

多目标野马算法求解开放同时送取货选址路径问题

吴 奥, 张惠珍*, 虎翼飞

上海理工大学管理学院, 上海

收稿日期: 2025年9月10日; 录用日期: 2025年9月18日; 发布日期: 2025年10月20日

摘 要

针对原始野马算法WHO在解决优化问题中存在的早熟收敛、易陷入局部最优等问题, 提出改进多目标野马算法MOIWHO。通过引入模拟二进制交叉算子与正态分布交叉算子以及自适应混沌变异机制, 提高算法跳出局部最优的能力。根据帕累托前沿评价解的质量, 引入均衡接近度确定最优解。进一步结合以最大满意度和最小成本为目标的模糊时间窗开放同时送取货选址路径问题, 提出了一种全新的融合编码解码方式, 通过6组算例证明了算法的有效性, 并通过13个标准算例与4种算法对比, 分析了算法性能。研究结果证明了所提出的多目标改进野马算法在各种规模问题上的有效性与实用价值。

关键词

多目标野马算法, 选址路径问题, 满意度, 开放同时送取货

Multi-Objective Wild Horse Algorithm for Solving the Open Simultaneous Pickup and Delivery Location-Routing Problem

Ao Wu, Huizhen Zhang*, Yifei Hu

Business School, University of Shanghai for Science and Technology, Shanghai

Received: September 10, 2025; accepted: September 18, 2025; published: October 20, 2025

Abstract

To tackle the issues of premature convergence and local optimum entrapment in the original Wild

*通讯作者。

文章引用: 吴奥, 张惠珍, 虎翼飞. 多目标野马算法求解开放同时送取货选址路径问题[J]. 建模与仿真, 2025, 14(10): 195-209. DOI: 10.12677/mos.2025.1410617

Horse Optimizer (WHO) for optimization problems, this paper proposes a Multi-objective Improved Wild Horse Optimizer (MOIWHO). By integrating simulated binary crossover (SBX), normal distribution crossover (NDCO) operators, and an adaptive chaos mutation mechanism, the algorithm's ability to escape local optima is enhanced. Solution quality is evaluated using the Pareto front, with a balanced closeness metric identifying optimal solutions. A novel integrated encoding-decoding scheme is developed for the open simultaneous pickup and delivery location-routing problem with fuzzy time windows, targeting maximum customer satisfaction and minimum operational cost. Effectiveness is validated via 6 test instances, and performance is analyzed by comparing with 4 algorithms on 13 standard benchmarks. Results demonstrate MOIWHO's effectiveness and practical value across problem scales.

Keywords

Multi-Objective Wild Horse Algorithm, Location-Routing Problem, Customer Satisfaction, Open Simultaneous Pickup and Delivery

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

物流环节是确保企业运营效率的关键。科学进行配送中心选址和规划配送路线对于企业提升效率至关重要，这也是众多企业特别是物流公司急需解决的关键问题。

独立的配送中心选址问题(Location-Allocating Problem, LAP) [1]和路线规划问题(Vehicle Routing Problem, VRP) [2]研究已有广泛成果，但传统研究往往将问题分开处理，忽略了它们之间的相互依赖和对整体优化的影响。选址路径问题(Location-Routing Problem, LRP) [3]将二者进行结合，核心目标是通过协同决策设施选址与路径规划，以最小化系统总成本(包括设施开设成本、车辆固定成本及运输成本等)，同时满足客户需求、容量约束等条件，对于物流和供应链管理尤为重要。

求解 LRP 的算法主要分为两大类：1) 精确算法，包括分支切割算法[4]、分支定价算法[5]、列生成算法[6]等。由于 LRP 是 NP-hard 问题，随着问题规模不断扩大，计算量会呈指数增长，因此精确算法难以求解大规模问题。2) 智能优化算法，最经典的如禁忌搜索算法[7]、粒子群算法等。其对 LRP 的应用已相对成熟，在中大型问题上表现良好。对于多目标选址路径问题，则是非支配遗传算法[8]和多目标粒子群算法[9]较多。

野马算法(Wild Horse Optimizer, WHO)是 Naruei 等[10]在 2021 年提出的一种新的智能优化算法，其局部搜索能力强、迭代迅速。目前已被用于无人机路线等工程优化问题，且已有研究[11]对其进行了改进，但对于多目标 LRP 的研究暂无，因而有待深入研究。

本文深入研究并组合了开放式同时送取货等选址路径问题模型，同时以最大化客户满意度为第二目标，提出了多目标选址路径问题模型。设计了多目标改进 WHO 进行求解。最后通过求解标准算例验证算法的有效性。

2. 问题描述与数学模型

2.1. 问题描述

企业在城市物流中的挑战是建立一定数量的设施，以满足区域内客户的需求。客户的送取货需求和

位置、潜在设施的开放成本、最大负荷和位置、车辆载重和启用成本均已知。客户存在他们期望的服务时间窗。企业需要在潜在设施中选择设施开放，配置适当数量的车辆完成运输任务，实现最小化总成本并尽量满足客户服务时间需求，最终不返回设施。根据问题特点和约束，建立了数学模型。模型参数及说明如表 1 所示。

Table 1. Model parameters and descriptions

表 1. 模型参数及说明

参数	参数说明
I	设施集合
J	客户集合
K	车辆集合
U	所有节点集合
c_i	在 i 处开放设施的成本
c_{ij}	点 i 到 j 的运输成本
d_{ij}	点 i 到 j 的距离
s_k	车辆 k 的指派成本
Q_k	车辆 k 的运载能力
Q_i	设施 i 的最大容量
p_j	客户 j 送货需求
q_j	客户 j 取货需求
Q_{ijk}	车 k 从 i 到 j 载货量
u_{ik}	车辆 k 访问客户 i 的顺序
M	一个足够大的整数
T_{jk}	车辆 k 开始服务客户 j 的时刻
$[a_j, b_j]$	客户 j 的满意时间窗
t_{ij}	车辆从 i 行驶到 j 的时长
t_p	客户单位送货时间
t_q	客户单位取货时间
x_{ijk}	车辆 k 从节点 i 驶向节点 j 则为 1，否则为 0
y_i	第 i 个候选设施开放为 1，否则为 0
y_{ij}	设施 i 与客户 j 之间存在指派关系 1，否则为 0
S_{jk}	表示服务是否在客户的期望时间窗内完成则为 1，否则为 0

