

# 考虑转移策略的物料配送和空箱回收集成优化

陆志强, 王韵熹

(同济大学 机械与能源工程学院, 上海 201804)

**摘要:** 为了降低飞机移动装配线物料供给成本, 构建了考虑转移策略的物料配送和空箱回收集成决策模型并设计了解算法。在配送和回收任务调度的基础上引入了转移任务和线边物料存放的决策, 以最小化小车出行成本为目标建立了数学模型并设计了一种以混沌差分进化算法为框架的双层迭代算法。算法上层对配送物料的分批进行搜索, 综合考虑小车装载能力和线边空间约束, 通过启发式算法对配送小车的发车时间和线边存储位置进行决策, 结合转移小车的线边循环运作方式, 设计了修复算子对调度和存储方案进行再优化; 算法下层对无法即时装载的空箱组批进行搜索, 使用转移调度算法进行解码。通过数值实验验证了该模型和算法的有效性。

**关键词:** 飞机移动装配线; 空箱转移; 物料配送; 线边存储; 混沌差分进化算法

中图分类号: TP29

文献标志码: A

## Integrated Optimization of Material Delivery and Container Pickup Problem Considering Transfer Strategy

LU Zhiqiang, WANG Yunxi

(College of Mechanical Engineering, Tongji University, Shanghai 201804, China)

**Abstract:** An integrated material delivery and container pickup decision-making model considering transfer strategy and corresponding algorithm were constructed to reduce the material supply cost of aircraft moving assembly line. The decision of transfer task and line-side storage was introduced on the basis of delivery and recycle task scheduling. A mathematical model was established to minimize the travel cost, and a two-level iterative algorithm based on chaotic differential evolution algorithm was designed. The upper layer of the algorithm searches the batches and makes decisions on the delivery time and storage location of the jobs through heuristic

algorithms comprehensively by taking the delivery capability and line-side space constraints into consideration. Combined with the line-side cycle mode of the transfer vehicle, repairing algorithms were designed to re-optimize the scheduling and storage scheme of materials. The lower layer of the algorithm searches for batches of container that cannot be loaded immediately, and decodes them using the transfer scheduling algorithm. Numerical experiments verify the effectiveness of the model and algorithm.

**Key words:** aircraft moving assembly line; container transfer; material delivery; line-side storage; chaotic differential evolution algorithm

飞机总装周期约占整个飞机制造周期的 25%~40%<sup>[1]</sup>。为了缩短总装周期, 近年来逐渐引入更加稳定和高效的移动装配技术<sup>[2]</sup>。但这种装配方式对精准配送提出了更高的要求, 亟需建立一个有效而可靠的配套物料供给系统。

飞机装配过程中所需的物料可以分为 4 类: 通用标准件、大型结构件、装配件和自制件。通用标准件(如螺栓等)一般在线边存储, 采用定期补货的方式供给; 大型结构件(如发动机、机翼等)通常使用叉车和机械臂等其他方式直接配送。本文研究问题中的物料主要指体积中等、种类复杂的装配件, 由于零部件组装的相关性, 通过齐套的方式使用标准料箱以准时化的方式进行配送, 并在作业完成后对空箱进行回收。

目前以飞机移动装配线为背景研究物料供给问题的文献较少, 而以汽车装配线为背景的研究相对成熟且物料供给过程与飞机类似, 因此可以借鉴汽车装配线中物料供给问题的模型与算法作为参考。Sali 等<sup>[3]</sup>对库存、齐套和顺序配送这三种常用的物料

收稿日期: 2022-01-13

基金项目: 国家自然科学基金(61473211, 71171113)

第一作者: 陆志强(1968—), 男, 教授, 工学博士, 主要研究方向为物流与供应链建模与优化。

E-mail: zhiqianglu@tongji.edu.cn。



论文  
拓展  
介绍

配送策略进行了描述并总结了各种配送方式的适用环境。Souza等<sup>[4]</sup>研究装配线准时化物料配送问题,以最小化搬运费用和库存费用为目标建立数学模型,提出了具有贪婪随机自适应搜索的启发式算法。Boysen等<sup>[5]</sup>以最小化最大库存水平为目标函数建立了汽车装配线叉车配送模型,并设计了动态规划算法进行求解。但文献[5]假定每个料箱的物料量为定值,在实际应用中可能出现缺料的情形,因此Zhou等<sup>[6]</sup>对假设进行了改进,在决策变量中引入了每次配送物料量的选择,同时设计了反向动态规划算法对该问题进行求解。上述问题均考虑的是点对点的配送问题,每次只配送一个工位。Fathi等<sup>[7]</sup>提出了milk-run配送方式下多载量小车装载和调度问题,即小车每次可以装载多个工位所需物料。Emde等<sup>[8]</sup>在此基础上对小车发车时间与配送工位进行了集成决策。Golz等<sup>[9]</sup>研究了多辆小车的组批和调度问题,设计了两阶段分解算法最小化配送人员数量。Zhou等<sup>[10]</sup>考虑了以电动机器人作为送料工具的情况,提出了一种二次搬运的策略,通过物料在不同机器人之间的转运降低了物料供给过程的能耗以及机器人的数量。

上述文献对汽车装配线的物料供给问题进行了充分的研究,但汽车装配线模型与飞机装配线模型存在区别,将其直接应用到飞机移动装配线中存在局限性。汽车装配线每个工位装配作业单一,一般采用固定料箱,研究的重点集中于小车的装载和调度问题;而飞机移动装配线装配周期长且并行作业多,不对线边空间进行合理的分配会造成物料存储空间不足等问题。胡鑫铭等<sup>[11]</sup>建立了飞机移动装配线物料配送与线边存储集成优化模型,在物料组批和小车调度的基础上,引入了物料在线边空间的存储决策。但文献[11]假设空箱在装配完成后统一回收,忽略了空箱对线边空间的占用以及回收所需要的物流成本。陆志强等<sup>[12]</sup>建立了飞机移动装配线物料配送与空箱回收集成决策模型,考虑让小车在单次出行中同时执行配送和回收任务,相比独立回收策略减少了出行次数。但是由于小车在线边只能按位置顺序执行各线边单元上的任务,因此需要在小车从中心仓库出发时就预留好空箱的存放空间,提高了小车返回仓库时的装载率却降低了配送物料时的装载率,相比独立回收策略成本优化有限。

事实上,对飞机移动装配线的物料配送和空箱回收进行集成决策存在如下权衡:在时间上,对物料分批和发车时间进行决策时需要协调配送和回收任

务的时间约束;在线边空间上,料箱和空箱共享线边存放空间,不对料箱到达线边的时间以及空箱离开线边的时间进行科学的决策就会导致存放空间不足;在小车的装载能力上,小车装载率受到空箱和料箱存放位置前后关系的影响,不对线边存储进行合理的规划会降低两类任务调度的集成度,增加出行成本。因此,对物料配送和空箱回收任务的集成决策展开研究对于保障飞机移动装配线物料供给系统的稳定性和效率具有重要意义。

