

# 混流装配线物料配送机器人协同调度方法

周炳海<sup>†</sup>, 徐佳惠

(同济大学 机械与能源工程学院, 上海 201804)

**摘要:** 为了提高混流装配线物料配送效率并降低送料能耗和成本, 引入新型线边集成超市和送料机器人, 并考虑送料机器人之间的协同调度; 结合送料机器人相互协作送料的特点, 以最小化送料机器人的投入成本和送料能耗成本为优化目标, 建立数学模型, 并明确送料机器人的数量和有效运行时间的临界值; 在此基础上提出多阶段自适应搜索算法, 以聚类启发式方法构建初始解, 借鉴传统自适应大邻域搜索算法设计符合协同调度特点的拆分和修复规则, 并引入自适应选择和扰动策略来增加解的多样性, 提高解的质量; 最后, 通过规则测试和算法对比, 验证了协同送料调度在降低送料机器人数量和能耗方面的优势, 以及拆分和修复规则、扰动策略等对提高解质量的作用。

**关键词:** 物料配送; 送料机器人; 线边集成超市; 能源消耗; 带转运的装卸一体化问题; 自适应大邻域搜索算法  
**中图分类号:** TP391      **文献标志码:** A

## Co-scheduling of mobile robots in mixed-model assembly lines

ZHOU Bing-hai<sup>†</sup>, XU Jia-hui

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

**Abstract:** To improve the efficiency of parts supply in mixed-model assembly lines and reduce energy consumption and cost at the meantime, novel line-integrated supermarkets and mobile robots are introduced, and co-scheduling of mobile robots is considered. Combined with the characteristics of the cooperation between robots, mathematical programming formulations are given with the aim to minimize the total cost consisting of applying robots and related energy consumption. On this basis, a multiple-phases adaptive search algorithm is developed, which constructs the initial solution with a “*k*-means” clustering heuristics, develops appropriate destroy and repair rules with reference to that of the traditional adaptive neighborhood search, and introduces adaptive selection and perturbation strategy to improve the diversity and effectiveness. Finally, simulation and comparison with other algorithms are carried out to evaluate the proposed algorithm. The results show the superiority of the proposed algorithm for reducing the number and energy consumption of mobile robots and the good performance of the specified rules, adaptive selection and perturbation strategy.

**Keywords:** part feeding; mobile robots; line-integrated supermarkets; energy consumption; pickup and delivery problem with transfers; adaptive large neighborhood search

## 0 引 言

物料配送作为汽车制造过程中的重要辅助性活动, 常常导致运作成本居高不下. 如何权衡生产效益、成本和能耗等已成为企业面临的一大新挑战<sup>[1-3]</sup>.

目前, 考虑能源消耗的混流装配线物料配送的研究较少, 且主要以最小化耗电量、耗油量或排放量为目标. 如 Qiu 等<sup>[4]</sup> 构建了电力消耗与自动导引运输车 (Automatic guided vehicle, AGV) 载重量的函数, 并对

仓库中不同类型的 AGV 进行路径规划. 除了考虑载重量, 文献 [5-6] 认为能耗与车辆运行时间或距离有关, 并构建了相关线性回归函数.

近年来, 随着如德国物流公司 Sigloch 的机器人 Toru<sup>[7]</sup>、CoBots 和 CreBots<sup>[8]</sup> 的成功应用, 汽车制造业逐渐采用可移动式机器人 (Mobile robots) 代替传统的 AGV、多载量小车<sup>[9]</sup> 等进行物料配送. Dang 等<sup>[1]</sup> 以最小化机器人能耗 (即总运行时间) 为目标, 结合

收稿日期: 2017-06-21; 修回日期: 2018-03-08.

基金项目: 国家自然科学基金项目 (71471135).

责任编辑: 黄敏.

作者简介: 周炳海 (1965—), 男, 教授, 博士生导师, 从事制造系统的建模、调度等研究; 徐佳惠 (1992—), 女, 硕士生, 从事物流系统建模、仿真与调度的研究.

<sup>†</sup>通讯作者. E-mail: bhzhou@tongji.edu.cn.

