

Dynamic Path Planning for Multi-Robot in Warehousing Operations

Xinmiao Wang

School of Economics and Management, Beijing Jiaotong University, Beijing
Email: 17125512@bjtu.edu.cn

Received: Apr. 26th, 2019; accepted: May 13th, 2019; published: May 21st, 2019

Abstract

Handling robots have been widely used in the picking system of “part-to-picker” in un-manned warehouse. However, there are still problems such as unreasonable path seeking planning and path conflict in the collaborative operation of multi-handling robot. Considering the load balance of the path and the turning time of the handling robot, the path planning model of the multi-handling robot with time window is established. Combined with the obstacle avoidance strategy of adjusting the priority, an improved “offline-online” two-stage dynamic path planning algorithm was proposed to realize the combination of global off-line path planning and online conflict avoidance, thus improving the operation efficiency of the un-manned warehouse system. Finally, three sets of simulation experiments are designed to verify the effectiveness of the improved algorithm by comparing the total running time of the handling robots.

Keywords

Multi-Handling Robot, Path Planning, Time Window, Two-Stage, Dynamic

仓储作业中多搬运机器人动态路径优化

王鑫淼

北京交通大学经济管理学院, 北京
Email: 17125512@bjtu.edu.cn

收稿日期: 2019年4月26日; 录用日期: 2019年5月13日; 发布日期: 2019年5月21日

摘要

搬运机器人在无人仓的“货到人”拣选系统中得到了广泛应用, 然而多个搬运机器人协同作业还存在着

路径寻优规划不合理、路径冲突等问题。考虑路径的负载平衡及搬运机器人转弯耗费的时间,建立了带有时间窗的多搬运机器人路径规划模型;结合调整优先级的避障策略,提出了一种改进的“离线-在线”两阶段动态路径规划算法,实现全局离线路径规划和在线冲突规避,提升了无人仓系统的运行效率。最后,设计三组仿真实验,通过对比搬运机器人的总运行时间,验证了改进算法的有效性。

关键词

多搬运机器人, 路径规划, 时间窗, 两阶段, 动态

Copyright © 2019 by author(s) and Hans Publishers Inc.

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

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



Open Access

1. 引言

人工智能、大数据等新一代信息技术提高了智慧物流系统的思维、感知和分析决策能力。智慧物流的三大主要研究领域包括仓储、运输和配送,三个最基本特征是操作的无人化、运营的智能化和决策的智慧化。以京东、亚马逊为代表的电商企业正在加速布局智慧物流体系,推动着智能化的搬运机器人在“货到人”系统中的应用。在拣选作业中,随着搬运机器人规模数量的增加,需要同时考虑障碍避碰、拥堵协调、冲突处理等问题,大大增加了路径规划的复杂程度。由于电商企业对于拣选的时效性要求较高,为搬运机器人规划出一条无冲突的时间最优路径对其作业效率有着很大的影响。

“货到人”拣选系统的优化对提升电商企业的市场竞争力具有重要作用[1]。部分研究者对智能仓库系统中的一些简化子问题开展了初步的研究工作,包括 Hazard [2]等学者研究静态货位分配问题、Li [3]等研究静态任务指派和路径规划问题等。但这些研究工作都是针对静态情况的,并没有考虑实际订单拣选过程的连续性和各个环节之间的关联性。研究结果还无法解决“货到人”拣选系统中的实际问题,因此必须研究动态的路径规划策略[4]。

多搬运机器人路径规划(Multi-Robot Path Finding, MRPF)问题分为解耦的和耦合的。解耦化的方法是指将搬运机器人看作多个独立的个体,各自规划行驶路径。采用解耦化的多搬运机器人路径规划方法计算复杂性低、鲁棒性好。其缺陷在于当仓库中搬运机器人数量较大时,极易产生冲突和拥堵,造成复杂的路径再次规划问题。Khorshid [5]等针对解耦算法无法保证最优解及完备解,提出了一种树搜索的方法。Chang [6]使用 Dijkstra 算法为双向路径中的多搬运机器人设计了无冲突的最短行驶路径。Ma [7]等提出带有惩罚项的改进粒子群优化算法解决了双层仓库多搬运机器人路径规划问题。Umar [8]等提出了一种混合遗传算法,对柔性生产系统环境下的多机器人系统进行路径规划,并采用设立优先级的方式对柔性生产系统环境下的任务和机器人进行集成排序,解决路径冲突。Desrochers [9]等人提出了应用时间窗解决多台机器人的碰撞问题。Wang [10]提出了一种概率迭代路径协调的局部搜索算法(PIPC),为最晚完工的机器人找到最优解。Alejandro [11]考虑到路径安全性、路径长度和路径平稳性,提出了一种基于混合蛙跳算法的多目标解耦算法。

基于耦合的多搬运机器人路径规划是指在运行过程中考虑到机器人会相互影响,从而考虑将所有机器人看作一个整体,为其规划路径。基于耦合的多搬运机器人路径规划方法能够实现机器人间的紧密和最优协调,规划的路径通常是最优(次优)及完备的解。Ma [12]重点研究了基于集群的多机器人任务分配及路径规划、负载可交换的多机器人路径规划、及路径执行不完美的问题。Miao [13]提出了改进的模拟

退火算法，用于机器人在静态和动态障碍物环境下的路径规划。Standley 和 Korf [14]提出了一种动态分组的多机器人无冲突路径规划方法，以降低求解的复杂度。Sharon [15]基于冲突提出了新的一种两级路径寻优算法，在高层对有冲突的机器人进行搜索，在底层对每个机器人进行路径快速寻优，其求解速度较 A*算法高了一个数量级。

通过对文献进行梳理，发现对仓储作业中多个机器人路径规划的研究尚不成熟，关于动态环境下搬运机器人协同作业、动态避障的研究较少，单一的算法难以解决复杂程度渐高的多搬运机器人路径规划问题。并且现有研究中较少将运行时间最短作为优化目标，路径的繁忙程度、搬运机器人的起停、转向次数对于机器人的运行时间也有很大的影响，以距离最短作为目标函数求解路径规划问题难以满足电商企业的实际需求。因此，设计求解多个搬运机器人的动态路径规划算法是当前研究的重点及难点。

本文针对仓储作业中多个搬运机器人的动态路径优化问题，设计了符合电商企业实际作业需求的带有时间窗的 MRPF 数学模型，提出了改进两阶段路径规划算法，将原问题分解为离线路径规划和在线路径协调两个子问题，在离线路径规划中对传统的 Dijkstra 算法进行了改进，考虑了路径繁忙程度以及机器人转向时间，在线路径协调中基于机器人的优先级进行实时调度及冲突避碰。通过仿真实验，与使用传统算法时搬运机器人的总运行时间进行对比，验证了本文算法的有效性。