基于对上述文献研究成果的分析,本文在文献[12]提出的物料供给问题中引入转移策略和线边存放位置的决策,提出了考虑转移策略的物料配送和空箱回收集成优化问题(Integrated optimization of material delivery and container pickup problem considering transfer strategy, IOMDCP-TS)。以飞机的一个装配节拍为决策周期,综合考虑小车装载能力与线边空间重复共享等因素,对料箱和空箱的组批方式与其在线边存放的位置与时间进行决策,建立了以最小化小车出行成本为目标函数的数学模型并设计了以混沌差分进化算法为框架的启发式算法对该模型进行求解。其中结合了发车时间动态调度算法、存放位置决策算法和转移调度算法。最后通过数值实验验证了该算法的可行性和有效性。

## 1 问题描述和数学模型

飞机移动装配线物料供给问题,是通过优化各作业所需物料的配送、回收时间及其在线边空间的存放位置,以达到最小化小车出行成本的目标。图1为飞机移动装配线的物料供应模式,整条装配线被

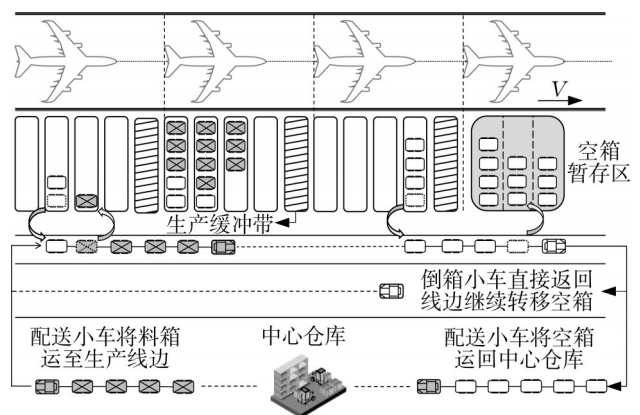


图1 考虑转移策略的物料配送及空箱回收模型示意图  
Fig.1 Material delivery and container pickup model considering transfer strategy

划分为多个“虚拟大工位”,每个大工位内存在大量的并行装配作业,在工位旁划定相关的区域用于存放各作业装配所需的物料,并由多载量小车将物料按照装配计划从中心仓库配送至对应的线边单元,同时将各线边单元上的空箱运回中心仓库。

小车装载率受到空箱和料箱在线边存放位置的相互关系的影响,因此。为了提高小车的装载率,减少小车出行成本,本文提出了转移策略,即允许小车将空箱放至线边末尾的空箱暂存区后直接返回装配线前端,由其他小车将暂存区的空箱运回中心仓库。通过该策略可以调整空箱的存放位置以及在线边的存放时间,减少小车从中心仓库到线边的往返次数并优化线边物料的存放状态。称执行料箱配送任务和空箱回收任务的小车为配送小车,执行空箱转移任务的小车为转移小车。

该问题的假设与符号说明如下:

(1)以飞机移动装配线的一个装配节拍时间为决策周期。假定该周期内某大工位包含  $n$  项装配作业,  $J = \{1, 2, \dots, n\}$ , 任意作业  $j \in J$  的执行工期为  $t_j$ , 开始时间为  $T_{S,j}$ , 完成时间  $T_{F,j} = T_{S,j} + t_j$ 。

(2)将时间离散化,时间集合为  $D = \{1, 2, \dots, Z\}$ ,  $Z$  表示所有作业操作时间总和,任意  $d \in D$  表示时间节点。  $t_{w,j} = \{dt = T_{DS,j} \dots T_{DF,j}\}$  表示作业  $j$  在线边存放的时间窗。  $J_l$  表示线边单元  $l$  的作业集合,  $J_l \in J_l =$

$\{j_m \dots j_n | \bigcap_{j=m}^n t_{w,j} \neq \Phi\}$  表示线边单元  $l$  第  $i$  个时间窗交集不为空集的作业集合。

(3)  $K_a$  表示配送小车集合,  $k \in K_a = \{1, 2, \dots, h\}$ ;  $K_b$  表示转移小车集合,  $k \in K_b = \{1, 2, \dots, h\}$ 。小车从仓库到线边的运输时间为  $t_1$ , 从装配线前端到末端的移动时间为  $t_b$ , 飞机移动速度为  $v$ , 表示单位时间内经过的线边单元数量。任意配送行程  $r$  不得超过装载能力上限  $C_m$ ,  $r \in R_k = \{1, 2, \dots, \pi\}$ ,  $R_k$  表示小车出行趟数集合。

(4)同一个作业的料箱由一辆小车一次性配送和回收,一辆小车一次可配送或回收多项作业所需的物料,作业  $j$  对物料的需求量为  $m_j$ 。作业  $j$  的物料抵达线边的时间为  $T_{DS,j}$ , 离开线边的时间为  $T_{F,j}$ , 配送提前期  $t_{p,j} = T_{S,j} - T_{DS,j}$ , 回收滞后期  $t_{l,j} = T_{F,j} - T_{l,j}$ 。

(5)将线边空间和作业空间离散化。如图 2 所示,作业空间描述了各作业在飞机上的装配位置,用集合  $\Gamma = \{1, 2, \dots, \varphi\}$  表示,  $\varphi$  表示飞机的长度,任意

$l_j \in \Gamma$  表示作业  $j$  在飞机上的装配位置;线边空间描述了物料在线边的存储位置,用  $l \in L = \{1, 2, \dots, \xi\}$  表示线边单元的集合,  $\xi$  表示最大线边单元的编号;用  $c_l = \{1, 2, \dots, \psi\}$  表示各线边单元内存储位置的集合,  $\psi$  表示线边单元的容量。对于任意一架飞机,状态 A 表示飞机开始进入装配线,移动到状态 B 的位置开始执行作业  $j$ , 移动到状态 D 的位置作业  $j$  执行结束。状态 C 表示该作业执行过程飞机的中间位置,定义该状态下作业  $j$  装配位置所对应的线边单元为该作业物料的中心存储位置,即  $l_j = \lceil l_j + vT_{S,j} + \frac{1}{2}vt_j \rceil$ ,  $\lceil \cdot \rceil$  表示向上取整。

(6)为减少取料距离,作业  $j$  的物料只能存放在  $l_{m,j}$  及其左右两侧各 1 个单位的线边空间内。为使线边物料存放整齐有序,要求任一作业的全部物料应连续存放于同一线边单元内。

(7)为了避免同一工段内前后两架飞机所需物料在线边发生重叠,在每个大工位之前设置生产缓冲带,该区间内不进行装配作业。配送小车发车时刻晚于最大提前量约束  $t_{\min}$ 。