不对称旅行商问题(Asymmetric travelling salesman problem, ATSP)的特点,对单个机器人的物料配送顺序进行决策. 基于相同的节能目标,Nielsen等<sup>[10]</sup>研究了配送能力约束下的单机器人调度问题,并采用改进遗传算法求解机器人每次的配送量. 但是,现有文献主要是对单个或多个相互独立的机器人进行调度,暂未发现对多个送料机器人进行协同调度的情况.

为此,本文考虑送料机器人的协同调度,即允许对配送的物料进行拆分、转运等操作,并构建多阶段自适应搜索算法对送料机器人进行调度,在保证高效、准时送料的同时,最小化送料机器人的投入成本和能耗成本.

## 1 数学建模

### 1.1 问题描述

图1所示是一个采用线边集成超市<sup>[11]</sup>的物料配送单元. 装配所需物料统一存放在工位旁的线边超市中,送料机器人根据生产计划和调度策略配送物料到各目标工位的JIS(Just-in-sequence)料箱. 为了避免送料机器人配送任务分配不均衡,即某些机器人运行效率不高的情况,本文考虑送料机器人的协同调度,即在调度过程中允许送料机器人在线边超市或工位处交换或转移待配送物料. 如图1所示,1号机器人负责工位1和工位5的两箱物料,2号机器人负责工位2和工位4的物料. 在协同调度的思想下,为节省机器人的总运行时间,所有待配送物料均可转移或互换,如图1中互换工位5和工位2的料箱,变成由1号机器人负责工位1和工位2,2号机器人负责工位4和工位5的物料配送. 若将送料机器人视为小车,各工位视为客户,则本文的协同送料调度问题类似于PDP-T(Pickup and delivery problem with transfer)问题. 但是,不同于PDP-T中出发地和目标客户的唯一性,本文中送料机器人将多次往返于工位和线边超市,对各工位进行小批量、多频次协同补料,其相互耦合,增加了调度的复杂性.

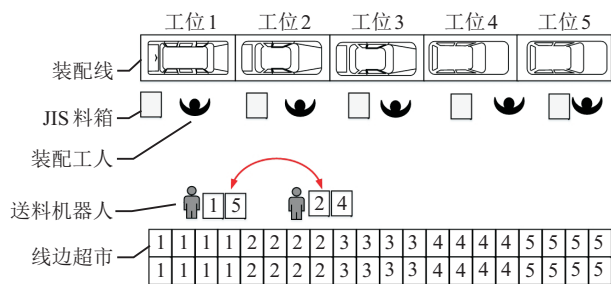


图1 线边集成超市物料配送单元示意图

为了有效描述该协同送料调度问题,作如下假设.

**假设1** 系统采用装配线节拍作为基本时间单位;

**假设2** 一个JIS料箱对应一种物料,且不允许发生过剩或缺料情况;

**假设3** 当前生产节拍补充的物料最早可在下一节拍用于装配;

**假设4** 各物料自线边超市出发,可经由多个机器人配送到目标工位;

**假设5** 物料拆分或转运只能在工位或线边超市发生;

**假设6** 机器人单次搬运能力有限;

**假设7** 物料装载、卸载时间与物料个数成正比;

**假设8** 工位间或到线边超市的行走时间为定值;

**假设9** 机器人在一个节拍内最多从线边超市出发一次;

**假设10** 送料机器人的能耗与运行时间线性相关.

### 1.2 数学模型

1) 问题参数:  $S$ 为装配线的工位集合,  $s = 1, 2, \dots, |S|$ ;  $T$ 为调度期间生产节拍数,  $t = 1, 2, \dots, T$ ;  $d_s^t$ 为工位  $s$  累积到节拍  $t$  时所需的物料总量;  $B_s^{\max}$ 为工位  $s$  处JIS料箱的容量;  $C_r^{\max}$ 为机器人  $r$  的搬运能力;  $O_r^{\max}$ 为机器人  $r$  允许的最长运行时间;  $\Delta_{ss'}$ 为工位  $s$  与  $s'$  之间所需的行走时间,且  $s, s' \in S \cup \{0\}$ ,  $0$  指线边超市;  $l^r$ 为机器人  $r$  装载单位物料的时间;  $u^r$ 为机器人  $r$  卸载单位物料的时间;  $\Gamma$ 为单个送料机器人的固定成本;  $\xi$ 为送料机器人单位运行时间的能耗.

