软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(9):2770−2784 [doi: 10.13328/j.cnki.jos.005944]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

智能仓储货位规划与 AGV 路径规划协同优化算法*

蔺一帅[1], 李青山[1], 陆鹏浩[1], 孙雨楠[1], 王 亮[2], 王颖芝[1]

[1](西安电子科技大学 计算机科学与技术学院,陕西 西安 710071)

[2](苏州明逸智库信息技术有限公司,江苏 昆山 215300)

通讯作者: 李青山, E-mail: qshli@mail.xidian.edu.cn

摘 要: 智能仓储的优化一般分为货架优化和路径优化两部分:货架优化针对货物与货架两者的关系,对货物摆放位置进行优化;而路径优化主要寻找自动引导小车(automated guided vehicle,简称 AGV)的最优路径.目前,大多的智能仓储优化仅对这两部分进行独立研究.在实际仓储应用中,只能以线性叠加的方式解决问题,导致问题的求解易陷入局部最优中.通过对智能仓储环节中各部分的关系进行耦合分析,提出了货位和 AGV 路径协同优化数学模型,将货架优化和路径规划归为一个整体;此外,提出了智能仓储协同优化框架的求解算法,包括货品相似度求解算法和改进的路径规划算法;并在以上两种算法的基础上,使用改进的遗传算法实现了货位路径协同优化.实验结果验证了所提出的智能仓储协同优化算法的有效性和稳定性.通过使用该算法,可有效提高仓储的出货效率,降低运输成本.

关键词: 智能仓储;货位规划;AGV 路径规划;协同优化;遗传算法

中图法分类号: TP18

中文引用格式: 蔺一帅,李青山,陆鹏浩,孙雨楠,王亮,王颖芝.智能仓储货位规划与 AGV 路径规划协同优化算法.软件学报, 2020,31(9):2770−2784. http://www.jos.org.cn/1000-9825/5944.htm

英文引用格式: Lin YS, Li QS, Lu PH, Sun YN, Wang L, Wang YZ. Shelf and AGV path cooperative optimization algorithm used in intelligent warehousing. Ruan Jian Xue Bao/Journal of Software, 2020,31(9):2770−2784 (in Chinese). http://www.jos.org.cn/1000-9825/5944.htm

Shelf and AGV Path Cooperative Optimization Algorithm Used in Intelligent Warehousing

LIN Yi-Shuai[1], LI Qing-Shan[1], LU Peng-Hao[1], SUN Yu-Nan[1], WANG Liang[2], WANG Ying-Zhi[1]

[1](School of Computer Science and Technology, Xidian University, Xi’an 710071, China)

[2](Suzhou Mingyi Intelligent Storage Information Company, Kunshan 215300, China)

Abstract: The optimization of intelligent warehousing is generally divided into shelf optimization and path optimization. Shelf optimization considers the position of goods and shelves, and optimizes the placement of goods. Path optimization mainly seeks the optimal path planning for automatic guided vehicles. At present, most of the studies focus on these two scenarios independently. In the actual warehousing application, the problem can only be solved by linear superposition, which makes the solution easy to fall into the local optimum. Based on the coupling analysis of the relationship between various sections in the intelligent warehousing process, this study proposes a mathematical model of cooperative optimization of shelf and position, which combines shelf optimization and path planning as a whole. In addition, a cooperative optimization framework, including a product similarity solving algorithm and an improved path planning algorithm, is proposed. Based on the above two algorithms, an improved genetic algorithm is proposed for the cooperative optimization of shelf and path. The experimental results verify the effectiveness and stability of the intelligent warehousing cooperative

* 基金项目: 国家自然科学基金(61672401, 61902039, 61902288); 西安市科技计划(2017073CG/RC036(XDKD004))

Foundation item: National Natural Science Foundation of China (61672401, 61902039, 61902288); Xi’an Science and Technology Program (2017073CG/RC036(XDKD004))

本文由“智能嵌入式系统”专题特约编辑王泉教授、吴中海教授、陈仪香教授、苗启广教授推荐.

收稿时间: 2019-07-02; 修改时间: 2019-08-18; 采用时间: 2019-11-02; jos 在线出版时间: 2020-01-13

CNKI 网络优先出版: 2020-01-14 11:27:19, http://kns.cnki.net/kcms/detail/11.2560.TP.20200114.1126.026.html

optimization algorithm proposed in this study. By using this algorithm, it can improve the shipping efficiency of storage and reduce transportation costs..

Key words: intelligent warehousing; shelf optimization; AGV path optimization; cooperative optimization; genetic algorithm

智能仓储系统是由立体货架、有轨巷道堆垛机、出入库输送系统、信息识别系统、自动控制系统、计算机监控系统、计算机管理系统等其他辅助设备组成的智能化系统.智能仓储的高度信息化、自动化,使得控制优化算法成为了智能仓储的灵魂所在[1].目前,智能仓储优化算法主要集中在货架优化和 AGV 车路径规划这两部分,且关于这两部分的研究都是各自独立的.货架优化是针对货物与货架两者的关系,对货物摆放位置的优化,其重点关注于总体出货代价、货架稳定性等因素[2].而路径优化主要针对于自动化运载车辆的路径规划的优化,要关注的重点在于完成任务的时间开销最小、避免车辆之间发生碰撞等.然而,基于对实际仓储的运维数据分析,我们发现传统的独立的智能仓储优化策略,即研究货架优化时只专注于货架摆放最优结果,研究路径规划时只关注于算法的执行效率和优化成果,完全忽视了货架规划和路径规划两者间的耦合关系,严重影响了智能仓储的出货效率最大化[3–5].因此,本文提出了基于遗传算法的货位规划与 AGV 路径规划协同优化算法(以下简称货位路径协同优化算法).该协同优化算法改进了经典遗传算法,其创新点在于将货位规划和路径规划协同考虑后,构造的适应性函数能够将相似度高的货架分散放置,将高频出库货物放置在易于出货的位置上,从而实现在大量同时出库任务到来时 AGV 调度不堵塞,从整体上提高仓库出货效率.

本文主要创新贡献如下:

(1) 本文通过对智能仓储环节中各部分的关系进行耦合分析,进而提出货位和 AGV 路径协同优化的数学模型.该模型与传统智能仓储优化算法的区别在于:将货架优化和路径规划归为一个整体,并用数学公式表达两者间的关系;

(2) 提出了智能仓储协同优化的求解算法,其中包括有货品相似度求解算法、改进的路径规划算法.在以上两种算法的基础上,使用改进适应性函数的遗传算法,实现了货位路径协同优化.同时,通过实验验证了该协同优化算法在不同场景下的表现情况.

1 国内外研究现状

本文研究的智能仓储货位与路径协同优化算法相关的公开文献资料很少,大多都是以货架优化和 AGV 路径规划独立研究为主.本文将分别就这两类问题分析国内外研究现状以及存在的相关问题.除此之外,对本文提出的协同优化算法的协同优化思想起源以及应用实例给予简要介绍.

1.1 智能仓储货架优化研究现状

货架(货位)优化研究是智能仓储领域的重要研究方向.货架优化就是计算货物摆放位置的算法,货架优化的目标是在保证仓储环境稳定的条件下提高仓库的出入库效率,同时优化货架空间,使仓储放入货品的数量最大化.

由 Cai 等人[6]针对仓储策略部分提出了基于存储的出入库频率和负载平衡的绝对的主通道存储策略与平均和最大值相关的控制策略,提升了库位使用率.Wang 等人[7]对传统的遗传算法进行改进,研究出基于遗传算法的库位分配方法.该方法以货品位置坐标为基因,使用权重和的方式来统一多目标函数为一个函数,使其成为适应性函数.通过基因遗传运算,得出优化后的库位摆放方案.刘峰等人[8]使用模糊化这一理念形成模糊算法,对货品的出货率、重量进行模糊化.但该优化算法每次只计算单个货品的位置摆放,对于一次性全部入库的运算会变得十分复杂.同时,模糊化方式存在使用精确度来换取计算时间的弊端.杨玮等人[9]基于混合粒子群算法,采用多色集合概念对货位进行分区,对不同对象使用相同形式的数学模型进行仿真.该算法的优点在于没有交叉变异,需要调整的参数较少.其缺点在于因缺乏动态调节,存在陷入局部最优的可能.

