

# 基于并行进化框架的多场景带容量的车辆路径优化算法研究

陈 磊, 潘安琪

东华大学信息科学与技术学院, 上海

收稿日期: 2025年3月11日; 录用日期: 2025年4月11日; 发布日期: 2025年4月18日

## 摘要

多场景优化问题涉及需要同时优化的多个场景, 每个场景都具有特定条件和多个待优化目标。其目标在于寻找一组公共折衷最优解(PCOS): 这些解在所有场景中均可行, 且能达成折中的优化效果。带容量约束的多场景车辆路径问题(MSCVRP)考虑了现实配送中的不确定扰动, 目前针对该问题研究较少。本文提出一种基于并行进化框架的多场景带容量车辆路径优化算法(PEF-MSOA)。通过利用各场景的knee点和约束极值线构建不同场景间解的转移策略, 实现不同场景间的有效信息传递。此外, 设计自适应算子选择策略以提升PCOS解集的质量。对比实验中验证了所提算法的有效性与优越性。

## 关键词

带容量车辆路径问题, 多场景优化问题, 并行进化框架, Knee点

# Multi-Scenario Capacitated Vehicle Routing Optimization Algorithm Based on Parallel Evolution Framework

Lei Chen, Anqi Pan

College of Information Science and Technology, Donghua University, Shanghai

Received: Mar. 11<sup>th</sup>, 2025; accepted: Apr. 11<sup>th</sup>, 2025; published: Apr. 18<sup>th</sup>, 2025

## Abstract

Multi-scenario optimization problems involve multiple scenarios that need to be optimized simultaneously, each with specific conditions and multiple objectives to optimize. The goal is to find a set of public compromised optimal solutions (PCOS): feasible in all scenarios and achieving a balanced

文章引用: 陈磊, 潘安琪. 基于并行进化框架的多场景带容量的车辆路径优化算法研究[J]. 计算机科学与应用, 2025, 15(4): 186-196. DOI: [10.12677/csa.2025.154091](https://doi.org/10.12677/csa.2025.154091)

**optimization effect. The multi-scenario capacitated vehicle routing problem (MSCVRP) considers uncertainty disturbances in real-world delivery, with few effective measures for this problem in existing research. In this paper, a multi-scenario capacitated vehicle routing optimization algorithm based on the parallel evolution framework is proposed. The knee solutions and extremal line of each scenario are used to construct a solution transfer strategy, thus achieving information transfer between different scenarios. Besides, an adaptive operator selection strategy is designed to improve the quality of PCOS. The effectiveness and superiority of the proposed algorithm are verified through comparative experiments.**

## Keywords

**Capacitated Vehicle Routing Problem, Multi-Scenario Optimization Problem, Parallel Evolution Framework, Knee Solution**

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

多目标优化问题(Multi-objective optimization problems, MOPs)是一类特殊的优化问题，旨在同时优化多个相互冲突的目标函数。在多个领域均有广泛应用。例如，在工程学、生物学和经济学领域[1]-[4]。如果一个多目标优化问题考虑多个不同的场景，则被定义为多场景优化问题(Multi-scenario optimization problem, MSOP)，这是近年来出现的一个概念。MSOP 与多任务优化问题(Multi-task optimization problem, MTOP)类似，都涉及同时优化多个任务。然而，区别在于 MTOP 为每个任务提供一组独特的解决方案，而 MSOP 旨在找到一组共同的解决方案，以满足不同场景的目标。带容量约束的车辆路径问题(CVRP) [5] 是一个经典的多目标组合优化问题，旨在通过规划最优车辆路径来最小化总行驶距离或成本，同时满足客户需求和车辆容量约束。多场景带容量车辆路径问题(MSCVRP)是 CVRP 的扩展，考虑了不同现实场景中的配送情况。尽管每个场景可能不会同时发生，但它们在长时间内具有不同的发生概率。因此，有必要找到一组在所有场景中都可行的折中解，以便多个场景尽可能实现最优函数值。例如，由于良好天气和恶劣天气之间的配送条件(例如，驾驶速度、交通拥堵、成本条件)的差异，良好天气和恶劣天气可以被定义为两个不同的配送场景。对于 MSOP，能够在所有场景中处理每个场景并实现折中最优性的解集被标记为公共折中最优解(Public compromise optimal solution, PCOS) [6] [7]。

解决 MSCVRP 的关键点可以总结如下：

- 1) 哪些解可以被定义为 PCOS。在所有场景中表现出良好收敛性和多样性的解可以被定义为 PCOS，在一个场景中表现良好但在另一个场景中表现不佳的解不被视为 PCOS。
- 2) 如何获得 PCOS。并行进化算法通常用于同时优化多个场景，每个场景涉及多个复杂问题，不同场景之间的知识迁移有助于找到 PCOS。
- 3) 如何确保 PCOS 的最优性。对于同一个场景，算子选择策略的合理性直接影响 PCOS 的质量。不同交叉算子的设计目标和适用场景各不相同，对解的搜索能力和局部优化能力有显著影响。

为了解决上述关键问题，本文提出了一种基于并行进化框架的多场景带容量车辆路径优化算法(Multi-scenario capacitated vehicle routing optimization algorithm based on parallel evolution framework, PEF-MSOA)，该算法用于并行优化多个场景。在 PEF-MSOA 的一次迭代中，设计了一种高效的知识转移策

略, 以在不同场景间选择出 PCOS。此外, 构建了一种自适应选择交叉算子策略, 以提高 PCOS 的质量。通过 PEF-MSOA, 实现了不同场景之间的信息交换, 提高了优化效果和效率, 能用于有效地解决 MSCVRP。本文的贡献可以总结如下:

- 1) 建立了多场景带容量车辆路径问题的数学模型, 其中良好天气和恶劣天气被定义为两个场景。分别为每个场景设计了成本和碳排放的多目标函数模型。
- 2) 构建了一种基于并行进化框架的多场景带容量车辆路径优化算法(PEF-MSOA)。设计了一种高效的知识转移策略, 以选择在所有场景中具有良好优化性能的 PCOS。
- 3) 使用三种交叉算子设计了一种基于距离的自适应选择交叉算子策略, 可以显著提高 PCOS 的质量。此外, 设计了一种存档更新过程, 以动态更新和存储 PCOS, 在控制存档大小的同时确保 PCOS 的最优性。

本文的其余部分组织结构如下。在第二章中, 我们简要讨论了相关概念与理论基础。第三章详细介绍了 MSCVRP 模型的构建。第四章详细介绍了提出的 PEF-MSOA。第五章列出了 PEF-MSOA 的实验结果。第六章对本文所做工作进行了总结。

## 2. 相关概念与理论基础

在本章中, 首先介绍了多场景优化问题(MSOP)的数学表达式。然后, 讨论了当前 MSOP 的研究方法。最后, 介绍了 knee 点的定义。

### 2.1. 多场景优化问题

多场景优化问题(MSOP)专注于寻找一组在不同场景下均具有优良性能的 PCOS, 广义上, MSOP 可定义为包含  $I$  个独立场景的  $M$  目标优化问题。与传统单场景优化不同, 其解需在全部  $I$  个场景中进行评估。每个场景均具有独特的参数、约束条件及对应的帕累托最优解集。MSOP 的最终目标是通过考虑各场景的需求, 找到在所有场景中表现良好的解。其核心挑战在于协调不同场景之间的差异, 并确保选出的解在每个场景中表现良好。这需要评估解在多个场景中的表现, 并基于这些评估进行选择和优化。不失一般性地, 对于含有  $I$  个场景、每个场景中包含  $M$  个目标函数和  $Y$  个不等式约束的多场景优化问题, 数学模型可描述如下:

$$\begin{aligned}
 \min : F^{(1)}(x) = & \left\{ f_1^{(1)}(x), \dots, f_m^{(1)}(x), \dots, f_M^{(1)}(x) \right\} \\
 & \dots \\
 \min : F^{(i)}(x) = & \left\{ f_1^{(i)}(x), \dots, f_m^{(i)}(x), \dots, f_M^{(i)}(x) \right\} \\
 & \dots \\
 \min : F^{(I)}(x) = & \left\{ f_1^{(I)}(x), \dots, f_m^{(I)}(x), \dots, f_M^{(I)}(x) \right\} \\
 s.t. & g_y^{(i)}(x) \leq 0, y = 1, 2, \dots, Y, i = 1, 2, \dots, I. \\
 x & \in \Omega
 \end{aligned} \tag{1}$$

其中,  $f_m^{(i)}(x)$  表示第  $i$  个场景的第  $m$  个目标函数,  $g_y^{(i)} \leq 0$  表示第  $i$  个场景的第  $y$  个不等式约束。

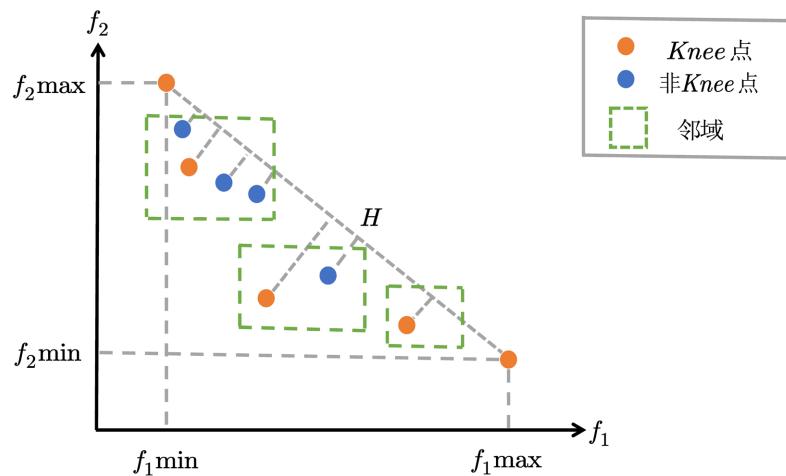
### 2.2. 现有 MSOP 研究方法

目前, 解决 MSOP 的主要方法分为两种: 第一种方法是基于聚合思想的算法, 常见的包括平均聚合方法[8]和最差情形聚合方法[9]。其核心思想是将各场景目标函数的平均值或最大值作为聚合函数, 将 MSOP 转化为单一场景下优化问题, 然后通过已有的多目标进化优化算法来进行求解。然而, 当场景间

差异较大时, 该方法可能导致局部最优解, 而非 MSOP 的全局最优解, 导致求解出来的最终解质量不高。第二种方法是场景导向法, 主要通过独立优化各场景的解集(定义为该场景的非支配解), 并采用预定的决策策略进行整合[10][11]。但该方法违背了多目标优化与多准则决策问题的基本原则, 且存在计算时间过长和复杂度高的问题。

### 2.3. Knee 点

在种群中表现出良好优化性能(收敛性与多样性)的解通常被定义为 knee 点。在每一代的进化过程中, 选择 knee 点进入下一代种群, 相较于非 knee 点, 能更有效地提升算法的收敛性。以双目标最小化问题为例, 如图 1 所示, 极值线  $H$  由两个目标的极值解计算得出, 其中一个解在第一个目标函数  $f_1$  上取得最大值, 另一个解在第二个目标函数  $f_2$  上取得最大值。对于每个解, 计算其到极值线  $H$  的距离, 若某解在其邻域内到  $H$  的距离最大, 则被识别为 knee 点。本文采用 KNEA [12] 选出各场景的 knee 点。



**Figure 1.** An illustration for determining knee solutions in KnEA for a bi-objective optimization problem  
**图 1.** 用于确定双目标优化问题中 knee 点的 KnEA 算法示意图