2. 带有时间窗的多搬运机器人路径规划模型

多搬运机器人的动态路径规划问题实质上是在仓库中的静态障碍物已知的情况下，为每个搬运机器人规划出从起始点到终点无冲突的最优或次优路径，使得多个搬运机器人的行驶路径不发生冲突。并且在搬运机器人行驶过程中可动态调整搬运机器人的行驶路径，从而对未知的动态障碍进行规避。

2.1. 问题描述

与“人到货”的拣选模式相反，“货到人”的拣选模式是指搬运机器人将任务货架移送至拣选台，拣选工人在拣选站将所需的商品从货架中取出即可。“货到人”拣选模式中搬运机器人的作业流程如下：

1) 为搬运机器人分配任务。由于一个搬运机器人不能同时执行多项订单任务，因此将任务分配给距离货架最近的空闲搬运机器人，为其规划自起点至终点的最优行走路径。并根据任务下达的时间为机器人设置优先级，任务下达越早，机器人的优先级越高；

2) 领取任务后的搬运机器人由空闲变为工作状态，自所在位置按照系统规划好的路径行驶至任务货架处，将任务货架顶起，机器人由空载变为负载状态；

3) 带有任务货架的搬运机器人行驶至目标拣选台，等待拣选。当前拣选任务完成后检查是否存在其他拣选任务，若有，移动至指定拣选台等待拣选；否则，根据系统给出的目标位置，为搬运机器人设计回程最优路径，将任务货架运回。搬运机器人此时的状态变为空载、空闲；

4) 判断搬运机器人是否需要充电，如需充电，行驶至充电桩处进行充电，否则释放搬运机器人；

5) 转至步骤 1)，重复执行以上操作，直至无拣选任务。

图 1 是由实际仓库作业环境抽象出的电子栅格地图，栅格地图由拣选台、搬运机器人、可移动货架以及充电站组成。搬运机器人可以在栅格图中沿着东、南、西、北四个方向行驶，栅格中的每一条边都是搬运机器人行驶的车道，一个搬运机器人在运行时只能占用一条边，同一时刻不允许两个搬运机器人在一条边上相向行驶。

为简化模型，对多搬运机器人路径规划问题做出如下假设：

1) 每条边上运行的搬运机器人数量不能超过其能承载的最大数量；

2) 每个搬运机器人在行驶过程中按匀速行驶，机器人遇到障碍时的停车减速时间忽略不计；

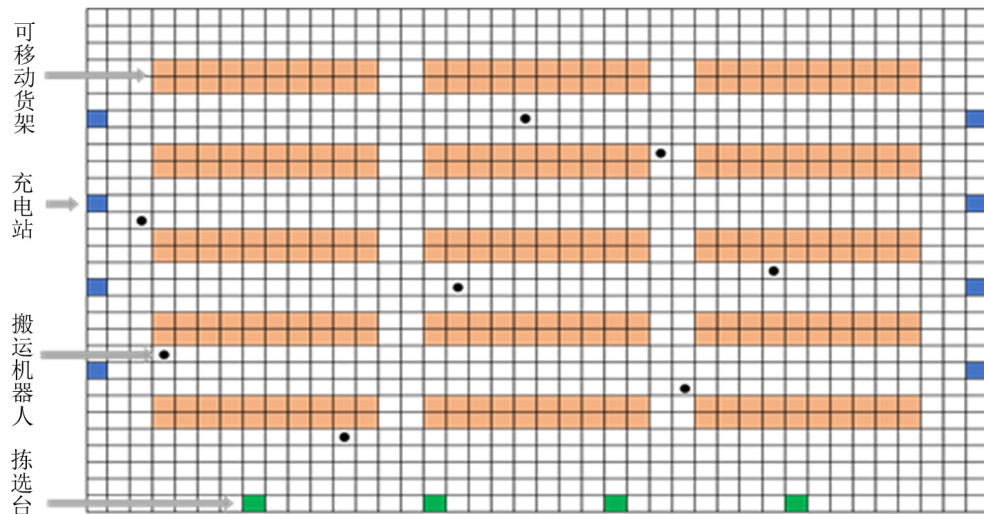


Figure 1. Environment map of warehouse operation

图 1. 仓储作业环境平面图

3) 同向行驶时, 为防止前面的机器人因突然停止(故障)导致两车相撞, 规定机器人在行驶过程中需与前车保持安全距离。本文将机器人的安全车距定为, 即车身长度 + t 秒内车辆的行驶长度;

4) 搬运机器人在运行过程中遇到的随机动态障碍物(人员进入)由搬运机器人的车载传感器和定位系统解决, 模型及算法只面向仓库中的货架障碍以及运行期间搬运机器人之间的冲突。

2.2. 符号说明

1) 相关参数

i, j, k : 路径节点, $i, j, k \in V$

w_{jk} : 边的权重

m_b : 运输任务标识, $b = 1, 2, \dots, p$

h_a : 搬运机器人标识, $a = 1, 2, \dots, q$

t_{jk}^m : 搬运机器人在路径 e_{jk} 上的时间窗

$t_{h_a}^{jk}$: 车辆经过路径 e_{jk} 的时间

$N_{kj}^{t_{jk}^m}$: 路径 e_{jk} 上, t_{jk}^m 时间窗内节点 k 向节点 j 方向运行的搬运机器人数量

$N_{jk}^{t_{jk}^m}$: 路径 e_{jk} 上, t_{jk}^m 时间窗内节点 j 向节点 k 方向运行的搬运机器人数量

$e_{jk} = \begin{cases} 1, & \text{节点 } j \text{ 到节点 } k \text{ 之间的路径存在} \\ 0, & \text{其他} \end{cases}$

$x_{h_a}^{t_{jk}^m} = \begin{cases} 1, & \text{由节点 } j \text{ 向 } k \text{ 行驶的搬运机器人在时间窗 } t_{jk}^m \text{ 内到达} \\ 0, & \text{其他} \end{cases}$

$z_{h_a}^{t_{jk}^m} = \begin{cases} 1, & \text{由节点 } k \text{ 向 } j \text{ 行驶的搬运机器人在时间窗 } t_{jk}^m \text{ 内到达} \\ 0, & \text{其他} \end{cases}$

2) 决策变量

t_{jk}^s : 经过节点 j, k 上的直线路段所耗费的时间

t_{jk}^c : 经过节点 j, k 上的拐弯路段所耗费的时间

t_w : 搬运机器人 h_a 的耗费的额外等待时间。

2.3. 模型建立

2.3.1. 时间窗

在道路有向网络图 $G=(V,E)$ 中, $V=\{1,2,\dots,n-1,n\}$ 表示图中节点的集合, $E=\{e_{12},e_{23},\dots,e_{mn}\}$ 表示两个相邻节点构成的边的集合, $W=\{w_{ab},w_{bc},\dots,w_{mn}\}$ 表示每条边的权重的集合。搬运机器人进入该条边的时间和驶出的时间构成了该条边的时间窗[16]。搬运机器人 h_a 在边 e_{jk} 上的时间窗为:

$$t_{jk}^a = [t_{jk}^{ai}, t_{jk}^{ao}] \quad (1)$$

q 个搬运机器人在边 e_{jk} 上时间窗集合为:

$$t_{jk}^m = \{t_{jk}^1, t_{jk}^2, \dots, t_{jk}^q\} \quad (2)$$

其中, $t_{jk}^a - t_{jk}^{ai} \geq t_{h_a}^{jk}$, 即只能在边 e_{jk} 的空闲时间窗内规划搬运机器人 h_a 的路径。