(8)在装配线末端设置空箱暂存区,转移小车将空箱从线边单元取下暂存在该区域内,由配送小车运回中心仓库。

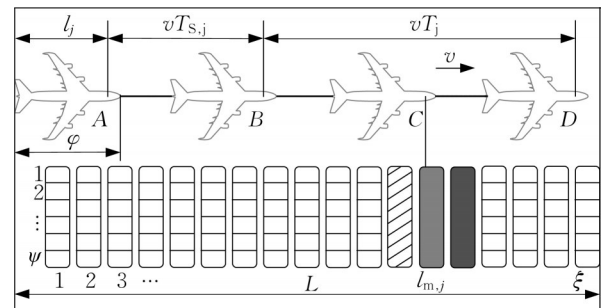


图2 飞机作业装配点及物料存放位置

Fig.2 Assembly positions and material storage on aircraft moving assembly line

定义决策变量如下:

$x_{j,r_k} \in \{0, 1\}$ , 作业  $j$  的料箱在配送小车  $k$  第  $r$  趟出行配送时则  $x_{j,r_k}$  等于 1, 否则取 0。

$y_{j,r_k} \in \{0, 1\}$ , 作业  $j$  的空箱在配送小车  $k$  第  $r$  趟出行时回收则  $y_{j,r_k}$  等于 1, 否则取 0。

$z_{j,r_k} \in \{0, 1\}$ , 作业  $j$  的空箱在转移小车  $k$  第  $r$  趟出行时转移则  $z_{j,r_k}$  等于 1, 否则取 0。

$(t_{a,r_k}, t_{b,r_k}, l_j), t_{a,r_k}$  表示配送小车  $k$  第  $r$  次出行的时

间,  $t_{b,r_k}$  表示转移小车  $k$  第  $r$  次出行的时间,  $l_j$  为作业  $j$  的存放单元。

定义辅助变量如下:

$\lambda_{a,r_k} = 1 - (1 - \sum_{j=1}^n x_{j r_k})(1 - \sum_{j=1}^n y_{j r_k})$ , 若配送小车  $k$  在第  $r$  次出行中有配送或者回收物料则  $\lambda_{a,(r_k)}$  等于 1, 否则取 0。

$$\lambda_{b,r_k} = \begin{cases} 0, & \sum_{j=1}^n z_{j r_k} = 0 \\ 1, & \text{其他} \end{cases}, \text{若转移小车 } k \text{ 在第 } r \text{ 次出}$$

行中有转移物料则  $\lambda_{b,r_k}$  等于 1, 否则取 0。

$y_{d,j r_k} = (1 - \sum_{r_k=1}^R \sum_{k=1}^K z_{j r_k}) \sum_{r_k=1}^R \sum_{k=1}^K y_{j r_k}$ , 若转移小车  $k$  在第  $r$  次出行中回收作业  $j$  的空箱而且该作业的空箱不需要转移则  $y_{d,j r_k}$  等于 1, 否则取 0。

$f_k = 1 - (1 - \sum_{j=1}^n \sum_{r_k=1}^R x_{j r_k})(1 - \sum_{j=1}^n \sum_{r_k=1}^R y_{j r_k})$ , 若小车  $k$  投入使用则  $f_k$  等于 1, 否则取 0。

根据上述模型假设和符号定义, 对该问题建模如下:

$$\text{Min } Z = C_a \sum_{k=1}^h \sum_{r=1}^{\pi} \lambda_{a,r_k} + C_b \sum_{k=1}^h \sum_{r=1}^{\pi} \lambda_{b,r_k} + C_c \sum_{k \in K_a \cup K_b} f_k \quad (1)$$

$$s.t. (t_{a,r_k} + t_l) \leq T_{s,j} + M(1 - x_{j r_k}), \forall j \in J, \forall k \in K, \forall r_k \in R_k \quad (2)$$

$$M(1 - y_{j r_k}) + t_{a,r_k} + t_l \geq T_{F,j}, \forall j \in J, \forall k \in K, \forall r_k \in R_k \quad (3)$$

$$M(1 - z_{j r_k}) + t_{b,r_k} + t_l \geq T_{F,j}, \forall j \in J, \forall k \in K, \forall r_k \in R_k \quad (4)$$

$$t_{b,r_k} \leq M(2 - y_{j r_k} - z_{j r_k}) + t_{a,r_k}, \forall j \in J, \forall k \in K, \forall r_k \in R_k \quad (5)$$

$$t_{a,r_k} \geq t_{\min}, \forall k \in K, \forall r \in R \quad (6)$$

$$\sum_{r=1}^{\pi} \sum_{k=1}^K x_{j r_k} = 1, \forall j \in J \quad (7)$$

$$\sum_{r=1}^{\pi} \sum_{k=1}^K y_{j r_k} = 1, \forall j \in J \quad (8)$$

$$\sum_{r=1}^{\pi} \sum_{k=1}^K z_{j r_k} \leq 1, \forall j \in J \quad (9)$$

$$\sum_{j=1}^n x_{j r_k} m_j + \sum_{l=1}^{\xi} \sum_{j \in J_l} (y_{d,j r_k} - x_{j r_k}) m_j \leq C_m, \forall k \in K, \forall r \in R \quad (10)$$

$$\sum_{j=1}^n z_{j r_k} m_j \leq C_m, \forall r \in R, \forall k \in K_b \quad (11)$$

$$\sum_{j \in J_l} m_j \leq \psi, \forall J_l \in J_l, \forall l \in L \quad (12)$$

$$l_{m,j} = l_j + v T_{s,j} + t_j \frac{v}{2}, \forall j \in J \quad (13)$$

式(1)为目标函数, 最小化小车出行成本, 包括配送小车和转移小车的出行次数以及车辆的固定成本。其中,  $c_a, c_b, c_c$  分别是三者的单位成本系数; 式(2)表示每个作业的料箱要在作业开始时间之前送达; 式(3)~(5)表示每个作业的物料只能在作业完成之后回收或转移, 同时转移时间要早于回收时间; 式(6)表示作业配送时间要满足最大提前量约束。式(7)~(9)表示单个作业物料的运输任务均只能由同一辆小车执行; 式(10)~(11)表示小车在整个行驶过程中都不能超过其载量上限; 式(12)表示任意时刻线边单元物料暂存量均不应超过其最高限度; 式(13)表示物料中心存放位置为其装配过程中的中心位置。

## 2 算法设计

Boysen 等<sup>[13]</sup>研究了 IOMDCP-TS 问题的一个子问题(从仓库向单工位组批配送物料, 目标为优化小车配送次数), 并证明了该问题为强 NP-hard 问题。而 IOMDCP-TS 问题除了小车的配送调度外, 还要考虑物料的存放以及小车的转移调度, 具有较高的复杂度。精确算法在求解这类大规模问题时效率较低。差分进化算法具有速度快、鲁棒性好等优点, 被广泛应用于生产调度等问题<sup>[14-15]</sup>。因此本文针对 IOMDCP-TS 问题的特点设计了以混沌差分进化算法为框架的启发式算法(HBCDE)。该算法上层对物料的组批方式进行搜索, 下层对每个包含物料组批方案的个体依次采用发车时间动态调度算法、存放位置决策算法和转移调度算法进行解码; 最后使用两种修复算子对不可行解进行修复。算法迭代过程中适应度较高的个体所对应的物料组批方案较优, 将作为优质解保留在记忆池中; 低适应度的个体将被淘汰或通过变异更新。设定  $N_m$  为最大迭代次数,  $g$  为当前迭代次数。算法框架如图 3 所示。