Tinelli 等人[10]使用多标准工具来优化智能仓储货位分配,应用层次分析法,以目标按照一定方式分成不同的标准,标准间相互组合形成不同的解,即对不同的目标函数进行求解并组合.Arnaout 等人[11]用蠕虫优化来解

决多层仓库库位布局问题.蠕虫类似于群体智能算法,使用连接图的方式来表现,通过迭代、限界来搜寻最优解.在 Arnaout 等人描绘的特定仓储算例中,与基本遗传算法相比,基于蠕虫算法的仓储布局算法计算时间较短.

基于对上述国内外研究现状的分析,可以看出,库位优化问题是一个 NP 难问题,目标函数较多,目前解决方法主要是来自于自启发算法的各类变种算法,这也是解决 NP 难问题的常规方式.但是,多数方法的优化思路都集中在根据物品和货架的关系来进行货位优化,忽略了出货路径对货位摆放优化的影响.

1.2 自动导引车路径规划研究现状

AGV(automated guided vehicle,自动导引车)是智能仓储的重要组成部分,在仓储系统中,主要扮演智能物流搬运的角色,配合任务分配和调度系统实现物品的智能配送、周转、出入库等操作[12].现阶段,针对 AGV 的优化算法主要分为两个方面:一方面是对路径进行优化,另一方面是针对 AGV 的任务调度进行优化.虽然侧重点不同,但其目标都在于提高 AGV 的使用效率.

Zhang 等人[13]在 A*算法的基础上加入 AGV 转弯次数影响因素,通过减少 AGV 的转弯次数,降低 AGV 转弯的时间消耗,有效地提高了 AGV 的使用效率.但算法忽略了 AGV 的冲突问题.姜康等人[14]研究了改进的遗传算法,在基因片段里,以三元为一组,整个任务序列为全部任务.对此基因段进行交叉、变异,通过适应度函数进行筛选.由于该算法每次都会计算全部的任务序列,频繁的新任务添加会大大增加此算法的计算次数.同时,该方法也忽略了小车的碰撞以及充电对调度带来的影响.张素云等人[15]在解决 AGV 冲突方面展开研究,提出了通过加减速控制 AGV 的碰撞避免方法.该方法先预估哪些节点会产生碰撞,在该节点部分使用加减速控制,以此来避免冲撞.在实际运行中,该算法需进一步考虑 AGV 承重、重心高低等因素对加减速幅度的影响.

Bilge 等人[16]使用时间窗法解决行驶冲突,对所有 AGV 的行动进行预估,可以预知所有车辆在不同时段所占用的不同路段,以此避免 AGV 车辆的碰撞.Mousavi 等人[17]采用混合遗传算法和粒子群算法优化多目标 AGV 调度.该算法在初期选用粒子群算法,在粒子进行移动后,更新粒子位置找到最佳粒子位置,再使用遗传算法对粒子的位置进行再优化,而后再回到粒子群算法中进行下一次粒子移动计算,直到收敛.此算法在收敛速度方面与遗传算法相近,最终结果优于单独使用遗传算法或粒子群算法.其缺点在于混合算法计算耗时较长.

上述国内外研究现状为本文的研究提供了很多值得借鉴的思路,比如遗传算法中的编码方式、A^*算法中关于 AGV 转弯会影响效率的提示、时间窗法通过静态规划路径来避免冲突等.同时,分析当前国内外的研究现状不难发现:现有 AGV 的优化算法中,对冲突问题的时间代价没有充分考虑.更重要的是,缺少将货架优化和路径规划联系起来进行统一优化的方法.这种忽略货架和路径的内在联系、两者分开优化、仅将两部分各自的最优解的串行组合的方式,很难找到一个全局最优解.

1.3 协同进化算法研究现状

协同进化算法框架的形成较早,是 Hillis 从自然界捕食、竞争以及共生关系得到启发,于 1990 年将这种协同的思想引入到进化算法中[18].Potter 将协同进化算法进行了进一步的研究,提出了合作式协同进化算法[19].此算法主要提出了一个算法框架,将完整的系统根据一定规则分为子系统,以“分而治之”的思想对子系统分别求解,将求解结果与其他子系统互动,达到协同进化的目的.

目前,协同算法在国内主要运用于空间布置,如 Wang 等人[20]、Huo 等人[21]在卫星舱的布局问题上使用协同进化的思想,配合相对应的算法,较好地解决了空间布局问题.Wang 等人在协同进化的基础上加入散射搜索法,更加贴合卫星舱的特点.而 Huo 等人则通过使用协同进化遗传算法,取得较好的布局优化结果.梁静等人[22]则通过协同进化的思想,提出使用粒子群算法来解决高维度问题.

比起具体的解决方案,协同进化更多的是一种解决问题的思想.上述的研究者在此思想上,结合适合的算法用来解决不同种类的问题,都取得一定成果.因此,结合智能仓储的特点,我们研究团队尝试将该思想引入仓储领域,运用协同进化的思想来对货架优化和路径规划进行集成研究[23].

综合分析上述国内外研究现状,可以看出:关于货架优化和 AGV 路径规划的相关解决方案和算法的独立研究都有重大进展,而对两问题的集成研究极少.大多数在进行货架优化时根本不考虑路径问题,忽略路径对最终

仓储出库效率的影响.举个简单例子:最优的货架优化可能会带来出货时的 AGV 拥堵,给 AGV 路径规划带来严重问题,从而影响到仓储出库效率.针对这种现象,本文将对 AGV 路径和货架优化进行集成研究,使用协同优化的思想将货物、货架、路径三者结合起来,使得在进行货架优化时,除了考虑到货物,也将路径纳入到影响因素中,从而提高仓储的整体效率,进一步降低仓储成本.

2 货位规划与 AGV 路径规划协同优化算法

现有智能仓储的主要优化方法为采用传统货架优化算法进行货位计算后,执行路径规划算法.这种优化方法容易导致一定区域内的车辆堵塞问题,大大降低了出货效率.针对这个问题,本文基于协同进化思想的启发,分析货物、货架、AGV 车路径规划的相互影响关系,提出了货架规划和路径规划协同进化算法,实现货架规划和 AGV 路径规划协同优化.该协同优化算法基于经典遗传算法上进行改进,其创新点在于将货位规划和路径规划协同考虑后,构造的适应性函数能够将相似度高的货架分散放置,且高频出库货物放置在易于出货的位置上,从而实现在大量同时出库任务到来时,AGV 调度不堵塞,从整体上提高仓库出货效率.

具体来说,本文提出的货位规划与 AGV 路径协同优化算法(简称协同优化算法)旨在通过设计合理的货位摆放,为出货路径的规划提供辅助,使用自启发算法产生一个考虑货位摆放因素的优化运输方案.其详细方案如下:协同优化算法首先根据历史出货批次对零散货物进行编码处理;利用货物批次产生的编码计算“货品间”相似度,并因此对零散货物进行组合,生成货品组;每一个货品组为待入库状态,视为一个货架单元,与货位一一对应.其次,在货品组入库之前,先为每一个货位计算相应的出货路径,记录并保存货架位置对应的出货路径.基于上述计算结果,算法进入协同优化模块,即货品组放入货位的顺序是自由的,这种随机放置的方案构成协同优化过程的货位因素;而货品组不同的摆放方案会在出货任务到来时生成不同的运输路线,这种不确定的运输路线就是优化过程的路径因素;最后,利用遗传算法框架对货位因素编码生成初始种群,计算不同出库方案下适应度函数的值,在不断迭代的过程中搜寻最优方案,在确定货品组入库摆放的同时确定出库路线.综上,这种包含两方面因素的迭代寻优过程就是货位-路径协同优化.