2.2. 数学模型

基于上述假设和参数描述，构建了由模糊时间窗、容量约束、多重成本最小以及满意度最大为目标的开放式同时送取货选址路径数学模型。

$$\text{Min} z_1 = \sum_{i \in I} c_i y_i + \sum_{i \in U} \sum_{j \in U} \sum_{k \in K} c_{ij} x_{ijk} + \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} s_k x_{ijk} \quad (1.1)$$

$$\text{Max}Z_2 = \sum_{j \in J} \sum_{k \in K} \frac{S_{jk}}{n} \quad (1.2)$$

s.t.

$$\sum_{i \in U} \sum_{k \in K} x_{ijk} = 1, \forall j \in J \quad (1.3)$$

$$\sum_{j \in U} x_{ijk} - \sum_{j \in U} x_{jik} = 0, \forall i \in U, \forall k \in K \quad (1.4)$$

$$\sum_{i \in U} \sum_{j \in J} x_{ijk} \leq 1, \forall k \in K \quad (1.5)$$

$$\sum_{o \in J} x_{io} + \sum_{o \in \frac{U}{\{J\}}} x_{ojk} \leq 1 + y_{ij}, \forall i \in I, \forall j \in J, \forall k \in K \quad (1.6)$$

$$u_{ik} - u_{jk} + Mx_{ijk} \leq M - 1, \forall i \in J, \forall j \in J, \forall k \in K \quad (1.7)$$

$$\sum_{j \in J} p_j y_{ij} \leq Q_i y_i, \forall i \in J \quad (1.8)$$

$$\sum_{j \in J} q_j y_{ij} \leq Q_i y_i, \forall i \in I \quad (1.9)$$

$$\sum_{j \in J} Q_{jik} - \sum_{j \in J} Q_{ijk} = q_i - p_i, \forall i \in I, \forall k \in K \quad (1.10)$$

$$Q_{ijk} \leq Q_k x_{ijk}, \forall i \in U, \forall j \in U, \forall k \in K \quad (1.11)$$

$$\sum_{k \in K} \sum_{j \in J} x_{jik} = 0, \forall i \in I \quad (1.12)$$

$$T_{jk} = T_{ik} + t_{ij} + t_p p_j + t_q q_j, \forall i \in U, \forall j \in J, \forall k \in K \quad (1.13)$$

$$x_{ijk} \in \{0, 1\}, \forall i \in U, \forall j \in U, \forall k \in K \quad (1.14)$$

$$y_i \in \{0, 1\}, \forall i \in I \quad (1.15)$$

$$y_{ij} \in \{0, 1\}, \forall i \in I, \forall j \in J \quad (1.16)$$

式(1.1)表示总成本最小, 包括设施开放成本、运输成本、车辆启用成本; (1.2)表示总满意度最大, 即需要满足尽量多客户的满意时间窗; (1.3)表示每个客户仅能且必被访问一次; (1.4)表示进入节点流量平衡; (1.5)表示车次使用限制; (1.6)表示每个客户仅能被一个设施服务; (1.7)表示消除子回路; (1.8)和(1.9)分别表示送取货不超过设施容量限制; (1.10)是车辆装卸平衡约束; (1.11)表示车辆不能超载; (1.12)表示车辆运送完成后不返回设施; (1.13)表示车辆到达下一节点的时间等于到达上一节点时间加上在该节点的送取货服务时间与路上所用时间; (1.14)~(1.16)为决策变量取值范围。

3. 求解方法

3.1. 编码及编码方式

3.1.1. 编码

由于 LRP 为离散问题, 因此采用实数编码方式, 为仓库、车辆的选用、路径的构建三个决策进行染色体编码并融合, 以统一规则同时在解空间搜索更优解。具体而言, 每个基因的取值为 0~1 范围内的实数。例如, 给定具有可选设施 5 个, 客户 20 个, 备选车辆 7 辆的 LRP, 染色体长度为 $5 + 7 + 20 = 32$ 。如图 1 所示, 蓝色、紫色、绿色部分分别为 5 个候选设施、7 辆车、20 个客户的染色体片段。

0.60012	0.51460	0.66014	0.30570	0.04219	0.60383	0.27985	0.17209
0.32881	0.79134	0.63529	0.61353	0.07956	0.47632	0.64734	0.78117
0.21645	0.96952	0.32551	0.52779	0.14661	0.15841	0.27246	0.59710
0.06919	0.37873	0.85815	0.54952	0.84724	0.45762	0.65898	0.75052

Figure 1. A schematic diagram of a complete chromosome containing 32 genes.

图 1. 含有 32 个基因的完整染色体示意图

3.1.2. 解码

当产生新的完整染色体后，以下列规则进行解码：

(1) 前 5 个基因计算建设哪些设施 i 进行，本文选择小于 0.5 的位置，即该例中选择开放候选设施 i_4 和 i_5 。

(2) 随后 7 个基因代表备选车辆，分别乘以可建设设施 i 的数目 2 ((1)中得出)并向上取整，得到备选车辆所选择开放的设施的编号，编号 1、2 代表开放的第一、二个候选设施 i_4 、 i_5 ，如图 2 所示。

(3) 剩余 20 个基因代表客户，用于计算客户所选择的车辆及运输顺序。将其乘以备选车辆数量 7 并向上取整可得每个客户所选车辆编号。之后分别按基因大小排序得到车辆服务客户顺序。以第四辆车为例，共有结果编号 2、8、16、18 需其服务，如图 3 所示，对以上结果进行升序排列即可得服务顺序为 $i_4 \rightarrow j_{18} \rightarrow j_2 \rightarrow j_8 \rightarrow j_{16}$ 。

