

包裹智能分拣系统分拣路径动态规划方法

李琿^a,肖海宁^b,周临震^b,王雨轩^b,顾海航^b

(盐城工学院 a. 优培学院; b. 机械工程学院,江苏 盐城 224051)

摘要: 为了提升包裹智能分拣系统的效率,设计一种包裹定向分拣路径动态规划方法。介绍一种新型包裹智能分拣系统的结构和工作原理;针对包裹智能分拣系统的特点,构建包裹定向分拣路径动态规划数学模型,并设计对应的规划算法;开发包裹智能分拣系统仿真平台,以验证包裹定向分拣路径动态规划方法效果。仿真结果表明:所设计的包裹定向分拣路径动态规划方法能够显著减少包裹平均分拣时间,从而提升包裹智能分拣系统的效率。

关键词: 包裹智能分拣;路径规划;Dijkstra 算法;仿真分析;Plant Simulation 软件

中图分类号: TP274+.3;TP391.9 **文献标志码:** A **文章编号:** 1671-5276(2023)03-0083-04

Dynamic Sorting Path Planning Method for Parcel Intelligent Sorting System

LI Hui^a, XIAO Haining^b, ZHOU Linzhen^b, WANG Yuxuan^b, GU Haihang^b

(a. You Pei College; b. School of Mechanical Engineering, Yancheng Institute of Technology, Yancheng 224051, China)

Abstract: In order to improve the efficiency of parcel intelligent sorting system, a dynamic parcel directional sorting path planning method is proposed. The structure of a new parcel intelligent sorting system and its working principle are introduced. According to the characteristics of parcel intelligent sorting system, the mathematical model of dynamic planning of parcel directional sorting path is constructed and a corresponding planning algorithm of parcel sorting path is designed. The simulation platform of parcel intelligent sorting system is developed to verify the effect of the directional sorting path dynamic planning method. Simulation results show that the proposed method can substantially reduce the average sorting time, so as to improve the sorting efficiency of parcel intelligent sorting system.

Keywords: intelligent parcel sorting; path planning; Dijkstra algorithm; simulation analysis; Plant Simulation software

0 引言

近年来,随着电子商务的快速发展,快递包裹数量与日俱增,使得分拣工作日益繁重^[1]。然而,目前国内快递包裹分拣工作依然依赖于人工,不仅分拣成本高、效率低,且易造成包裹的损坏或丢失。为了提升快递包裹分拣的自动化程度和效率,国内外在分拣作业流程及方法优化、新型分拣系统开发等方面开展了一系列的研究工作。张阳等^[1]借鉴生产线平衡思想,对快递集散中心分拣作业进行了优化。翟龙真^[2]采用人因工程中的设计原则与方法对半自动化分拣作业进行了优化。贺学成等^[3]针对应用于快递分拣的高密集度 AGV 系统,设计了能够避免拥堵的路径规划算法。黄梦涛等^[4]对用于快递分拣的二维码定位与校正方法进行了研究。王立平^[5]对基于柔性分拣单元的高效智能分拣系统中的分拣信息识别技术、分拣优化策略等进行了研究。德国不莱梅大学生产与物流研究所开发了一款模块化的智能物流系统 Celluveyor,使得快递分拣和传送变更为高效、灵活^[6]。然而,国内分拣系

统依然基于滚筒、皮带等较为落后的输送技术,在分拣效率及灵活性上难以取得重大突破。为了提高包裹分拣的灵活性,本文设计了一种基于全方位轮(omnidirectional wheel)^[7]的新型模块化包裹智能分拣系统。首先介绍该系统的结构及工作原理,并以经典 Dijkstra 算法为基础,考虑各分拣模块分拣负荷流量,设计了一种权值动态调整的多目标优化包裹分拣路径规划方法。

