

一种基于AGVs的智能仓储批量分拣模型及方法

胡炜晟, 李淑月

南京邮电大学物联网学院, 江苏 南京

收稿日期: 2025年3月5日; 录用日期: 2025年4月14日; 发布日期: 2025年4月22日

摘要

本文研究了一种基于AGVs的智能仓储批量分拣模型及两阶段改进批量分拣优化方法。分析了受能量约束的AGV行驶路径问题, 考虑以最小化AGV行驶距离为优化目标, 构建多AGVs批量分拣数学模型。在迭代改进分拣优化的第一阶段利用待分拣订单位置, 订单特征生成高质量初始解。第二阶段考虑使用智能优化算法, 分别利用模拟退火算法和遗传算法对第一阶段生成的优质初始解进一步改进, 并对所适用的智能仓储规模进行分析与验证。本文构建的分拣模型充分考虑了货架布局的多样性、订单构成的通用性、以及AGV行驶路径的规范性, 更加适用于实际智能仓储分拣需求。提出的方法对于具有不同规模、不同分布的货架及订单都能够高效地进行分拣, 在提升AGV利用率的同时降低智能仓储总能耗, 为实际智能仓储应用提供可适配的柔性支撑。

关键词

智能引导小车, 智能仓储, 模拟退火算法, 遗传算法

A Batch Sorting Model and Method for Intelligent Warehousing Based on AGVs

Weisheng Hu, Shuyue Li

School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing Jiangsu

Received: Mar. 5th, 2025; accepted: Apr. 14th, 2025; published: Apr. 22nd, 2025

Abstract

This study investigates an intelligent warehouse batch picking model based on AGVs, along with a two-stage enhanced batch picking optimization method. The AGV travel path problem under energy constraints is analyzed. With the objective of minimizing AGV travel distance, a mathematical model

for batch picking with multiple AGVs is constructed. In the first stage of iterative sorting optimization, high-quality initial solutions are generated using the positions and characteristics of orders. In the second stage, intelligent optimization algorithms, simulated annealing algorithm and genetic algorithm, are applied to further improve the high-quality initial solutions obtained in the first stage. Analysis and verification of applicable intelligent warehousing scales are also conducted. The sorting model developed in this study takes into account the diversity of shelf layouts, the generality of order compositions, and the standardization of AGV travel paths, making it more suitable for practical intelligent warehousing sorting requirements. The proposed method efficiently handles sorting for shelves and orders with varying scales and distributions. It enhances AGV utilization while reducing the overall energy consumption of intelligent warehousing, thus providing adaptable and flexible support for practical intelligent warehousing applications.

Keywords

Intelligent Guided Vehicles, Intelligent Warehousing, Simulated Annealing Algorithm, Genetic Algorithm

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

全球经济一体化和新兴信息技术的快速发展,使仓储与物流在构建准则、规划方案和实施技术方面面临重大挑战。传统物流存在人力成本高、效率低、运营风险大等问题,难以满足日益增长的需求。智能化仓储与物流依托物联网和自动化机器人技术,能够减少人工操作,提高货物存取速度,降低错误率,从而节省时间和成本,提升运输效率和安全性。因此,向智能化转型已成为行业发展的必然趋势。

为了满足这一需求,自动导引小车(Automated Guided Vehicle, AGV)应运而生,成为了仓储与物流向智能化转型的重要一环。AGV 小车作为一种集成了多种先进技术的柔性智能物流装备,其自动化、高柔性、高效率、高可靠性以及并行作业等特点能很好地满足各种物流与制造需求[1],并在物流行业与智能制造行业中逐渐发挥着重要作用。

对 AGV 小车的研究也存在许多痛点。其中,调度和路径规划是一个相对困难的问题。在复杂的仓库环境中,存在多个 AGV 小车同时执行任务的情况,如何合理分配任务和规划路径是一个复杂的优化问题。需要考虑到不同 AGV 小车的位置、目标、速度等因素,并避免碰撞和冲突。这对于实现高效的调度和路径规划算法提出了挑战。

在智能仓储中,订单分配是一个关键环节。AGV 系统可以根据订单信息,智能地将订单分配给合适的 AGV 小车进行处理。这样的智能分配可以确保订单的及时处理,最大程度地提高仓储效率和响应速度。良好的 AGV 调度与路径规划可以有效地提升 AGV 在智能仓储中批量分拣的效率。针对现有技术的不足,本文以最小化 AGV 行驶路径为优化目标构建符合智能仓储实际运营特点的批量分拣数学模型,通过分析订单特征及订单间关联关系,基于订单内聚度和离散度分别从多个角度设计 AGV 订单智能分拣优化方法,通过减少 AGV 行驶距离提高货物传送效率,降低仓储成本的同时提升了资源利用率。

2. 相关研究分析

AGV 的调度问题可以分为单目标优化问题 and 多目标优化问题。许多学者针对以上提出的各种 AGV

调度问题进行了研究, 针对 AGV 的静态调度问题, Boccia 等[2]提出了一种基于瓶颈广义分配问题的原始混合整数线性规划模型; Dang 等[3]提出了一种结合局部搜索的混合自适应大型邻域搜索算法; Goli 等[4]提出了包含启发式算法的混合遗传算法以及一种鲸鱼优化算法优化总成本; Riazi 等[5]提出了一种基于逻辑的启发式 Benders 分解方法和单步分解启发式算法来解决 AGV 的调度问题; Zou 等[6]提出了一种多目标贪婪遗传算法; Zou [7]提出了一种新的解表示方法和一种加速解的评价的方法; Singh 等[8]提出了一种新的数学方法, 其利用自适应的大型邻域搜索算法和线性规划来求解问题; 目前, 大多数任务调度问题通常假设为静态情况, 即根据当前已知的信息进行任务分配和调度。然而, 在自动仓库系统运行时, 情况可能会随着时间的推移而发生变化, 为了应对系统的突发变化, AGV 动态调度也是急需研究的一个重要问题[9], 针对 AGV 的动态调度问题, Maoudj 等[10]提出了一种分散的多智能体方法优化 AGV 行驶的最大距离; Hu 等[11]结合行为树和强化学习提出了一种自适应交通控制模型, 以提高 AGV 的决策能力从而执行更好的调度; Lin 等[12]提出了基于特征的状态-动作-奖励-状态-动作算法; Lin 等[13]提出了一种基于 NSGA-II 和优劣解距离法改进的任务调度优化方法解决多目标的 AGV 调度优化问题, 并在原有 NSGA-II 算法的基础上进行了改进; Zheng 等[14]针对利用水运 AGV 实现完全自主的码头间运输问题, 提出了一种基于禁忌搜索的高效启发式求解方法; Lian 等[15]考虑时间成本, 提出了时间约束不均匀的虚拟网络图, 基于路径网络模型的特点, 实现了一种改进的 A*路径规划算法; Li 等[16]针对动态调度问题, 提出了一种参数自适应和计算时间自适应的离散入侵杂草优化算法; Hu 等[17]将 AGV 实时调度问题表述为马尔可夫决策过程, 并进一步开发了一种新的深度 Q 网络方法来实现最优混合规则策略。针对 AGV 的联合调度问题, 邹裕吉等[18]提出了一种基于时间窗和 Dijkstra 算法的多目标自适应聚类遗传算法; 范厚明等[19]以岸桥能耗作为第一阶段的优化目标, AGV 运输过程能耗作为第二阶段的优化目标, 建立了一个两阶段优化模型, 并设计枚举法求解第一阶段模型, 使用改进遗传算法求解第二阶段优化模型; Xin 等[20]提出了一种定制的高效遗传算法来解决所研究的以运输集装箱的完成时间和动能消耗作为双优化目标的混合整数非线性规划问题。