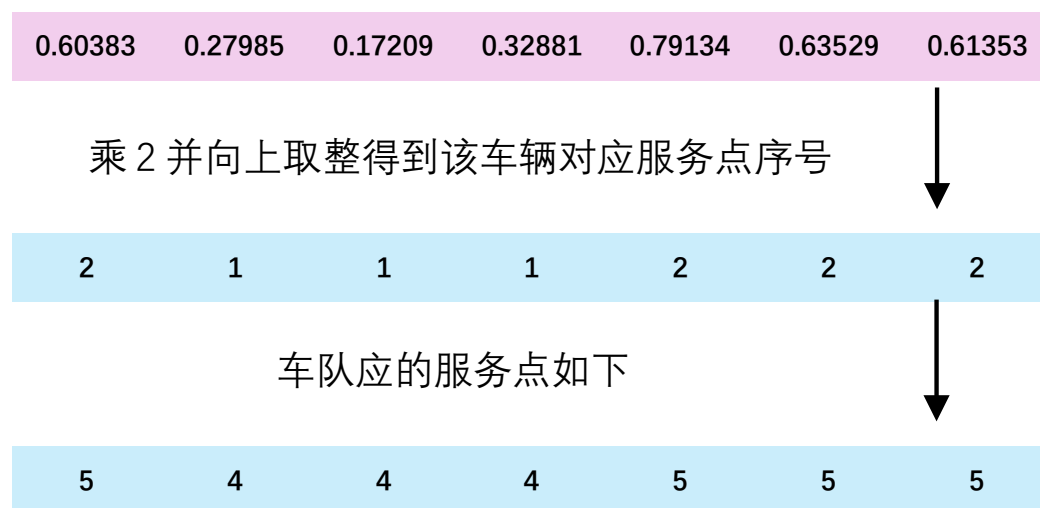


Figure 2. Alternative vehicle gene selection facility process

图 2. 备选车辆基因选择设施流程

1	4	5	6	2	7	3	4	2	2
2	5	1	3	7	4	6	4	5	6

Figure 3. The service vehicle corresponding to the customer

图 3. 客户对应的服务车辆

3.2. 基本 WHO

基本 WHO 由五个步骤组成：1) 创建初始种群和挑选种马；2) 放牧和交配；3) 种马领导和带领群体；4) 交流选拔种马；5) 保存解决方案。

初始种群被分为多组， N 是初始种群个体数量， PS 是种马的比例， $G = [N \times PS]$ 为种马群体， $(N - G)$ 为其余环绕种马的马群。

放牧行为表现为式(2.1)：

$$\bar{X}_{i,G}^j = 2Z \cos(2\pi RZ) \times (Stallion^j - X_{i,G}^j) + Stallion^j \quad (2.1)$$

式中， $\bar{X}_{i,G}^j$ 是马群当前位置， $Stallion^j$ 是种马位置， Z 为适应机制系数， R 是 $[-2, 2]$ 范围内的随机数， $\cos$ 函数使原函数在小范围内扰动，产生新的 $\bar{X}_{i,G}^j$ 以表示种群更新后的位置。其中， Z 由式(2.2)确定：

$$IDX = (\bar{R}_1 < TDR = 0); Z = R_2 \Theta IDX + \bar{R}_3 \Theta (\sim IDX) \quad (2.2)$$

式中， $\bar{R}_1$ 和 $\bar{R}_3$ 为 $[0, 1]$ 空间内均匀分布的随机向量； R_2 为 $[0, 1]$ 之间的随机数； IDX 为维度与 $\bar{R}_1$ 相同的向量，若 $\bar{R}_1$ 第一个元素小于 TDR ，则 IDX 第一个元素为 0，反之为 1，依次类推； TDR 为迭代次数从 1 线性递减到 0 的自适应因子； Θ 为点乘； $\sim$ 为二进制取反。交配行为由临时组中来自不同种群的小马产生，表达式为(2.3)：

$$X_{G,k}^p = \text{Crossover}(X_{G,i}^q, X_{G,j}^z) \quad i \neq j \neq k, p = q = \text{end} \\ \text{Crossover} = \text{Mean} \quad (2.3)$$

式中， $X_{G,k}^p$ 表示种群 k 中个体 p 离群后再次进入到种群 k 中的个体位置， $X_{G,i}^q$ 和 $X_{G,j}^z$ 分别表示其来自不同组的父代 q 与母代 z 。之后种马带领种群前往合适的水洼，但若该水洼已被另一个种群占领，后来的种马将寻找其他水洼。具体计算公式为(2.4)：

$$\overline{Stallion}_{G_i} = \begin{cases} 2Z \cos(2\pi RZ) \times (WH - Stallion_{G_i}) + WH, R_3 > 0.5 \\ 2Z \cos(2\pi RZ) \times (WH - Stallion_{G_i}) - WH, R_3 \leq 0.5 \end{cases} \quad (2.4)$$

式中， $\overline{Stallion}_{G_i}$ 为 i 组种马的新位置， WH 为水洼的位置， $Stallion_{G_i}$ 为 i 组种马现在的位置， Z 、 R 同上。当种马适应度被其他成员超越时，执行种马交换，公式为(2.5)：

$$Stallion_{G_i} = \begin{cases} X_{G_i} & \text{if } \text{cost}(X_{G_i}) < \text{cost}(Stallion_{G_i}) \\ Stallion_{G_i} & \text{if } \text{cost}(X_{G_i}) > \text{cost}(Stallion_{G_i}) \end{cases} \quad (2.5)$$

3.3. 多目标改进野马算法 MOIWHO

WHO 参数少、寻优能力强、时间复杂度低，但鲁棒性较弱，容易陷入局部最优。故需对 WHO 改进以适应多目标 LRP。为增强全局搜索能力，防止陷入局部最优，对 WHO 进行以下改动：