2) 决策变量:  $R$ 为送料机器人集合,  $r = 1, 2, \dots, |R|$ ,  $|R|$ 为送料机器人数量,其大小未知;  $x_s^{tr} \in \{0, 1\}$ ,若  $r$ 在节拍  $t$  服务工位  $s$ ,则为1,否则为0,且  $s \in S \cup \{0\}$ ;  $z_r^{ss'} \in \{0, 1\}$ ,若  $r$  服务完工位  $s$  后前往工位  $s'$ ,则为1,否则为0,且  $s, s' \in S \cup \{0\}$ ;  $P_r^t = \{s_1, s_2, \dots, s_n\}$ 为机器人  $r$  在节拍  $t$  计划经过的工位集合;  $PN_r^t = \{k_1, k_2, \dots, k_n\}$ 对应  $P_r^t$  的送料量;  $Q_r^t = \{s'_1, s'_2, \dots, s'_n\}$ 为机器人  $r$  在节拍  $t$  实际经过的工位集合;  $QN_r^t = \{k'_1, k'_2, \dots, k'_n\}$ 对应  $QN_r^t$  的送料量.

3) 数学模型: 目标函数为

$$\min f = \Gamma \cdot |R| + \xi \cdot OT. \tag{1}$$

约束条件为

$$OT = \sum_{r \in R} \sum_{t=1}^T \left( \sum_{s \in Q_r^t} \sum_{s' \in Q_r^t} z_r^{ss'} \cdot \Delta_{ss'} + \sum_{k \in QN_r^t} k \cdot (l^r + u^r) \right); \tag{2}$$

$$d_s^t \leq \sum_{r \in R} \sum_{t'=1}^t \sum_{k \in \text{QN}_r^{t'}} k \cdot x_s^{t'r} \leq d_s^t + B_s^{\max}, \quad \forall s \in S, t = 1, 2, \dots, T; \quad (3)$$

$$\sum_{s \in P_r^t} \sum_{k \in \text{PN}_r^t} k \leq C_r^{\max}, \quad \forall r \in R, t = 1, 2, \dots, T; \quad (4)$$

$$\sum_{t=1}^T \sum_{s \in Q_r^t} \sum_{s' \in Q_s^t} z_r^{ss'} \cdot \Delta_{ss'} + \sum_{t=1}^T \sum_{k \in \text{QN}_r^t} k \cdot (l^r + u^r) \leq O_r^{\max}, \quad \forall r \in R; \quad (5)$$

$$\sum_{s \in S} x_s^{tr} \leq 1, \quad \forall r \in R, t = 1, 2, \dots, T; \quad (6)$$

$$\sum_{s' \in Q_r^t} z^{0s'} = \sum_{s \in Q_r^t} z^{s0} = 1, \quad \forall r \in R, t = 1, 2, \dots, T; \quad (7)$$

$$0 \leq \sum_{t=1}^T x_0^{tr} - \sum_{t'=1}^T x_0^{t'r} \leq 1, \quad \forall r \in R; \quad (8)$$

$$t \cdot \sum_{r \in R} x_s^{tr} + u^r \cdot \sum_{k \in \text{QN}_r^t} k \cdot x_s^{tr} \leq t' \cdot \sum_{r \in R} x_s^{t'r}, \quad \forall s \in S, 1 \leq t < t' \leq T; \quad (9)$$

$$x_s^{tr} \cdot (t + u^r \cdot k) + \Delta_{ss'} - x_s^{t'r} \cdot t' \leq M(1 - z_r^{ss'}), \quad \forall r \in R, t, t' = 1, 2, \dots, T, t \neq t', s, s' \in Q_r^t, M \rightarrow +\infty. \quad (10)$$

其中:式(1)为目标函数,即最小化机器人投放和能耗总成本.在实际求解过程中,送料机器人的投放量也是待决策变量,且其值越小越好;而能耗量与各送料机器人的有效运行时间成正比,时间表达式如式(2)所示,OT即所有送料机器人在调度期间内的总运行时间.式(3)~(5)为能力约束,分别限制了JIS料箱容量、机器人搬运能力和允许运行时间.式(6)确保任意机器人在某一时刻最多在一个工位补料.式(7)和(8)约束了机器人每次从线边超市出发送料,并最终返回.式(9)和(10)为时间约束,前者避免了同个工位在同一时间的并行补料,后者保证了机器人前后两次补料的时间.