以上文献仅仅针对于 AGV 的调度问题, 而 AGV 调度与路径规划的协同优化可以有效提高系统的作业效率。针对 AGV 调度与路径协同优化的问题, 余娜娜等[21]提出一种聚类-协同优化算法, 其中设计协同进化遗传算法对包裹组进行指派和排序, 并将无冲突路径规划算法引入到协同进化遗传算法的解码方案中实现协同优化; 李静等[22]以最小化 AGV 的能量消耗为优化目标, 设计了两个阶段的算法对问题模型求解; Riazi 等[23]采用 Benders 分解方法的几个改进和加速策略同时解决 AGV 调度与无冲突路径问题。并提出了一种新的启发式方法, 可以快速产生高质量的解决方案。

由上述文献可知, AGV 调度与路径规划问题多采用混合整数规划建模, 并通过智能优化算法求解。智能优化算法凭借其全局搜索能力和适应性, 已成为解决该类问题的重要工具。

3. 问题描述与数学模型

3.1. 问题描述

智能仓储结构是一种利用先进技术和自动化系统来提升仓储效率 and 管理的仓储方案, 旨在实现更高效、更准确的仓储操作和库存管理。智能仓储结构通常采用各种自动化设备, 如自动小车, 机器人、输送带等, 用于货物的存储、装卸和运输。这些设备能够根据系统的指令和预定规则, 高效地完成仓库内的操作。自动化设备通过数据采集和传感器检测仓库信息数据, 这些数据通过物联网连接到中央控制系统, 为实时监控和决策提供支持, 智能物流仓储结构如图 1 所示。

如图 2 所示, 具体来说, 智能仓储中货架并行排列, 分为三个维度。从前到后货架号从 1 至最大货架范围依次递增, 从上至下分为不同层次, 同一货架同一层次分为多个存储槽位, 货物按类别分布在不

同的货架上。货架左右两侧设置叉车自动滑行轨道，货物在左侧存储在右侧取出。存货时叉车滑动至存储货架后，由升降杆搭载叉车托送货物至指定层次，叉车臂可以水平伸缩，在同一货架同一层次不同槽位存入货物。取货按类似操作将货物取出。

图 3 所示为本文 AGV 行驶布局图，货架间通道为 AGV 可以行驶的路径，货架两侧由多台 AGVs 依序轮替负责拣货。操作不同货架的 AGV 可以并行在路径上行驶，操作同一货架 AGV 需按调度顺序依次行驶避免冲突。接收到订单后 AGV 从停靠区域出发，拣货系统根据 AGV 拣货单中的订单货架号、层号、槽位控制 AGV 在对应货架可达路径前后或上下行驶。当 AGV 停靠在指令发出的货架前，货架区域内负责搬卸的机器人感知 AGV 停靠位置，协同将货物装卸到叉车上，叉车托载货物升至指定的货架层，并延伸叉车臂到指定槽位进行存货取货。