1) 交叉算子改进

原算法在交配的过程中，使用单点/多点交叉。本文在实验过程中发现，当数据维度过大时效果较差，故使用模拟二进制交叉算子与正态分布交叉算子，结合二者的优势，同时增强局部搜索与全局搜索能力。自适应交叉算子公式为：

$$\begin{cases} x_1 = \frac{t}{2T}[M + \alpha N] + \frac{T-t}{2T}[M + \beta N] \\ x_2 = \frac{t}{2T}[M - \alpha N] + \frac{T-t}{2T}[M - \beta N] \end{cases} \quad (2.6)$$

其中:

$$\begin{cases} M = p_1 + p_2 \\ N = p_1 - p_2 \end{cases} \quad (2.7)$$

式中, p_1 和 p_2 为父代基因, x_1 和 x_2 为子代基因; t 为当前迭代次数, T 为总迭代次数; α 和 β 分别为二进制交叉 SBX 和正态分布交叉 NDX 的作用因子, 其计算公式为:

$$\alpha = \begin{cases} (2\mu)^{\frac{1}{\eta+1}}, & \mu \leq 0.5 \\ [2(1-\mu)]^{\frac{1}{\eta+1}}, & \mu > 0.5 \end{cases} \quad (2.8)$$

$$\beta = |N(1, 0.2)| \quad (2.9)$$

式中, η 为非负数, 一般取 10; μ 为(0,1)的随机数; $|N(1, 0.2)|$ 为均数为 1、标准差为 0.2 的正态分布随机变量。

2) 自适应混沌变异

针对 WHO 容易过早陷入局部最优的问题, 提出了最大失败统计数 λ (经多次试验验证设定为迭代次数的 1/10 较有效), 当算法连续运行 λ 次都没有找到更优解时, 意味着局部收敛, 进行混沌变异操作。具体为: 删除种群中最差的 m 个个体, 然后通过混动映射产生新的 m 个个体替换, 经多次实验证明该方法有效, 且当 $m = 10$ 时, 性能较为明显。

改进多目标 WHO 使用中心 tent 混动映射产生新个体, 具体公式如下:

$$x_{n+1} = \begin{cases} 2x_n, & 0 < x_n < \frac{0}{5} \\ 2(1-x_n), & 0.5 \leq x_n < 1 \end{cases} \quad (2.10)$$

式中, x_n 和 x_{n+1} 分别代表旧个体和新个体。

3.4. 适应度函数

多目标问题中子目标可能会相互冲突, 不能简单按照成本对解决方案进行排序, 需采用不同的适应度函数来评估解的质量。

对此, 本文采用了帕累托优化理论中的非支配排序法进行评估, 通过比较解集之间的占优关系来判断它们的相对优劣, 从而筛选出一组在所有目标上都不被其他解支配的解。为了进一步评估这些解的质量, 引入灰关联度和均衡接近度, 通过两种指标来综合考虑解决方案在各个目标上的表现。最终根据计算得到的均衡接近度来确定每个解的适应度值, 以便在解决方案之间进行选择和优化。

传统灰关联度数可以确定两组序列之间的接近程度, 但存在局部点关联倾向, 黄丽萍等[12]在此基础上, 通过引入灰熵理论, 提出了均衡接近度方法, 改善了这一问题。

灰关联度定义如下:

设序列集 $X = \{x_i | x_i = (x_i(1), x_i(2), \dots, x_i(k)), i \in N\}$ 为关联因素集, 其中

$N = \{1, 2, \dots, n\}$, $k \in K$, $K = \{1, 2, \dots, k\}$ 。参考序列 $X_0 = \{x_0(k) | k \in K\}$ 与比较序列 $X_i = \{x_i(k) | k \in K\}$ 的灰关联度有:

$$\gamma(x_0(k), x_i(k)) = \frac{\min_i \min_k |x_0(k) - x_i(k)| + \xi \max_i \max_k |x_0(k) - x_i(k)|}{|x_0(k) - x_i(k)| + \xi \max_i \min_k |x_0(k) - x_i(k)|} \quad (2.11)$$

$$\gamma(X_0, X_i) = \frac{1}{k} \left(\sum_{k \in K} \gamma(x_0(k), x_i(k)) \right) \quad (2.12)$$

式中, ξ 为分辨系数, $\xi \in (0, 1)$; $\min \min$ 为最最小值; $\max \max$ 为最最大值。式(2.11)计算了第 i 个比较序列 X_i 与参考序列 X_0 在第 k 个目标函数上的灰关联系数; 式(2.12)计算了第 i 个比较序列 X_i 与参考序列 X_0 的灰关联度。

熵关联度定义如下:

$R(i)$ 为比较序列 i 的关联系数数列, $R_i = \{\gamma(x_0(k), x_i(k)), k \in K\}$, 有:

$$\rho_i(k) = \frac{\gamma(x_0(k), x_i(k))}{\sum_{k \in K} \gamma(x_0(k), x_i(k))} \quad (2.13)$$

$$B(R_i) = \frac{-\sum_{k \in K} \rho_i(k) \ln \rho_i(k)}{\ln n} \quad (2.14)$$

其中, 式(2.12)为第 i 个比较序列 X_i 在第 k 个目标函数上的灰关联度系数分布映射, 也叫灰关联密度值; 式(2.14)为第 i 个比较序列 X_i 与灰关联熵最大值的比值, $\ln n$ 是最大熵值, 即为第 i 个比较序列 X_i 的均衡度, 又称熵关联度。

均衡接近度定义如下:

已知灰关联度与均衡度, 较序列 X_i 与参考序列 X_0 之间的均衡接近度 $B(X_0, X_i)$ 即为:

$$B(X_0, X_i) = \gamma(X_0, X_i) \times B(R_i) \quad (2.15)$$

以上均衡接近度运用于多目标优化可以有效改善因传统灰关联度强局部点关联倾向导致的关联性倾斜, 使得所求结果的有效性进一步提高。

本节中, 带模糊时间窗开放式同时送取货选址路径问题两个目标分别为最小化总成本与最大化满意度。此两者值组成的序列即为参考序列, 其他可行解的目标值序列为比较序列。现引入帕累托支配关系评判解的优劣, 更新策略如下:

$$X^{t+1} = \begin{cases} X^t, X^t \prec X^{t+1} \\ X^t, X^t \text{ 与 } X^{t+1} \text{ 互不支配, } B(X^t) > B(X^{t+1}) \\ X^{t+1}, X^{t+1} \prec X^t \\ X^{t+1}, X^t \text{ 与 } X^{t+1} \text{ 互不支配, 且 } B(X^{t+1}) > B(X^t) \end{cases} \quad (2.16)$$