### 2.1 差分进化算法框架

本文采用离散型差分进化算法, 使用双层整数编码。编码的列数为作业的数量  $n$ , 编码数字为不大于作业数量的整数。上层数字代表该作业物料配送的批次数, 下层数字代表该作业物料回收的批次数。如图 4 所示作业 1 的物料在第二批配送, 在第三批回收。为了提高算法搜索的效率, 在初始种群

中加入了两条分别基于作业开始时间(FCFS)和作业结束时间(LFT)进行排序的个体。

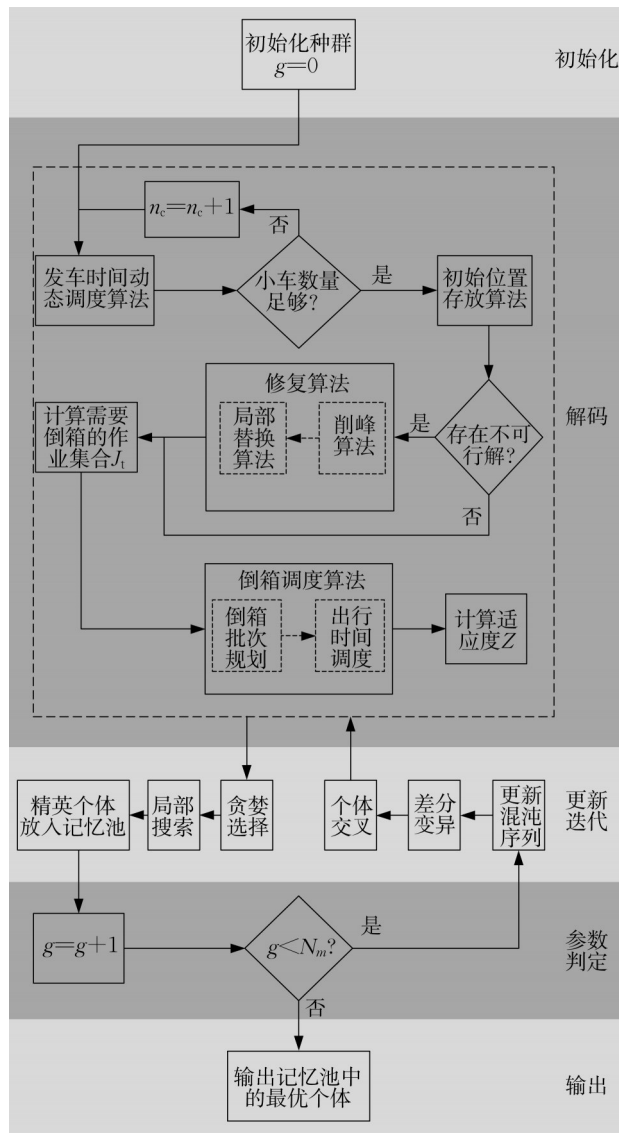


图3 算法结构

Fig.3 algorithm structure

对于违反载量或时间约束的作业,从编号最小的开始执行就近转移规则,将该作业移入离当前批次最近的批次,直到该作业满足约束。如果所有批次都无法存放该作业则为该个体添加惩罚值。

混沌序列具有随机性、遍历性等特点<sup>[16]</sup>。为了防止算法早熟收敛,本文在差分变异算子中引入混沌序列对比例因子进行动态调整,根据 logistic 方程  $y_{n+1,d} = \mu y_{n,d}(1 - y_{n,d})$  产生混沌序列,然后通过载波方式将混沌变量映射到比例因子的取值范围  $[0, 2]$ 。式中:  $n \in [1, N_{\max}]$ ,  $N_{\max}$  取 100,  $d$  为变量的维度,  $\mu$  是混沌状态的控制参数,本文取 4,该参数下 logistic 方程完全进入混沌状态。在离散差分进化算法中缩放比例因子根据  $F_{ch} = y_{n,d} \exp(-t/t_{\max})$  动态产生,差分进化中的变异算子如式(14)所示,其中  $r_1, r_2, r_3$  为与  $i$  不同的互不相同的 3 个随机个体。关于差分进化算法的通用部分不再赘述。

$$v_{r,j}^{g+1} = x_{r_1,j}^g + F_{ch}(x_{r_2,j}^g - x_{r_3,j}^g) \quad (14)$$

### 2.2 发车时间动态调度算法

在物料分批确定的情况下,对批次发车时间进行优化可以减少物料在线边的占用时间。各批次的发车时间会受到最早作业开始时间和最晚空箱回收时间的约束。在该范围内,若料箱较多而发车时间偏早会导致配送提前期过长;若空箱较多而发车时间偏晚则会导致回收滞后期长。二者均会使料箱在线边的占用时间延长,导致线边存放空间不足。在参考文献[13]提出的反向动态调度技术的基础上,基于以下定理提出了发车时间动态调度算法,该算法能保证每个批次物料在线边的总占用时间最短。应用该算法得到的物料占用时间作为中间解,为后续存放位置决策算法的应用奠定基础。

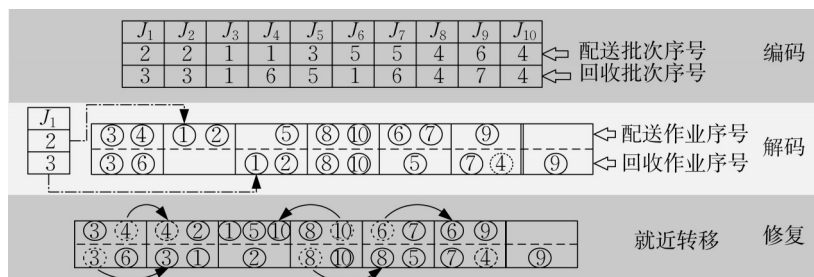


图4 编解码及修复示意图

Fig.4 Encoding, decoding and repairing representation

定理:对于包含配送作业集合  $J_x$  和回收作业集合  $J_y$  的作业批次  $B_i$ , 若满足  $\min\{t_{ea,(r+1)_k} -$

$2t_0 | \forall k \in K, \forall r_k \in R_k\} < \min\{t_{s,j} - t_0 | \forall j \in B_i\}$ , 那么使得该批次物料在线边占用最小的发车时间为

$$t_{\text{opt},i} = \begin{cases} \max \{ t_{i,j} - t_0 | \forall j \in B_i \}, i=1 \\ \min \{ t_{s,j} - t_0 | \forall j \in B_i \}, \sum_{j \in J_s} m_j - \sum_{j \in J_k} m_j > 0 \\ \max \{ \max \{ t_{i,j} - t_0 | \forall j \in B_i \}, \min \{ t_{\text{opt},(r+1)_k} - 2t_0 | \forall k \in K, \forall r_k \in R_k \} \}, \text{其他} \end{cases}, i > 1$$