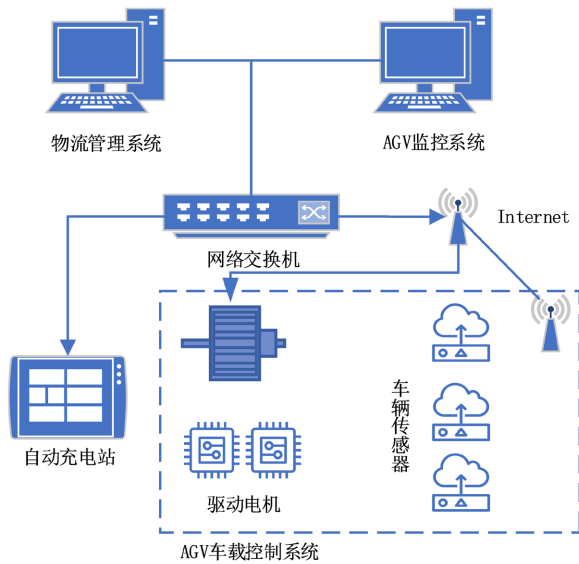


Figure 1. Warehouse structure of intelligent logistics
图 1. 智能物流仓储结构

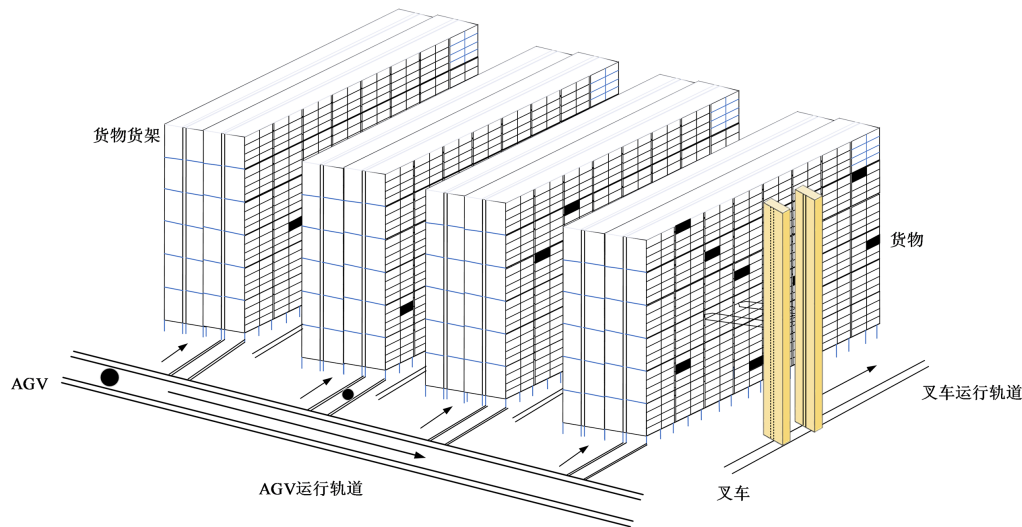


Figure 2. Warehouse shelf structure
图 2. 仓储货架结构

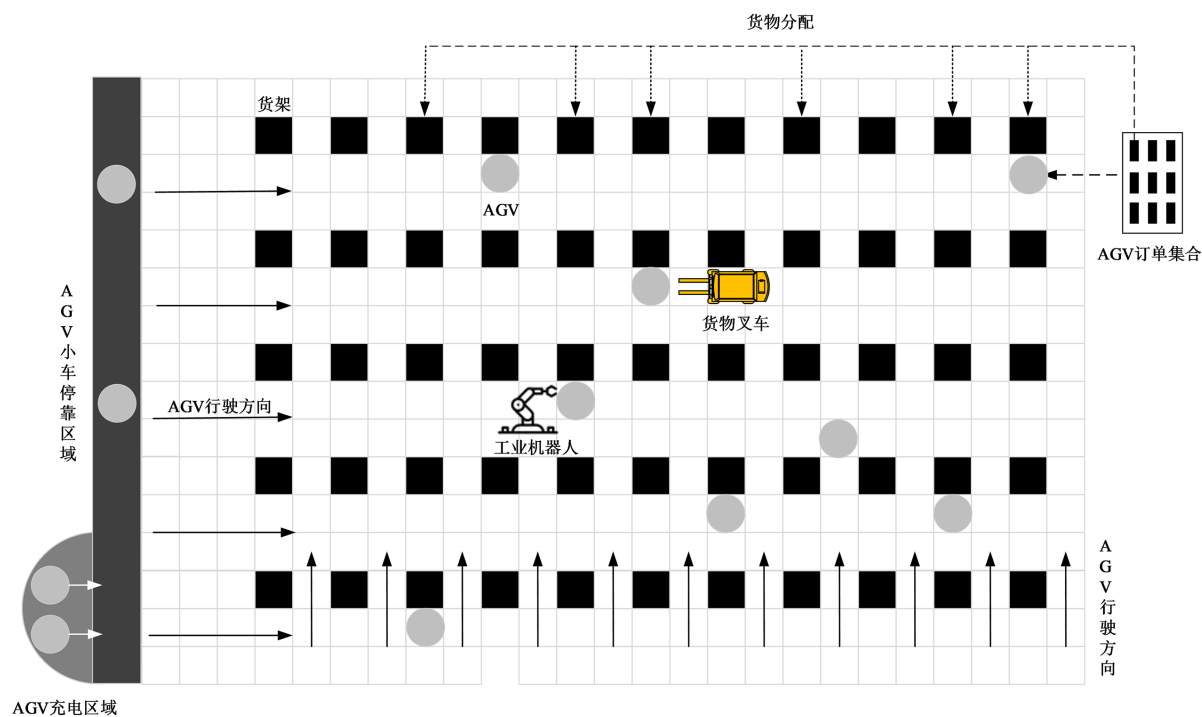


Figure 3. AGV driving layout
图 3. AGV 行驶布局图

3.2. 数学模型

为更加清晰地描述构建的数学模型以及后续的算法，本文涉及的符号及含义如表 1 所示。

Table 1. System resulting data of standard experiment

表 1. 标准试验系统结果数据

符号	含义
O_i	第 i 个订单
O_i^{min}	第 i 个订单中的最小货架号
O_i^{max}	第 i 个订单中的最大货架号
O_a^i	第 a 个小车的第 i 个订单
AGV_i	第 i 个 AGV 小车
$oNum$	订单总数
$vNum$	AGV 小车总数
x_{ij}	0-1 变量，当 O_i 分配到第 j 台 AGV 时， x_{ij} 值为 1，否则为 0

基于表 1 给出的符号以及描述，本文建立的数学模型如下。

$$\min \sum_{i=1}^{vNum} (H_i - L_i) \quad (1)$$

$$H_i = \max \{O_j^{max} \mid x_{ij} = 1, 1 \leq j \leq oNum\} \quad (2)$$

$$L_i = \min \{O_j^{min} \mid x_{ij} = 1, 1 \leq j \leq oNum\} \quad (3)$$

$$\sum_{j=1}^{oNum} x_{ij} = \left\lceil \frac{oNum}{vNum} \right\rceil \quad (4)$$

$$\sum_{i=1}^{vNum} x_{ij} = 1 \quad (5)$$

公式(1)表示基于 AGV 的批量分拣模型的优化目标, 公式(2)所示 H_i 是第 i 台 AGV 所达最大货架号, 公式(3)所示 L_i 是第 i 台 AGV 所达最小货架号, 公式(4)表示每台 AGV 均等分批的订单数量, 公式(5)表示每个订单只能分配给一台 AGV 运输。

4. 算法设计

4.1. 基于贪心策略的动态边界初始解生成

当 $vNum = 100$, $oNum = 2000$ 时, 问题解空间 $|\Theta| = 2.6 \times 10^{59}$, 无法在多项式的时间内求出最优解, 因此本文的问题具有 NP-难的性质。为了在有限的时间成本下深入搜索问题最优解, 一个好的初始解具有重要的作用, 其可以给予算法一个好的搜索开端, 有效加快算法的收敛速度, 优化最终解的质量。本文使用两种方法生成高质量的初始解。首先介绍基于贪心策略的动态边界初始解生成方法。

基于贪心策略的动态边界初始解生成算法的基本思想是: 在待分配订单选择具有最小覆盖区间的订单为核心订单, 以核心订单的最小货架号和最大货架号为边界, 计算其余订单边界相对于核心订单边界的增量, 每次选择具有最小增量的订单加入同一 AGV 并对边界进行更新。其中, 某个订单 i 的覆盖区间等于 $O_i^{max} - O_i^{min}$ 。算法的具体步骤如下:

Step 1. 计算每个订单的覆盖区间;

Step 2. 设置各 AGV 的分拣任务为空;

Step 3. 为第 i 台 AGV 分配拣货单;

Step 4. 在待分配订单中选择具有最小覆盖区间的订单 $oMLength$;

Step 5. 将订单 $oMLength$ 添加至第 i 台 AGV 的拣货单;

Step 6. 获取第 i 台 AGV 拣货单包含的最小货架号 MIN_i 和最大货架号 MAX_i ;

Step 7. 以 MIN_i 和 MAX_i 作为第 i 台 AGV 拣货单的边界;

Step 8. 计算每一待分配订单 O 与第 i 台 AGV 拣货单边界的增量 Δd ;

Step 9. 将具有最小增量的待分配订单添加至第 i 台 AGV 拣货单;

Step 10. 执行 Step 6 和 Step 7 更新第 i 台 AGV 拣货单边界;

Step 11. 如果第 i 台 AGV 拣货单包含订单数量为 $\lceil oNum/vNum \rceil$, $i = i + 1$ 且 $i < vNum$, 执行 Step 3; 否则, 执行 Step 13;

Step 12. 如果 AGV 拣货单包含订单数量少于 $\lceil oNum/vNum \rceil$, 执行 Step 8;

Step 13. 结束分配。

初始解生成阶段 Step 8 中待分配订单与第 i 台 AGV 拣货单边界的增量按公式(6)~(8)进行计算。

$$\Delta d = \Delta d_L + \Delta d_R \quad (6)$$

$$\Delta d_L = \begin{cases} MIN_i - oMin, & \text{if } MIN_i > oMin \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

$$\Delta d_R = \begin{cases} oMax - MAX_i, & \text{if } MAX_i < oMax \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

4.2. 基于订单特征的多角度迭代改进算法初始解生成

在智能仓储系统中,合理的订单分配对于提高 AGV 调度效率至关重要。传统的订单分配方法往往采用随机或简单的启发式规则,可能导致 AGV 的行驶路径过长,从而降低拣货效率。为此,本文提出了一种基于订单特征的多角度迭代优化方法(MAIOF),以最小化 AGV 的总行驶距离为目标,通过分析订单的离散度与内聚度,动态调整订单分配方案,从而优化初始解质量。其基本思想是:

(1) 初始解生成:采用随机方法为 AGV 分配订单,并计算初始行驶路径;