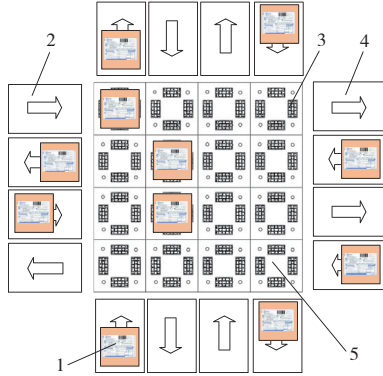
1 包裹智能分拣系统简介

所设计的包裹智能分拣系统示意图如图 1 所示。四周分布有多个入口传送带与出口传送带,中间为分拣区域,一般每个出口传送带均对应一个地区包裹,用于将目的地为该地区的已分拣包裹输送至对应的包裹周转箱。待分拣包裹在入口传送带依次进入分拣区域,在入口传送带末端装有快递条码采集装置,通过采集快递条码可获知快递所需送达的出口传送带。分拣区域的结构示意图如图 2 所示,整个分拣区由多个分拣模块整齐填充,其功能是将来自各入口传送带的物流包裹,按照各包裹目的地进

基金项目: 国家自然科学基金资助项目(52005427);江苏省高校自然科学基金资助项目(19KJB510013,18KJA460009);江苏省大学生创新创业训练计划资助项目(201910305089H)

第一作者简介: 李琿(1999—),男,江苏盐城人,本科,研究方向为物注销仿真与优化技术。

行分拣输送。各分拣模块均装有可感知包裹位置的传感装置,可根据包裹位置和为其规划的分拣路径确定其输送方向。通过多个相邻分拣模块的接力输送,可将包裹输送至对应的出口传送带,从而实现包裹的分拣。



1—快递包裹;2—分拣区域入口传送带;3—分拣区域;
4—分拣区域出口传送带;5—分拣模块。
图1 包裹智能分拣系统示意图

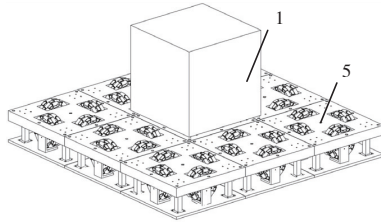


图2 分拣区域结构示意图

对分拣区域控制系统而言,其首要解决的是包裹分拣路径动态规划问题,即已知某包裹所处的入口传送带以及根据快递条码信息获知的快递所需送达的出口传送带,确定完成分拣所需途径的分拣模块。由于所有包裹共享同一分拣区域,因此,在为各包裹规划分拣路径时,不仅需要优化该包裹自身的性能指标(如:途径的分拣模块数最少、最小化包裹分拣时间等),还需考虑其他包裹的已规划分拣路径,避免包裹间发生路径冲突从而引发分拣区域的拥堵。

2 包裹定向分拣路径动态规划数学模型

首先介绍建模所需的符号。

$\Omega_m^b = \{G_m^b \mid 1 \leq m \leq N_c^b\}$,为分拣区域中所有包裹的集合。其中: G_m^b 为第 m 个正在分拣的包裹; N_c^b 为分拣区域中包裹数。

$\Omega_m^f = \{G_m^f \mid 1 \leq m \leq N_c^f\}$,为已完成分拣的包裹集合。其中: G_m^f 为第 m 个已分拣包裹; N_c^f 为已完成分拣的包裹数。

$\Omega_M = \{M_i \mid 1 \leq i \leq N_M\}$,为所有分拣模块的集合。其中: M_i 为第 i 个分拣模块; N_M 为分拣区域中分拣模块数。

$P_m^b = \{M_s M_i \cdots M_j \cdots M_k M_d\}$,为控制系统为包裹 G_m^b 规划的分拣路径,即自入口传送带至出口传送带先后途经的分

拣模块序列。

$Q_{m,i}^f$ 为 P_m^b 中分拣模块 M_i 的出现次数,即 G_m^b 分拣过程中需要经过分拣模块 M_i 的次数。

H_m^i 为 G_m^b 分拣路径是否经过分拣模块 M_i 的标识符,定义如下:

$$H_m^i = \begin{cases} 0 & M_i \in P_m \\ 1 & M_i \notin P_m \end{cases} \quad (1)$$

t_m^f 为 G_m^f 分拣耗时,即包裹 G_m^f 分拣过程中在分拣区域的停留时间。

\bar{T} 为所有包裹的平均分拣耗时,计算公式为

$$\bar{T} = \sum_{m=1}^{N_c^f} t_m^f \quad (2)$$

O_m^f 为 G_m^f 分拣过程中途径的分拣模块数量。

\bar{O} 为所有包裹定向分拣过程中途径的平均分拣模块数,计算公式为

$$\bar{O} = \sum_{m=1}^{N_c^f} O_m^f \quad (3)$$

$e_{i,j} = (M_i, M_j)$,为两相邻的分拣模块 M_i 与 M_j 所构成的有向边, M_i 为边 $e_{i,j}$ 的起点, M_j 为边 $e_{i,j}$ 的终点。