证明如下:

发车时间规划这一子问题的目标函数是在使得物料在线边的占用在时间和空间两个维度上最小,也就是使得非必需存放时间和物料量的乘积  $\sum_{j \in B_i} (t_{p,j} + t_{l,j}) m_j$  最小。

设  $t_{b,s,i} = \min \{ t_{s,j} - t_0 | \forall j \in B_i \}$ ,  $t_{b,i} = \min \{ t_{l,j} - t_0 | \forall j \in B_i \}$ , 则该函数可以表示为

$$\begin{aligned} & \sum_{i \in I_s} (t_{b,s,i} - t_i) m_i + \sum_{i \in I_l} (t_i - t_{b,i}) m_i + \sum_{i \in I_k} (t_{b,i} - t_i) m_i + \sum_{i \in I_r} (t_i - t_{b,i}) m_i \\ & = (\sum_{i \in I_s} m_i - \sum_{i \in I_k} m_i) t_i + \sum_{i \in I_k} m_i t_{b,s,i} - \sum_{i \in I_k} m_i t_{b,i} \end{aligned}$$

该函数为  $t_i$  的单调函数, 取值由  $t_i$  的系数 ( $\sum_{i \in I_s} m_i - \sum_{i \in I_k} m_i$ ) 决定。

当  $\sum_{i \in I_s} m_i - \sum_{i \in I_k} m_i \leq 0$  时, 总占用量为单调递减的函数, 需要满足批次开始时间约束  $t_{\text{opt},i} \leq \min \{ t_{s,j} - t_0, \forall j \in B_i \}$ , 令  $t_i = \min \{ t_{s,j} - t_0, \forall j \in B_i \}$  可使得总占用量最小。

当  $\sum_{i \in I_s} m_i - \sum_{i \in I_k} m_i > 0$  时, 总占用量为单调递增的函数, 需要满足批次完成时间约束  $t_{\text{opt},i} \geq \max \{ t_{l,j} - t_0, \forall j \in B_i \}$  和发车时间间隔约束  $t_{\text{opt},i} \geq \min \{ t_{\text{ea},(r+1)_k} - 2t_0 | \forall k \in K, \forall r_k \in R_k \}$ 。令  $t_i = \max \{ \max \{ t_{l,j} - t_0, \forall j \in B_i \}, \min \{ t_{\text{ea},(r+1)_k} - 2t_0 | \forall k \in K, \forall r_k \in R_k \} \}$ , 总占用最小。

发车时间的动态规划算法步骤如下:

(1) 获取作业分批方案  $\Omega = \{ B_1, B_2 \dots B_i \dots \}$ 。初始化小车数量  $n_c = 1$ , 最早可用时间  $t_{\text{ea},r_c} = t_{\text{min}}$ 。

(2) 令  $i = 1$ , 选择首个批次的物料集合  $B_i$ , 安排其发车时间  $t_{B_i} = \max \{ t_{l,j} - t_0 | \forall j \in B_i \}$ 。

(3) 若  $\min \{ t_{\text{ea},(r+1)_k} - 2t_0 | \forall k \in K, \forall r_k \in R_k \} \geq \min \{ t_{s,j} - t_0 | \forall j \in B_i \}$ ,  $n_c = n_c + 1$ ,  $k = n_c$ ,  $t_{B_i} = \max \{ t_{l,j} - t_0 | \forall j \in B_i \}$ ,  $t_{\text{ea},(r+1)_k} = t_{B_i} + 2t_1$ ; 否则  $k = k^*$ , 对于任意  $k \in K_a$ , 有  $t_{\text{ea},r_k} < t_{\text{ea},(r+1)_k}$ ,  $t_{B_i} = t_{\text{opt},i}$ 。

(4) 更新该车的最早可用时刻为  $t_{\text{ea},(r+1)_k} = t_{r_k} + 2t_0$ 。

(5) 如果  $i = |\Omega|$ , 转(6), 否则  $i = i + 1$ , 转(3)。

(6) 输出该组批方案下各批次物料的配送时刻  $T_\Omega = \{ t_{B_1}, t_{B_2} \dots t_{B_{|\Omega|}} \}$ 。

### 2.3 存放位置决策算法

在确定了物料在线边的占用时间之后, 可以将带时间维度的线边存放区看作一个长、宽、高分别对应于装配节拍、线边长度和单位线边单元容量的三维容器。胡鑫铭等<sup>[15]</sup>将物料在线边的存放过程抽象为了一类带单边时间窗的三维装箱问题。但该模型仅考虑了物料提前配送对于线边的非必需占用, 而在实际情况中回收的滞后也会导致空箱对于线边空间的占用, 因此本文考虑了一个带双边时间窗的三维装箱问题并为该问题设计了物料初始存储位置生成算法并为无法存放的物料设计了修复算法。

物料的初始存储位置生成算法如下: 首先将线边单元  $l_i$  的作业集合  $J_{l_i}$  中的所有作业按照  $m_j (T_{D_{r,j}} - T_{D_{s,j}})$  降序排列, 按照该顺序依次挑选作业放入其对应线边单元, 判断是否满足三个维度上的约束, 满足则输出该作业存放位置; 若其存放范围内的线边单元均不满足则将其放入无法存放作业集合。

由于在配送小车发车时间的调度的过程中松弛了线边容量的约束, 因此可能会出现作业无法存放的情况。对于无法存放的作业集合  $N_p$ , 采取局部替换和削峰算法两种算子对不可行解进行修复。首先使用不会增加额外成本的局部替换算子, 若不能修复则采用存在额外成本的削峰算子进行修复。

(1) 局部替换: 通过在局部范围内对作业存放位置进行互换调整使线边存量满足约束。主要步骤为: 对于无法存放的作业  $j_{\text{un}}$ , 从其可存放范围中按照剩余空间降序选择线边单元, 按物料量升序选择作业直到可以替换  $j_{\text{un}}$ 。对新的  $j_{\text{un}}$  首先判断能否直接放入, 若没有则重复相同操作直到不存在无法存放的作业。

(2) 削峰算法: 参考文献[17]提出的削峰操作, 利用转移策略将空箱离开线边的时间提前, 等同于

削减物料占用立方体的长,解决因作业存放时间重合导致空间不足的问题。

削峰算法步骤如下:

(1)获取无法存放的作业集合  $N_p$ ,令未被选择的作业集合  $N_o = N_p$ 。

(2)从  $N_o$  中随机取出一个作业物料  $j$ ,令  $j_{um} = j, l_{ip} = l_{m-1,j}$ 。

(3)将线边单元  $l_p$  的作业集合的子集  $J_{l_i}$  按照  $\sum_{j \in J_{l_i}} m_j (T_{D_{F,j}} - T_{D_{S,j}})$  升序排列。