(2) 计算随机解的行驶总距离 $initDistance$ 后,计算每台 AGV 拣货单的离散度,第 i 台 AGV 拣货单中订单 j 的离散度 $disDegree_i^j$ 的计算方式如公式(9)所示:

$$disDegree_i^j = \left| \frac{O_j^{max} - O_j^{min}}{2} - cohDegree_i \right| \quad (9)$$

(3) 其中第 i 台 AGV 拣货单的内聚度 $cohDegree_i = (H_i - L_i)/2$,第 i 台 AGV 拣货单的离散度 $disDegree_i$ 如公式(10)所示:

$$disDegree_i = \sum_{k=1}^{\left\lfloor \frac{oNum}{vNum} \right\rfloor} disDegree_i^k \quad (10)$$

(4) 计算完离散度之后,将所有 AGV 按离散度从大到小排序,MAIOF 具体包括以下步骤:

Step 1. 选定具有最大离散度的 AGV_a ;

Step 2. 在剩余 AGV 中选定具有最小离散度的 AGV_b ;

Step 3. 计算 AGV_a 与 AGV_b 的行驶距离之和 $distance_{ab}$;

Step 4. 对于 AGV_a 中的每一个订单 O_a^i ,如果 $\exists O_b^j \in AGV_b$,使得 O_a^i 与 O_b^j 交换后 AGV_a 与 AGV_b 的新行驶距离 $newdistance_{ab} < distance_{ab}$,则将 O_b^j 加入 O_a^i 的可交换订单候选集 $O_a^i canSet$;

Step 5. 对于 AGV_a 中的每一个订单 O_a^i ,计算其与 AGV_b 的可交换订单候选集 $O_a^i canSet$ 中每一个订单

O_b^j 的交换成本 $cost(O_a^i, O_b^j) = \frac{|disDegree_a^i - newdisDegree_a^i|}{disDegree_a^i}$;

Step 6. 在 AGV_a 中选择具有最小交换成本的订单与 AGV_b 进行交换,执行 Step 8;

Step 7. 如果 AGV_a 中所有订单与 AGV_b 的可交换订单候选集都为空,且剩余未交换 AGV 不为空,则执行 Step 2;

Step 8. 结束选择交换。

往往一次迭代无法获得较优的初始解,为生成较优的初始解,一般对随机解进行 N 次迭代($N > 5$)。

4.3. 改进模拟退火算法

模拟退火算法借鉴了固体退火的过程,在温度较高的时候,初始解逐步向局部最优解更新,但在温度下降到一定程度后,解空间会陷入局部最优解,而模拟退火算法给予空间中不好的解一个接受的概率。在迭代的过程中,不好的解可能被接受为新解从而跳出了局部最优继续向整体最优更新。

本文所述的改进模拟退火算法具有 3 个基本参数:起始温度 T ,终止温度 T_{min} ,温度变化率 ΔT 。在模拟退火迭代的过程中,为避免过早陷入局部最优解,使用了两个策略并设置了策略选择概率 r_{switch} 来随机选择策略。 p_{switch} 随机生成且 $p_{switch} \in [0,1]$ 。在算法开始时,首先获取初始解的初始行驶总距离 $initDistance$,以模拟退火为搜索框架综合利用订单间的最大最小货架关系引导解的搜索方向,在每次迭代后,计算 $\Delta E_k = updateDistance - initDistance$ 用以衡量当前阶段解改进的优劣,若改进向好的方向更新,则继续;否则以一定概率考虑改变搜索区域。改进模拟退火算法流程见图 4。

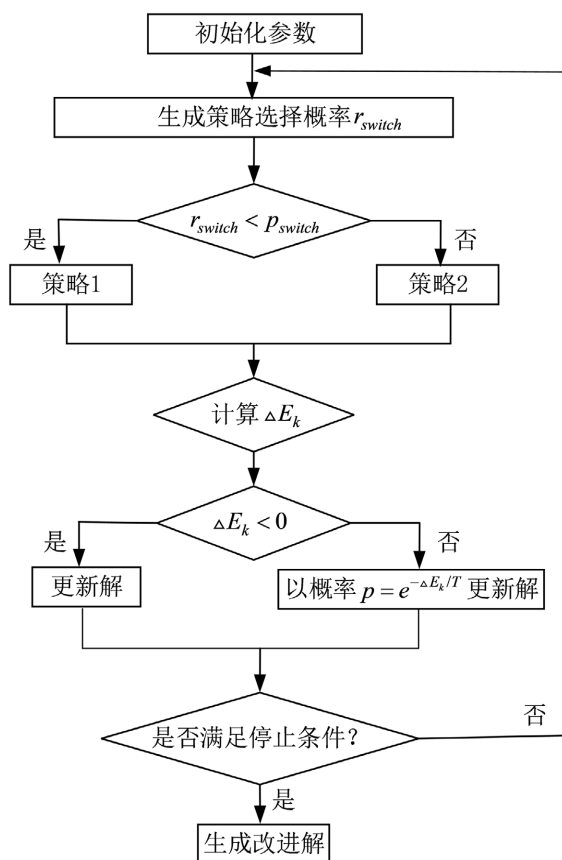


Figure 4. Flowchart of the improved simulated annealing algorithm

图 4. 改进模拟退火算法流程图

所述的策略 1 首先获取模拟退火当前解的行驶总距离 $initialDistance$ ，通过内聚度 $cohDegree$ 大小对每一台 AGV 进行聚类，在每一个聚类的集合中进行订单交换操作，策略 1 具体包括以下步骤：

Step 1. $improvedDistance = initialDistance$ ；

Step 2. 利用 K-Means 基于内聚度将 AGV 聚类为 N 组， $(G_1, G_2, \dots, G_N)$ ；

Step 3. 计算第 i 组 G_i 中每台 AGV 拣货单的行驶距离，即第 i 组第 j 台 AGV 的行驶距离为 $jDisG_i$ ，选取 G_i 中具有最大行驶距离的 AGV 拣货单 $maxDisG_i$ 和具有最小行驶距离的 AGV 拣货单 $minDisG_i$ ；

Step 4. 将 $maxDisG_i$ 中具有最大货架号订单与 $minDisG_i$ 中具有最大货架号订单交换，计算交换后两台 AGV 拣货单的行驶距离 $newmaxDisG_i$ ， $newminDisG_i$ ；

Step 5. 若 $newmaxDisG_i + newminDisG_i$ 小于 $maxDisG_i + minDisG_i$ ，则更新两台 AGV 拣货单，更新 $improvedDistance$ ；否则，执行 Step 6；

Step 6. $i = i + 1$ 且 $i \leq N$ ，执行 Step 3；否则，结束策略 1。

策略 2 的步骤与策略 1 基本一致，但是在 Step 4 的关键订单交换中，策略 2 选择将 $maxDisG_i$ 中具有最小货架号订单与 $minDisG_i$ 中具有最小货架号订单交换。

改进模拟退火算法的参数包括起始温度 T ，终止温度 T_{min} ，温度变化率 ΔT ，策略选择概率 r_{switch} 。通过改变参数进行对比实验，分别将各项参数设为 $T = 50$ ， $T_{min} = 1 \times 10^{-9}$ ， $\Delta T = 0.6$ ， $r_{switch} = 0.5$ 。初始解的接受概率较高，使得算法能够在较大的搜索空间内进行全局探索。设定 $T = 50$ 以确保初期较大的扰动能力，防止算法过早收敛。控制算法停止的条件，通常设定一个足够小的温度， $T_{min} = 1 \times 10^{-9}$ 以保证算法