### 1.3 问题性质

**性质 1** 当  $S, T, C_r^{\max}, d_s^T (d_s^0 = 0)$  已知时,假设参数  $\mu = \max_{t=1,2,\dots,T} \left\{ \sum_{s \in S} d_s^t - \sum_{s \in S} d_s^{t-1} \right\}$ , 则该装配区域所需送料机器人数量  $|R|$  的下界为

$$|R|_{lb} = \lceil \mu / C_r^{\max} \rceil.$$

**证明**  $\forall t = 1, 2, \dots, T, \mu = \max \left\{ \sum_{s \in S} d_s^t - \sum_{s \in S} d_s^{t-1} \right\}$  表示所有工位

在节拍  $t$  内物料总需求量的最大值.因此,为确保装配线不发生缺料,在任意节拍  $t$ , 机器人均能及时对各工位补料,结合假设 3 和约束 (4) 可得该装配单元最少需要  $\lceil \mu / C_r^{\max} \rceil$  个送料机器人,即  $|R|_{lb} = \lceil \mu / C_r^{\max} \rceil$ .  $\square$

**性质 2** 若  $S, R, l^r, u^r, \Delta_{0s}$  已知,  $\forall r \in R, s \in S$ , 服务时间记为  $\delta_s^r = l^r + u^r + 2\Delta_{0s}$ . 若送料机器人每次仅配送单个工位在某一节拍所需物料,则机器人的有效运作时间的上界为  $\frac{|S| \cdot T + |R| - 1}{|R|} \max_{r \in R, s \in S} \{\delta_s^r\}$ .

**证明** 设送料机器人  $r' (r' \in R)$  最后配送的是工位  $s$  在  $t$  节拍所需物料,完成时间记为  $F_{st}^{r'}$ , 则  $\forall r \in R, r \neq r'$ , 在最优调度方案中应满足  $F_{st}^{r'} \leq F_{st}^r + \delta_s^r$ , 对其求和可得

$$(|R| - 1)F_{st}^{r'} \leq \sum_{r \in R} F_{st}^r - F_{st}^{r'} + \sum_{r \in R, r \neq r'} \delta_s^r,$$

即

$$F_{st}^{r'} \leq \frac{1}{|R|} \left( \sum_{r \in R} F_{st}^r + \sum_{r \in R, r \neq r'} \delta_s^r \right).$$

此外,所有送料机器人的总运作时间等同于物料所需的总服务时间,因此可得  $\sum_{r \in R} F_{st}^r \leq |S| \cdot T \cdot \max_{r \in R, s \in S} \{\delta_s^r\}$ ; 同时,又因为  $\sum_{r \in R, r \neq r'} \delta_s^r \leq (|R| - 1) \cdot \max_{r \in R, s \in S} \{\delta_s^r\}$  成立,所以

$$F_{st}^{r'} \leq \frac{1}{|R|} [ |S| \cdot T \cdot \max_{r \in R, s \in S} \{\delta_s^r\} + (|R| - 1) \cdot \max_{r \in R, s \in S} \{\delta_s^r\} ],$$

$$\text{即 } F_{st}^{r'} \leq \frac{|S| \cdot T + |R| - 1}{|R|} \cdot \max_{r \in R, s \in S} \{\delta_s^r\}. \quad \square$$

为减少物料配送总成本,应对送料机器人的投放量和运行时间(能耗)进行权衡.结合算例参数,性质 1 可确定该装配区域所需的最少送料工人数,性质 2 限制机器人的最大运行时间,因此在后续的算法运行时有效减少了搜索空间和运行时间.