2.3.2. 带有时间窗的多搬运机器人路径规划模型

由于电商企业对于订单的时效性要求较高, 因此本文将搬运机器人的总运行时间作为优化目标。在满足各个约束条件的情况下, 多搬运机器人路径规划的目标函数为机器人在直行路段上的运行时间与机器人转向的时间, 以及机器人因遇到冲突或路径拥堵而额外耗费的等待时间的总和。即:

$$\min Z = \sum_{a=1}^p h_a (t_{ji}^s * w_{ji} + t_{ji}^c + t_{ik}^s * w_{ik} + t_w) \quad (3)$$

s.t.

$$\sum_{a=1}^p h_a \leq p \quad (4)$$

$$\sum_{a=1}^p \sum_{b=1}^q x_{ab} = q \quad (5)$$

$$\sum_{a=1}^p x_{ab} = 1 \quad (6)$$

$$t_{jk}^m = \{[t_{jk}^{ai}, t_{jk}^{ao}]\} \quad (7)$$

$$N_{jk}^{t_{jk}^m} = e_{jk} * \sum_{i=1}^q x_{h_a}^{t_{jk}^m} \quad (8)$$

$$N_{kj}^{t_{jk}^m} = e_{jk} * \sum_{i=1}^q z_{h_a}^{t_{jk}^m} \quad (9)$$

$$N_{kj}^{t_{jk}^m} + N_{kj}^{t_{jk}^m} \leq \frac{L_{jk}}{L_v} \quad (10)$$

$$N_{kj}^{t_{jk}^m} * N_{jk}^{t_{jk}^m} = 0 \quad (11)$$

$$e_{jk}, x_{h_a}^{t_{jk}^m}, z_{h_a}^{t_{jk}^m} = \{0,1\} \quad (12)$$

式(4)表示执行任务的搬运机器人的车辆数应少于车辆总数; 式(5)表示 a 个搬运机器人共同完成 b 个任务; 式(6)表示每个任务只能由 1 个搬运机器人完成; 式(7)表示路径 e_{jk} 上的时间窗; 式(8)和式(9)表示在时间窗 t_{jk}^m 内, 从节点 j 到节点 k 的路径上运行的搬运机器人数量; 式(10)表示在时间窗 t_{jk}^m 内, 节点 j 到节点 k 的路径上搬运机器人运行的机器人数量应少于路径可容纳的最大数量; 式(11)表示在时间窗 t_{jk}^m 内, 节点 j 到节点 k 的路径上不存在相向行驶的车辆; 式(12)表示搬运机器人 h_a 是否在时间窗 t_{jk}^m 内按时通过路径 e_{jk} 。

3. 带有时间窗的多搬运机器人两阶段动态路径规划算法

与单一的单阶段优化方法比,多阶段优化有利于降低问题的求解难度,提高搜索效率[17]。两阶段算法经常用于车辆的路径规划问题中, Samia M. [18]针对双向单车道上经常出现多个机器人死锁的情况,提出了一种两阶段控制策略。Jung Hoon Lee [19]等人对多机器人之间的冲突进行了研究,并采用两阶段交通控制法实现了多机器人系统的无碰规划。孟祥虎[20]等设计了两阶段算法来增强全局搜索和局部开发能力。本文对传统两阶段路径规划方法进行了改进,在离线阶段基于时间窗原理,在 Dijkstra 算法中加入了路径热度,生成全局时间最优路径;在线阶段依据搬运机器人的优先级进行实时调度,从而规避和解决路径冲突。带有时间窗的多搬运机器人两阶段动态路径规划算法流程如图 2 所示。

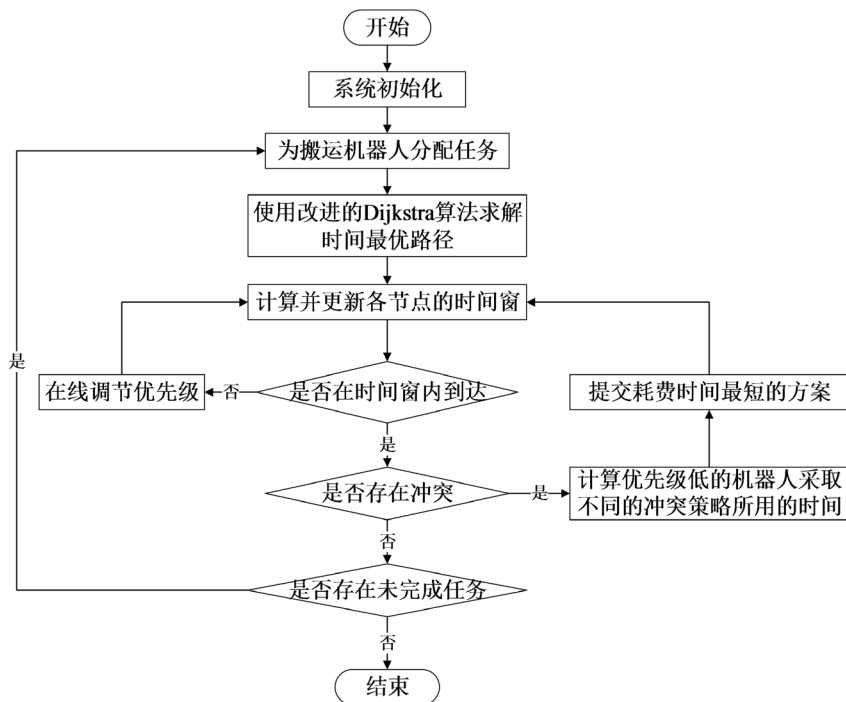


Figure 2. Flowchart of multi-handling robot path planning algorithm
图 2. 多搬运机器人路径规划算法流程图

3.1. 离线阶段 - 带有路径热度的 Dijkstra 算法求解最优路径

为了使路径负载得到平衡,在传统 Dijkstra 算法的路径权重中加入了热度值,对算法进行了改进。现定义在时间周期 Δ 内,各路径的热度为 $fb = \{fb_{12}, fb_{23}, \dots, fb_{mn}\}$,运行过程中每条路径上的搬运机器人的实时数量为 $b_g = \{b_{12}, b_{23}, \dots, b_{mn}\}$, E 为总路径数量,具体计算流程如下:

- 1) 利用传统的 Dijkstra 算法为各个搬运机器人从任务起始点到终点的路径进行初步规划;
- 2) 对 1) 中的路径进行分解,统计时间周期 δ 内各个路段的利用频次 h_k ;
- 3) 当搬运机器人通过当前路径时, b_g 自动减 1;
- 4) 构造路径繁忙程度对 Dijkstra 算法选择路径的影响系数函数:

$$F(fb) = (b_g + 1)h_k / \sum_{g=1}^p \sum_{k=1}^E b_g h_k \quad (13)$$

则加入路径热度后的路径权重值为:

$$w_{jk} = 1 + F(fb) \quad (14)$$

5) 利用 Dijkstra 算法, 根据加权后的路径长度对搬运机器人的路径进行规划, 直至完成所有的任务。

3.2. 在线阶段 - 基于调节优先级的在线交通控制策略

定义参数含义如下:

$v_{h_a}^i$ v 点是搬运机器人 h_a 经过的第 i 个节点

$v_{h_a}^j$ a 车是经过节点 v 的第 j 辆搬运机器人

$A = \{v_{h_a}^1, v_{h_a}^2, \dots, v_{h_a}^n\}$, 表示搬运机器人所经过的节点, $B = \{h_{av}^1, v_{av}^2, \dots, v_{av}^p\}$, 表示经过某节点的搬运机器人车辆, 数组 A 、 B 中的节点在搬运机器人经过某一节点后更新。

1) 车辆 h_a 行驶至节点 v , 检测 v 是否为 h_a 的第 i 个节点, h_a 是否为通过此节点的第 j 辆车;

2) 若节点 v 未被其他搬运机器人占用, h_a 是第 j 个通过该节点的搬运机器人, 即在该点的优先级最高, 则 h_a 继续行驶, 通过节点 v ;

3) 若节点 v 未被其他搬运机器人占用, h_a 不是第 j 个通过该节点的搬运机器人, 即前一辆车 h_{a-1} 未通过此节点, 需判断搬运机器人 h_{a-1} 所处的位置是否处于 h_a 经过节点 v 的下一行驶路段范围外。若是, 执行步骤①; 若否, 执行步骤②:

①调整节点 v 上 h_a 与 h_{a-1} 的优先级, 改变车辆的通过顺序;

②停车等待, 设定等待时长为 T_{w1} , 若在 T_{w1} 范围内 h_{a-1} 通过节点 v , 则 h_a 继续按照原始路径行走, 通过节点 v ; 若在 T_{w1} 范围内 h_{a-1} 未通过节点 v , 则需重新规划 h_a 至终点的路径。若路径存在, 则将其更改为 h_a 的行驶路径, 若不存在, 则继续停车等待处理;

4) 若节点 v 被其他搬运机器人占用, 则停车等待, 设定等待时长为 T_{w2} 。若等待时长超出 T_{w2} , 节点 v 仍处于被占用状态, 则考虑重新规划路径, 若不存在其他可行路径, 也就是说 v 是搬运机器人到达终点的必经节点, 则停车, 状态显示为故障状态。

多搬运机器人在线调节流程见图 3。

3.3. 在线冲突规避

由于道路类型为双向单车道, 在实际行驶的过程中, 两个搬运机器人在同一条路段上朝着相反的方向行驶, 容易出现机器人死锁、相互碰撞的现象, 发生相向冲突。根据冲突特点, 本文提出了在线相向冲突的解决策略。解决策略融合了基于机器人优先级的等待策略及二次规划路径策略, 在线调节冲突, 有效地节省了搬运机器人因冲突而额外耗费的时间。具体解决策略如下:

①若两辆相向行驶的搬运机器人均按原始路径匀速行驶时, 存在着相向冲突, 那么判断两个搬运机器人的优先级。优先级高的搬运机器人正常通过, 另一辆车则进行在线调节, 判断等待、二次规划路径所需的时间, 选择耗时较少的策略。

②若两辆相向行驶的搬运机器人中存在着一辆搬运机器人在停车等待, 存在着相向冲突, 那么判断两个搬运机器人的优先级。若暂停的搬运机器人优先级最高, 那么继续等待, 另一搬运机器人进行在线调节, 判断等待、二次规划路径所需的时间, 选择耗时较少的策略; 若行驶的搬运机器人优先级最高, 则搬运机器人正常通过, 暂停等待的搬运机器人行驶至下一空闲节点执行等待命令。

③若两辆相向行驶的搬运机器人中存在着一辆搬运机器人在执行拣选操作, 存在着相向冲突, 那么搬运机器人继续执行拣选工作, 另一搬运机器人选择进行在线调节策略。在冲突解决策略中, 执行拣选工作的搬运机器人优先级最高。

相向冲突解决策略如图 4 所示:

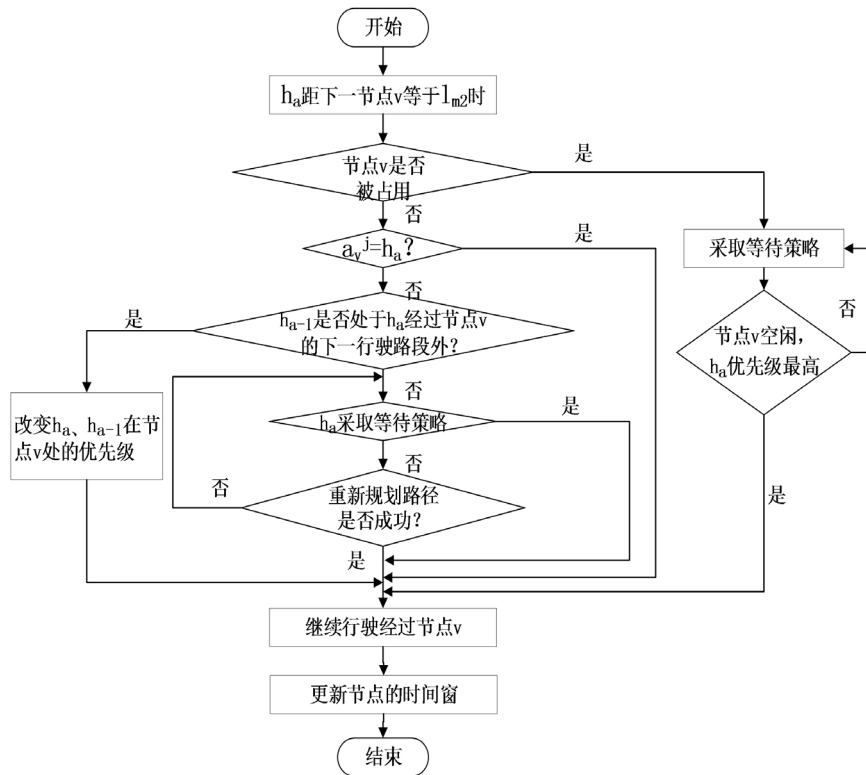


Figure 3. The flow chart of online adjust the priority strategy

图 3. 在线调节优先级策略流程图

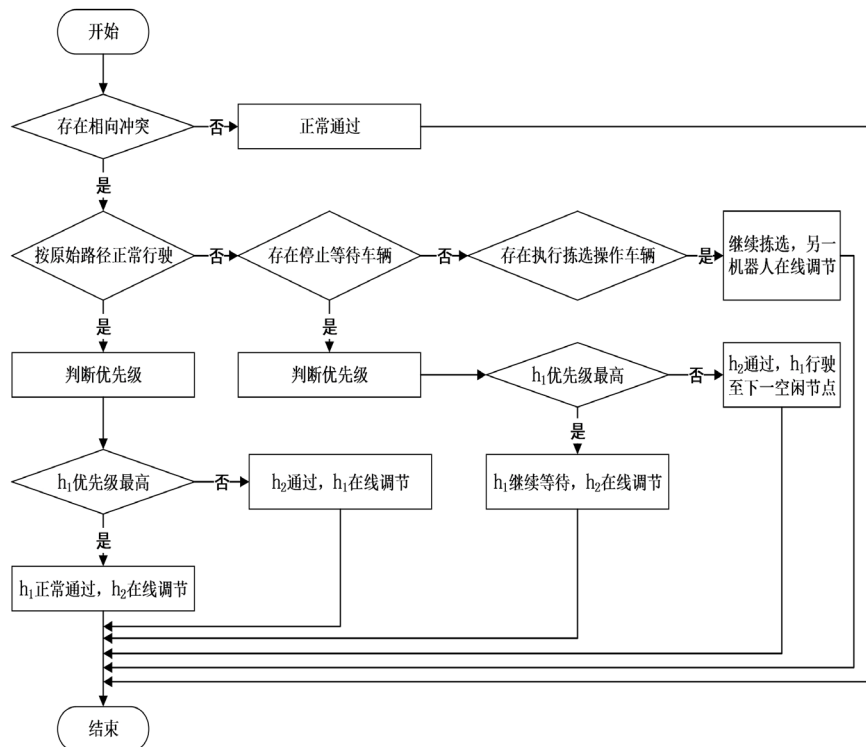


Figure 4. Strategy for resolving conflicts in the opposite direction

图 4. 解决相向冲突

4. 仿真实例

采用 Matlab 软件在 40 * 30 的栅格地图中设计了三组对比实验。第一组实验对比了使用传统 Dijkstra 算法与改进的 Dijkstra 算法时搬运机器人的总行驶距离及运行时间；第二组实验对比了在不同的任务规模下，使用传统 Dijkstra 算法与改进的 Dijkstra 算法时搬运机器人的总行驶距离及运行时间；第三组实验对比了在运行过程中，搬运机器人遇到相向冲突时，采取不同在线解决策略所耗费的时间。三组实验分别对离线阶段和在线阶段的动态路径规划算法的有效性进行了验证。

4.1. 第一组对比实验

在考虑路径繁忙程度及转弯时间的基础上，为搬运机器人设定 4 个任务，对比使用传统 Dijkstra 算法与改进的 Dijkstra 算法时搬运机器人的总行驶距离及运行时间。

搬运机器人任务参数如表 1 所示。

Table 1. Task parameter settings of the handling robot system

表 1. 搬运机器人系统任务参数设置

序号	优先级	起始坐标	结束坐标
1	1	(10,18)	(8,1)
2	2	(21,10)	(16,1)
3	3	(19,23)	(24,1)
2	4	(29,8)	(32,1)

在离线状态下，使用传统 Dijkstra 算法和改进 Dijkstra 算法规划 4 个搬运机器人的行走路径如图 5、图 6 所示。

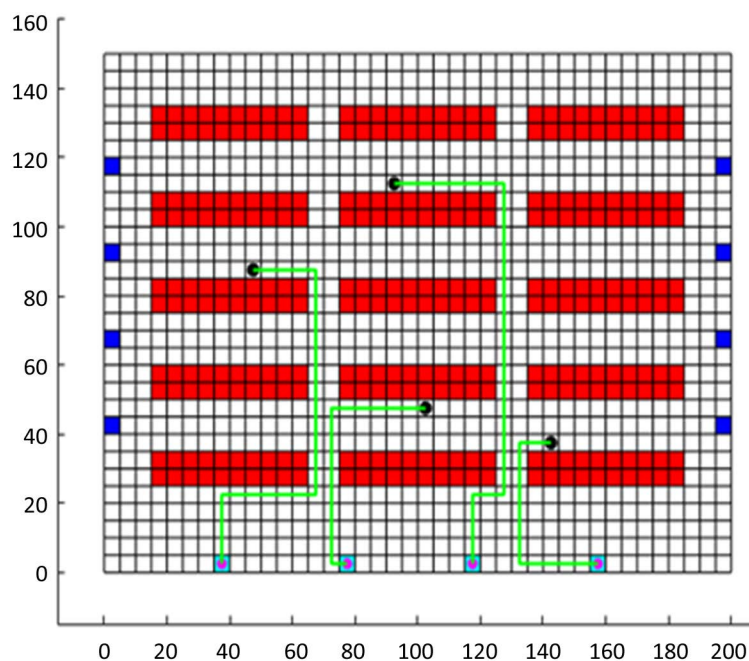


Figure 5. Traditional Dijkstra algorithm for path planning

图 5. 传统 Dijkstra 算法规划路径图

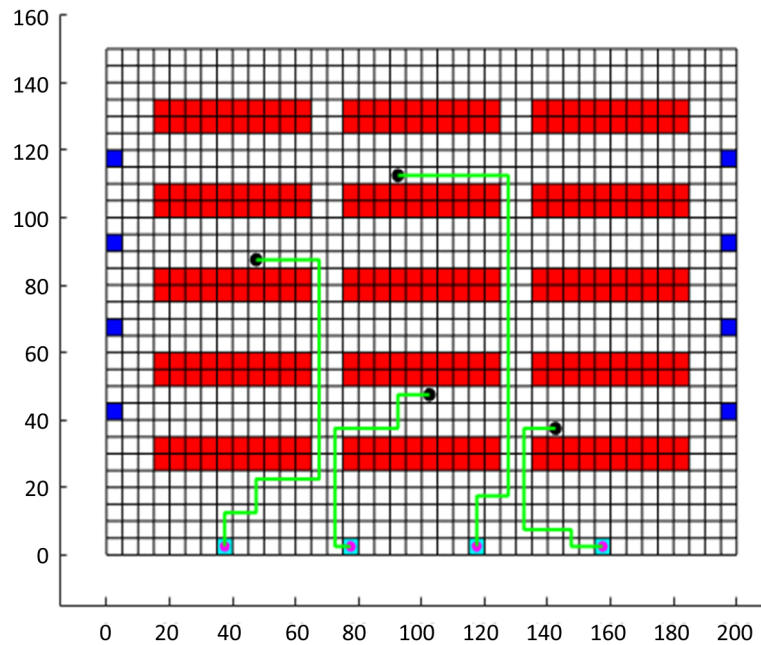


Figure 6. Improved Dijkstra algorithm for path planning
图 6. 改进 Dijkstra 算法规划路径图

4 个任务的总运行时间和总运行距离对比见图 7。实验结果表明,若综合考虑各条路径的热度以转向所需时间后,使用改进的 Dijkstra 算法求解的路径,搬运机器人的总运行距离比传统算法节约了 14.45 米;总运行时间节省了 8.45 秒,搬运机器人的效率与传统算法相比有明显的提升,体现了改进算法的有效性。