2.1 协同优化数学模型

本文通过对智能仓储环节中各部分的关系进行耦合分析,提出了货位路径协同优化的数学模型.该模型与传统智能仓储优化算法的区别在于:将路径规划和货架优化归为一个整体,并用数学公式表达两者间的关系.具体变量及变量约束条件描述如下:$f(x)$为协同优化的总目标函数,f_{path}为所有任务出库总时间花费,f_{other}为除 AGV 车辆路径规划外算法其他部分的开销,α和β分别为影响系数,且$\alpha+\beta=1$.n 为 AGV 车辆总数.n_i 为当前 AGV 车辆编号, $i=1,2,\ldots,n,i\in N$.M 为出货任务总数.m_i 为当前出货任务编号,$i=1,2,\ldots,M,i\in N$.G 为货品总数.g_i 为当前货品编号, $i=1,2,\ldots,G,i\in N$.(i,j)表示当前坐标点位置,$i,j\in N,i\leqslant I,j\leqslant J$.$S$ 为货架总数.μ为车辆从终点返回起点的惩罚系数.

具体来说,本文协同优化的总体目标如公式(1)表示:

$$f(X)=\alpha f_{path}+\beta f_{other},\alpha+\beta=1 \tag{1}$$

其中,α和β为影响系数:α系数为“路径最短”的权重,关注算法寻找出来的路线最短;β系数为“额外开销”的权重,关注碰撞、冲突、转弯等额外时间消耗,即关注多个 AGV 选择的运输路径相互之间尽可能不重合,以免发生冲突等.

当 $M>N$ 时,即任务数大于车辆总数时,车辆从终点返回起点需要一定时间,见公式(2):

$$f_{path}=\sum_{M=1}^{N} f_{m,n}+\mu\sum_{M=N+1}^{M} f_{m,n} \tag{2}$$

当 $M\leqslant N$ 时,即车辆数大于任务数时,每个任务一辆 AGV 车,见公式(3):

$$f_{path}=\sum_{M=1}^{N} f_{m,n} \tag{3}$$

$f_{m,n}$ 为某次调度时,车辆运行花费的时间,其中,(i,j)为目标点,f_c 为车辆出现堵塞后的等待时间,详见公式(4):

$$f_{m.n}=\frac{d}{v}(f_i+f_j+f_{I-i}+f_{J-j})+f_c \tag{4}$$

f_{other} 主要包括两个部分:一个是物品间相关度计算,用于物品的分类;另外一个就是货架优化算法的时间消耗,此处 $i+1<G$,见公式(5):

$$f_{other}=\sum_{i=1}^{G}\sum_{j=i+1}^{G}f_{g_i,g_j}+f_s \tag{5}$$

下面我们将从货品相似度算法、多 AGV 路径规划算法、货位规划和 AGV 路径规划协同优化算法这 3 部分详细描述本文提出的货位规划与 AGV 路径规划协同优化算法的实现思路.货品相似度算法和多 AGV 路径规划算法将为最终的货位路径协同优化算法提供支持.

2.2 货品相似度算法

本文对实际正在运维的仓储货物数据进行处理,清洗去除操作时间、货品名称等无关数据,抽取货品编号(即货品的唯一标示)、批次(同一批次货品可理解为在同一时间内执行出入库)等数据进行数据分析,以 0 和 1 标识是该编号货品否在该批次出货,每个货品都是一个由批次数各维度组成的向量.通过余弦相似度算法计算该向量间的相似度,获得货品相似度.具体来说,基于余弦相似度的货品相似度计算方法具体流程如图 1 所示.

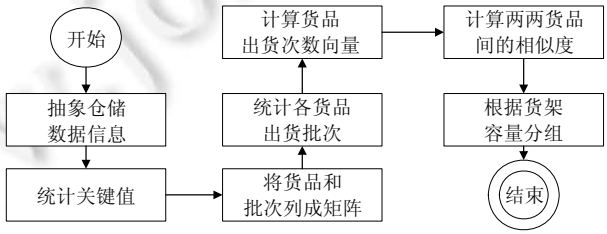


Fig.1 Flowchart of proposed product similarity solving algorithm

图 1 货品相似度算法流程图

首先,根据仓储数据信息计算最大出货批次数.将货品的出货信息记录下来,形成向量表.例如,1 号货物的向量值为[0,1,1,0,1],分别代表在 2 批次、3 批次、5 批次出货,其他货品以此类推.在此基础上累加货品出库总次数,计算出库频率.基于余弦公式(公式(6))计算两两货物间的余弦值:

$$\cos(x,y)=\frac{\sum_{i=1}^{n}x_i*y_i}{\sqrt{\sum_{i=1}^{n}x_i^2}*\sqrt{\sum_{i=1}^{n}y_i^2}} \tag{6}$$

由于货品和它本身和的余弦值为 1,代表两货物最为相似.因此,我们将货物余弦值减去 1 再求其绝对值,和 1 的差值越小,证明相似度越高.以这个新的值代表两货物间的相似度,最终会获得一个货品和货品相似的上三角矩阵.根据这个矩阵的值,即可计算出一个货品组.

2.3 路径规划算法

基于对遗传算法的改进,本文设计的基于遗传算法的多 AGV 路径规划算法,将路径长度、转弯数加入适应度函数,即转弯次数越少,路径总长越短,该个体越优秀.同时,采用改进的静态地图法来解决多 AGV 路径冲突问题.路径的规划是货位路径协同优化的基础,是本文的重要内容之一,主要使用遗传算法执行规划路径的操作,其伪代码如下所示.

输入:地图信息、起点、终点;

输出:规划完成的路径点集合.

```
funcation GA_path(Map,start,end)
    P_k←init(·)                    /*初始化种群*/
    k←0                            /*初始化遗传代数*/
```

```
    fit←fit_val(P_k)                    /*计算适应值*/
    Best_fit←fit                        /*初始化最佳个体*/
    for k to Max_Gerenation
        selectoperator                  /*选择算子*/
        crossings                       /*交叉操作*/
        variation                       /*变异操作*/
        P_k←Rebuild(P_k)                /*再次优化路径*/
        if Best_fit<fit_val(P_k)        /*判断是否为最佳个体*/
            then Best_fit←fit_val(P_k)  /*代替最佳个体*/
        end if
    end for
    return P_Best
end funcation
```

具体来说,在遗传算法计算过程中,首先对目标结果进行编码,路径编码方式为排列编码.例如,[1,2,5,6,7,8]表示从 1 号点出发,途径 2,5~7 这些点,最终到达 8 号点.路径坐标点序号作为基因参与算法运算.根据起始点到目标点序号,生成随机可行的路径序列,称为初始种群.

然后对此种群进行适应值计算,如公式(7)所示:

$$fit=\frac{1}{\left(path_{length}*\left(1+\frac{1}{\sqrt{node_{num}-1}}\right)\right)} \tag{7}$$

其中:$path_{length}$参数与 fit 适应度值成反比,表示路径越短,适应度值越优秀,选择出来的路线也越贴近最优解.适应值越大,意味着这个个体越优秀.$node_{num}$ 表示转弯次数,取值大于等于 1.$node_{num}$ 参数的添加,是考虑到转弯因素对路径规划的影响,$1+\frac{1}{\sqrt{node_{num}-1}}$ 系数对 $path_{length}$参数的微调,使算法在路径选择中尽可能避免选择拐弯次数过多的路线.经过实验验证:$node_{num}$ 的加入,有效地减少了算法选择转弯次数过多的路线.将基于该公式计算出的初始种群中所有个体的适应值记录下来,再对此种群进行交叉和变异操作.

本文基于轮盘赌法对交叉和变异操作的对象进行选取,以使得适应度越大的个体被选中的概率越大.轮盘赌选算子可以根据个体适应值在群体中所占比例,结合该算子被选中的累积概率,再取一个 0 到 1 之间的随机值,比较子代个体的适应度比例和随机值大小,选取子代适应值高的个体参与到遗传运算中,进一步提高整个种群的适应值,从而取得更优的最终结果,或者让该种群向更优的结果发展.具体来说,我们先计算该种群的适应值总和 fit_sum,然后,计算每个个体的适应值占比 $rfit=fit/it_sum$ 和各部分累积概率 $cfit$(如公式(8)所示):

$$cfit_{popnext_i}=cfit_{popnext_{i-1}}+rfit_{popnext_i} \tag{8}$$

其中,$i\leqslant popsize$(种群大小),$popnext_i$ 为第 i 代种群,$popcurrent_i$ 为第 i 代种群的副本.

随后遍历种群副本 $popcurrent_i$,而后生成一个 0-1 之间的随机数 p.比较子代个体的 $cfit$ 和 p 的值,直到 $cfit$ 比 p 大时,该个体代替 $popcurrent_i$ 中此位置的个体.最后,将 $popcurrent_i$ 的所有值赋给 $popnext_i$.