在收敛时仍能搜索到较优解。为确保算法在初期探索较大搜索空间, 在后期细化搜索, 设定 $\Delta T = 0.6$ 。为了在两个不同的搜索策略之间进行选择, 本文采用自适应双策略机制, 策略选择概率设定 $r_{switch} = 0.5$, 使两种策略均衡作用。

4.4. 改进遗传算法

在智能仓储系统中, 合理的订单分配能够有效减少 AGV(自动导引车)的总行驶距离, 提高拣货效率。针对传统方法容易陷入局部最优的问题, 本文提出一种基于改进模拟退火的多角度迭代优化方法(MAIOF)。

由于较大的种群规模可以提高解的多样性, 但计算成本较高。本实验设定 $popNum = 20$, 保证一定的多样性, 同时控制计算开销。最大迭代次数 $MaxIter$ 用于设定算法的终止条件, 避免陷入死循环。本实验设置 $MaxIter = 50$, 确保算法有足够的进化代数以逼近最优解。交叉变异概率控制父代个体之间的基因交换频率, 通常设定在 0.6 到 0.9 之间。本实验采用 $p \leq p_{com} = 0.8$, 较高的交叉变异概率可以加快种群进化, 提高算法的全局搜索能力。交叉方式选择概率为 p_{way} 设为 0.5。在改进遗传算法迭代的过程中, 若 $p \leq p_{com}$ 则执行交叉操作。

以模拟退火(SA)为搜索框架, 并结合订单特征(内聚度、离散度)进行聚类优化。通过设计双策略自适应搜索机制, 在模拟退火过程中随机选择不同的优化策略, 以增强解的探索能力, 提高算法的全局搜索能力。具体步骤如下。

1) 染色体编码

为便于表示订单在 AGV 小车上的分配情况, 本文采用二维 0-1 矩阵对染色体进行编码, 一个染色体的编码如图 5 所示:

$$\begin{vmatrix} x_{11} & \cdots & x_{1 \ oNum} \\ \vdots & \ddots & \vdots \\ x_{vNum \ 1} & \cdots & x_{vNum \ oNum} \end{vmatrix}$$

Figure 5. Chromosome encoding structure
图 5. 染色体编码结构

染色体矩阵每行代表对应 AGV 的小车, 每列代表订单的编号。如图所示, 若 $x_{13} = 1$, 代表编号为 3 的订单被分配到了第 1 个 AGV 小车。图 5 表示的矩阵必须严格满足问题的约束条件, 即矩阵每行决策变量 x 为 1 的数量必须恒等于 $oNum/vNum$, 每列决策变量 x 为 1 的数量必须恒等于 1。

2) 种群初始化

设种群规模等于 $popNum$, 在种群初始化时, 首先选择一个 2.1 或 2.2 节生成的高质量初始解加入初始种群, 接着随机生成 $popNum - 1$ 个随机初始解加入初始种群。

3) 交叉与变异

本文的改进遗传算法在每次交叉变异前会更新两个参数 α 和 p 。参数 p 为 $[0, 1]$ 随机数, 用于交叉和变异的概率, 参数 α 用于选择不同的交叉方式, 计算方式如公式(11) (12)所示:

$$\alpha' = 2 - t \times (2 / MaxIter) \quad (11)$$

$$\alpha = 2\alpha' \times r - \alpha' \quad (12)$$

其中 t 代表当前迭代次数, $MaxIter$ 代表预先设定的最大迭代次数, r 为 $[0, 1]$ 随机数。

假设交叉变异概率为 p_{com} , 交叉方式选择概率为 p_{way} 。在改进遗传算法迭代的过程中, 若 $p \leq p_{com}$ 则

执行交叉操作, 在交叉操作中, 计算参数 α , 若 $|\alpha| < p_{way}$, 则使用 MAIOF 算法对种群中前 $popNum/2$ 个个体进行一次优化。若 $|\alpha| \geq p_{way}$ 则使用逼近最优解交叉方法如图 6 所示; 若 $p > p_{com}$ 则执行逼近随机解变异方法如图 7 所示。

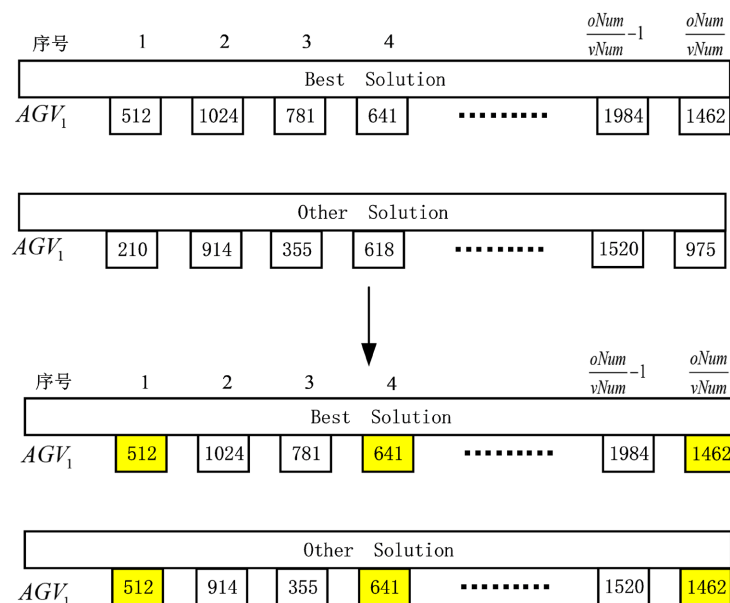


Figure 6. Crossover process structure

图 6. 交叉流程

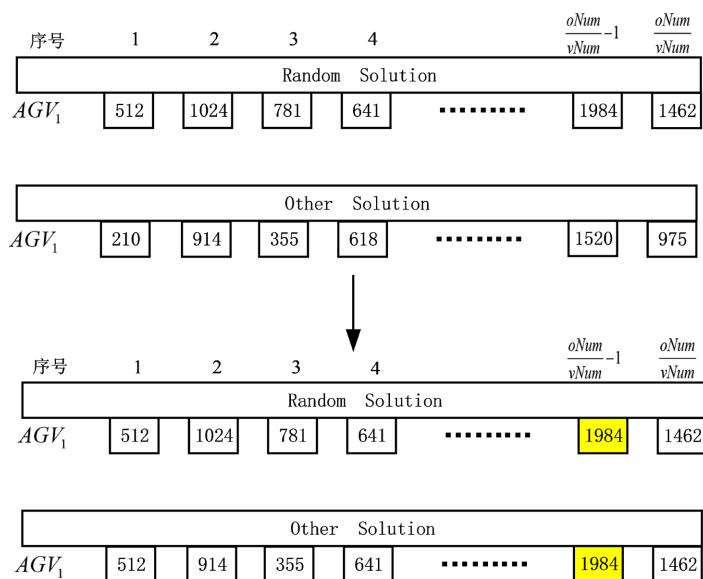


Figure 7. Mutation process

图 7. 变异流程

如图所示, 逼近最优解交叉方法包括以下步骤:

Step 1. 在交叉变异前, 根据每个个体的适应度值从小到大对个体种群进行排序, 取出最优解 p_0 ;