### 3. MSCVRP 建模

车辆配送条件因环境不同而有所差异。例如, 由于良好天气与恶劣天气在交通拥堵和成本条件上的差异, 距离权重因子和成本权重因子也会有所不同。本文将良好天气和恶劣天气定义为 MSCVRP 下的两个独立场景, 其中良好天气为场景 1, 恶劣天气为场景 2。每个场景包含两个目标函数: 成本  $f_1^{(I)}$  和碳排放  $f_2^{(I)}$ 。两个场景同时优化, 构成了多场景带容量车辆路径优化问题。MSCVRP 模型构建如下:

$$TD = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} W_{ij} d_{ij} x_{ijk} \quad (2)$$

$$f_{fuel} = TD.c \quad (3)$$

$$f_{labor} = \sum_{k \in K} L_k u_k \quad (4)$$

$$T_{pjk} = q_{ijk} + v_{ijk1} + v_{ijk2} \quad (5)$$

$$f_{penalty} = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} P_e T_{pjk} x_{ijk} \quad (6)$$

$$C_{ijk} = p_0 + (p - p_0) r_{ijk} \quad (7)$$

$$r_{ijk} = \frac{Q_{ijk}}{Q} \quad (8)$$

$$f_{carbon} = \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} C_{ijk} x_{ijk} W_{fij} d_{ij} \quad (9)$$

$$f_1^{(I)} = f_{fuel} + f_{labor} + f_{penalty} \quad (10)$$

$$f_2^{(I)} = f_{carbon} \quad (11)$$

$$\sum_{k \in K} \sum_{j \in V} x_{ijk} = 1, \forall i \in C \quad (12)$$

$$\sum_{j \in V} v_{0jk} = \sum_{i \in V} v_{i0k} = 1, \forall k \in K \quad (13)$$

$$\sum_{i \in V} x_{ihk} - \sum_{j \in V} x_{hjk} = 0, \forall h \in C, k \in K \quad (14)$$

$$\sum_{i \in V, i \neq j} x_{ijk} - \sum_{i \in V, i \neq j} x_{jik} = 0, \forall j \in V, k \in K \quad (15)$$

$$x_{ijk} \in \{0, 1\}, \forall (i, j) \in A, k \in K \quad (16)$$

$$\sum_{i \in C} \sum_{j \in V} q_i x_{ijk} \leq Q, \forall k \in K \quad (17)$$

其中, 等式(2)表示车辆在第 I 个场景下车辆行驶总距离。等式(3)表示在第 I 个场景下车辆的燃油消耗成本。等式(4)表示第 I 个场景下配送的劳动成本。等式(5)表示路径中容量约束违反的总和。其中,  $v_{ijk1}$  表示从节点  $i$  到节点  $j$  是否违反载重约束,  $v_{ijk1}$  表示若  $v_{ijk1} = 1$ , 则从  $i$  到  $j$  的容量约束违反总和。等式(6)表示不同场景下容量约束违反惩罚成本。等式(7)表示第  $k$  辆车的单位距离碳排放量。等式(8)表示车辆载重的变化率系数。等式(9)表示两个场景中的总碳排放量, 准确反映了碳排放与载重和行驶距离的正比关系。等式(10)表示第 I 个场景下的第一个目标函数总成本。等式(11)表示第 I 个场景下的第二个目标函数总碳排放量。等式(12)~等式(17)为车辆路径问题的标准约束, 本文不再赘述。

通过上述对目标函数与约束模型的构建, 准确描述了在良好天气与恶劣天气下的基于成本与碳排放目标函数的多场景带容量车辆路径问题。

#### 4. 基于并行进化框架的多场景带容量车辆路径优化算法

图 2 为 PEF-MSOA 算法的总体流程框架图, 由于本文建立的 MSCVRP 中每个场景包含两个优化目标函数, 故以下描述基于求解包含 I 个场景, 每个场景中包含两个目标函数的最小化多场景优化问题。在初始化阶段, 对于每个情景  $S_i$ ,  $i \in \{1, 2, \dots, I\}$ , 随机生成并评估一个包含 N 个解的种群  $P_i$ , 并评估种群  $P_i$  中的每一个解在当前场景中的目标函数值。将  $P_i$  分成三个子种群  $P_{i-j}$ ,  $j \in \{1, 2, 3\}$ , 每个子种群使用不同的交叉算子。在进化迭代过程的每一轮迭代中, 首先, 使用 KnEA 算法找到子种群  $P_{i-j}$  的 knee 点  $K_{i-j}$  与约束极值线  $H_{i-j}$ , 每个场景的 knee 点与极值线由各子种群合并构成。其次, 在基于 knee 点的跨场景知识迁移策略步骤中, 评估每个场景的 knee 点是否可以被其他情景作为可接受解。迁移策略的评估基于每个情景中构造的约束极值线, 如果来自其他情景的 knee 点位于约束极值线以下, 则该 knee 点被认为是相应情景的可接受解  $T_i$ 。被所有情景接受的 knee 点定义为公共折中最优解(PCOS)。随后, 在基于距离的自适应选择交叉算子策略步骤中, 在每一轮的迭代中动态更新算子优化种群比例矩阵  $M_i$ , 以确定下一代进化过程中分配给每个算子的子种群数量。同时, 使用环境选择动态更新档案 Ar 中的 PCOS, 以确保它们的最优性, 同时控制存档数量为 N。最后, 对于每个情景, 其可接受解  $T_i$  参与其种群  $P_i$  的后代

生成过程, 以促进不同情景之间的有效信息传递, 帮助其他场景快速找到决策空间中最优区域, 加速求解过程。当达到迭代终止条件时, 此时档案中的解即为最终的公共折中最优解。图 2 为 PEF-MSOA 算法的总体流程框架图, 详细的算法策略步骤流程描述如后文所述。