其中, t 为算法迭代次数, X^t 为第 t 次迭代的解, $B(X^t)$ 为 X^t 的均衡接近度, 与适应度值正相关。

3.5. 算法步骤

改进 WHO 步骤如图 4 所示:

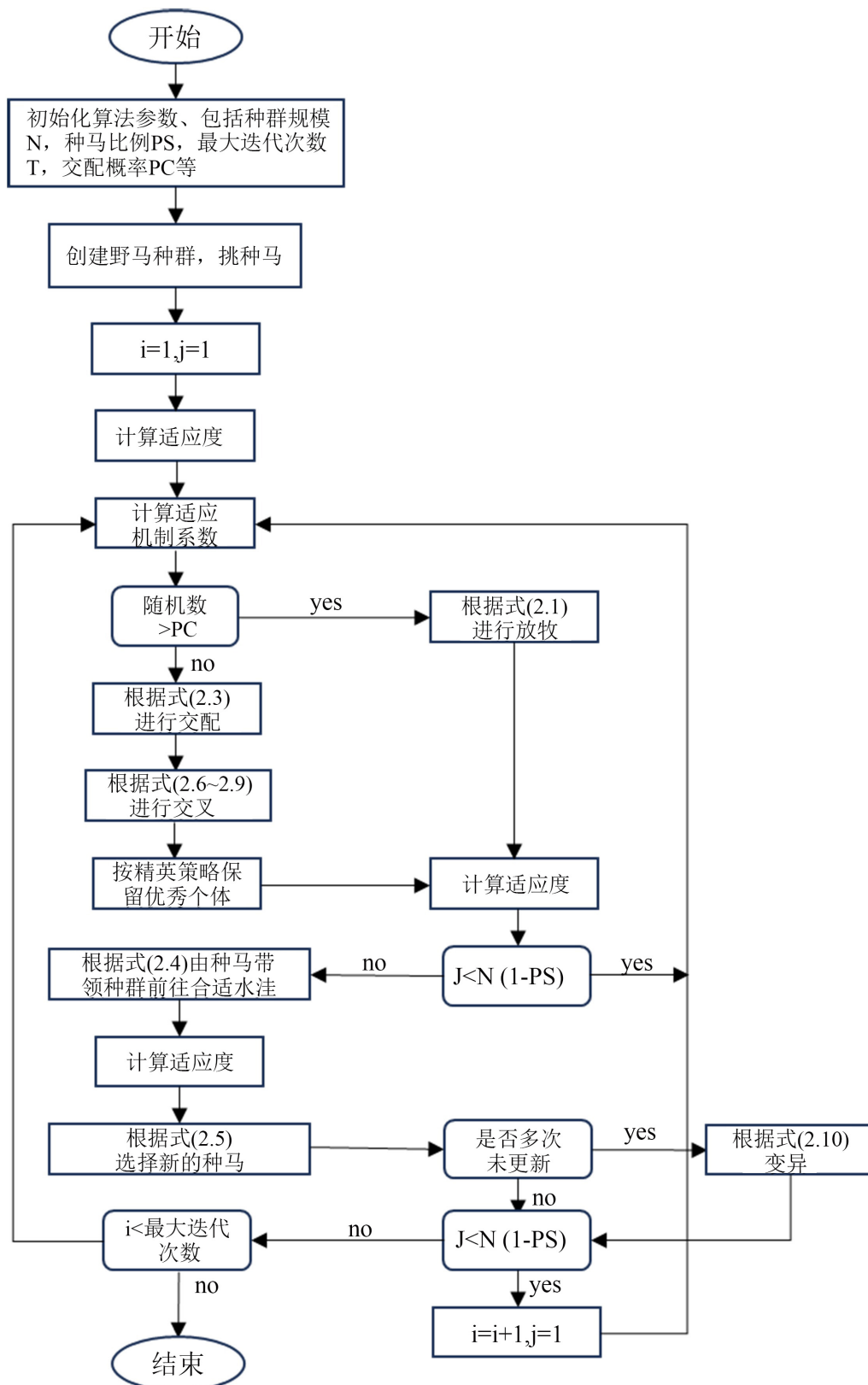


Figure 4. Flowchart of the multi-objective improved wild horse algorithm
图 4. 多目标改进野马算法流程图

4. 实验分析

4.1. 实验数据

为评估改进多目标 WHO 的求解性能, 使用 Matlab R2021a 实现算法, 并在 64 位 Microsoft Windows 10 系统下, 配备英特尔 12th Gen Intel(R) Core(TM) i5-12600KF(3700 MHz)处理器和 32.00 GB (3600 MHz) 内存的主机进行了大量数据实验。

本文在 Prins 等[13]提出的 CLRP 算例数据上做了相应改进, 利用 AM 分离法生成客户的送取货需求, 即令客户点送货需求为原需求, 取货需求 p_i 为:

$$p_i = \begin{cases} (1+\lambda)q_i & i \text{ 为奇数} \\ (1-\lambda)q_i & i \text{ 为偶数} \end{cases} \quad (3.1)$$

其中, q_i 为算例中原始需求, λ 取 0.7。

为验证算法有效性, 本文算例增加了时间窗数据, 具体公式为: $a_j = randi(0,400)$; $b_j = a_j + randi(30,60)$, 其他数据如表 2 所示。

Table 2. 6 sets of benchmark instances
表 2. 6 组基准算例

算例编号	名称	设施数量	客户数量	车辆启用成本	车辆容量
1	coord20-5-1	5	20	1000	70
2	coord20-5-1b	5	20	1000	70
3	coord50-5-1	5	50	1000	70
4	coord50-5-1b	5	50	1000	150
5	coord100-5-1b	5	100	1000	150
6	coord100-10-1	10	100	1000	150