## 2 算法构建

### 2.1 转运操作

**定义 1** 对于送料机器人协同调度问题,定义转运操作  $\Upsilon = (\Upsilon^-, \Upsilon^+)$ . 其中:  $\Upsilon^- = (r_1, s_T, s, k_T)$  表示转出,  $\Upsilon^+ = (r_2, s_T, s, k_T)$  表示转入,且  $\Upsilon^-$  和  $\Upsilon^+$  成组出现.

转运操作  $\Upsilon = (\Upsilon^-, \Upsilon^+)$  表示送料机器人  $R_1$  在工位  $s_T (s_T \in S)$  将工位  $s$  所需的  $k_T$  个物料转交机器人  $r_2 (r_2 \in R)$  继续配送.当执行转运操作后,分别调整  $Q_{r_1}^t, \text{QN}_{r_1}^t, Q_{r_2}^t, \text{QN}_{r_2}^t$ , 并验证约束 (3)~(10).

为说明协同送料的优越性,针对图1的装配单元,考虑以下两种情景.其中: $\forall r \in R, s, s' \in S, \Delta_{ss'} = \Delta, C_r^{\max} = 3, l^r = u^r = 0$ ,各工位物料需求均为1.

情景1:  $P_1^t = \{1\}, P_2^t = \{2\}, PN_1^t = PN_2^t = \{1\}$ ,即机器人1和机器人2分别对工位1和工位2配送1单位物料,则共需2个机器人,总运行时间为 $4\Delta$ .采用转运操作进行重组,则  $Q_1^t = \{1, 2\}, QN_1^t = \{1, 1\}, Q_2^t = \emptyset, QN_2^t = \emptyset$ ,即机器人1负责工位1和工位2,此时共需1个机器人,总运行时间为 $3\Delta$ .

情景2:  $P_1^t = \{1, 5\}, P_2^t = \{2, 4\}, PN_1^t = PN_2^t = \{1, 1\}$ ,即机器人1负责为工位1和工位5送料,机器人2负责送料到工位2和工位4,则共需2个机器人,总运行时间为 $6\Delta + 4\Delta = 10\Delta$ .考虑转运操作,  $s_T = 2$ ,则  $Q_1^t = \{1\}, QN_1^t = \{1\}, Q_2^t = \{2, 4, 5\}, QN_2^t = \{1, 1, 1\}$ ,即机器人1将工位5的物料转交给机器人2,此时仍需2个机器人,但总运行时间为 $2\Delta + \Delta + 5\Delta = 8\Delta$ .

上述内容呈现了两种考虑转运操作的优化情况.情景1的转运发生在机器人离开线边超市前,充分利用机器人的搬运能力,减少机器人的投放.情景2中,为减少运行时间,机器人在工位2转交物料.因此,考虑转运操作或以协同送料的思想促使机器人进行小范围连续工位的补料有利于降低送料机器人的投放成本和能耗成本.

## 2.2 多阶段自适应搜索算法

由问题描述可知机器人协同调度问题在一定程度上是PDP-T问题.文献[8]已证明PDP问题及其所有变体(如PDP-T)均为NP难问题,故存在更多耦合情况的协同调度问题也是NP难问题.

目前,自适应大邻域搜索算法(Adaptive large neighborhood search, ALNS)已广泛应用于求解VRP、PDP-T等组合优化问题<sup>[8,12-13]</sup>.本文构建多阶段自适应搜索算法(Multiple-phase adaptive search, MPAS),其在ALNS算法基础上,引入启发式聚类算法构造初始解,根据协同调度的特征设计破坏和修复规则,采用自适应选择方法重构可行解,设置精英库储存较优解,并构造扰动策略避免算法陷入局部最优.