$e_{i,j}$ 为边 $e_{i,j}$ 的规划属性值,定义如下:

$$e_{i,j} = \begin{cases} 1 & \text{边 } e_{i,j} \text{ 属于某正在分拣包裹的路径} \\ 0 & \text{边 } e_{i,j} \text{ 不属于任一正在分拣包裹的路径} \end{cases} \quad (4)$$

$A = [a_{i,j}]_{N_M \times N_M}$,为分拣模块间的邻接矩阵。其中 $a_{i,j}$ 表示 M_i 与 M_j 间的邻接关系,定义如下:

$$a_{i,j} = \begin{cases} 0 & i=j \\ 1 & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 直接相邻} \\ \infty & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 非直接相邻} \end{cases} \quad (5)$$

$D = [d_{i,j}]_{N_M \times N_M}$,为分拣模块间的方位矩阵。其中 $d_{i,j}$ 表示 M_i 与 M_j 间的方位关系,定义为

$$d_{i,j} = \begin{cases} 0 & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 直接相邻,且 } M_i \text{ 位于 } M_j \text{ 左侧} \\ 90 & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 直接相邻,且 } M_i \text{ 位于 } M_j \text{ 下侧} \\ 180 & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 直接相邻,且 } M_i \text{ 位于 } M_j \text{ 右侧} \\ 270 & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 直接相邻,且 } M_i \text{ 位于 } M_j \text{ 上侧} \\ \infty & \text{分拣模块 } M_i \text{ 与 } M_j \text{ 不相邻} \end{cases} \quad (6)$$

$U = [f(i)]_{1 \times N_M}$,为分拣模块负荷流量矩阵。其中 $f(i)$ 表示 M_i 的负荷流量,表示当前分拣区域中尚需通过 M_i 的包裹数,公式为

$$f(i) = \sum_{m=1}^{N_c^b} H_m^i \quad (7)$$

构建的多目标优化数学模型为:

决策变量:

对 Ω_m^b 中每个待分拣包裹 G_m^b ,已知与包裹入口传送带连接的分拣模块为 M_s ,与包裹目的出口传送带连接的分拣模块为 M_d ,确定其分拣路径 P_m^b ,具体包括 G_m^b 分拣路径是否经过各分拣模块 M_i 的标识符 H_m^i 以及包裹分拣途径分拣模块的序列,即路径

$$P_m^b = \{M_s M_i \cdots M_j \cdots M_k M_d\} \quad (8)$$

优化目标:

以最小化途径分拣模块数和各分拣模块分拣负荷流量为综合优化目标,并根据分拣系统实时运行参数 \bar{T} 和 \bar{O} 对权值优化调整,表示为

$$\min\{(O_m^F) + \sigma \times \sum_{\forall M_i \in P_m^B} f(i)\} \quad (9)$$

式中 σ 为动态负荷权值,计算公式为

$$\sigma = \frac{\bar{T}}{\bar{O}} \quad (10)$$

约束条件:

$$Q_{m,i}^F \leq 1 \quad \forall M_i \in \Omega_M \quad (11)$$

$$a_{i,j} = 1 \quad \forall e_{i,j} \in P_m^B \quad (12)$$

$$d_{i,j} \neq \infty \quad \forall e_{i,j} \in P_m^B \quad (13)$$

$$e_{i,j} + e_{j,i} \leq 1 \quad \forall e_{i,j} \in P_m^B \quad (14)$$

式(11)表示包裹在分拣过程中每个分拣模块至多只经过一次;式(12)和式(13)表示包裹分拣路径必须为相邻分拣模块构成的连续有序序列;式(14)表示所有包裹在任意两相邻分拣模块间的运行方向必须一致,避免两包裹在分拣过程中出现相向冲突。

3 包裹定向分拣路径动态规划算法

由所构建的数学模型可知,包裹定向分拣路径动态规划问题可归为单源最短路径规划问题。Dijkstra 算法^[8]作为求解单源最短路径规划问题的经典算法,不仅能够搜索到全局最优路径,而且算法复杂度低。但经典 Dijkstra 算法是单目标优化算法,且无法考虑式(11)一式(14)所示的各类约束。因此,本文根据所构建模型的特点,对经典 Dijkstra 算法加以改进,设计相应的包裹定向分拣路径动态规划算法,算法整体流程图如图3所示。