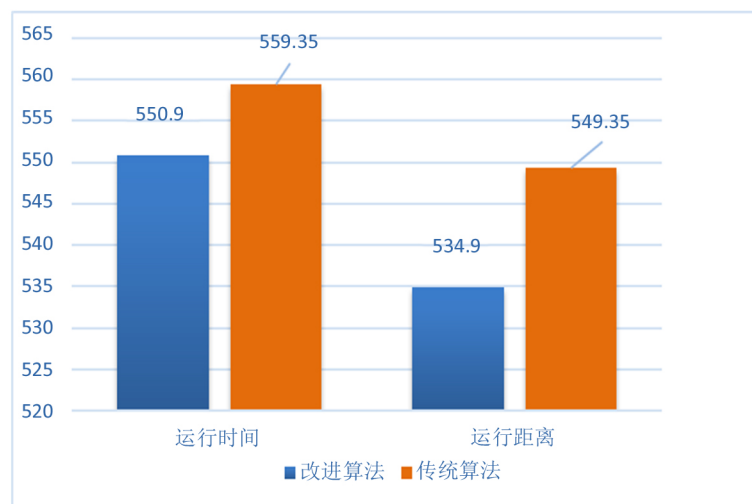


Figure 7. Comparison of results before and after algorithm improvement
图 7. 算法改进前后运算结果对比

4.2. 第二组对比实验

本组以搬运机器人的总运行时间、总行驶距离为评价标准,对比在不同任务规模的情况下,改进后的多搬运机器人路径规划算法的稳定性。搬运机器人的任务参数如表 2 所示。

Table 2. Task parameter settings of the handling robot system
表 2. 搬运机器人系统任务参数设置

序号	优先级	起始坐标	结束坐标
1	1	(10,18)	(8,1)
2	2	(21,10)	(16,1)
3	3	(19,23)	(24,1)
4	4	(29,8)	(32,1)
5	5	(32,25)	(17,13)
6	6	(8,23)	(11,13)
7	7	(24,1)	(16,15)
8	8	(29,25)	(25,8)
9	9	(7,5)	(29,5)
10	10	(38,27)	(32,1)
11	11	(6,15)	(19,8)
12	12	(18,8)	(30,15)

不同算例规模下基于传统 Dijkstra 算法和改进 Dijkstra 算法的路径规划输出结果如表 3、表 4 所示。

Table 3. Path planning results based on traditional Dijkstra algorithm under different example sizes
表 3. 不同算例规模下基于传统 Dijkstra 算法的路径规划输出结果

任务规模	总行驶距离	总运行时间	机器人平均行驶距离	机器人平均行驶时间
4	549.35	559.35	137.34	139.84
8	1107.71	1131.72	138.46	141.47
12	1692.41	1708.42	141.03	142.37

Table 4. Path planning results based on improved Dijkstra algorithm under different example sizes
表 4. 不同算例规模下基于改进 Dijkstra 算法的路径规划输出结果

任务规模	总行驶距离	总运行时间	机器人平均行驶距离	机器人平均行驶时间
4	534.9	550.9	133.73	137.73
8	1085.24	1119.45	135.66	139.93
12	1650.87	1697.78	137.57	141.48

通过仿真结果绘制在不同任务规模下，改进算法前后每辆搬运机器人的平均运行距离和平均运行时间对比图，对比图如图 8、图 9 所示。

由图 8、图 9 可看出，随着任务的增加，搬运机器人的总运行距离和总运行时间均有所增加。但是使用改进后的 Dijkstra 算法得出的每辆搬运机器人的平均运行时间和平均运行距离均小于传统算法的计算结果，体现出算法的有效性与稳定性。

4.3. 第三组对比实验

本组以搬运机器人在遇到相向冲突时，对比采取不同在线解决策略时多搬运机器人所耗费的时间。两个任务的参数见表 5。搬运机器人的仿真运行界面如图 10 所示。

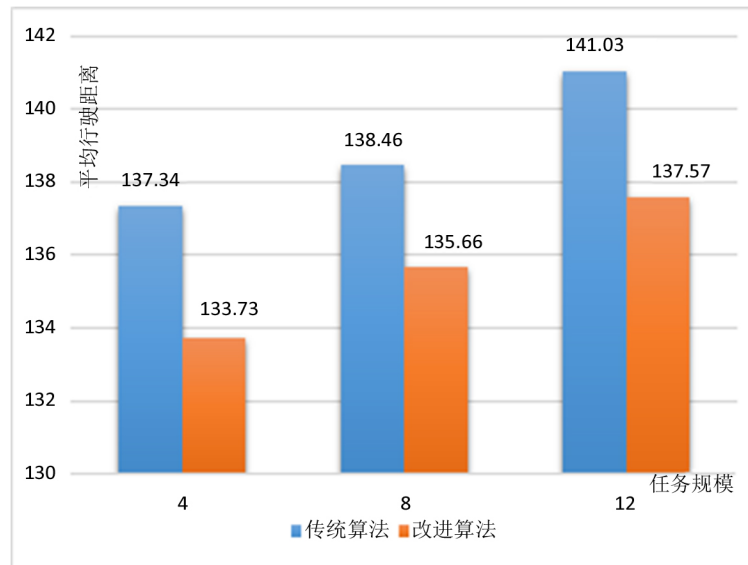


Figure 8. Average distance comparison of the handling robots
图 8. 搬运机器人平均行驶距离对比图

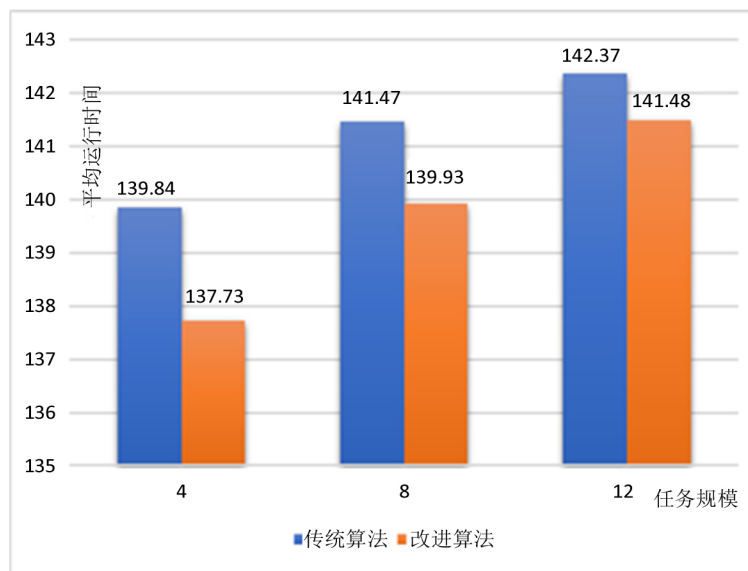


Figure 9. Average running time comparison of the handling robots
图 9. 搬运机器人平均运行时间对比图