在交叉操作的实现中,交叉次数是路径点总数的 1/4,具体计算出的次数向下取整.每次交叉操作方法相同.首先,随机选取此种群中的某一条路径,在此路径上找到一个随机点设为交叉点 q,然后寻找另一条也通过此点的基因.设 $p1$,$p2$ 为原选出基因.将 $p1$ 中 q 点以后的点被 $p2$ 中 q 点以后的点代替生成新的基因 $p1$,再将 $p2$ 中 q 点以后的点由 $p1$ 中 q 点以后的点代替生成新的基因 $p2$.对 $p1$ 和 $p2$ 进行路径再优化,然后计算并更新 $p1$ 和 $p2$ 的适应值、路径长度和转弯次数,完成交叉操作.

在变异操作中,首先判定是否发生变异.基于预设变异几率($mutationRate$),根据 $random=Math.random(\cdot)*(1.0/mutationRate)$计算变异率($Math.random(\cdot)$为随机数生成函数).此时,若 $random$ 为 0,则执行变异.变异时,先随

机挑选种群中的一条染色体(即一个完整的可行路径),再在此路径上随机选取一个路径点,分别计算此条染色体上该路径点的前一个点 *pre* 和后一个点 *next*.而后在地图上分别搜索 *pre* 和 *next* 所有相邻点,并分别寻找所有相邻点中的重合点.我们将这些重合点视为可变异点,以保证变异后路径为通路.在重合点中随机选取一个点作为新的连接点,代替一开始选中的点,生成新的染色体.计算并更新新染色体的适应值、路径长度、转弯次数,完成变异操作.最后检查是否迭代到最大代数:若没有,则继续进行变异和交叉;若达到最大代数,则对种群进行筛选,即比较所有染色体(个体)的适应值,适应值最大即为最优结果.

除此之外,本文基于改进的静态地图法来解决多 AGV 路径冲突问题,以保证算法在多任务下各 AGV 车辆不会相撞.本文对已有的静态地图法进行了改进,在假设车辆保持匀速运行时,估算了车辆的运行位置,以此将车辆已走过的路径实时释放掉,车辆未走过的路径保持封锁状态,弥补静态地图法存在的缺陷.具体来说,在计算新的任务路径时,先获取之前所有未完成的任务执行情况,即在运行中的 AGV 的当前位置,并获取他们未来将要走的路径.通过对每一个新任务执行路径封锁,在优化最初避免冲突,使得冲突不可能发生.对于可能会造成的当前时间点车辆无法搜索到可行路径的情况,设置车辆等待.详细来说,算法中使用的地图不是创建的原始地图,而是将其他 AGV 车已占用的道路排除后的新地图,这意味着每个 AGV 车进行路径规划时都有一个属于他自己的地图.该地图将其他 AGV 已规划但未走过的路径点进行封锁,已封锁路径上的点和货架、充电桩等一样视为不可通过的障碍物,以此保证新的车辆规划避开这些障碍物,解决 AGV 的路径冲突问题.

2.4 货位路径协同优化算法

本文对正在运维的智能仓储出入口数据进行分析,发现:如果相似度高的货物摆放集中且密集,极有可能导致路径堵塞;相反,相似度高的货物摆放分散,则容易出现高频货物所在货架出库距离长这一问题.而分散与否,则是由最佳货架出库路径的重合程度判断的.因此,将货架最佳出库路径和货品的相关性结合起来,可有效地找到一个缓解 AGV 路径拥堵、提高出库效率的货品摆放结果.基于以上数据分析,本文提出的货位规划与路径规划协同优化算法的适应度函数包含两个参数:货架的出库代价以及该货架和周边货架的冲突代价.实验表明:该设置的适应值可以用来筛选出合格的个体,从而产生合适的解.

本文提出的货位规划与路径规划协同优化算法的基本思路为:首先计算货品间的相似度,按照货架最大载货种类数对货品进行聚类操作;再计算聚类后各个类别间货品的相似度,之后再计算各个货架的最佳出库路径.利用计算完成的数据以及当前地图信息,使用改进的遗传算法计算出货架布局情况,也就是货架的摆放方式,从而完成货位布局和路径规划两者的共同优化.

货位路径协同优化算法的具体流程如图 2 所示:首先,基于上文货品相似度算法的描述,使用余弦相似度算法将实验数据的货品间的相似度计算出来,并统计出库次数;然后结合相似度,依据货架的载货种类数,对当前货品进行分类,并且赋予每个分类属性.分类的出库频率为当前分类物品中出库频率最高的物品的出库频率,分类的出库批次为当前分类物品中出库频率数前 20%的货品出库批次总和.

举个例子,假设 1 号~10 号物品在一个分类中,此时,若 1 号、4 号货品的出库频率最高,且 1 号、4 号货品的出库批次向量分别为[1,0,1,1,1,0,1,0]和[1,0,1,1,1,1,1,0],则该分类的出库批次向量为[1,0,1,1,1,1,1,0].此时,一个分类有着货品的属性,且集成了更多的货品,将这个分类看作是一个未被放入货架位置的待入库货架,即将所有货品转化为待入库货架.

计算货架间的相似度,同样使用余弦相似度算法来进行计算.在计算出相似度后,对所有值减去 1 再取其绝对值,绝对值越小越相似.设定一个阈值η,此时,计算出的相似度一切小于η的货架认定为是高相似度货架.高相似度货架会参与到适应值的计算中,η过大,会认定多数货架是都是相互相似的,算法会难以选出真正优秀的解;η过小会导致选择不到足够的相似度货架,虽然算法可以给出一个它认为“优秀”的解.但在任务单来临时,可能仍旧会造成拥堵的状况,这也标志着算法的失败.本文以仓储实际运维数据(200 件货品,14 个批次的数据)为数据样本对仓储数据进行实验,调整η取值,观察了余弦值的取值规律,最终确选取 0.25 为本文提出的协同优化算法中η的值.

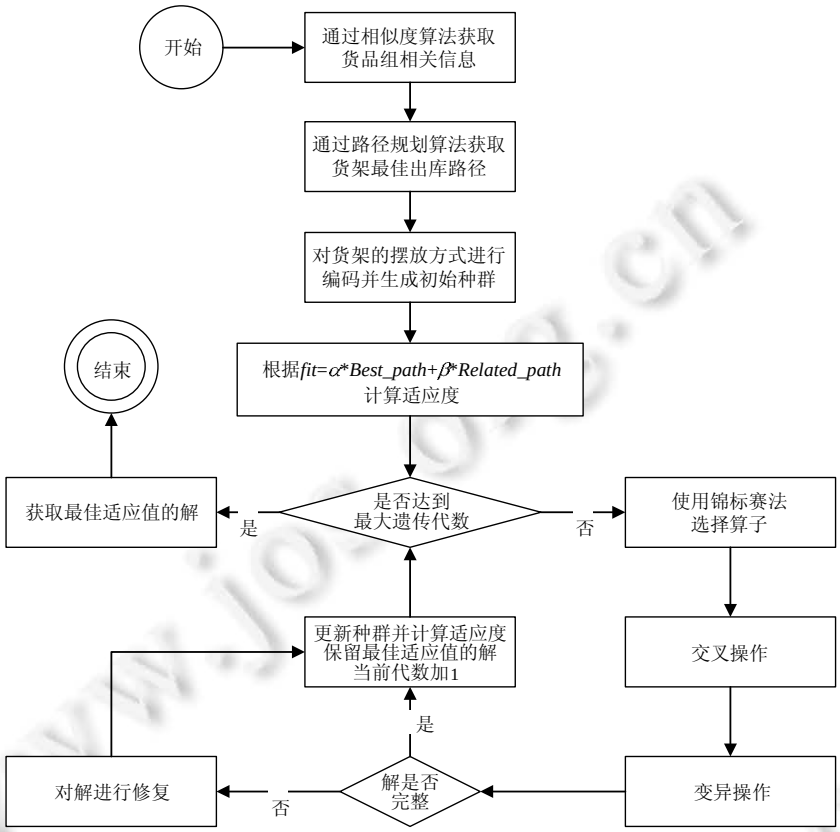


Fig.2 Flowchart of proposed shelf and AGV path coopeartive optimization algorithm

图 2 货位规划与路径规划协同优化算法流程图

货位路径协同优化的第 2 步为计算每个应摆放货架的位置.基于上文给出的路径规划算法,我们计算从该位置单独出货时的最佳出货路径,以获得每个可能摆放货架的位置和其出库的最佳路径,即最快出库方式.在获取并记录了各个货架位置的最佳出库路径后,开始对货架位置进行综合运算.