Step 2. 取出第 i 个个体 p_i , 其中 $i = (1, 2, \dots, popNum-1)$;

Step 3. 从 p_0 中的 AGV_k 小车中选出 $outNumber$ 个订单替换掉 p_i 中 AGV_k 对应位置的 $outNumber$ 个订单, 并执行订单修复算子, 将生成的新个体加入个体种群; 其中 $outNumber = \lceil oNum / (4 \times vNum) \rceil$;

Step 4. $k = k + 1$, 若 $k < vNum$ 执行 Step 3; 否则执行 Step 5;

Step 5. $i = i + 1$, 若 $i < popNum$ 执行 Step 2; 否则结束逼近最优解交叉。

逼近随机解变异方法包括以下步骤:

Step 1. 在交叉变异前, 根据每个个体的适应度值从小到大对个体种群进行排序, 取出随机解 p_r , $p_r \neq p_0$;

Step 2. 取出第 i 个个体 p_i , 其中 $i = (1, 2, \dots, popNum - 1), i \neq r$;

Step 3. 从 p_r 中的 AGV_k 小车中选出 $outNumber$ 个订单替换掉 p_i 中 AGV_k 对应位置的 $outNumber$ 个订单, 并执行订单修复算子。其中 $outNumber = 1$;

Step 4. $k = k + 1$, 若 $k < vNum$ 执行 Step 3; 否则执行 Step 5;

Step 5. $i = i + 1$, 若 $i < popNum$ 执行 Step 2; 否则结束逼近最优解交叉。

在上述交叉变异的过程中, 订单的交换会导致生成的新个体无法满足问题的约束条件, 所以必须设置一个修复算子来调整新生成的个体以满足约束条件。订单修复算子的步骤如图 8 所示:

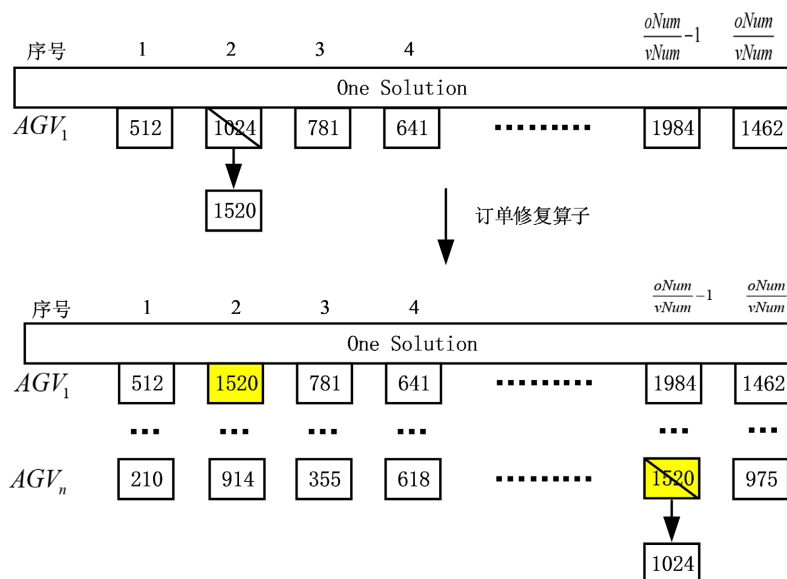


Figure 8. Schematic diagram of order repair operator

图 8. 订单修复算子示意图

如图所示 AGV_1 中 1024 号订单被交叉或变异操作替换为 1520 号订单, 因为 1520 号订单以及被分配到其他 AGV, 所以将会导致违反约束条件。执行订单修复算子时, 遍历剩余的订单, 直到找到引发冲突的订单(1520 号)所在的 AGV 小车编号以及位置, 如图所示 1520 号订单在 AGV_n 的 $\frac{oNum}{vNum} - 1$ 号位置, 将该位置的订单设为之前被替换掉的订单(1024 号)。

4) 选择

选择操作保留当前种群中最优秀的解, 在剩余的个体中使用轮盘赌的方式选择 $popNum - 1$ 个个体。

改进遗传算法的伪代码如下所示:

```
1. Generate the initial population  $p_i (i = 1, 2, \dots, popNum)$ 
2. Evaluate the fitness for each candidate solutions
3. Sort population by fitness from smallest to largest
4. for  $i = 1$  to  $MaxIter$  do
5.   Update  $\alpha$  and  $p$ 
6.   If  $p \leq p_{com}$  then
7.     If  $|\alpha| < p_{way}$ 
8.       for  $j = 0$  to  $popNum/2$  do
9.         UseMAIOFfor  $p_j$ 
10.      End for
11.     else if  $|\alpha| > p_{way}$ 
12.       逼近最优解交叉
13.     End if
14.   If  $p > p_{com}$  then
15.     逼近随机解变异
16.   End if
17.   Update fitness for each candidate solutions
18.   Select individual in population
19. End for
```

5. 仿真实验及结果分析

为评估提出方法的性能, 本文采用仓储真实数据集与随机生成数据集两种场景进行仿真实验, 根据实验结果分别对动态边界初始解生成策略与智能进化分拣方法的性能进行全面的比较分析。算法采用 python3.7 进行实现, 在 i7-8750H 2.20GHz 处理器、16GB 内存、64 位 win11 操作系统的计算机上进行测试。

5.1. 实验数据

本文首先采用仓储真实数据集对算法性能进行验证, 如表 2 所示:

Table 2. Features of actual warehouse order data

表 2. 仓储实际订单数据特征

	订单数量	货架范围	最小货架		最大货架	
			均值	标准差	均值	标准差
D1	2000	4~2030	286.25	278.84	736.65	422.19
D2	2000	4~2032	240.79	281.34	686.98	460.54

真实仓储数据集包含两组数据 D1 与 D2, 订单数量均为 2000。D1 货架范围在 4~2030 之间, 订单最小货架号最大货架号分别为 4 和 2030, 最小货架号均值为 286.25, 标准差为 278.84, 最大货架号均值为 736.65, 标准差为 422.19。D2 货架范围在 4~2032 之间, 订单最小货架号最大货架号分别为 4 和 2032, 最小货架号均值为 240.79, 标准差为 281.34, 最大货架号均值为 686.89, 标准差为 460.54。表 2 所示仓储实际订单数据特征表明实际订单包含货物数量不固定, 货物所在位置离散且覆盖货架范围大。

随着智能仓储应用领域趋于广泛, 现有仓储规模显著扩大, 为确保实验结果的全面性与客观性, 本文以真实仓储数据特征为依据, 随机生成三种(小规模、中规模、大规模)不同订单数量规模的数据集, 如表 3 所示, 小规模、中规模、大规模数据集分别包含 2000、5000、10000 个订单, 每种规模数据集的货架号范围均为 1~5000。

Table 3. Features of three randomly generated datasets**表 3.** 三种随机生成数据集特征

	小规模	中规模	大规模
订单数量	2000	5000	10000
货架范围		[1, 5000]	

为进一步体现实际仓储货架布局的多样性以及订单构成的通用性, 每种规模数据集中货架均涵盖均匀分布, 正态分布及指数分布三种形态, 各形态数据生成参数设置如表 4 所示。