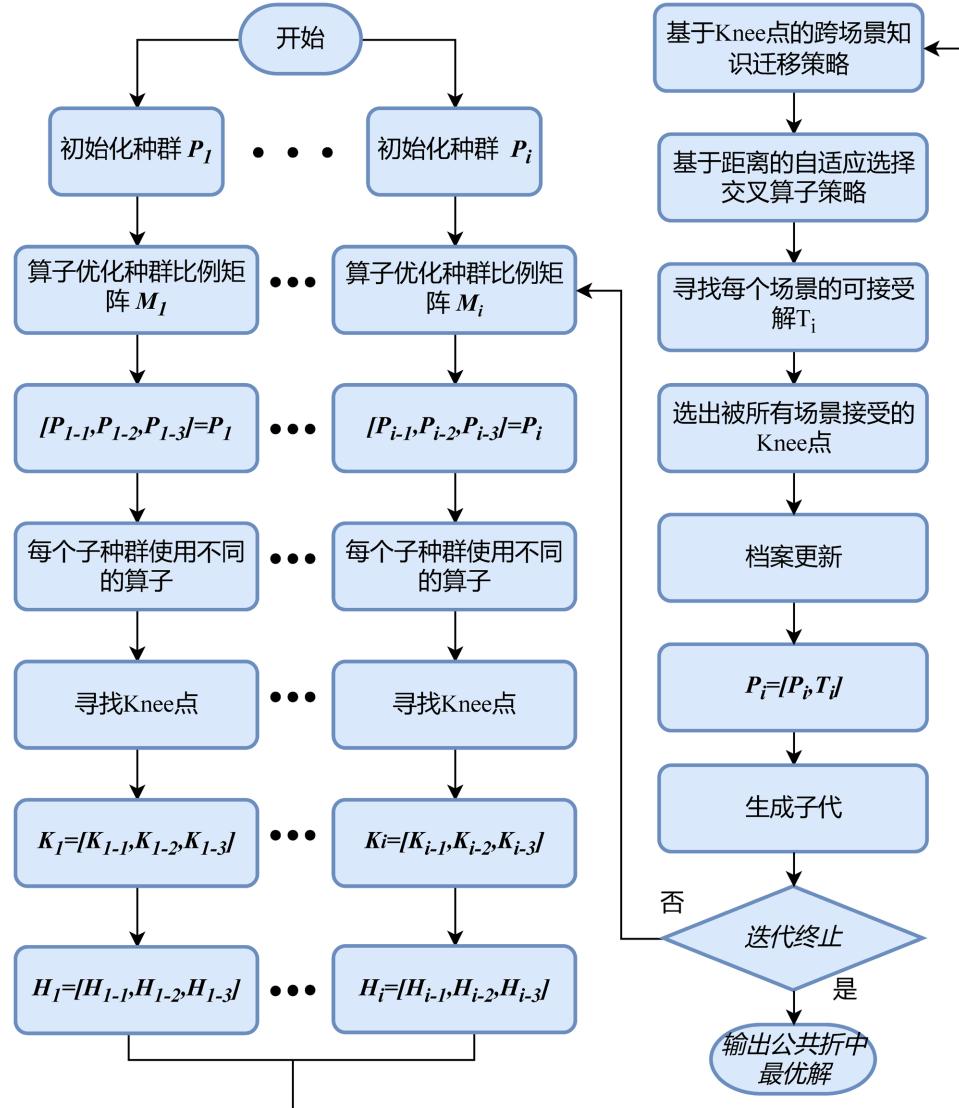


Figure 2. The flowchart of multi-scenario optimization algorithm based on parallel evolution framework  
 图 2. 基于并行进化框架的多场景优化算法流程图

#### 4.1. 基于 Knee 点的跨场景知识迁移策略

PCOS 应在所有场景中表现出良好的收敛性与多样性, 并整合来自各场景的有价值信息。因此, 设计了一种基于 knee 点的跨场景迁移策略。首先, 对于场景  $S_i$ , 其约束极值线  $H_i$  可被定义为  $a_i x + b_i y + c_i = 0$ , 用于判断其他场景的 knee 点是否可以作为该场景的可接受解。参数  $a$ 、 $b$ 、 $c$  的值由每个场景中 knee 点中的极值解确定。以双目标最小化问题为例, 一个极值解在目标一  $f_1^{(l)}$  上取得最大值, 另一个极值解在目标二  $f_2^{(l)}$  上取得最大值。设解  $A(x_A, y_A)$  为某一个场景的 knee 点集合中的一个 knee 点, 其到  $H_i$  的距离可通过公式(18)和(19)进行计算。若  $d(A, H_i) > 0$ , 表示解  $A$  位于  $H_i$  的下方, 则  $A$  被定义为场景  $S_i$  的可接

受解  $T_i$ ; 否则, 解不能被定义为场景  $S_i$  的可接受解。基于 knee 点的跨场景迁移策略实现了 knee 点的跨场景迁移与每个场景可接受解的识别。被所有场景接受的 knee 点被定义为 PCOS。图 3 展示了两场景间的迁移策略示意图。尽管迁移策略能够得到所有的 PCOS, 但其最优性仍需提升, 需设计措施以进一步提高 PCOS 的质量。

$$d(A, H_i) = \frac{|a_i x_A + b_i y_A + c_i|}{\sqrt{a_i^2 + b_i^2}}, a_i x_A + b_i y_A + c_i < 0 \quad (18)$$

$$d(A, H_i) = -\frac{|a_i x_A + b_i y_A + c_i|}{\sqrt{a_i^2 + b_i^2}}, a_i x_A + b_i y_A + c_i \geq 0 \quad (19)$$