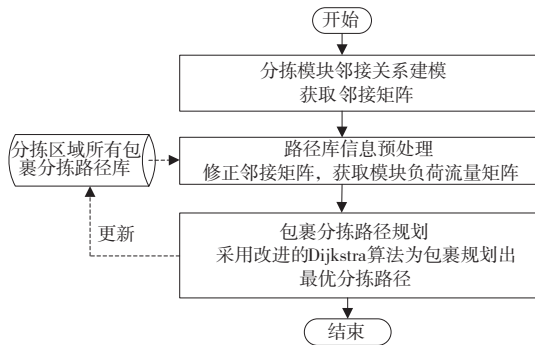


图3 包裹定向分拣路径动态规划算法

具体方法如下:

1) 分拣模块邻接关系建模

Step1: 对分拣区域内的 N_M 个分拣模块编码。

Step2: 按照式(5)一式(6)的定义获取分拣模块间的邻接矩阵 $\mathbf{A}=[a_{i,j}]_{N_M \times N_M}$ 和方位矩阵 $\mathbf{D}=[d_{i,j}]_{N_M \times N_M}$ 。

2) 路径库信息预处理

Step1: 为了避免两包裹在分拣过程中出现相向冲突,根据分拣区域所有包裹定向分拣路径库,按照式(14)所示约束条件的要求对邻接矩阵 $\mathbf{A}=[a_{i,j}]_{N_M \times N_M}$ 进行修正,

修正方法如下,若 $e_{i,j}=1$,则设置 $a_{j,i}=\infty$,从而避免新规划的包裹定向分拣路径中出现有向边 $e_{j,i}=(M_j, M_i)$ 。

Step2: 根据分拣区域所有包裹定向分拣路径库,按照式(7)计算各分拣模块的分拣负荷流量,获得分拣模块负荷流量矩阵 $\mathbf{U}=[f(i)]_{1 \times N_M}$ 。

3) 路径规划

假定与包裹入口传送带连接的分拣模块为 M_s ,与包裹目的出口传送带连接的分拣模块为 M_d ,采用改进的Dijkstra 算法为包裹规划一条从 M_s 至 M_d 的最优路径,具体步骤如下:

Step1: 初始化,设置分拣模块集合 $\Omega_s = \{M_s\}$ 和分拣模块集合 $\Omega_D = \Omega_M \setminus \Omega_s$ 。其中: Ω_s 为已搜索到与 M_s 间最优路径的分拣模块集合; Ω_D 为尚未搜索到与 M_s 间最优路径的分拣模块集合;设置最优路径矩阵 $\mathbf{R}=[R_{s,i}]_{1 \times N_M}$, $R_{s,i}$ 为 M_s 至物流模块 M_i 的最优路径,尚未搜索到最优路径时为 \emptyset ,初始化时仅有 $R_{s,s} = \{M_s\}$;设置代价矩阵 $\mathbf{W}=[W_{s,i}]_{1 \times N_M}$, $W_{s,i}$ 是包裹途径物流模块 M_i 所需付出的代价,根据式(9)所示的目标函数,将 $W_{s,i}$ 定义如下:

$$W_{s,i} = a_{s,j} + \sigma f(i) \quad (15)$$

Step2: 搜索 Ω_D 中拥有最小代价的分拣模块,假定该模块为 M_k ,最小代价为 $W_{s,k} = a_{s,k} + \sigma f(k)$,则将分拣模块 M_k 从 Ω_D 移入 Ω_s ,即设置: $\Omega_s = \Omega_s \cup \{M_k\}$, $\Omega_D = \Omega_D \setminus \{M_k\}$;并设置 $R_{s,k} = \{M_s, M_k\}$,完成后判断新加入 Ω_D 中的分拣模块 M_k 是否为 M_d 。若否,转 Step3,若是,转 Step4,

Step3: 更新 Ω_D 中每个物流模块的代价和最优路径,更新方法是对 Ω_D 中的任意分拣模块 M_n ,其新的代价 $W_{s,n}$ 更新为

$$W_{s,n} = \begin{cases} W_{s,k} + W_{k,n} + \sigma f(n) & W_{s,k} + W_{k,n} < W_{s,n} \\ W_{s,n} & \text{else} \end{cases} \quad (16)$$