4.2. 参数设置

为得到更佳参数组合, 对算例 1、2 进行了分析。通过比较算例最优目标函数值和平均运行时间, 对种群规模 N、种马数量 G 进行参数分析。在研究其中一个参数性能时, 其他参数保持不变。

表 3 为种群规模在 6 种不同取值情况下对 2 组算例的平均运行结果, 加粗部分为最优结果。图 5 为两组算例对应的变化曲线图。当种群规模超过 40 时, 两组算例都取到最优解。平均时间受种群规模影响不大, 但并非随种群规模的提高而提高, 两组算例的最优运行时间分别在种群规模 120 (算例 1)、种群规模 60 (算例 2)处, 但波动不大。综合来看, 本文对改进多目标 WHO 种群规模的选择为 40。

Table 3. The impact of population size on the solution
表 3. 种群规模对解的影响

种群规模	总成本		平均运行时间(s)	
	20-5-1	20-5-1b	20-5-1	20-5-1b
20	42,706	30,479	5.07	3.34
40	35,876	28,611	4.94	3.30
60	36,799	29,642	4.99	3.11

续表

80	38,405	30,582	4.96	3.45
100	39,574	32,268	4.66	3.23
120	36,780	31,937	4.55	3.32

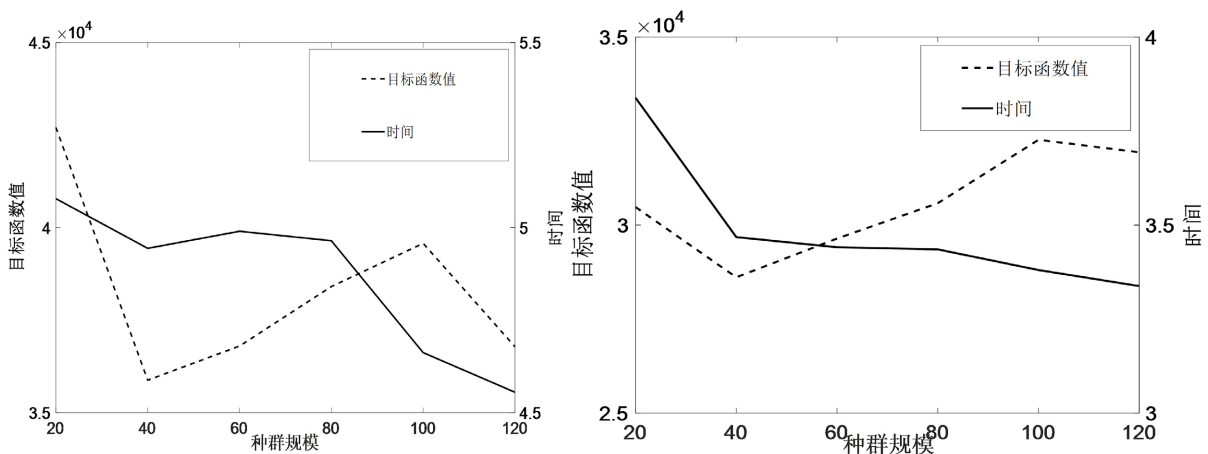


Figure 5. Changes in objective function values and computation time with respect to population size for case 1 (left) and case 2 (right)

图 5. 算例 1 (左)、2 (右)目标函数、时间随种群规模变化情况

表 4 为种马数量在 6 种不同取值情况下对 2 组算例的平均运行结果，加粗部分为最优结果。图 6 为两组算例对应的变化曲线图。

对于两组算例，目标值基本随着种马数量增加而变优，算例 1 在种马数量 30 处略显波动，算例 2 在种马数量 36 处取到较好值，但两组算例都在种马数量 36 处取到最好值。对运行时间而言，两组算例运行时间都随种马数量增多而严格增多，且有随指数增多的趋势。综合来看，种马数量取 36 时，总成本最低，且运行时间可接受。故本文多目标改进 WHO 种马数量取值为 36。

Table 4. The impact of stallion count on the solution

表 4. 种马数量对解的影响

种马数量	总成本		平均运行时间(s)	
	20-5-1	20-5-1b	20-5-1	20-5-1b
6	46,770	30,226	1.74	1.46
12	41,478	35,044	4.85	3.32
18	37,885	31,177	10.24	7.17
24	34,772	29,245	17.54	12.37
30	36,313	26,959	26.67	18.44
36	33,854	26,304	37.09	25.86

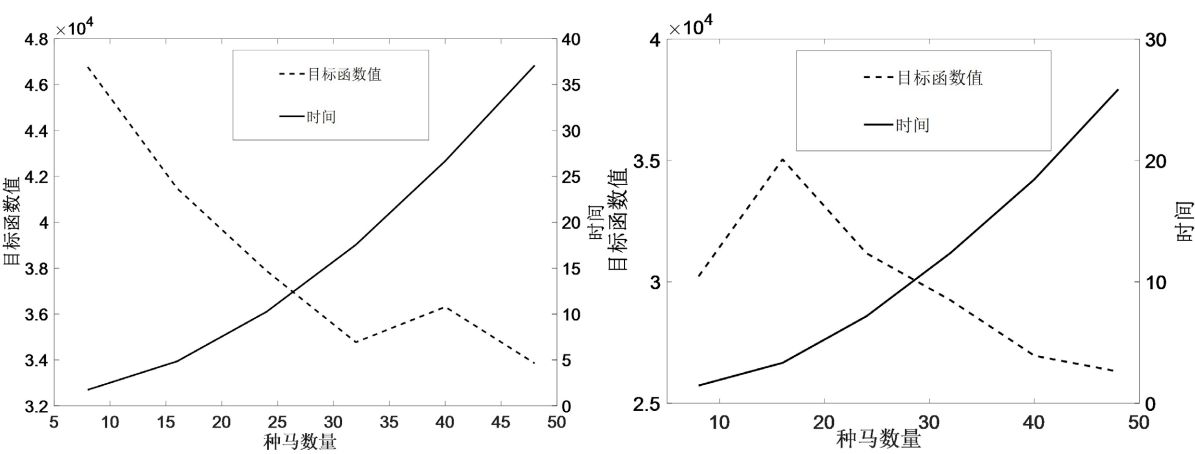


Figure 6. Changes in objective function values and computation time with respect to stallion count for case 1 (left) and case 2 (right)