货位路径协同优化算法的求解算法基于遗传算法设计,首先对遗传算法进行编码.根据需要解决的问题,需要把未入库的货架,摆放到货架位置上,需要计算的是货架如何入库的问题.在编码上,选用排列编码即可.在排列上,选取货架位置作为空位,将未入库货架放入其中.例如,[5,4,2,3,1]意为将 5 号未入库货架放入 1 号货架位,将 4 号未入库货架放入 2 号货架位,以此类推.完成编码的选择和实现后,即可以生成初始种群.初始种群的建立是生成一组随机数,随机数区间在未入库货架号区间范围内.具体实现为:先获取所有未入库的货架号放入集合 A;再生成随机数,将其放入到个体的基因中记为集合 B.此时,该货架号从之前的集合 A 除去,防止再次选中.经过不断的生成,直到取完集合 A 中所有数.在随机生成的种群中,考虑个体重合问题,使用生成新的个体来替代重合个体,直到该种群中个体的数量满足设定的值.完成初始种群的生成后,计算该种群中个体的适应值.适应值 fit 的计算方式如公式(9)所示:

$$fit = f \times \left(\alpha \times Bestpath + \beta \times \sum_{i=1}^{N} Relatpath_i \right) \tag{9}$$

其中,f 是该货架的出入库频率,此参数可以放大整体适应度,使出库频率影响到货架摆放位置;$Bestpath$ 为当前货架出库的最佳路径;$\sum_{i=1}^{N} Relatpath_i$ 为与所有当前货架相似度较高的货架的出库路径重合量的和,该参数可用来降低冲突发生的情况,将避免冲突考虑到货架排放中;α和β为权重系数,在满足$\alpha+\beta=1$ 的约束下,调节两个参

数间的权重比,基于后续实验调整,本文的α系数取值为 0.8,β系数取值为 0.2.此外,根据适应度函数分析来看,适应度与该未入库货架的出库频率、最佳出库路径长度以及与其相关性高的其他货位路径冲突数成正相关.

完成适应度的计算,算法开始代数迭代.首先使用锦标赛法选取算子,基于已计算出的适应度,在种群中随机挑选个体并比较适应度,适应度大的被淘汰,最终选择出一定量的个体进行交叉生成下一代.在基本的锦标赛基础上,本文加入了被选择系数以提高锦标赛选择算子的效率,有效消弱了为了减少劣质个体被多次比较从而导致选择出不优秀个体的情况发生.具体来说,我们为每个个体赋予一个被选择系数,其默认初值为 1.一开始,各个个体的被选择系数相等,开始选择时为每个个体生成区间为[0,1]的随机数,再用该随机数乘以被选择系数,最终获得的按从小到大顺序取进入锦标赛.一旦某一个体被淘汰,则减少它的被选择系数.

基于上述获得的算子,算法开始依次进行交叉和变异操作.交叉操作时,选取集合 B 中任意两个体,进行交叉操作.本文提出协同优化算法选用有序交叉方法进行交叉,将父代中的某一段截取出来留给子代,再将另一个父代的基因按其顺序,在保证解的完整性下,依次放入子代中.例如两父代 x 为[1,2,3,4,5],y 为[3,2,5,1,4]进行交叉.截取 x 的中间 3 个为子代部分,当前子代状态为[·,2,3,4,·].再将 y 的基因按顺序,在不重复的情况下,依次填入其中,最终子代结果为[5,2,3,4,1].变异操作选用离散变异的方式进行,变异率 P 取 0.7/*chrom_length*(*chrom_length* 为编码长度).同时,在变异的过程中,考虑变异检测和变异位数因素.对种群中的个体进行变异检测即产生随机值,看是否小于变异率,当小于时执行变异.在变异位数方面,如果货位数大于货架数,取奇数;如果货位数等于货架数时,变异位数取偶数.基于交叉、变异的执行,新的个体重新放入种群中.算法判断是迭代代数否满足迭代终止要求:若满足,选取最优个体为最终解.

3 实验结果及分析

基于昆山某智能仓储的实际运维数据,我们在智能仓储仿真平台对本文提出的货位规划和路径规划协同优化算法进行了实验,并与传统遗传算法[24]、基于出货频率的贪心算法[25]进行了对比分析,实验结果验证了本文所提出的智能仓储货位规划与路径规划协同优化算法的有效性.

3.1 实验数据和评价指标

本次实验数据来自于昆山某生产厂商仓储物流数据,并额外加入一些干扰数据来模拟突发情况.实验数据涉及库存总货品 200 种,其中,物品间的相似度比例大致为 6:2:2:1:1,分别代表货物间两种货品间强相关、3 种货品间强相关直到 6 种货品间强相关,该比例的获取是从该厂商 15 天内共计 14 个批次的出库记录中统计获取的.实验中,货架、地图等设置是根据单个货架可放置货品种类数、地图预留的货位位置、路径点数等信息来配置的.基本的环境设置即为实际仓储环境,货位设置为单列连续摆放,总计 32 个货架,仓库入口和仓库出口分别位于仓储两侧.

本文所有的实验基于以下假设.

1) AGV 以匀速进行直线行走、转弯、运输等动作,且不出现偏离轨道的情况;
2) 路径点间间距等长.在衡量算法运算效果时,设 AGV 运行过两路径点间长度的时间为单位时间,即单位时间=路径点间距/AGV 速度(本文假定 AGV 平均运行速度为 0.5m/s,路径点间距为 1m,即单位时间为 2s);
3) AGV 车数量充足,能够保证每个任务都有独立的车辆立即执行.

设置的对比算法为其他货架优化算法,一个为传统遗传算法[24],另一个为基于出货频率的贪心算法[25],采用的路径计算统一为本文所实现的路径规划算法.本文的货位路径协同优化算法(以下实验数据分析中简称协同优化算法)和传统货架优化遗传算法,设定变异率为 0.07,交叉率为 0.68,迭代代数均为 300.为了让遗传算法可以和本文所设计的算法形成对比,在遗传算法的适应性函数中添加了货品相关性β的影响,并为其设置了影响系数 0.2,出货频率的系数α为 0.8.

在实验结果的分析中,主要评价指标包括为完成出库任务进行路径规划的算法自身耗时(记为出库路径规划耗时)、AGV 车辆按规划路径执行完出库任务的预估耗时(记为 AGV 运行耗时)、完成出库任务总耗时(即出

库路径规划耗时和 AGV 运行耗时之和,记为出库总耗时)、为完成出库任务所需调用的 AGV 车辆数目(记为动用车辆数)等.

3.2 实验结果及分析

下面我们将基于 14 个批次的出库数据,从不同的维度来对比分析货位路径协同优化算法的表现情况.

首先,我们分析各算法在完成所有批次货物出库的情况下的总体实验结果,即包含各类货品特征的仓储综合场景,该场景对应于实际仓储的综合运行状况;

其次,为了分析智能仓储协同优化算法在特定特征场景下的优化效果,我们筛选出符合不同特点的出货批次,分成下列 3 个特征场景进行各算法间的对比分析,具体包括:1) 以货品间相似度高、出货频率高为特点的场景 1;2) 以货品间相似度高、出货频率低为特点的场景 2;3) 以货品间相似度低、出货频率高为特点的场景 3.

下面分别对以上场景的实验结果进行具体描述和分析.

a) 总体实验结果及分析

本文对所有批次数据分别进行了实验,统计了 3 种算法在完成各批次货品时的出库任务路径规划时间(见表 1 中“出库路径规划耗时”列)和 AGV 车辆执行完该批次任务所需时间(见表 1 中“AGV 运行耗时”列).从而计算出各算法在实际运行过程中完成出货任务的总时间消耗(见表 1 中“出库总耗时”列),即路径规划时间和 AGV 执行任务时间之和.具体实验结果数据见表 1.