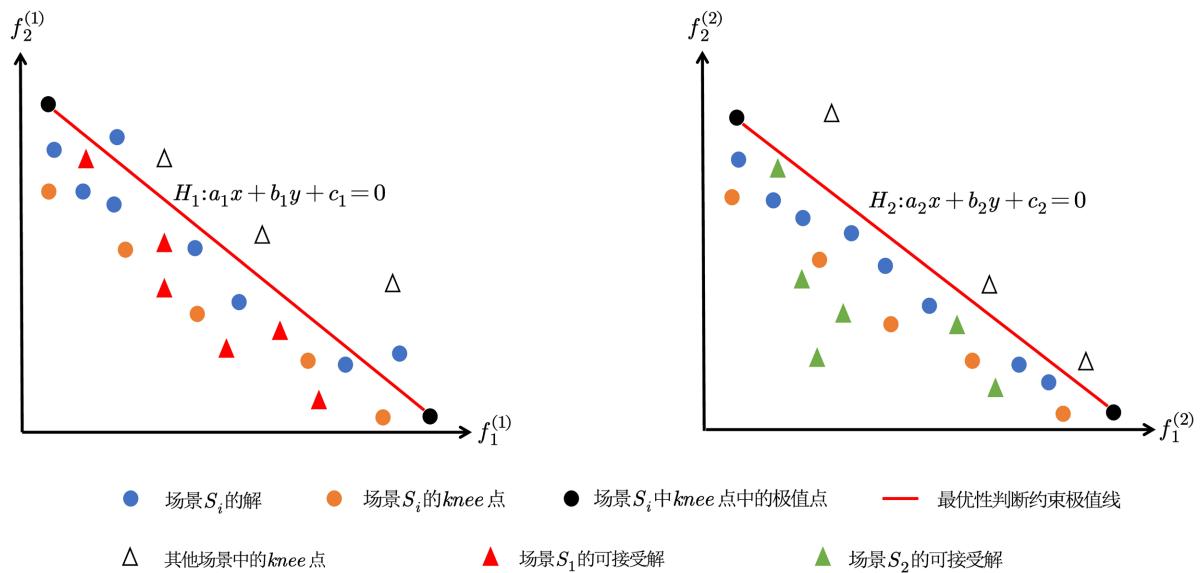


Figure 3. Schematic of the transfer strategy of knee points between two scenarios

图 3. 两场景间 knee 点的迁移策略示意图

#### 4.2. 基于距离的自适应选择交叉算子策略

为了提高 PCOS 的质量, 设计了一种基于距离的自适应算子选择策略。对于场景  $S_i$ ,  $i \in \{1, 2, \dots, I\}$ , 其种群  $P_i$  被划分为三个子种群  $P_{i-j}$ ,  $j \in \{1, 2, 3\}$ 。初始化不同交叉算子优化子种群数量矩阵  $M_i$ ,  $M_i$  决定每种交叉算子优化子种群数量。在这些子种群的进化过程中, 顺序交叉算子(OX)、基于距离的交叉算子(CM1)和基于断裂的交叉算子(CM2)被应用于进化这些子种群,  $P_{i-1}$  使用 OX 作为交叉算子,  $P_{i-2}$  使用 CM1 作为交叉算子,  $P_{i-3}$  使用 CM2 作为交叉算子。

OX 从一个父代中选择一个基因片段, 保留其顺序, 并用另一个父代的基因按相对顺序填充剩余位置 [13]。CM1 在生成子代时考虑了父代解的距离信息和容量约束。它结合最优路径段以保留较短路径的特征, 从而生成更好的子代解, 其流程图如图 4 所示。CM2 是用于解决车辆路径问题的交叉算子, 它随机选择两条路径, 并在随机断点处将每条路径拆分为头部(从配送中心开始)和尾部(返回配送中心), 其示意图如图 5 所示。基于 knee 点的跨场景迁移策略也应用于不同场景中具有相同索引的子种群之间, 例如  $P_{i-1}$  和  $P_{j-1}$ ,  $i \neq j$ 。 $S_i$  的第  $j$  个子种群获得的可接受解和 knee 点分别定义为  $T_{i-j}$  和  $K_{i-j}$ 。 $T_{i-j}$  和  $K_{i-j}$  到  $H_i$  的距离之和定义为  $D_{i-j}$ , 用于更新交叉算子优化子种群数量矩阵  $M_i$ 。根据之前对 knee 点和约束极值线的定义, 解离极值线越远, 其收敛性越好。因此,  $D_{i-j}$  越大, 相应算子的优化效果越好。 $D_{i-j}$  也可以通

过公式(18)和(19)进行计算。自适应算子选择策略决定了下一代进化过程中分配给每个算子的子种群数量, 引导搜索方向以提高 PCOS 的质量。每个场景的可接受解  $T_i$  定义为来自相应子种群的  $T_{i-j}$  集合, 定义如下:

$$T_i = [T_{i-1}, T_{i-2}, T_{i-3}] \quad (20)$$

场景  $S_i$  的  $D_{i-j}$  之和定义如下:

$$W_i = D_{i-1} + D_{i-2} + D_{i-3} \quad (21)$$

交叉算子优化子种群数量矩阵  $M_i$  的定义如下:

$$M_i = \left[ \frac{D_{i-1}}{W_i} \frac{D_{i-2}}{W_i} \frac{D_{i-3}}{W_i} \right] \quad (22)$$