图 6. 算例 1 (左)、2 (右)目标函数、时间随种马数量变化情况

4.3. 算例结果分析

为验证多目标改进 WHO 求解的有效性，选取 4 组算例各运行 15 次，每组算例都会得到帕累托解集。4 组算例根据客户数量 20、50 分为两种规模，以下对每种规模选取一组帕累托解集进行分析。

1) 客户数量为 20 的算例：

Table 5. Comparison of Pareto solution sets, grey relational degrees, equilibrium degrees, and equilibrium proximity for 20 customers

表 5. 20 客户数的帕累托解集、灰关联度、均衡度、均衡接近度比较

序号	总成本	满意度	灰关联度	均衡度	均衡接近度
1	33,854	81.10%	0.667	0.811	0.541
2	34,525	81.30%	0.588	0.863	0.507
3	34,561	84.06%	0.603	0.892	0.538
4	35,628	84.95%	0.526	0.948	0.499
5	35,889	86.14%	0.520	0.965	0.502
6	36,151	86.55%	0.510	0.974	0.496
7	36,198	87.11%	0.512	0.978	0.501
8	36,524	87.34%	0.499	0.985	0.491
9	36,703	87.54%	0.493	0.989	0.487
10	37,217	90.31%	0.503	1.000	0.503
11	37,325	91.07%	0.510	1.000	0.510
12	37,589	91.29%	0.504	0.999	0.503
13	38,328	91.65%	0.486	0.994	0.483
14	38,428	91.92%	0.488	0.992	0.484
15	38,639	92.72%	0.495	0.986	0.488
16	38,843	92.49%	0.486	0.985	0.479

续表

17	38,949	94.44%	0.520	0.968	0.503
18	39,032	94.85%	0.527	0.962	0.507
19	39,053	95.11%	0.532	0.959	0.510
20	39,436	96.54%	0.560	0.931	0.521
21	39,505	97.46%	0.587	0.913	0.536
22	39,543	98.07%	0.607	0.900	0.546
23	39,560	98.36%	0.617	0.893	0.552
24	40,936	100%	0.667	0.811	0.541

针对算例 1 和 2，每次实验都会生成相应的帕累托解集，共 30 次实验产生了 30 组帕累托解集，表 5 为其中一组来自算例 1 的解集，根据总成本由小到大排列。

在表 5 中，满意度会随着总成本的增加而增加，符合实际情况。表 5 中的 24 个解都属于帕累托非支配解。为了判断其优劣，参照引进的均衡适应度方法进行数据加工，数据如表 5 所示。可知，均衡接近度最大的解序号为 23。根据均衡度理论，该解最接近参考解，即本例中视为最优解。

2) 客户数量为 50 的算例

Table 6. Comparison of Pareto solution sets, grey relational degrees, equilibrium degrees, and equilibrium proximity for 50 customers

表 6. 50 客户数的帕累托解集、灰关联度、均衡度、均衡接近度比较

序号	总成本	满意度	灰关联度	均衡度	均衡接近度
1	72,188	80.00%	0.667	0.811	0.541
2	73,485	80.76%	0.617	0.852	0.525
3	75,693	80.98%	0.549	0.898	0.493
4	75,779	82.57%	0.557	0.912	0.508
5	76,703	83.92%	0.543	0.937	0.509
6	76,704	84.13%	0.545	0.938	0.511
7	77,593	84.55%	0.528	0.952	0.503
8	77,593	85.39%	0.535	0.958	0.513
9	78,249	88.00%	0.546	0.980	0.535
10	82,939	90.47%	10	0.505	1.000
11	85,345	90.93%	11	0.486	0.996
12	87,347	92.20%	12	0.488	0.983
13	87,763	93.35%	13	0.504	0.973
14	88,518	94.25%	14	0.515	0.961
15	89,863	96.48%	15	0.558	0.922
16	92,207	96.72%	16	0.550	0.900
17	93,522	96.77%	17	0.545	0.889
18	93,559	100.00%	18	0.667	0.811

同理，针对算例 3 和 4，共 30 次实验产生了 30 组帕累托解集，以下为其中一组来自算例 3 的解集，根据总成本由小到大排列，如表 6 所示。

由表 6 可知，50 客户算例中，均衡接近度最大的解序号为 1。根据均衡度理论，该解最接近参考解，即视为最优解。

4.4. 算法性能分析

为进一步评估多目标改进 WHO 的性能，与其他算法进行对比。仍使用 CLRP 标准算例进行分析，即设置成本权重为 1。对比算法为 GRASP [13]、MAPW [14]、LRGTS [15]、BKR 为已知最优解。

表 7 为 Barreto 数据集的 13 个 CLRP 标准算例，MOIWHO 列为对每个算例运行 20 次的最优结果。第一列是算例名称，名称后缀 50×5 代表该算例为 50 个客户、5 个设施的 CLRP 算例。第二列为已知最优解，为后四列中的最小值。三到五列为上述算法求得的结果。可以看出，针对这些算例，MOIWHO 可以求得 12 组已知最优解，唯一一组未求得已知最优解的算例 Daskin95-150x10 包含 150 个客户，为特大型算例。实验结果证明了改进策略的有效性，显著提升了算法的收敛精度、寻优能力等综合性能。

Table 7. Solving 13 Barreto datasets using 4 algorithms
表 7. 4 种算法求解 13 个 Barreto 数据集