Table 1 Experimental data of all shipment batchs

表 1 各批次出库情况实验数据汇总表

算法 / 批次	货位路径协同优化算法(s)			遗传算法(s)			贪心算法(s)		
	出库路径规划耗时	AGV 运行耗时	出库总耗时	出库路径规划耗时	AGV 运行耗时	出库总耗时	出库路径规划耗时	AGV 运行耗时	出库总耗时
1	15.32	66	81.32	42.66	70	112.66	48.47	68	116.47
2	16.32	70	86.32	44.31	72	116.31	67.32	72	139.32
3	16.31	78	94.31	37.65	80	117.65	38.95	86	124.95
4	18.22	68	86.22	49.31	72	121.31	78.22	70	148.22
5	16.5	74	90.5	20.71	68	88.71	45.66	68	113.66
6	14.21	70	84.21	13.43	68	81.43	38.67	72	110.67
7	17.34	70	87.34	35.62	70	105.62	40.51	68	108.51
8	12.54	62	74.54	13.41	62	75.41	30.87	64	94.87
9	11.26	68	79.26	27.89	70	97.89	14.67	70	84.67
10	13.42	64	77.42	24.56	62	86.56	23.51	70	93.51
11	11.58	66	77.58	35.67	74	109.67	64.58	80	144.58
12	13.64	76	89.64	44.56	70	114.56	30.95	78	108.95
13	14.56	66	80.56	73.76	68	141.76	72.64	68	140.64
14	16.79	78	94.79	18.43	72	90.43	67.84	70	137.84

首先,基于表 1 中“出库路径规划耗时”列所示实验数据,本文从路径规划时间角度对实验结果进行分析.

如图 3 所示,本文提出的协同规划算法在在路径规划时间上比较稳定,且用时较短;而其他两种算法用时不稳定,且在大多数批次中,用时明显大于协同优化算法.探究其原因主要在于:

- 在多 AGV 场景下,路径规划的最好情况是每个 AGV 从起点出发,按规划路径行驶,到终点无冲突,那么路径规划只需进行一次即可;
- 相反,若多 AGV 在车辆规划路径时发生冲突时,算法会进行 3 件事:等待 1 个单位时间计算路径;计算绕路的路径;从两种方案中决策较优方案.因此,一旦多 AGV 发生冲突,路径规划算法的工作量会急剧攀升,这就是导致其他两算法路径规划时间不稳定的原因.

而协同优化算法将货位路径重合变成影响货架摆放的因素,对货架摆放结果产生影响.实验数据证明:此方法可明显地减少并避免了冲突,显著降低了出库路径规划用时的消耗.

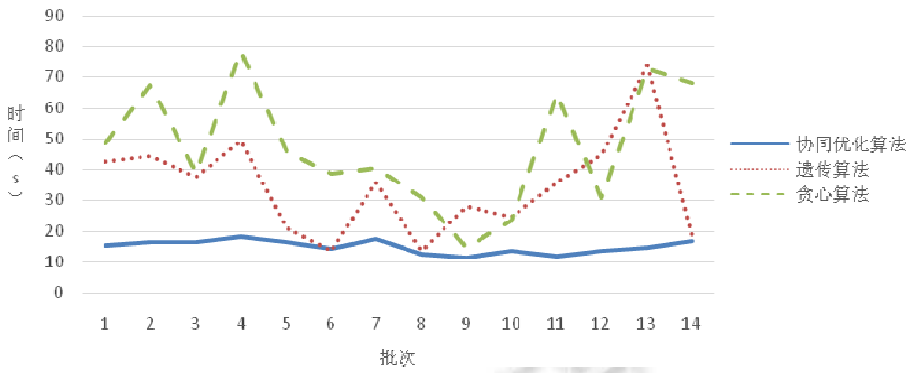


Fig.3 Time-consuming comparison chart of outbound path planning

图 3 出库路径规划耗时对比图

其次,从 AGV 完成出货任务的时间消耗角度分析,本文统计了 AGV 按规划路线实际执行任务所需时间.对于每一批次,我们将该批次中执行出货任务用时最长的 AGV 运行时间计为该批次的 AGV 执行出货任务耗时.在进行具体预估计算时,按照上文所述的假设给定的单位时间为 2s.同时,考虑到实际运行中,车辆的停止、转向和启动这个过程会额外耗时.因此,每一次车辆转弯额外增加 1 个单位时间用时.再者,当 AGV 发生冲突时,计算车辆间冲突视为 1 个单位时间,和转弯类似,AGV 在避免冲突时存在停止再启动,等待时间系统设置为 1 个单位时间,即每一次 AGV 冲突共需要消耗 2 个单位时间.综合上述时间代价,计算出各算法在 AGV 完成出货任务耗时的总时间,实验结果如基于表 1 中“AGV 运行耗时”列实验数据所示.基于实验数据生成了 AGV 运行预估耗时折线图.如图 4 所示,按 3 种算法所规划出的 AGV 路径执行出货任务时,AGV 的预估耗时差距不明显.本文对该实验结果进行了进一步分析认为:协同优化算法在计算货架位置时,不是以出入库频率作为唯一的计算依据,为了避免碰撞,会适当将相似度高的货架分开放置,以有效避免高相似度货品同时出货时 AGV 路径冲突的现象.但其缺点在于,不能够把高频率的货架放在最容易出货的位置.而实验中的预估出货时间时会以同一批次多个 AGV 中最长运行时间为准,因此从运行时间的角度来看,协同优化算法的优势不大.还有两个因素存会影响最终时间,即转弯时间惩罚和冲突等待时间惩罚,惩罚力度的改变也会对结果产生比较大影响.本文设置的惩罚力度比较柔和,所以各算法在 AGV 运行预估时间上比较相近.

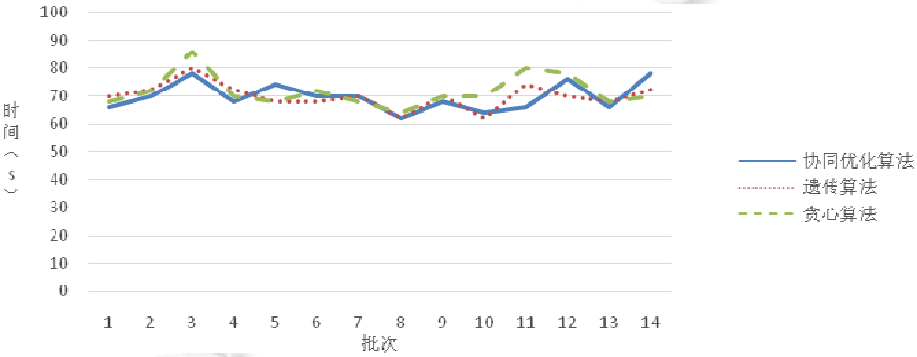


Fig.4 Comparison chart of AGV running time

图 4 AGV 运行耗时对比图

最后,基于各算法的出库任务路径规划时间和 AGV 车辆执行完该任务所需时间之和,本文计算出各算法在实际运行过程中完成出货任务的总时间消耗(实验数据见表 1“出库总耗时”列所示).如图 5 所示,协同优化算法的出库任务总耗时,在各个批次中都表现稳定,且耗时较短.