将分拣模块 M_n 的最优路径 $R_{s,n}$ 同步更新为

$$R_{s,n} = \begin{cases} R_{s,k} \cup R_{k,n} & W_{s,k} + W_{k,n} < W_{s,n} \\ R_{s,n} & \text{else} \end{cases} \quad (17)$$

完成上述更新后转 Step2。

Step4: 输出最优路径 $R_{s,d}$,并设置 $P_m^B = R_{s,d}$ 。

4 方法验证

4.1 包裹智能分拣系统仿真平台的开发

采用 Plant Simulation15.0^[9]开发的包裹智能分拣系统仿真平台界面如图4所示。该仿真平台以某公司规划建设分拣实验系统为原型,整个分拣区域由400个分拣模块组成,分拣区域四周分布有入口传送带和出口传送带各40条,分拣模块边长为0.6m,分拣模块对包裹的输送速度为0.8m/s,快递条码采集装置采用 Zebra DS457,单个快递条码的平均采集耗时小于0.5s。采用 Source 对象表示入口传送带,入口传送带最快以1个/s的频次向连接的分拣模块输送待分拣包裹,并在各对象的入口触发包裹计数事件,从而将分拣区域的包裹数 N_c^b 维持在某恒定值,避免分拣区域包裹数过多引发阻塞。采用 Drain 对象表示出口传送带。分拣模块采用 Station 对象表示。路径规

划功能在 Method 对象中编制 SimTalk 程序实现,当新包裹产生后,自动触发路径规划算法程序,并在各分拣模块入口中触发包裹流向控制事件,从而保证包裹按照各自分拣路径运行。

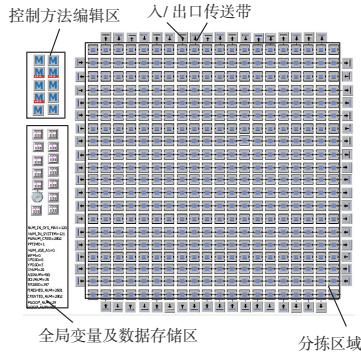


图4 包裹智能分拣系统仿真平台

4.2 对比实验及分析

对比3种包裹定向分拣路径规划方法:

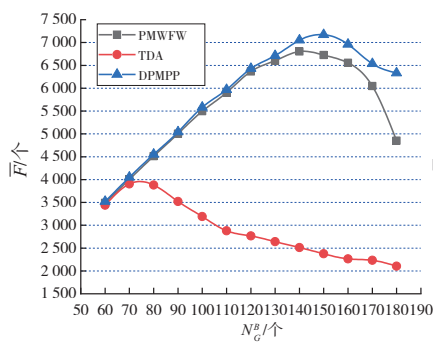


图5 各 N_c^b 下的系统单位小时平均分拣包裹数

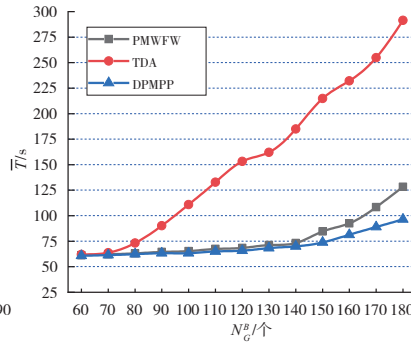


图6 各 N_c^b 下的所有包裹平均分拣耗时

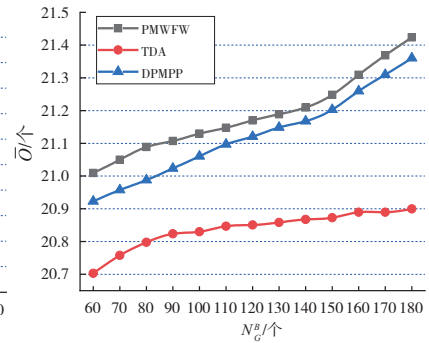


图7 各 N_c^b 下的所有包裹平均途径分拣模块数

如图5所示,当 N_c^b 少于70时,各包裹间出现路径冲突的概率较小,因此3种方法的单位小时平均分拣包裹数 \bar{F} 无明显差异。随着 N_c^b 的增多,各包裹间发生路径冲突的概率显著增加,而TDA仅以最小化途径分拣模块数为目标。由图7可知,TDA在任何 N_c^b 下的 \bar{O} 均最低,但由图6可知,所有包裹平均分拣耗时 \bar{T} 显著增加,这说明TDA未考虑各分拣模块负荷流量,导致各包裹在各分拣模块的等待时间较长。由图5可知,随着 N_c^b 的上升,TDA的单位小时平均分拣包裹数 \bar{F} 反而下降,性能最差。

