前风速预测主要有两种技术路线：一是基于历史风速数据的统计预测；二是基于风动力学的气象模型预测。

第一种方法在现有文献中已有不少应用案例^[2]。之所以

1. 华电（云南）新能源发电有限公司 云南昆明 650100

[6] He Pengfei, Hao J K, Xia Jinhui. Learning-guided iterated local search for the minmax multiple traveling salesman problem[J]. Computers & Operations Research, 2026, 185: 107255.

[7] Zheng Jiongzhi, Hong Yawei, Xu Wenchang, et al. An effective iterated two-stage heuristic algorithm for the multiple traveling salesmen problem[J]. Computers & Operations Research, 2022, 143: 105772.

[8] Park J, Kwon C, Park J. Learn to solve the min-max multiple traveling salesmen problem with reinforcement learning[C]// AAMAS'23: Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems. Richland, SC: IFAAMAS, 2023: 878-886.

[9] Son J, Kim M, Choi S, et al. Equity-transformer: solving NP-hard min-max routing problems as sequential generation with equity context[PP/OL].(2024-02-05)[2026-06-15].https://doi.org/10.48550/arXiv.2306.02689.

[10] Zuo Binbin, Li Weifan, Zhao Jiankuo, et al. CMIP: combining constructive model with improvement policy for large-scale min-max multiple traveling salesman problem[J]. IEEE Transactions on Intelligent Transportation Systems, 2025, 27(1): 1550-1564.

[11] Cordeau J F, Gendreau M, Laporte G. A tabu search heuristic for periodic and multi-depot vehicle routing problems[J]. Networks, 1997, 30(2): 105-119.

【作者简介】

徐若易（1994—），男，浙江舟山人，博士研究生，助教，研究方向：演化计算。

陈实（2005—），男，浙江嘉兴人，本科，研究方向：智能优化算法。

王麒向（2006—），男，浙江杭州人，本科，研究方向：智能优化算法。

（收稿日期：2026-05-16 修回日期：2026-06-18）