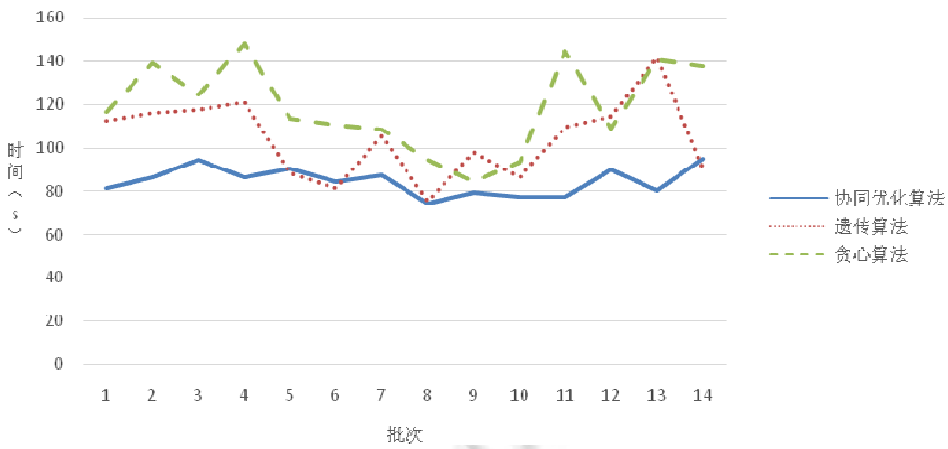


Fig.5 Comparison chart of total time-consuming of outbound tasks

图 5 出库总耗时对比图

在以上实验数据分析的基础上,本文进一步计算了 3 种算法在所有批次中出库任务总耗时方面的最优值、平均值和方差(见表 2).从计算结果可以看出:综合所有批次的实验数据,协同优化算法的耗时最优值低于遗传算法 1.5%,低于贪心算法 11.9%;在平均值方面,协同优化算法耗时低于遗传算法 18.8%,低于贪心算法 28.7%;此外,分析方差数据可以看出,协同优化算法的稳定性远远高于其他算法.因此,同其他两种算法相比,本文提出的协同优化算法在有效性和稳定性方面具有显著优势.

Table 2 Data analysis of the effectiveness and stability of proposed algorithm

表 2 算法有效性和稳定性数据分析表

	最优值	平均值	方差
协同优化算法	74.54	84.559 29	40.569 31
遗传算法	75.41	104.140 7	332.301 9
贪心算法	84.67	118.626 4	420.352

b) 特征场景实验结果及分析

特征场景实验分析,旨在验证智能仓储协同优化算法在不同场景下的表现情况.按照仓储货物可能出现的特点,我们从所有出货批次中筛选出包含该特点的出货批次,分成下列 3 个场景再进行算法间的对比分析.

- 场景 1:货品间相似度高、出货频率高的货品需求任务单;
- 场景 2:货品间相似度高、出货频率低的货品需求任务单;
- 场景 3:货品间相似度低、出货频率高的货品需求任务单.

在实验中,我们分别对货位路径协同优化算法、传统遗传算法、贪心算法在特定场景下的路径规划算法耗时、AGV 运行时间、AGV 转弯次数、冲突等待时间、完成出货任务所需 AGV 数量、综合耗时进行了计算.下面分别对 3 个场景的实验结果进行分析.

场景 1 的高相似度、高出货频率的货品集中出货是一般仓储常规的出货情况,一般而言,发生这种情况时也会伴随出货数量大这一特点,出货量大更能够考验算法的优化效果.这种常规情景下的算法优化效果也是最值得注意和研究的.表 3 描述了在该场景下,3 种算法的具体表现情况.从路径规划的时间消耗比较上来看:协同优化算法任务路径规划时间消耗最短,贪心算法任务路径规划时间消耗最长;从优化结果上分析,三者优化结果运行时间相近,货位路径协同优化算法和传统货架优化算法为 31 个单位时间,转弯次数均为 2 次,贪心算法为 30 个单位时间.但是除了货位路径协同优化算法外,其他两个算法均存在 AGV 运行时存在冲突.从综合用时来看,货位路径协同优化算法的表现更为出众一些,也符合预期的估计,没有造成车辆冲突的情况.因此,从货品出库效率的角度讲,协同优化算法优于其他两算法.

Table 3 Experimental data of Scenario 1

表 3 场景 1 实验结果汇总表

	路径规划时间消耗(s)	运行时间(单位时间)	转弯次数(次)	冲突次数(次)	动用车辆数(辆)	综合用时(s)
协同优化算法	11.58	31	2	0	3	77.58
遗传算法	35.67	31	2	2	3	109.67
贪心算法	64.58	30	2	4	4	144.58

场景 2 中,货品的主要特征为货品间相似度高、出货频率低,即高相似度、低出货频率的货品集中出货的场景,其主要应用于突发性缺少某些货物而进行的少量货物出库的情况.在该场景下,3 种算法的表现情况见表 4.具体来说,从路径规划时间上分析,货位路径协同优化算法表现最好,遗传算法其次,贪心算法较弱.其原因在于:贪心算法的关注点为出入库频率,不涉及相关性问题,在高相关货品出库时,货品分散在其他货架,导致其更有可能需要规划更多的货架进行出库,更多出库车辆数带来的是更高的车辆部署成本以及更久的路径规划时间.从 AGV 路径冲突分析,货位路径协同优化算法依然有效地避免了 AGV 冲突.同时,与场景 1 相比,在该场景下,传统遗传算法和贪心算法的冲突次数都有多所降低.其原因是:在这两种算法所计算的货架位置中,低频率货品安排位置一般孤立,出库时不会总是占用主要出库道路,因此相对冲突发生的机会变小.而货位路径协同优化算法则是在高频率货品放在优质出库位和集中出库时防止碰撞之间做平衡,在处理低频率货物时更可能将其和高频率货物穿插放置,其优点在于低频出货时也可能使其更近,缺点在于有些高频货物无法放在最佳出库位置上.从整体优化效果来看,尽管协同优化算法仍旧比其他算法优化效果更好,但优势不如场景 1 中明显.

Table 4 Experimental data of Scenario 2

表 4 场景 2 实验结果汇总表

	路径规划时间消耗(s)	运行时间(单位时间)	转弯次数(次)	冲突次数(次)	动用车辆数(辆)	综合用时(s)
协同优化算法	16.31	35	4	0	2	94.31
遗传算法	37.65	35	3	1	2	117.65
贪心算法	38.95	39	4	0	4	124.95

场景 3 中,货品的主要特征为货品间相似度低、但货品本身历史出货频率高.该场景主要发生于零散货品补货,一般货品出货量较小,AGV 路径冲突问题相对不严重.在该场景下,3 种算法的表现情况见表 5.从实验结果可以看出:在该场景下,协同优化算法的综合表现稍弱于传统遗传算法.其原因在于:协同优化算法关注货品间的相似度的适应性函数在设计之初就是为了应对集中出货导致 AGV 出货路径规划冲突严重这个问题,其更为关注货品间的相似度;而传统遗传算法正好相反,它更为关注货品的出货频率.因此,对于以货品低相似度、高频率为特征的场景 3 中,AGV 冲突处理能力不再成为决定算法表现的核心因素,协同优化算法不具优势.

Table 5 Experimental data of Scenario 3

表 5 场景 3 实验结果汇总表

算法	路径规划时间消耗(s)	运行时间(单位时间)	转弯次数(次)	冲突次数(次)	动用车辆数(辆)	综合用时(s)
协同优化算法	16.79	35	4	0	3	94.79
遗传算法	18.43	31	3	1	3	90.43
贪心算法	67.84	27	4	2	3	137.84

综上所述,我们在综合场景和特定特征场景下各算法的表现情况均进行了相关实验及结果分析.实验结果表明:从出库效率的角度来看,在不同特定特征场景下,货位路径协同优化算法的表现有所差异,在高相似度、高出货频率的特征场景中最具优势.在综合场景下,货位路径协同优化算法的表现明显优于其他算法.因此,在实际的智能仓储系统中,本文所提出的货位路径协同优化算法可有效提高仓储的出库率.

4 总结与展望

智能仓储的优化是目前整个未来仓储发展的重要方向之一.本文根据实际问题需求,参考协同优化思想,提出了智能仓储货位路径协同优化的数学模型和相关求解算法,包括货品相似度求解算法和改进适应度函数的

路径规划算法;并在以上两种算法的基础上,基于货位路径协同优化思想实现了货位路径协同优化.同时,基于真实仓储运维数据,本文从不同的维度、 场景分别对货位路径协同优化算法的表现情况进行实验并分析,实验结果表明,本文提出的智能仓储协同优化算法在算法有效性和稳定性上具有显著优势.该算法可有效提高仓储的出货效率,降低运输成本.

在后续工作中,我们将在以下方面继续展开研究:(1) 本文所提出的解决方案主要针对于网格式 AGV 布局,在其他布局下能否适用有待进一步考察和验证;(2) 将货品的相关性、体积、质量均引入货位路径协同优化算法中,以此保证货架的稳定性和放置货品的效率,扩大本文所提出的货位路径协同优化算法的适用条件,使其可以扩展至更多的应用场景;(3) 考虑当 AGV 小车数量不充足时,即待执行的出货任务数多余可支配的 AGV 数量时,智能仓储协同优化算法的研究.