与TDA不同,DPMP和PMFW以降低包裹途径分拣模块数和各分拣模块负荷流量为综合优化目标,因此,由图7可知,虽然包裹平均途径分拣模块数 \bar{O} 略高于TDA,但仅增加不到5%,且由图6可知,随着 N_c^b 的上升,DPMP和PMFW的包裹平均分拣耗时 \bar{T} 远低于TDA。

1) 本文设计的分拣路径动态规划方法 (the dynamic planning method proposed in this paper, DPMP);

2) 基于传统单目标Dijkstra算法 (traditional Dijkstra algorithm, TDA)的规划方法,即仅以最小化途径分拣模块数为目标,但考虑了式(11)~式(14)所示约束的分拣路径规划方法;

3) 权值固定的分拣路径规划方法 (the planning method with fixed weight, PMFW),即令式(15)中的权值 $\sigma=1$ 。

上述3种包裹定向分拣路径规划方法均源于Dijkstra算法,算法复杂度均为 $O(N_M^2)$,以拥有400个分拣模块的仿真平台为例,运行仿真平台的工作站配置为:CPU E3-1535M 3.1 GHz, DDR4 64 G内存。3种包裹定向分拣路径规划方法自获取包裹信息至完成路径规划的耗时均小于0.1 s,连续运行仿真平台48 h,统计24~48 h间仿真数据,各方法在各 N_c^b 下的单位小时平均分拣包裹数 \bar{F} 、所有包裹平均分拣耗时 \bar{T} 和平均途径分拣模块数 \bar{O} 分别如图5~图7所示。

由图5可知,DPMP和PMFW的单位小时平均分拣包裹数 \bar{F} 远高于TDA。这说明考虑各分拣模块负荷流量后,能够有效降低包裹在各分拣模块的等待时间,进而提高系统分拣效率。其次,在任何 N_c^b 数量下,DPMP所能实现的单位小时平均分拣包裹数 \bar{F} 均高于PMFW,而所有包裹平均分拣耗时 \bar{T} 和平均途径分拣模块数 \bar{O} 均低于PMFW,证明了与固定权值相比,本文所提出的动态负荷权值计算方法更为合理。

5 结语

本文以新型包裹智能分拣系统为研究对象,介绍了其结构及工作原理;构建了包裹定向分拣路径动态规划数学模型,根据该数学模型对经典Dijkstra算法加以改进,设计相应的包裹定向分拣路径动态规划算法;最后采用西门

(下转第116页)

3 结语

封隔器胶筒在坐封过程中存在材料、几何以及接触非线性,导致有限元求解难以收敛且不准确。本文基于非线性有限元理论,提高了非线性仿真的收敛能力与准确度。仿真结果如下所述。

1)随着坐封载荷的线性增加,胶筒变形量、等效应力、接触压力非线性增加。

2)在相同的坐封载荷下,单胶筒、双胶筒、三胶筒中的上胶筒接触压力变化相近,密封效果差异较小。但是,随着胶筒个数增加,沿着远离压缩端的方向,胶筒的接触压力依次减小,胶筒密封能力依次减弱。

3)随着摩擦因数的增加,胶筒表面接触压力降低。通过“低应力分布带”和“接触压力分布曲线斜率”可评价胶筒密封性能。

4)胶筒硬度越小,则胶筒的变形量越大,与接触面的接触压力分布越大且更为均匀。在设计中,可选用硬度较小的橡胶材料作为密封胶筒,选用硬度较大的胶筒作为起保护作用的胶筒。

参考文献:

- [1] 王早祥,张砚雯,车家琪,等. 锦纶帘线封隔器胶筒工作性能及其影响因素[J]. 天然气工业,2020,40(1):97-103.
- [2] TAYLOR D M. Packer & safety valve development for ultra high pressure high temperature test & production wells[C]//All Days. Houston, Texas, USA: OTC,2012:1-13.
- [3] LEKONTSEV Y, TEMIRYAEVA O A, SAZHIN P V. Laboratory tests of the packer sealing elements[J]. IOP Conference Series:

Earth and Environmental Science,2019,262:012040.