(4)按顺序遍历各子集,判断将能否通过将该子集中作业的离开线边时间提前至作业结束时间的方式来将该作业放入。若  $\exists J_{l_i}$ ,加入作业  $j$  后仍然满足对于  $\forall t, m_{l_i,t} < \varphi$ ,对于  $\forall j \in J_{l_i}$ ,令  $T_{F,j} = T_{D_{F,j}}, J_{l_i} = J_{l_i} \cup j_{um}, N_p = N_p - j_{um}$ ,转 Step5;若  $\nexists J_{l_i}$  且  $l_p \neq l_p + 1$ ,令  $l_p = l_p + 1$ ,转(3);否则转(5)。

(5)若  $N_o \neq \Phi$ ,转(2),否则转(6)。

(6)若  $N_p \neq \Phi$ ,添加惩罚值到目标函数中;否则输出存放位置集合  $L = \{l_1, l_2, \dots, l_n\}$  和需要削减存放时间的作业集合  $J_p$ 。

如图5所示,有7个作业的物料被存放在线边,立方体的长代表该作业的物料在线边存放的时间窗  $[T_{D_{S,j}}, T_{D_{F,j}}]$ ,高代表该作业的物料量。矩形中深色的部分代表作业的在线边存放的硬时间窗  $[T_{S,j}, T_{F,j}]$ ,此时间窗无法调整;带斜线的浅色部分代表作业的物料的配送提前期  $t_{p,j}$  以及存放滞后期  $t_{l,j}$ ,可以通过更改发车时间来对其进行调整。如图5所示,按照初始存储位置,作业5的物料会超出线边单元的容量。使用削峰算法将作业7在线边的时间窗从  $[T_{D_{S,7}}, T_{D_{F,7}}]$  缩短为  $[T_{D_{S,7}}, T_{F,7}]$ ,避免了作业5和作业7的物料在时间上的重合,修复后使该线边单元满足了最大容量约束。

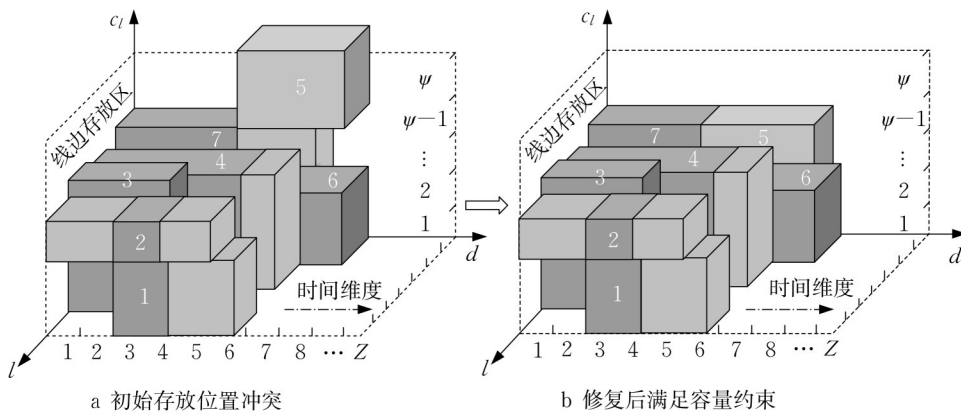


图5 削峰算法

Fig.5 Peak clipping algorithm

### 2.4 转移调度算法

通过2.2节和2.3节中的算法解码可以得到需要转移的作业集合  $J_t = J_p \cap J_u, J_p$  为超出线边容量约束需要减少回收滞后期的作业集合,由2.4节中的削峰算法计算得到。 $J_u$  表示无法即时装载的作业集合。计算  $J_u$  的算法步骤如下:

(1)获取出行批次  $\Omega = \{B_1, B_2 \dots B_i \dots\}$ ,按出行时间升序排列。

(2)选中第一个还未被选择的批次  $B_i$ ,初始化批次  $B_i$  的载量  $m_{B_i} = \sum_{j \in B_i} m_j$ 。

(3)遍历各线边单元,若该线边单元存在配送作业,  $m_{B_i} = m_{B_i} - m_j$ ;若存在回收作业且  $m_{B_i} + m_j \leq C_m$ ,则  $m_{B_i} = m_{B_i} + m_j$ ;若  $m_{B_i} + m_j > C_m$ ,将该作业加入转移作业集合。

(4)若所有批次都已被选择,输出无法即时装载的作业集合  $J_u$ ;否则转(2)。

将  $J_t$  按完成时间进行升序排列,使用最小批量法生成转移批次的初始解,以出行次数最少为目标对最优转移批次进行搜索。转移小车出发时间的决策问题是配送小车出发时间决策问题的一个特例,即各转移批次中只包含回收作业,不包含配送作业,因此最优发车时间依然适用于2.2节中所证明的定理。

### 3 数值实验

本文使用文献[11]所构建的算例集进行数值实验,参考了文献[18]关于出行成本的参数设置,设定单次配送小车行驶成本45\$,单次转移小车行驶成

本9\$,单辆小车固定成本45\$。小车容量 $C_m=20$ ,线边单元容量 $\xi=20$ ,中心仓库到线边装卸货的时间 $t_1=4$ ,线边运行时间 $t_b=1$ ,飞机移动速度 $v=0.5$ 。构建了30、60、120等3个规模的算例来进行实验,每个规模的算例集包括3组,每组10个算例。本文在Pycharm 2021.2.1(community edition)平台上使用Python语言进行编程测试,计算机配置为Ryzen 74 800H处理器,2.9GHz主频,16G内存。

### 3.1 策略对比

将本文提出的转移策略TS和文献[12]所提出的集成调度策略CHBG以及传统的两阶段调度策略CAS进行对比。CAS策略是指将物料配送和空箱回收作为两个独立的问题,分别进行调度,最后将总出行成本相加;CHBG策略是指对物料配送和空箱回收进行集成调度,一个批次可以同时执行这两种任务;TS策略是在CHBG策略的基础上,允许空箱在线边进行转移以优化料箱和空箱的存放次序。对三种策略采用相同的解码算法。对比的结果如表1所示,表格中 $C_i$ 代表策略 $i$ 物料配送的总成本, $G_i$ 代表策略 $i$ 的成本和三种策略中的最低成本之间的差距,计算如下:

$$G_i = \frac{C_i - C_{\text{best}}}{C_{\text{best}}} \times 100\% \quad (18)$$

对表1的数据进行分析可以得到如下实验结论:

(1)在成本上,在3种规模的算例下,TS策略都拥有最低的总成本,而且随着算例规模的上升,相比另外两种策略的优化幅度也进一步提升。在120的算例规模下,TS策略相比CHBG策略和CAS策略优化比例分别达到了19.40%和29.70%。