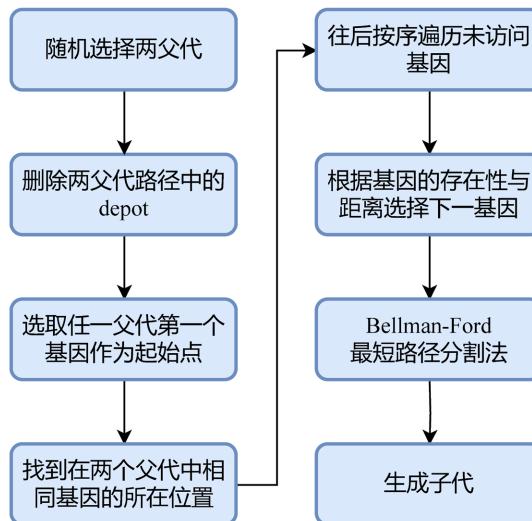


Figure 4. The flowchart of CM1 operator

图 4. CM1 交叉算子流程图

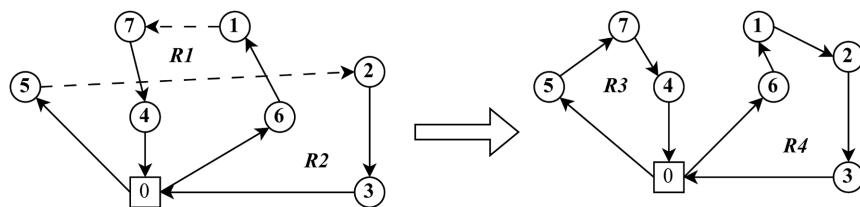


Figure 5. The flowchart of CM2 operator

图 5. CM2 交叉算子流程图

## 5. 实验部分

在本节中, 通过实验分别验证了 PEF-MSOA 中基于 knee 点的跨场景迁移策略和自适应选择交叉算子策略的有效性。使用 HV [14] 和 Spacing [15] 指标评估所获得解集的质量, 并分析其在两个场景目标空间中的表现。

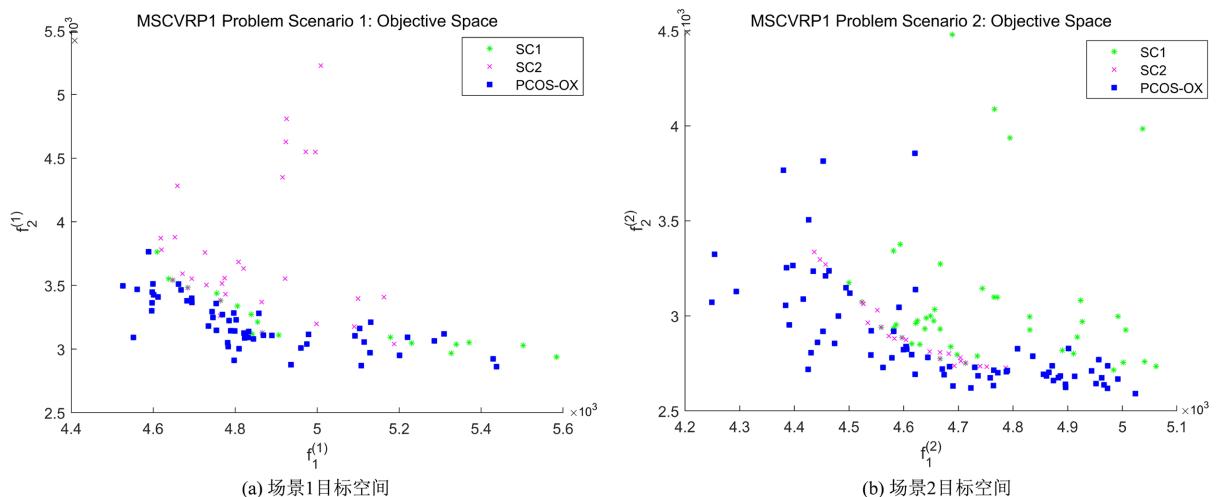
### 5.1. 测试问题

为了验证基于 knee 点的跨场景迁移策略的性能, 设计了两种对比算法。使用场景间 knee 点进行迁

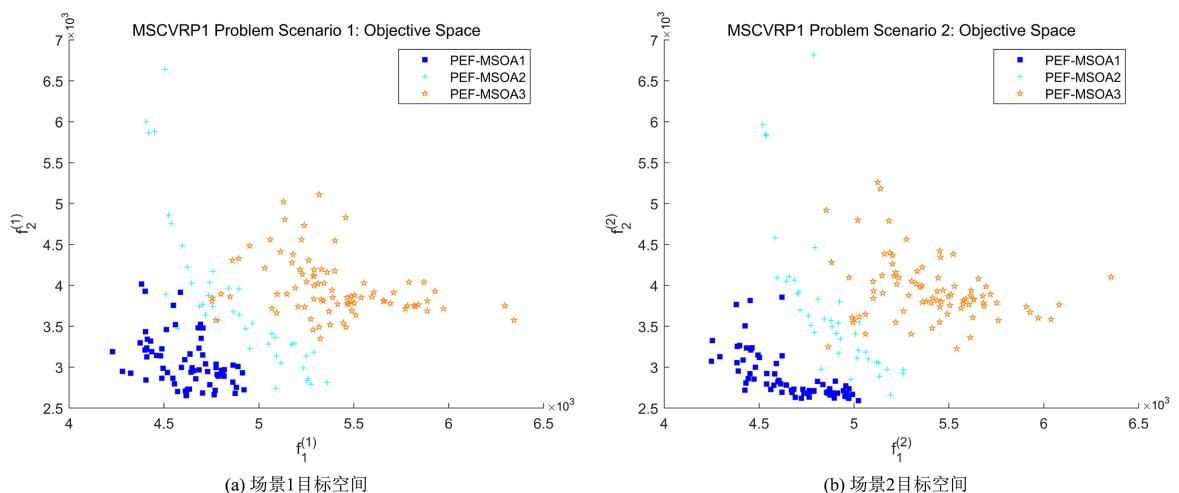
移且仅采用 OX 作为交叉算子的算法定义为 PEF-MSOA1, 其得到的 PCOS 命名为 PCOS1。在场景间迁移所有解的算法定义为 PEF-MSOA2, 其得到的 PCOS 命名为 PCOS2。在场景间无解迁移的算法称为 PEF-MSOA3, 其得到的 PCOS 命名为 PCOS3。所有算法均使用 MSCVRP 作为测试问题, 本实验使用上海真物流数据, 所有场景均只有 1 个配送中心, 其位置坐标为(40, 50)。在本文中, MSCVRP 设置为两场景。在场景 1 中, 客户数量为 100, 车辆数为 24, 车辆载重为 100; 场景 2 中, 客户数量为 59, 车辆数为 15, 车辆载重为 80; 违反载货量约束惩罚系数均设为 10。