Table 4. Parameters for generating three types of data shapes**表 4.** 三种形态数据生成参数

	最小货架号	最大货架号
均匀分布	$U[1, 2500]$	$U[2501, 5000]$
正态分布	$N(715, 800)$	$N(1840, 1250)$
指数分布	$E(715)$	$E(715)$

在随机生成一个订单内包含的货架号之前, 首先要确定每个订单中包含的货架数。通过对实际数据进行分析, 确定表 1 中 D1 数据集里每个订单包含的货架数的均值为 2.81, 标准差为 2.16, 本文以此参数作为参考, 通过 $N(2.81, 2.16)$ 的正态分布生成订单中包含的货架数 N_s 。生成货架数 N_s 后, 使用 $p = 0.5$ 的伯努利分布随机生成长度为 N_s 的 0-1 数组。从数组第一个位置开始, 若数组当前位置值为 0 则按照表 4 给出的参数生成对应分布(均匀分布/正态分布/指数分布)并满足货架范围的最小货架号, 若数组当前位置值为 1 则同理生成最大货架号, 每当生成一个货架号时, 将货架号加入当前订单, 直到生成 N_s 个货架号后, 转入下一个订单。

改进模拟退火算法的参数包括起始温度 T , 终止温度 T_{min} , 温度变化率 ΔT , 策略选择概率 r_{switch} 。通过改变参数进行对比实验, 分别将各项参数设为 $T = 50$, $T_{min} = 1 \times 10^{-9}$, $\Delta T = 0.6$, $r_{switch} = 0.5$ 。改进遗传算法的种群规模 $popNum$ 设为 20, 最大迭代次数 $MaxIter$ 设为 50, p_{com} 设为 0.8, p_{way} 设为 0.5。3 组不同分布的数据, 共 9 组数据。3 种分布生成数据的参数如表 4 所示: 测试数据来进行仿真实验。本文的智能仓储随机生成测试数据特征如表 1 所示, 其中 AGV 数量 $vNum$ 均为 100。

5.2. 动态边界初始解生成策略性能分析

为验证基于动态增量的初始解生成算法以及基于订单特征的多角度迭代改进算法(MAIOF)生成初始解的有效性, 进行了以下实验: 首先通过随机分配订单生成初始随机解, 再使用基于动态增量的初始解生成算法生成初始解, 使用基于订单特征的多角度迭代改进算法(MAIOF)改进随机初始解, 对比两种算法对初始解的改进程度。

表 5 所示是在 3 种规模以及 3 种分布下随机生成 100 次初始解并取平均值的初始解大小。表 6 为使用动态增量算法生成初始解的大小及生成时间, 表 7 为使用基于订单特征的多角度迭代改进算法(MAIOF)迭代 10 次生成初始解的大小及生成时间。由表 6 与表 7 可知, 使用动态增量算法生成的初始解以及使用 MAIOF 改进的初始解在所有规模与分布下对比随机初始解都有较大的优化。说明考虑订单区间覆盖并动态修改订单边界同时考虑订单离散特征可以对解进行有效的改进。

Table 5. Values of random solutions (time/s)
表 5. 随机解数值大小数值(时间/s)

	小规模	中规模	大规模
均匀分布	480,269 (0.004)	492,262 (0.023)	495,869 (0.086)
正态分布	388,970 (0.004)	417,350 (0.022)	441,519 (0.085)
指数分布	421,316 (0.004)	463,939 (0.023)	481,719 (0.086)

Table 6. Parameters for generating three types of data shapes
表 6. 三种形态数据生成参数

	小规模	中规模	大规模
均匀分布	189,070 (6.188)	194,026 (80.047)	193,992 (605.981)
正态分布	129,845 (6.143)	132,764 (82.013)	133,626 (621.827)
指数分布	131,115 (6.150)	133,029 (84.669)	133,842 (648.136)

Table 7. Parameters for generating three types of data shapes
表 7. 三种形态数据生成参数

	小规模	中规模	大规模
均匀分布	283,194 (31.378)	309,278 (185.983)	305,358 (735.871)
正态分布	175,412 (31.253)	190,967 (187.451)	213,204 (698.998)
指数分布	175,368 (28.862)	180,510 (175.179)	187,435 (660.201)

5.3. 智能优化算法有效性测试

为验证所提出的改进模拟退火算法及改进遗传算法的有效性,使用动态增量算法生成的初始解以及 MAIOF 改进的初始解作为两种智能优化的初始解。

表 8 为改进模拟退火算法在使用动态增量初始解下的优化结果,表 9 为改进模拟退火算法在使用 MAIOF 初始解下的优化结果。

Table 8. Results of SA improvement (dynamic increment initial solutions) (time/s)
表 8. SA 改进(动态增量初始解)结果数值(时间/s)

	小规模	中规模	大规模
均匀分布	188,653 (52.733)	193,921 (137.892)	193,863 (265.283)
正态分布	128,744 (55.801)	131,838 (127.483)	132,356 (258.363)
指数分布	130,196 (54.759)	132,510 (134.966)	133,336 (268.530)

Table 9. Results of SA improvement (MAIOF initial solutions) (time/s)
表 9. SA 改进(MAIOF 初始解)结果数值(时间/s)

	小规模	中规模	大规模
均匀分布	242,692 (58.096)	251,200 (127.649)	257,848 (438.236)
正态分布	156,983 (53.842)	163,286 (122.604)	186,253 (250.529)
指数分布	152,526 (55.220)	165,953 (145.696)	184,205 (324.276)

由表 8 可以看出,改进 SA 算法在使用动态增量的初始解的情况下对每组数据都有着一定程度的改进,特别在正态分布和指数的分布的数据下,改进幅度更为明显。由表 9 可知,改进 SA 算法在使用 MAIOF

初始解的情况下对各组数据都有较大的改进, 证明了聚类 - 边界交换算法以及模拟退火算法在搜索解空间上的有效性。

表 10 为改进遗传算法在使用 MAIOF 初始解下的优化结果。

Table 10. Results of GA improvement (MAIOF initial solutions)

表 10. GA 改进(MAIOF 初始解)结果

	小规模	中规模	大规模
均匀分布	243,000 (128.511)	276,112 (360.347)	274,816 (820.031)
正态分布	149,409 (164.391)	163,714 (460.784)	171,742 (876.203)
指数分布	158,977 (147.143)	160,575 (424.795)	153,991 (881.331)

由表 10 可知, 在初始种群中加入 MAIOF 初始解的改进遗传算法有着较大的优化幅度, 在正态分布和指数分布的数据下, 改进 GA 算法的优化程度基本优于所提出的改进的 SA 算法, 而在均匀分布下, 改进 GA 算法的优化程度并没有改进 SA 算法好。

6. 结论

本文考虑一种基于 AGVs 的智能仓储批量分拣模型及两阶段改进批量分拣优化方法, 第一阶段考虑订单特征生成两种高质量的初始解, 第二阶段使用模拟退火算法和遗传算法对高质量的初始解进行优化。在不同规模不同分布的数据上进行实验, 表明所提出的初始解生成算法与智能优化算法两阶段改进可以生成优秀的分配结果。