表1 不同策略出行成本对比

Tab.1 Comparison of travel cost between different models

算例规模/N	组别	TS		CHBG		CAS	
		C/\$	G/%	C/\$	G/%	C/\$	G/%
30	1	917.1	0	1 026	11.93	1 107	20.76
	2	929.7	0	1 062	14.27	1 143	23.02
	3	919.8	0	1 044	13.49	1 139	23.78
	平均值	922.2	0	1 044	13.23	1 130	22.52
60	1	1 826.1	0	2 111	15.57	2 318	26.91
	2	1 824.3	0	2 149	17.69	2 305	27.30
	3	1 804.5	0	2 070	14.78	2 270	26.38
	平均值	1 818.3	0	2 110	16.01	2 298	26.86
120	1	3 516	0	4 289	21.99	4 649	32.22
	2	3 636	0	4 260	18.39	4 669	28.91
	3	3 621	0	4 208	17.80	4 601	27.97
	平均值	3 591	0	4 252	19.40	4 640	29.70

(2)在可行解比例上,TS策略也拥有最高的求解成功率。如图6所示,在30规模的算例下,3种策略都能对所有算例成功求解;在60规模的算例下,TS策略依然能够成功求解所有算例,而CHBG策略和CAS策略都下降到了93.33%;在120的规模下,TS策略求解成功率达到了86.67%,而CHBG策略和CAS策略分别下降到了76.67%和73.33%。

综上所述,转移策略的引入能够有效降低物料配送和空箱回收集成模型的小车出行成本和不可行解比例,说明该策略在物料批次的划分和存放位置的决策上都具有优越性。

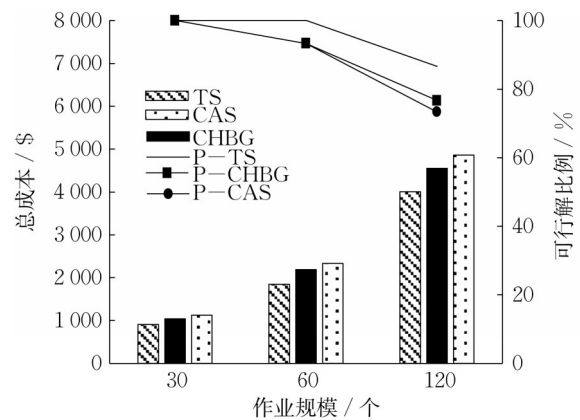


图6 不同策略成本及可行解比例对比

Fig.6 Comparison of cost and proportion of feasible solution between different strategies

### 3.2 算法对比

为了验证HBCDE算法的性能,将该算法与去除修复算法的HBCDE-RA,标准DE算法和CMAES算法进行了对比。在30、60、120等3个规模的算例下进行了实验,每个算例集包括30个算例。用可行解比例和总成本两个指标作为衡量标准,分别用 $P$ 和 $C$ 进行表示。分析表2中的实验结果可以得到如下结论:

表2 不同算法结果对比

Tab.2 Comparison of experimental results between different models

算法	30		60		120	
	C/\$	P/%	C/\$	P/%	C/\$	P/%
HBCDE-RA	905.40	100.00	1 812.27	100.00	3 521.4	86.67
HBCDE	921.00	100.00	1 883.45	86.67	3 673.8	56.67
CMAES	935.10	100.00	1 932.14	80.00	3 755.4	56.67
DE	940.20	100.00	1 966.91	83.33	3 855.6	50.00

(1)在算法框架的对比中,HBCDE算法在3种规模的算例下,都拥有更低的成本和更高的求解成



功率,随着作业规模的增大,优势更加明显。在规模为120的算例下,相比HBCDE算法和CMAES算法,该算法在成本上分别优化了6.65%和4.49%,在求解成功率上与CMAES算法持平,比DE算法高出6.67%。说明本文使用的混沌差分进化算法在求解物料组批配送这一类组合优化问题上具有优越性,可以得到更高的求解质量和可行解比例。

(2)在修复算法的效果验证实验中,加入修复算法的HBCDE-RA算法比HBCDE算法成本降低了4.33%,求解成功率提高了20%,说明该修复算法在三维装箱这类问题的不可行解修复上具有良好的效果。

### 3.3 算例错位度对转移策略效果的影响

为了探究算例中影响转移策略有效性的因素,本文综合考虑小车运输机制和作业信息提出了一项以错位度(Dislocation degree, dld)为核心的评价指标,该指标表征了算例中空箱的回收任务与配送任务相互冲突的可能性。定义辅助变量 $D_{l,ij}$ ,代表两个算例之间的错位关系,如果作业 $j$ 完成时间和中心存放位置都小于作业 $i$ 则 $D_{l,ij}=1$ ,否则 $D_{l,ij}=0$ ,如式(19)所示。对于任意作业 $j$ ,错位度计算方式如式(20)所示, $|J|$ 表示作业数量。

$$D_{l,ij} = \begin{cases} 1, T_{F,j} < T_{S,i} \text{ 和 } l_{m,j} < l_{m,i} \\ 0, \text{其他} \end{cases} \quad (19)$$

$$dld = \frac{\sum_{j=1}^J \frac{m_j \sum_{i=1}^J D_{l,ij}}{\psi |J|}}{|J|} \quad (20)$$

分别绘制81个成功求解的算例的错位度和转移策略的优化幅度的折线图,如图7所示可以看出两条曲线的形状高度吻合。计算两个指标的回归方程为 $G = -0.0901 + 1.809dld$ 。方差分析表如3所示, $P$ 值表示样本间的差异由抽样误差所致的可

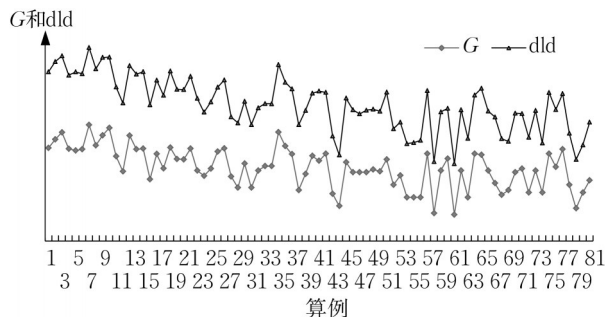


图7 错位度和优化幅度曲线的对比

Fig.7 Curve comparison of misalignment and optimization amplitude

能性,其值小于0.001说明错位度对于转移策略的优化幅度有显著的影响,而Pearson相关性系数为0.735,说明二者的关系为正相关。对于错位度高的算例采用转移策略可以降低更多的成本。该评估方法也可以应用到其他采用循环配送方式的料箱配送和回收集成调度问题中。

表3 错位度与优化幅度的回归分析

Tab.3 Regression analysis of misalignment and optimization amplitude

来源	自由度	Adj SS	Adj MS	F	P
回归	1	0.0849	0.084897	92.76	<0.001
错位度	1	0.0849	0.084897	92.76	<0.001
误差	79	0.0723	0.000915		
合计	80	0.1572			