## 5.2. 实验参数设置

种群规模  $N$  和存档  $Ar$  的大小均设置为 100, 交叉概率  $P_c = 0.9$ , 变异概率  $P_m = 1/D$ , 其中  $D$  为决策空间的维度。最大迭代次数设置为 5000, 对应 50 代。在 KNEA 算法中, 帕累托前沿  $F_i$  中 knee 点的比例阈值  $T = 0.5$ , 邻域宽度  $r$  以及帕累托前沿  $F_i$  中 knee 点的比例  $t$  随进化过程实时更新。实验在 PLATEMO [16]平台上进行。



**Figure 6.** Optimized solutions of PEF-MSOA1 in objective space  
**图 6.** PEF-MSOA1 在两场景目标空间中的优化解图



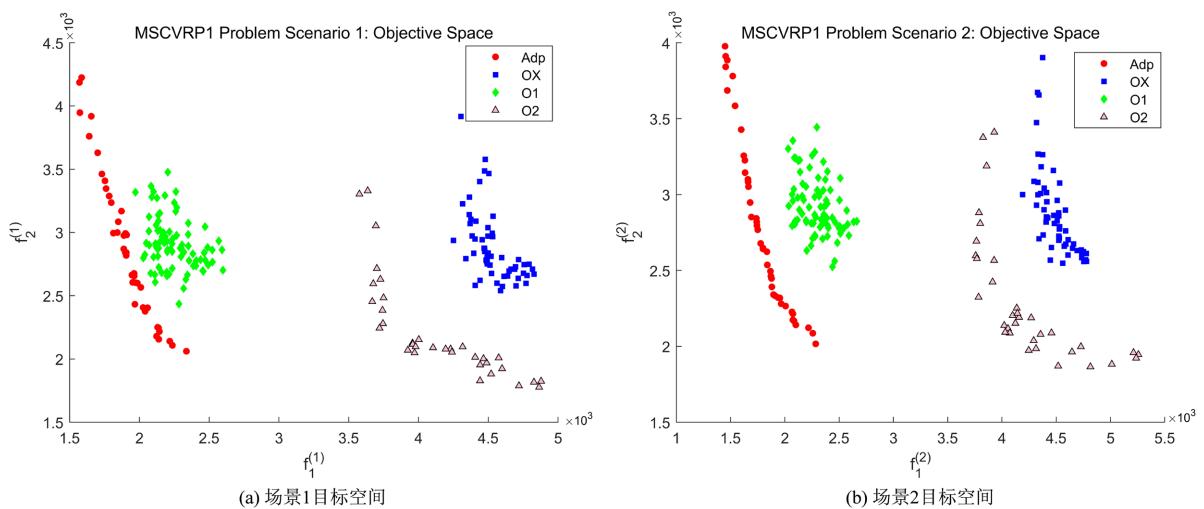
**Figure 7.** Optimized solutions for different transfer strategy in objective space  
**图 7.** 不同迁移策略在两场景目标空间中的优化解图

MSCVRP 涉及两个场景。由于计算 HV 值时需要设置参考点, 且不同场景的参考点不同, 因此需要分别为每个场景设置参考点并计算 HV 值。最终, 取两个场景 HV 值的平均值作为最终结果。对于 Spacing 指标, 无需设置参考点, 因此无需分别在两个场景中运行。每个算法在 MSCVRP 上独立运行 20 次。

**Table 1.** Comparison results of HV and Spacing value with three approaches under MSCVRP in two scenarios

**表 1.** 两种场景下三种算法的 HV 与 Spacing 值对比结果

算法	PEF-MSOA1			PEF-MSOA2			PEF-MSOA3		
	SC1	SC2	AVG	SC1	SC2	AVG	SC1	SC2	AVG
HV	8.29e - 2	8.22e - 2	<b>8.26e - 2</b>	6.51e - 2	6.76e - 2	6.64e - 2	4.64e - 2	4.89e - 2	4.77e - 2
Spacing	—	—	<b>5.77e + 1</b>	—	—	8.37e + 1	—	—	1.03e + 2



**Figure 8.** Optimized solutions for different operators in objective space  
**图 8.** 不同交叉算子在两场景目标空间中的优化解图

### 5.3. 实验结果分析

图 6 展示了 PEF-MSOA1 在两场景目标空间中的优化解, 其中 SC1 和 SC2 分别表示场景 S1 和 S2 的解集, PCOS1 在两个场景中的收敛性和多样性均优于 SC1 和 SC2。表 1 中展示了三种算法获得的 PCOS 的 HV 和 Spacing 值, 图 7 展示了不同迁移策略在两场景中获得的 PCOS 的目标空间图。从上述结果可以得出结论, PEF-MSOA1 获得的 PCOS 质量最佳, 而 PEF-MSOA3 获得的 PCOS 质量最差。这表明场景间的信息交换可以提高 PCOS 的质量, 特别是使用 knee 点进行跨场景迁移时效果最好, 验证了本文中所提出的基于 knee 点的跨场景迁移策略的有效性。