算例	BKR	GRASP	MAPW	LRGTS	MOIWHO
Christofides69-50×5	565.6	599.1	565.6	586.4	565.6
Christofides69-75×10	861.6	861.6	866.1	863.5	863.5
Christofides69-100×10	842.9	861.6	850.1	842.9	842.9
Daskin95-88×8	355.8	356.9	355.8	368.7	355.8
Daskin95-150×10	44,011.7	44,625.2	44,011.7	44,386.3	44,876.7
Gaskell67-21×5	424.9	429.6	424.9	424.9	424.9
Gaskell67-22×5	585.1	585.1	611.8	587.4	585.1
Gaskell67-29×5	512.1	515.1	512.1	512.1	512.1
Gaskell67-32×5	571.7	571.9	571.9	584.6	571.7
Gaskell67-32×5	504.3	504.3	534.7	504.8	504.3
Gaskell67-36×5	460.4	460.4	485.4	476.5	460.4
Min92-27×5	3062	3062	3062	3065.2	3062
Min92-134×8	5809	5965.1	5950.1	5809	5809

5. 结论

本文在开放式同时选取货选址路径问题模型上加入了模糊时间窗以衡量客户满意度，建立了以满意度为第二目标的多目标选址路径问题模型。针对 WHO 鲁棒性差、容易陷入局部最优的问题，改进了交叉方式，并重新设计了混沌变异，新的变异方式有效改善了算法后期容易陷入局部最优的问题。另外，本文设计了一种新的解码方式，每条车辆路线清晰可见。最后，通过参数分析确定 MOIWHO 最佳参数，并与 CLRP 标准算例进行对比，证明了改进的 MOIWHO 求解中小规模选址路径问题具有良好性能。但是，本文仍存在许多不足，其中假设过多，而且未完全考虑到运输过程中的不确定性，如恶劣天气、交通拥堵等，后续研究将从不确定性入手，考虑整个模型系统的鲁棒性。

基金项目

国家自然科学基金(72101149); 教育部人文社会科学基金(21YJC630087)。

参考文献

- [1] Tran, T.H., Scaparra, M.P. and O'Hanley, J.R. (2017) A Hypergraph Multi-Exchange Heuristic for the Single-Source Capacitated Facility Location Problem. *European Journal of Operational Research*, **263**, 173-187. <https://doi.org/10.1016/j.ejor.2017.04.032>
- [2] Zhang, H., Zhang, Q., Ma, L., Zhang, Z. and Liu, Y. (2019) A Hybrid Ant Colony Optimization Algorithm for a Multi-Objective Vehicle Routing Problem with Flexible Time Windows. *Information Sciences*, **490**, 166-190. <https://doi.org/10.1016/j.ins.2019.03.070>
- [3] Cooper, L. (1972) The Transportation-Location Problem. *Operations Research*, **20**, 94-108. <https://doi.org/10.1287/opre.20.1.94>
- [4] Karaoglan, I., Altiparmak, F., Kara, I. and Dengiz, B. (2011) A Branch and Cut Algorithm for the Location-Routing Problem with Simultaneous Pickup and Delivery. *European Journal of Operational Research*, **211**, 318-332. <https://doi.org/10.1016/j.ejor.2011.01.003>
- [5] Ponboon, S., Qureshi, A.G. and Taniguchi, E. (2016) Branch-and-Price Algorithm for the Location-Routing Problem with Time Windows. *Transportation Research Part E: Logistics and Transportation Review*, **86**, 1-19. <https://doi.org/10.1016/j.tre.2015.12.003>
- [6] Farham, M.S., Süral, H. and Iyigun, C. (2018) A Column Generation Approach for the Location-Routing Problem with Time Windows. *Computers & Operations Research*, **90**, 249-263. <https://doi.org/10.1016/j.cor.2017.09.010>
- [7] Wang, X.F. (2013) An Location-Routing Problem with Simultaneous Pickup and Delivery in Urban-Rural Dual-Directions Logistics Network. *Advanced Materials Research*, **756**, 3423-3429. <https://doi.org/10.4028/www.scientific.net/amr.756-759.3423>
- [8] Wang, Y., Assogba, K., Liu, Y., Ma, X., Xu, M. and Wang, Y. (2018) Two-Echelon Location-Routing Optimization with Time Windows Based on Customer Clustering. *Expert Systems with Applications*, **104**, 244-260. <https://doi.org/10.1016/j.eswa.2018.03.018>
- [9] Marinakis, Y. (2015) An Improved Particle Swarm Optimization Algorithm for the Capacitated Location Routing Problem and for the Location Routing Problem with Stochastic Demands. *Applied Soft Computing*, **37**, 680-701. <https://doi.org/10.1016/j.asoc.2015.09.005>
- [10] Naruei, I. and Keynia, F. (2021) Wild Horse Optimizer: A New Meta-Heuristic Algorithm for Solving Engineering Optimization Problems. *Engineering with Computers*, **38**, 3025-3056. <https://doi.org/10.1007/s00366-021-01438-z>
- [11] 虎翼飞, 张惠珍, 陈曦. 改进野马算法求解低碳开放式送取货选址路径问题[J]. 包装工程, 2024, 45(1): 229-238.
- [12] 黄丽萍. 具有灰需求的同时取送货选址路径优化研究[D]: [硕士学位论文]. 福州: 福州大学, 2019.
- [13] Prins, C., Prodhon, C. and Calvo, R.W. (2006) Solving the Capacitated Location-Routing Problem by a GRASP Complemented by a Learning Process and a Path Relinking. *4OR*, **4**, 221-238. <https://doi.org/10.1007/s10288-006-0001-9>
- [14] Prins, C., Prodhon, C. and Calvo, R.W. (2006) A Memetic Algorithm with Population Management (MA|PM) for the Capacitated Location-Routing Problem. In: Gottlieb, J. and Raidl, G.R., Eds., *Proceedings of the European Conference on Evolutionary Computation in Combinatorial Optimization*, Springer, 183-194. https://doi.org/10.1007/11730095_16
- [15] Prins, C., Prodhon, C., Ruiz, A., Soriano, P. and Wolfier Calvo, R. (2007) Solving the Capacitated Location-Routing Problem by a Cooperative Lagrangean Relaxation-Granular Tabu Search Heuristic. *Transportation Science*, **41**, 470-483. <https://doi.org/10.1287/trsc.1060.0187>