### 3.4 成本系数对转移策略有效性的影响

成本系数之间的大小关系可能会影响该策略的有效性。为了探究二者比值处于什么范围内该策略有效,对三种规模的算例进行实验,绘制临界点的箱线图如图8所示。比值在临界点以下则该策略有效。临界点的计算方法为:以0.01为间隔生成总成本关于转移与配送成本系数比值的变化曲线,观察与CHBG策略成本的交叉点,以交叉点左侧的值为临界点。可以看出3组规模算例的临界值分布在[0.61,0.84]的区间内,规模越大,临界值越高。在具体装配线的决策过程中,可以通过计算这两者的比值是否达到临界值来判断采用转移策略是否有效。

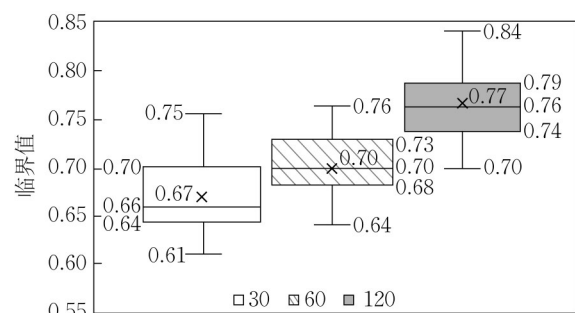


图8 不同规模算例下的临界值

Fig.8 Critical value of examples with different scales

## 4 结论与展望

(1)在飞机移动装配线中引入转移策略对物料供给模型进行优化。将料箱的配送、回收以及线边存储这三类互相影响的问题的决策整合到一个集成

模型当中,并设计了元启发式算法对该模型进行求解。通过数值实验验证了该模型和算法的有效性,为飞机移动装配线物料供给问题的研究提供了参考和借鉴。

(2)对影响转移策略有效性的两种因素进行了研究。首先提出了基于作业信息计算的预测指标“错位度”。使用回归方程对转移策略所能带来的优化幅度进行预测,通过方差分析验证了该指标的可靠性;其次研究了成本系数对调度性能的影响,通过数值实验界定了该策略的有效范围,为该策略的使用提供了参考。

(3)后续可以考虑使用机器学习对不同运行策略在不同算例下的效果进行预测,实现不同装配线情况下的动态调度,以应对具有高不确定性的作业环境。

#### 作者贡献声明:

陆志强:提出研究选题,设计研究思路和论文框架;

王韵熹:设计研究思路,完成实验设计并实施,分析数据,撰写论文。

#### 参考文献:

- [1] 温李庆. 基于Petri网的飞机总装配生产线建模及调度优化研究[D]. 杭州:浙江大学, 2015.  
WEN Liqing. Modeling and optimal scheduling for aircraft final assembly line based on Petri net [D]. Hangzhou: Zhejiang University, 2015.
- [2] ABDINNOUR S. Hawker beechcraft uses a new solution approach to balance assembly lines [J]. Interfaces, 2011, 41(2): 164.
- [3] SALI M, SAHIN E. An empirical assessment of the performances of three line feeding modes used in the automotive sector: line stocking vs. kitting vs. sequencing[J]. International Journal of Production Research, 2015, 53(5): 1439.
- [4] SOUZA M C, CARVALHO C R V, BRIZON W B. Packing items to feed assembly lines [J]. European Journal of Operational Research, 2008, 184(2): 480.
- [5] BOYSEN N, BOCK S. Scheduling just-in-time part supply for mixed-model assembly lines [J]. European Journal of Operational Research, 2011, 211(1): 15.
- [6] ZHOU Binghai, PENG Tao. Scheduling the in-house logistics distribution for automotive assembly lines with just-in-time principles[J]. Assembly Automation, 2017, 37(1): 51.
- [7] FATHI M, ALVAREZ M J, MEHRABAN F H, *et al.* A multiobjective optimization algorithm to solve the part feeding problem in mixed-model assembly lines [J]. Mathematical Problems in Engineering, 2014, 11(1): 809.
- [8] EMDE S, GENDREAU M. Scheduling in-house transport vehicles to feed parts to automotive assembly lines [J]. European Journal of Operational Research, 2017, 260(1): 255.
- [9] GOLZ J, GUJJULA R, GUENTHER H O, *et al.* Part feeding at high-variant mixed-model assembly lines[J]. Flexible Services and Manufacturing Journal, 2012, 24(2): 119.
- [10] ZHOU Binghai, FEI Qianran. Hybrid self-adaptive biobjective optimization of multiple robot scheduling problem for mixed-model assembly lines considering energy savings [J]. Proceedings of the Institution of Mechanical Engineers Part I Journal of Systems and Control Engineering, 2020:839.
- [11] LU Zhiqiang, ZHU Hongwei, HU Xingming. Optimization of material delivery problem for aircraft moving assembly line [C]//14th IEEE International Conference on Automation Science and Engineering.Munich:IEEE, 2018: 926-930.
- [12] 陆志强, 曾浚. 飞机移动生产线物料配送与空箱回收集成建模 [J]. 同济大学学报(自然科学版), 2019, 47(1): 136.  
LU Zhiqiang, ZEN Jun. Integrated modeling of material delivery and container pickup problem for aircraft moving assembly line [J]. Journal of Tongji University (Natural Science), 2019, 47(1): 136.
- [13] BOYSEN N, BRISKORN D, EMDE S. Just-in-time vehicle scheduling with capacity constraints [J]. IIE Transactions, 2016, 48(2): 134.
- [14] 陆志强, 牟小涵. 考虑质量衰退的批量调度与预防性维护的联合优化 [J]. 同济大学学报(自然科学版), 2021, 49(4): 535.  
LU Zhiqiang, MOU Xiaohan. Joint optimization of lot-streaming scheduling and condition-based maintenance based on a consideration of quality degradation [J]. Journal of Tongji University (Natural Science), 2021, 49(4): 535.
- [15] LAI Mingyong, CAO Erbao. An improved differential evolution algorithm for vehicle routing problem with simultaneous pickups and deliveries and time windows [J]. Engineering Applications of Artificial Intelligence, 2010, 23(2): 188.
- [16] 李兵, 蒋慰孙. 混沌优化方法及其应用 [J]. 控制理论与应用, 1997(4): 613.  
LI Bing, JIANG Weisun. Chaos optimization method and its application [J]. Control Theory & Applications, 1997(4): 613.
- [17] 马咏, 何正文, 江波, 等. 一种求解柔性资源约束前摄性项目调度问题的启发式算法 [J]. 运筹与管理, 2021, 30(8): 14.  
MA Yong, HE Zhengwen, JIANG Bo, *et al.* A heuristic algorithm for solving flexible resource constrained proactive project scheduling problem [J]. Operations Research and Management, 2021, 30(8): 14.
- [18] 彭涛, 周炳海. 车辆装配线边物料准时化配送算法 [J]. 控制理论与应用, 2016, 33(6): 779.  
PENG Tao, ZHOU Binghai. Just-in-time distribution algorithm of line-side parts for automobile assembly lines [J]. Control theory & Application, 2016, 33(6): 779.