为验证本文中所提出自适应选择交叉算子策略的有效性, 分别使用 OX、CM1 和 CM2 作为交叉算子进行实验, 并对比其组合使用效果。图 8 展示了使用不同交叉算子获得的 PCOS 在两场景中的目标空间图。由图可知, CM1 在目标  $f_1^{(1)}$  上表现更优, 而 CM2 在目标  $f_2^{(1)}$  上表现更优, 且两者均优于 OX 算子。其中使用基于距离的自适应选择交叉算子策略获得的 PCOS 几乎严格支配了使用单一交叉算子获得的 PCOS, 表明该策略能显著提升了 PCOS 的质量。

## 6. 总结

本文构建了一个多场景带容量车辆路径问题, 即 MSCVRP。为了有效解决 MSCVRP, 提出了一种基于

并行进化框架的多场景带容量车辆路径优化算法。在 PEF-MSOA 的每一次迭代中, 首先利用 KnEA 算法选出每个场景中具有良好收敛性和多样性的 knee 点。其次, 设计了基于 knee 点的跨场景迁移策略, 评估每个场景的 knee 点是否可以被其他场景定义为可接受解。被所有场景接受的 knee 点被定义为公共折中最优解(PCOS)。然后, 为了提高 PCOS 的质量, 设计了一种基于距离的自适应选择交叉算子策略, 用于更新交叉算子优化子种群数量矩阵, 以此确定下一代进化过程中分配给每个算子的子种群数量。同时, 对于每个场景, 将其可行解与子代种群结合, 促进不同场景之间的有效信息传递。最后, 使用环境选择确保 PCOS 的最优性并控制其规模。通过实验验证了本文所提出算法的优越性和新颖性, 能够有效地解决 MSCVRP。

## 参考文献

- [1] Herrero, J.G., Berlanga, A. and Lopez, J.M.M. (2009) Effective Evolutionary Algorithms for Many-Specifications Attainment: Application to Air Traffic Control Tracking Filters. *IEEE Transactions on Evolutionary Computation*, **13**, 151-168. <https://doi.org/10.1109/tevc.2008.920677>
- [2] Ishibuchi, H. and Murata, T. (1998) A Multi-Objective Genetic Local Search Algorithm and Its Application to Flowshop Scheduling. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, **28**, 392-403. <https://doi.org/10.1109/5326.704576>
- [3] Yeung, S.H., Man, K.F., Luk, K.M. and Chan, C.H. (2008) A Trapeziform U-Slot Folded Patch Feed Antenna Design Optimized with Jumping Genes Evolutionary Algorithm. *IEEE Transactions on Antennas and Propagation*, **56**, 571-577. <https://doi.org/10.1109/tap.2007.915473>
- [4] Handl, J., Kell, D.B. and Knowles, J. (2007) Multiobjective Optimization in Bioinformatics and Computational Biology. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **4**, 279-292. <https://doi.org/10.1109/tcbb.2007.070203>
- [5] Dantzig, G.B. and Ramser, J.H. (1959) The Truck Dispatching Problem. *Management Science*, **6**, 80-91. <https://doi.org/10.1287/mnsc.6.1.80>
- [6] Fadel, G., et al. (2005) Multi-Criteria Multi-Scenario Approaches in the Design of Vehicles. *6th World Congresses of Structural and Multidisciplinary Optimization*, Rio de Janeiro, 30 May-3 June 2005.
- [7] Deb, K., Zhu, L. and Kulkarni, S. (2018) Handling Multiple Scenarios in Evolutionary Multiobjective Numerical Optimization. *IEEE Transactions on Evolutionary Computation*, **22**, 920-933. <https://doi.org/10.1109/tevc.2017.2776921>
- [8] Zhu, L., Deb, K. and Kulkarni, S. (2014) Multi-Scenario Optimization Using Multi-Criterion Methods: A Case Study on Byzantine Agreement Problem. *2014 IEEE Congress on Evolutionary Computation (CEC)*, Beijing, 6-11 July 2014, 2601-2608. <https://doi.org/10.1109/cec.2014.6900637>
- [9] Deb, K., Zhu, L. and Kulkarni, S. (2015) Multi-Scenario, Multi-Objective Optimization Using Evolutionary Algorithms: Initial Results. *2015 IEEE Congress on Evolutionary Computation (CEC)*, Sendai, 25-28 May 2015, 1877-1884. <https://doi.org/10.1109/cec.2015.7257115>
- [10] Wiecek, M.M., Singh, V. and Blouin, V. (2007) Multi-Scenario Multi-Criteria Optimization in Engineering Design. Defense Technical Information Center.
- [11] Wiecek, M.M., Blouin, V.Y., Fadel, G.M., Engau, A., Hunt, B.J. and Singh, V. (2009) Multi-Scenario Multi-Objective Optimization with Applications in Engineering Design. In: Barichard, V., et al., Eds., *Multiobjective Programming and Goal Programming: Theoretical Results and Practical Applications*, Springer, 283-298. [https://doi.org/10.1007/978-3-540-85646-7\\_26](https://doi.org/10.1007/978-3-540-85646-7_26)
- [12] Zhang, X., Tian, Y. and Jin, Y. (2015) A Knee Point-Driven Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, **19**, 761-776. <https://doi.org/10.1109/tevc.2014.2378512>
- [13] Amri Sakhri, M.S. (2021) Comparative Analysis of Different Crossover Structures for Solving a Periodic Inventory Routing Problem. *International Journal of Data Science and Analytics*, **14**, 141-153. <https://doi.org/10.1007/s41060-021-00280-2>
- [14] Zitzler, E. and Thiele, L. (1999) Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, **3**, 257-271. <https://doi.org/10.1109/4235.797969>
- [15] Schott, J.R. (1995) Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.
- [16] Tian, Y., Cheng, R., Zhang, X. and Jin, Y. (2017) Platemo: A MATLAB Platform for Evolutionary Multi-Objective Optimization [Educational Forum]. *IEEE Computational Intelligence Magazine*, **12**, 73-87. <https://doi.org/10.1109/mci.2017.2742868>