在接下来的研究中, 应该注重于复杂化问题的数学模型, 并考虑 AGV 的电量, 冲突等对智能仓储生产产生明显影响的关键约束。

致 谢

本研究得到江苏省研究生科研与实践创新计划项目(No. KYCX23_1055)资助, 在此对该项目的支持表示诚挚感谢。

基金项目

江苏省研究生科研与实践创新计划项目(No. KYCX23_1055)。

参考文献

- [1] 付建林, 张恒志, 张剑, 等. 自动导引车调度优化研究综述[J]. 系统仿真学报, 2020, 32(9): 1664-1675.
- [2] Boccia, M., Masone, A., Sterle, C. and Murino, T. (2023) The Parallel AGV Scheduling Problem with Battery Constraints: A New Formulation and a Matheuristic Approach. *European Journal of Operational Research*, **307**, 590-603. <https://doi.org/10.1016/j.ejor.2022.10.023>
- [3] Dang, Q., Singh, N., Adan, I., Martagan, T. and van de Sande, D. (2021) Scheduling Heterogeneous Multi-Load Agvs with Battery Constraints. *Computers & Operations Research*, **136**, Article ID: 105517. <https://doi.org/10.1016/j.cor.2021.105517>
- [4] Goli, A., Tirkolaee, E.B. and Aydin, N.S. (2021) Fuzzy Integrated Cell Formation and Production Scheduling Considering Automated Guided Vehicles and Human Factors. *IEEE Transactions on Fuzzy Systems*, **29**, 3686-3695. <https://doi.org/10.1109/tfuzz.2021.3053838>
- [5] Riazi, S., Bengtsson, K. and Lennartson, B. (2021) Energy Optimization of Large-Scale AGV Systems. *IEEE Transactions on Automation Science and Engineering*, **18**, 638-649. <https://doi.org/10.1109/tase.2019.2963285>
- [6] Zou, W., Pan, Q., Wang, L., Miao, Z. and Peng, C. (2022) Efficient Multiobjective Optimization for an AGV Energy-Efficient Scheduling Problem with Release Time. *Knowledge-Based Systems*, **242**, Article ID: 108334. <https://doi.org/10.1016/j.knosys.2022.108334>

- [7] Zou, W., Pan, Q., Meng, L., Sang, H., Han, Y. and Li, J. (2023) An Effective Self-Adaptive Iterated Greedy Algorithm for a Multi-AGVs Scheduling Problem with Charging and Maintenance. *Expert Systems with Applications*, **216**, Article ID: 119512. <https://doi.org/10.1016/j.eswa.2023.119512>
- [8] Singh, N., Dang, Q., Akcay, A., Adan, I. and Martagan, T. (2022) A Matheuristic for AGV Scheduling with Battery Constraints. *European Journal of Operational Research*, **298**, 855-873. <https://doi.org/10.1016/j.ejor.2021.08.008>
- [9] Bao, B., Duan, Z. and Chen, W. (2020) Mission Scheduling of Multi-AGV System with Dynamic Simulation. 2020 *International Symposium on Autonomous Systems (ISAS)*, Guangzhou, 6-8 December 2020, 115-120. <https://doi.org/10.1109/isas49493.2020.9378846>
- [10] Maoudj, A., Kouider, A. and Christensen, A.L. (2023) The Capacitated Multi-AGV Scheduling Problem with Conflicting Products: Model and a Decentralized Multi-Agent Approach. *Robotics and Computer-Integrated Manufacturing*, **81**, Article ID: 102514. <https://doi.org/10.1016/j.rcim.2022.102514>
- [11] Hu, H., Jia, X., Liu, K. and Sun, B. (2021) Self-Adaptive Traffic Control Model with Behavior Trees and Reinforcement Learning for AGV in Industry 4.0. *IEEE Transactions on Industrial Informatics*, **17**, 7968-7979. <https://doi.org/10.1109/tii.2021.3059676>
- [12] Lin, C., Chen, K. and Hsieh, L. (2023) Real-time Charging Scheduling of Automated Guided Vehicles in Cyber-Physical Smart Factories Using Feature-Based Reinforcement Learning. *IEEE Transactions on Intelligent Transportation Systems*, **24**, 4016-4026. <https://doi.org/10.1109/tits.2023.3234010>
- [13] Lin, Y., Xu, Y., Zhu, J., Wang, X., Wang, L. and Hu, G. (2023) MLATSO: A Method for Task Scheduling Optimization in Multi-Load AGVs-Based Systems. *Robotics and Computer-Integrated Manufacturing*, **79**, Article ID: 102397. <https://doi.org/10.1016/j.rcim.2022.102397>
- [14] Zheng, H., Xu, W., Ma, D. and Qu, F. (2022) Dynamic Rolling Horizon Scheduling of Waterborne AGVs for Inter Terminal Transportation: Mathematical Modeling and Heuristic Solution. *IEEE Transactions on Intelligent Transportation Systems*, **23**, 3853-3865. <https://doi.org/10.1109/tits.2021.3102998>
- [15] Lian, Y., Yang, Q., Xie, W. and Zhang, L. (2021) Cyber-Physical System-Based Heuristic Planning and Scheduling Method for Multiple Automatic Guided Vehicles in Logistics Systems. *IEEE Transactions on Industrial Informatics*, **17**, 7882-7893. <https://doi.org/10.1109/tii.2020.3034280>
- [16] Li, Z., Sang, H., Pan, Q., Gao, K., Han, Y. and Li, J. (2023) Dynamic AGV Scheduling Model with Special Cases in Matrix Production Workshop. *IEEE Transactions on Industrial Informatics*, **19**, 7762-7770. <https://doi.org/10.1109/tii.2022.3211507>
- [17] Hu, H., Jia, X., He, Q., Fu, S. and Liu, K. (2020) Deep Reinforcement Learning Based AGVs Real-Time Scheduling with Mixed Rule for Flexible Shop Floor in Industry 4.0. *Computers & Industrial Engineering*, **149**, Article ID: 106749. <https://doi.org/10.1016/j.cie.2020.106749>
- [18] 邹裕吉, 宋豫川, 王馨坤, 等. 自动导向小车与加工设备多目标集成调度的聚类遗传算法[J]. 中国机械工程, 2022, 33(1): 97-108.
- [19] 范厚明, 郭振峰, 岳丽君, 等. 考虑能耗节约的集装箱码头双小车岸桥与 AGV 联合配置及调度优化[J]. 自动化学报, 2021, 47(10): 2412-2426.
- [20] Xin, J., Meng, C., D'Ariano, A., Wang, D. and Negenborn, R.R. (2022) Mixed-Integer Nonlinear Programming for Energy-Efficient Container Handling: Formulation and Customized Genetic Algorithm. *IEEE Transactions on Intelligent Transportation Systems*, **23**, 10542-10555. <https://doi.org/10.1109/tits.2021.3094815>
- [21] 余娜娜, 李铁克, 张文新, 等. 自动分拣仓库中多载量 AGV 调度与路径规划算法[J]. 计算机集成制造系统, 2024, 30(4): 1458-1471.
- [22] 李静, 朱小林. 集装箱码头上多自动引导车的调度和路径规划[J]. 计算机集成制造系统, 2022, 28(5): 1449-1461.
- [23] Riazi, S. and Lennartson, B. (2021) Using CP/SMT Solvers for Scheduling and Routing of AGVs. *IEEE Transactions on Automation Science and Engineering*, **18**, 218-229. <https://doi.org/10.1109/tase.2020.3012879>