References:

[1] Ding L,Zhang Y,Yin SC. Discussion on the development status and trends of my country's logistics and storage industry. Hoisting and Conveying Machinery, 2018(4):69−71 (in Chinese).

[2] Xu CX. Smart warehousing solutions from "people looking for goods" to "goods looking for people". China Strategic Emerging Industry, 2018(21):36-37 (in Chinese).

[3] Wu JP. Application analysis and prospects of Kiva robot in Amazon warehouse. Logistics & Material Handling, 2015,20(10): 159−164 (in Chinese).

[4] Wang XC. Talking about the "goods to people" technology of robots in warehouse management——based on the KIVA orange robot of Amazon logistics center. Modern Economic Information,2017(13):78,80 (in Chinese).

[5] Sun C, Liang Q, Wang ST. Investigation and research on JD warehouse intelligent system——Taking Shanghai "Asia One" warehouse as an example. China Logistics & Purchasing, 2017(18):70−71 (in Chinese).

[6] Cai J, Kuang X, Song S, *et al*. Automated warehouse storage assignment policy based on storage frequency and workload balance. In: Proc. of the Int'l Conf. on Logistics, Informatics and Service Sciences. IEEE, 2017. 1−6.

[7] Wang WL, Gao J, Gao TY, Zhao HT. Optimization of automated warehouse location based on genetic algorithm. In: Proc. of the Int'l Conf. on Control, Automation and Artificial Intelligence. 2017.

[8] Liu F, Shi Z, Liu Y. Location assignment algorithm for AS/RS based on fuzzy mathematics. Journal of University of Shanghai for Science and Technology, 2011,33(1):71−74 (in Chinese with English abstract).

[9] Yang W, Zhang WY, Chang YB, Qiu XH, Wang W. Optimization of location assignment in AS/RS. Modern Manufacturing Engineering, 2014,12:134−140 (in Chinese with English abstract).

[10] Tinelli LM, Becker M. Intelligent warehouse product position optimization by applying a multi-criteria tool. In: Proc. of the Robotics in Smart Manufacturing. Berlin Heidelberg: Springer-Verlag, 2013. 137−145.

[11] Arnaout JP. Worm optimization for the multiple level warehouse layout problem. Annals of Operations Research, 2017,4:1−23.

[12] Cservenák Á. Further development of an AGV control system. In: Proc. of the Vehicle and Automotive Engineering 2. Cham: Springer-Verlag, 2018.

[13] Zhang B, Li LW, Zhao YH, *et al*. The research on E-commerce logistics picking AGV path optimization method based on the improved A* algorithm. In: Proc. of the Int'l Conf. on Cybernetics, Robotics and Control. IEEE, 2017. 99−103.

[14] Jiang K, Sun HY, He QC. Research on AGV scheduling strategy assembly workshop considering with distributing area layout. Modular Machine Tool & Automatic Manufacturing Technique, 2013,10:141−144 (in Chinese with English abstract).

[15] Zhang SY, Yang YS, Liang CJ, Xu BW, Li JJ. Optimal control of multiple AGV path conflict in automated terminals. Journal of Transportation Systems Engineering and Information Technology, 2017, 17(2):83−89 (in Chinese with English abstract).

[16] Bilge Ü, Ulusoy G. A time window approach to simultaneous scheduling of machines and material handling system in an FMS. Operations Research, 2017,43(6):1058−1070.

[17] Mousavi M, Yap HJ, Musa SN,Tahriri F, Dawal SZM. Multi-Objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization. Plos One, 2017,12(3):e0169817.

[18] Hillis WD. Co-Evolving parasites improve simulated evolution as an optimization procedure. Physica D, 1990,42(1-3):228−234.

[19] Shi Y, Teng H, Li Z. Cooperative co-evolutionary differential evolution for function optimization. In: Proc. of the Int'l Conf. on Natural Computation. Berlin Heidelberg: Springer-Verlag, 2005.

[20] Wang, YS, Teng HF, Shi YJ. Cooperative co-evolutionary scatter search for satellite module layout design. Engineering Computations, 2009,26(7):761−785.

[21] Huo JZ, Teng HF, Sun W, *et al*. Human-Computer co-operative co-evolutionary method and its application to a satellite module layout design problem. The Aeronautical Journal, 2010,114(1154):209−223.

[22] Liang J, Liu R, Yu KJ, Qu BY. Dynamic multi-swarm particle swarm optimization with cooperative coevolution for large scale global optimization. Ruan Jian Xue Bao/Journal of Software, 2018,29(9):2595−2605 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/29/2595.htm [doi: 10.13328/j.cnki.jos.005398]

[23] Lu PH. Design and implementation of intelligent warehousing collaborative optimization algorithms [MS Thesis]. Xi'an: Xidian University, 2019 (in Chinese).

[24] Zhang YH, Qiu JD, Tang MA. Slotting optimization for the automated stereoscopic warehouse by shelf shared mode. Computer Applications and Software, 2017,34(07):262−266+272 (in Chinese with English abstract).

[25] Liu ZS, Tong LY, Wei LP, Zhu K, Qian XL. A slotting optimization model based on greedy algorithm. Logistics Sci-Tech, 2013,36(9):99−101 (in Chinese with English abstract).

附中文参考文献:

[1] 丁力,张玉,尹世琛.我国物流仓储行业发展现状与趋势探讨.起重运输机械,2018,(4):69−71.

[2] 徐晨曦.智能仓储的解决方案——从“人找货”到“货找人”.中国战略新兴产业,2018,21:36−37.

[3] 吴菁芃.亚马逊仓库 Kiva 机器人的应用分析与前景展望.物流技术与应用,2015,20(10):159−164.

[4] 王晓春.机器人在仓储管理中的“货到人”技术浅谈——基于亚马逊物流中心 KIVA 橙色机器人.现代经济信息,2017(13):78,80.

[5] 孙璨,梁沁,王姝婷.京东仓储智能化体系调查研究——以上海“亚洲一号”仓库为例.中国物流与采购,2017(18):70−71.

[8] 刘峰,施展,刘莹.自动化立体仓库的货位分配模糊算法研究.上海理工大学学报,2011,33(1):71−74.

[9] 杨玮,张文燕,常晏彬,邱小红,王雯.自动化立体仓库的货位分配优化.现代制造工程,2014,12:134−140.

[14] 姜康,孙海燕,何其昌.综合配料区布局的总装车间 AGV 系统调度研究.组合机床与自动化加工技术,2013,10:141−144.

[15] 张素云,杨勇生,梁承姬,许波桅,李军军.自动化码头多 AGV 路径冲突的优化控制研究.交通运输系统工程与信息,2017,17(2):83−89.

[22] 梁静,刘睿,于坤杰,瞿博阳.求解大规模问题协同进化动态粒子群优化算法.软件学报,2018,29(9):2595−2605. http://www.jos.org.cn/1000-9825/29/2595.htm

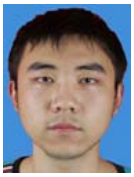
[23] 陆鹏浩.智能仓储协同优化算法的设计与实现[硕士学位论文].西安:西安电子科技大学,2019.

[24] 张衍会,邱建东,汤旻安.一种货架共用模式自动化立体仓库货位优化.计算机应用与软件,2017,34(07):262−266+272.

[25] 刘志帅,仝凌云,魏利鹏,朱凯,茜晓立.基于贪婪算法的货位优化模型.物流科技,2013,36(9):99−101.

蔺一帅(1987－),女,博士,讲师,CCF 专业会员,主要研究领域为面向 Agent 的软件工程,智能系统软件.

李青山(1973－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为软件自适应,软件演化,面向 Agent 的软件工程.



陆鹏浩(1994－),男,硕士,主要研究领域为分布计算,可信计算与信息安全.

孙雨楠(1996－),男,硕士生,主要研究领域为智能仓储协同优化.

王亮(1976－),男,硕士,工程师,主要研究领域为智能工厂,智能仓储物流,智能园区行业领域的新一代信息技术.

王颖芝(1996－),女,硕士生,主要研究领域为智能小车存取系统优化.