- [4] 张智,祝效华,许建波. 基于正交试验的压缩式封隔器胶筒的结构参数优化[J]. 天然气工业,2019,39(3):80-84.
- [5] 刘清友,黄云,湛精华,等. 井下封隔器及其零件工作行为仿真研究[J]. 石油机械,2005,33(6):23-26.
- [6] 刘清友,黄云,湛精华,等. 井下封隔器接触有限元模型研究[J]. 钻采工艺,2005,28(2):58-60,67.
- [7] 刘巨保,党黎明,付茂青,等. 封隔器胶筒轴向压缩大变形与双重接触力学分析[J]. 石油机械,2014,42(7):49-54.
- [8] 水浩澈,张付英,杨俊梅. 温度变化对压裂封隔器胶筒密封性能和疲劳寿命的影响研究[J]. 机械设计,2020,37(6):43-48.
- [9] 章娅菲,窦益华,罗睿. 基于 ZSM-5 型沸石-水系统的封隔器胶筒材料密封性能研究[J]. 润滑与密封,2018,43(10):12-17,22.
- [10] 张付英,水浩澈,张玉飞,等. 压裂用封隔器双胶筒组合结构的密封性能研究[J]. 机械设计,2019,36(11):14-19.
- [11] 李早元,胡光辉,刘健,等. 碳酸盐岩选择性固井管外封隔器失效原因探讨[J]. 石油钻采工艺,2014,36(1):100-103.
- [12] 刘奇,赵彬,高文磊. 基于 ABAQUS 的舌形密封圈结构参数分析[J]. 机械制造与自动化,2021,50(5):73-75.
- [13] 周梦雨,李凡珠,杨海波,等. 橡胶材料的非线性黏弹性本构方程[J]. 高分子材料科学与工程,2020,36(3):79-84.
- [14] 杨锋,郭然. 多边形应力杂交单元的接触算法研究[J]. 应用数学和力学,2019,40(10):1059-1070.
- [15] 张康,张仕民,丁庆新. 管内封堵器胶筒硬度对其自密封特性影响研究[J]. 石油矿场机械,2018,47(2):8-12.

收稿日期:2021-12-03

(上接第 86 页)

子公司的 Plant simulation 15.0 作为开发工具,开发了包裹智能分拣系统仿真平台,对本文所设计分拣路径动态规划方法的效果进行分析。仿真结果表明所设计包裹定向分拣路径动态规划方法,能够显著降低包裹平均分拣时间,从而提升包裹智能分拣系统的分拣效率。

参考文献:

- [1] 张阳,马如意,刘聪雨,等. 快递集散中心分拣作业优化研究[J]. 工业工程与管理,2018,23(6):95-100,124.
- [2] 翟龙真. 基于人因工程学的快递分拣作业优化研究[D]. 衡阳:南华大学,2016.
- [3] 贺学成,吕淑静,吕岳. 高密度度 AGV 快递包裹分拣系统的路径规划[J]. 计算机系统应用,2019,28(4):39-44.
- [4] 黄梦涛,黎译繁. 面向快递分拣的二维码定位与校正方

法[J]. 科学技术与工程,2019,19(3):153-157.

- [5] 王立平. 基于柔性分拣单元的高效智能分拣系统研究[D]. 南京:南京航空航天大学,2018.
- [6] 杨伟国. 基于全向轮的可编程输送平台的设计与研究[D]. 淮南:安徽理工大学,2020.
- [7] PALACÍN J, MARTÍNEZ D, RUBIES E, et al. Suboptimal omnidirectional wheel design and implementation[J]. Sensors, 2021,21(3):865.
- [8] 叶颖诗,魏福义,蔡贤资. 基于并行计算的快速 Dijkstra 算法研究[J]. 计算机工程与应用,2020,56(6):58-65.
- [9] 杨振泰,黎向锋,张立果,等. 基于 Plant Simulation 的柔性作业车间仿真研究[J]. 机械制造与自动化,2020,49(3):87-89,122.

收稿日期:2021-10-29