Table 5. Task parameter settings of the handling robots
表 5. 搬运机器人任务参数设置

车辆	优先级	起始坐标	结束坐标	通过路径顺序
h_1	1	(16,1)	(11,10)	(16,1)→(16,2)→(14,2)→(14,10)→(11,10)
h_2	2	(13,15)	(8,1)	(13,15)→(13,13)→(14,13)→(14,5)→(10,5)→(10,3)→(8,3)→(8,1)

由以上信息我们可知，在运行过程中搬运机器人与在(14,8)→(14,6)的路段时间窗有重叠部分，会产生相向冲突。由于的优先级低于，在此种情境下，我们对比采取单一的等待策略和基于在线调节策略所需要的总运行时间，如表 6 所示。

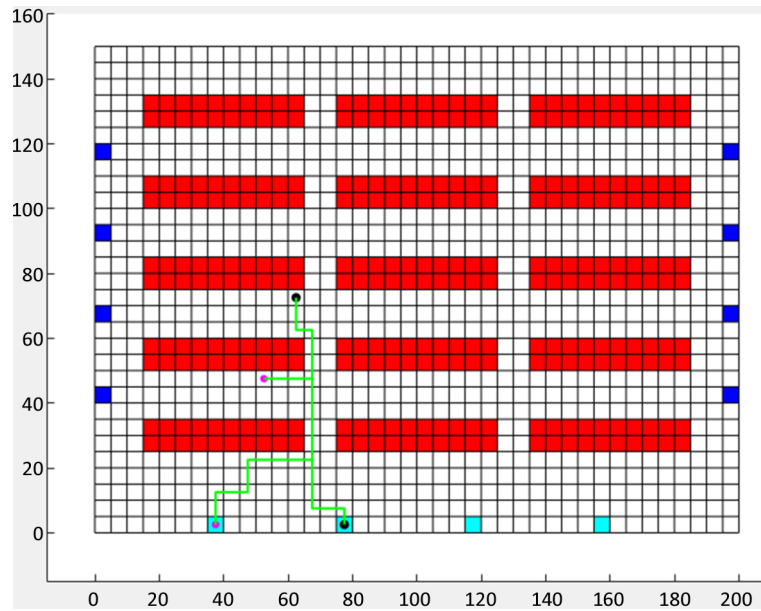


Figure 10. Simulation path diagram of handling robots
图 10. 搬运机器人仿真路径图

Table 6. Simulation results of handling robot under different strategies

表 6. 不同策略下搬运机器人的仿真输出结果

策略选择	路径通过次序	总运行时间	总运行距离
无相向冲突	(13,15)→(13,13)→(14,13)→(14,5)→(10,5)→(10,3)→(8,3)→(8,1)	132.39	126.39
采取单一等待策略	(13,15)→(13,13)→(14,13)→(14,5)→(10,5)→(10,3)→(8,3)→(8,1)	162.86	126.39
在线调节策略	(13,15)→(13,13)→(14,13)→(14,8)→(15,8)→(15,6)→(14,6)→(14,5)→(10,5)→(10,3)→(8,3)→(8,1)	146.25	142.55

当搬运机器人 h_1 和 h_2 存在相向冲突时, h_2 若在节点(14,8)处直接选择等待 h_1 经过后再按照原来的路径行驶, h_2 的运行时间为 162.86 秒, 此时 h_2 需要等待 30 秒的时间; 若 h_2 依据在线调节的策略, 重新规划行驶路径时, 只需比无冲突的情况多耗费 14 秒, 即可规避相向冲突。

分析可得, 采取在线调节策略中重新规划路径的方法对于调节 h_1 和 h_2 之间存在的相向冲突所得到的结果较优, 比直接停车等待的策略节约 16 秒。调节后的搬运机器人的路径如图 11 所示。

在第一组对比实验中我们分别采用传统算法和改进的 Dijkstra 算法为对 4 个搬运机器人进行离线路径规划, 实验结果表明了改进后的 Dijkstra 算法在总运行时间和运行距离上比传统算法分别节省了 14.45 米、8.45 秒; 在第二组对比实验中我们分别采用传统算法和改进的 Dijkstra 算法对不同数量规模的搬运机器人进行离线路径规划, 实验结果表明, 数量规模的增长对于改进的算法产生的影响不大, 在总运行时间和运行距离上依旧有所减少; 在第三组对比实验中我们对任务产生的在线路径冲突进行决策, 研究结果表明, 单一的冲突解决方式不一定是最优结果, 可能是等待策略最优, 也可能是二次规划路径的策略最优, 需要具体问题具体分析。通过计算产生冲突的时间窗, 实行在线调节冲突的策略不仅可以规避搬运机器人之间的冲突, 还可以节约规避冲突所耗费的时间, 验证了算法的有效性。

5. 结论

针对多搬运机器人的动态路径优化问题, 考虑了机器人的转向时间和路径的繁忙程度, 建立了带有

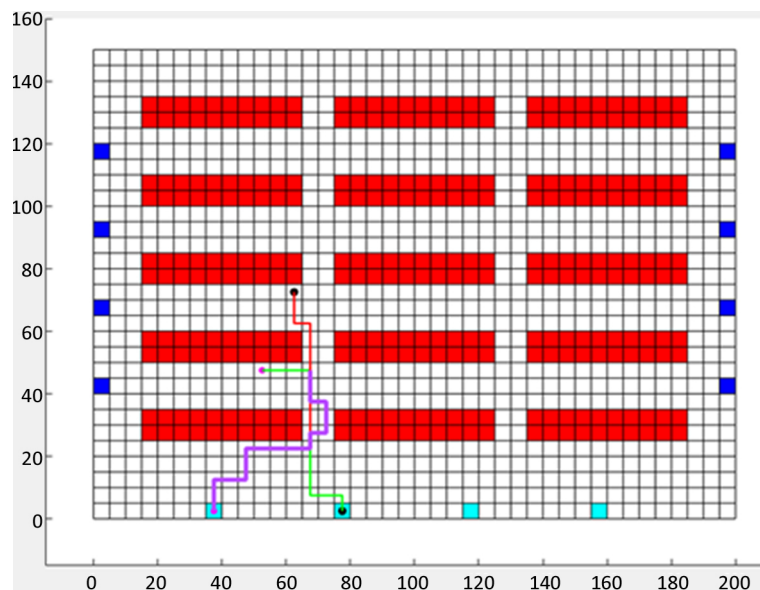


Figure 11. Path diagram for online adjustment strategy

图 11. 采取在线调节策略的路径图

时间窗的多搬运机器人路径规划模型，设计了带有时间窗的改进两阶段动态路径规划算法。在离线状态下使用改进的 Dijkstra 算法规划全局时间最优路径，在线状态下基于机器人的优先级对搬运机器人进行实时调度，并为可能存在的相向冲突提供了相应的解决策略。解决方案融合了车辆等待、二次规划路径以及基于优先级调整搬运机器人通行次序等方法，有效地节省了搬运机器人因冲突而额外耗费的等待时间。

通过仿真分析，提出的改进算法对搬运机器人的总运行时间有着明显的缩短，增强了搬运机器人路径的柔性，为仓储作业中多搬运机器人动态路径优化问题提供了解决方案。但是当机器人的数量增大到一定程度时，算法的响应时间和搜索速度则会变慢。因此，对大规模的搬运机器人进行动态路径优化，是未来的研究方向之一。

参考文献

- [1] 胡国盛. 我国电商企业物流体系优化分析[J]. 商业经济研究, 2017(19): 106-108.
- [2] Hazard, C.J. and Wurman, P.R. (2006) Alphabet Soup: A Testbed for Studying Resource Allocation in Multi-Vehicle Systems. *Proceedings of the 2006 AAAI Workshop on Auction Mechanisms for Robot Coordination*, Boston, 17 July 2006, 23-30.
- [3] Li, W., Jiang, L. and Li, Z. (2016) Research on the Task Assignment Problem of Warehouse Robots in the Smart Warehouse. *12th International Symposium on Operations Research and Its Applications in Engineering, Technology and Management*, Luoyang, 21-24 August 2015, 29-33.
- [4] Yeung, W.-K., Choi, T.M. and Cheng, T.C.E. (2011) Supply Chain Scheduling and Coordination with Dual Delivery Modes and Inventory Storage Cost. *International Journal of Production Economics*, **132**, 223-229. <https://doi.org/10.1016/j.ijpe.2011.04.012>
- [5] Khorshid, M.M., Holte, R.C. and Sturtevant, N.R. (2013) A Polynomial-Time Algorithm for Non-Optimal Multi-Agent Pathfinding. *Symposium on Combinatorial Search. Proceedings of the 4th Annual Symposium on Combinatorial Search*, Castell de Cardona, Barcelona, 15-16 July 2011, 76-83.
- [6] Chang, W.K. and Tanchoco, J.M.A. (1991) Journal of Production Research Conflict-Free Shortest Time Bidirectional AGV Routing. *International Journal of Production Research*, **29**, 2377-2391. <https://doi.org/10.1080/00207549108948090>
- [7] Ma, Y., Wang, H., Xie, Y. and Guo, M. (2014) Path Planning for Multiple Mobile Robots under Double-Warehouse. *Information Sciences*, **278**, 357-379. <https://doi.org/10.1016/j.ins.2014.03.058>

- [8] Umar, U.A., Ariffin, M.K.A. and Ismail, N. (2012) Priority-Based Genetic Algorithm for Conflict-Free Automated Guided Vehicle Routing. *Journal of Engineering Valuation and Cost Analysis*, **50**, 732-739.
- [9] Desrochers, M., Desrosiers, J. and Solomon, M. (1992) A New Optimization Algorithm for the Vehicle Routing Problem with Time windows. *Operation Research*, **40**, 199-415. <https://doi.org/10.1287/opre.40.2.342>
- [10] Wang, W. and Goh, W.B. (2015) A Stochastic Algorithm for Makespan Minimized Multi-Agent Path Planning in Discrete Space. *Applied Soft Computing*, **30**, 287-304. <https://doi.org/10.1016/j.asoc.2015.01.046>
- [11] Hidalgo-Paniagua, A., Miguel Vega-Rodríguez, A., Ferruz, J. and Pavón, N. (2015) MOSFLA-MRPP: Multi-Objective Shuffled Frog-Leaping Algorithm Applied to Mobile Robot Path Planning. *Engineering Applications of Artificial Intelligence*, **44**, 123-136. <https://doi.org/10.1016/j.engappai.2015.05.011>
- [12] Ma, H., Koenig, S., Ayanian, N., et al. (2017) Overview: Generalizations of Multi-Agent Path Finding to Real-World Scenarios. *Computer Science*, arXiv: 1702.05515.
- [13] Miao, H. and Tian, Y.C. (2013) Dynamic Robot Path Planning Using an Enhanced Simulated Annealing Approach. *Applied Mathematics and Computation*, **222**, 420-437. <https://doi.org/10.1016/j.amc.2013.07.022>
- [14] Standley, T.S. and Korf, R.E. (2011) Complete Algorithms for Cooperative Pathfinding Problems. International Joint Conference on Artificial Intelligence. *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, Barcelona, 16-22 July 2011, 223-229.
- [15] Sharon, G., Stern, R., Felner, A. and Sturtevant, N.R. (2015) Conflict-Based Search for Optimal Multi-Agent Pathfinding. *Artificial Intelligence*, **219**, 40-66. <https://doi.org/10.1016/j.artint.2014.11.006>
- [16] 何小锋, 马良. 带时间窗车辆路径问题的量子蚁群算法[J]. 系统工程理论与实践, 2013, 33(5): 1255-1261.
- [17] 张立峰, 易万里, 刘晓兰. 基于两阶段算法的大规模成品油二次配送优化[J]. 系统工程理论与实践, 2016, 36(11): 2951-2963.
- [18] Maza, S. and Castagna, P. (2005) A Performance-Based Structural Policy for Conflict-Free Routing of Bidirectional Automated Guided Vehicles. *Computers in Industry*, **56**, 719-733. <https://doi.org/10.1016/j.compind.2005.03.003>
- [19] Lee, J.H., Lee, B.H. and Choi, M.H. (1998) A Real-Time Traffic Control Scheme of Multiple AGV Systems for Free Minimum Time Motion: A Routing Table Approach. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, **28**, 347-358. <https://doi.org/10.1109/3468.668966>
- [20] 孟祥虎, 胡蓉, 钱斌. 求解带时间窗车辆路径问题的有效混合 PBIL 算法[J]. 系统工程理论与实践, 2014, 34(10): 2701-2709.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2326-3415, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>
期刊邮箱: airr@hanspub.org