### 2.2.1 编码

送料机器人的协同调度,其实质是确定机器人的送料路径及对应的出发时间、配送量等.为便于求解,本文对送料机器人进行双层变长编码,即 $\forall r \in R, \Omega_r = \{(P_r^t, PN_r^t) | t = 1, 2, \dots, (T-1)\}$ 为机器人 $r$ 的送料计划,其实质是一个 $2(T-1)C_r^{\max}$ 的矩阵.因此,综合所有送料机器人的调度计划得到可行解

$$X = \{\Omega_1, \Omega_2, \dots, \Omega_{|R|}\}.$$

### 2.2.2 初始解构造

本文采用 $k$ -means算法的思想构建MPAS算法的初始解,即将相同或相近工位视为一个整体,由同一送料机器人补料.其中,聚类中心个数等于送料机器人数量,且各聚类中心的物理位置均匀分布在装配线各工位连线上.结合问题特征和约束条件,构造过程中应兼顾送料机器人的搬运能力、装配线不缺料等情况,故在构造解的过程中,结合最近和次近距离进行聚类.算法具体过程描述如下.

Step 1: 根据性质1设置初始的聚类中心数量CN;

Step 2: 根据CN和装配线工位数 $|S|$ 确定聚类中心位置,并计算各工位到各聚类中心的行走时间 $cd_{ij}, i = 1, 2, \dots, |S|, j = 1, 2, \dots, CN$ ;

Step 3:  $\forall i = 1, 2, \dots, |S|$ ,将 $cd_{ij}(j = 1, 2, \dots, CN)$ 升序排列,将最小值对应工位 $i$ 的所有物料需求 $d_i^t$ 分配给机器人 $j$ 配送,并验证约束条件(3) ~ (10);

Step 4: 若Step 3违反约束条件,则根据 $cd_{ij}$ 考虑次小值分配,并验证约束,依次类推,直到所有工位均分配到某个聚类中心,否则 $CN \leftarrow CN + 1$ ,并重复Step 1 ~ Step 4.

当产生初始解后,为进一步提高解的质量,增加解的多样性,对现有解进行邻域搜索,结合破坏和修复规则改变现有解结构.

### 2.2.3 破坏规则

为改善现有解,先按照破坏规则删除 $E$ 条送料计划 $P_r^t$ 及相应 $PN_r^t$ ,本文共采用3种破坏规则.

1) 随机破坏规则RD.  $\forall r \in R, t = 1, 2, \dots, T$ ,若 $P_r^t \neq \emptyset$ ,则随机生成 $E$ 条非重复的待删除记录,并从相应解向量中删除.其可能是对同一机器人送料计划的多次破坏,也可能是对不同机器人送料计划的调整.

2) 最差破坏规则WD. 此规则关注送料机器人的行走时间,其删除当前可行解中配送单位物料行走时间最长的记录.  $\forall r \in R, t = 1, 2, \dots, T$ ,计算 $P_r^t$ 中送料平均行走时间 $AT_r^t$ .  $\forall r \in R$ ,根据 $AT_r^t$ 降序排列,并选取前 $E$ 条记录加入待选择集合.最终,从集合中随机选择 $E$ 条记录,且 $AT_r^t$ 值越大,对应记录删除的概率越高.

3) 相似破坏规则SD. 相似破坏规则的基本思想是删除相似的记录,以增加解的多样性.本文对相似性的衡量主要涉及4个方面:平均行走时间 $AT_r^t$ ,送料

出发时间 $t$ ,配送量 $AQ_r^t$ 以及是否由同一个机器人送料 $\phi(r, r')$ .若 $m = (P_r^t, PN_r^t)$ 和 $n = (P_{r'}^{t'}, PN_{r'}^{t'})$ 表示两条不同的送料记录,则相似度 $\psi(m, n)$ 计算方式如下:

$$\frac{1}{\psi(m, n)} = \rho_1 \cdot |AT_r^t - AT_{r'}^{t'}| + \rho_2 \cdot |t - t'| + \rho_3 \cdot |AQ_r^t - AQ_{r'}^{t'}| + \rho_4 \cdot \phi(r, r'). \quad (11)$$

其中:若由同一个机器人送料,则 $\phi(r, r')$ 为1,否则为0;参数 $\rho_1 \sim \rho_4$ 为权重系数,其取值参考文献[12],即 $\rho_1 = 9, \rho_2 = 3, \rho_3 = 2, \rho_4 = 5$ .该破坏规则首先随机选择一条记录 $(P_{r'}^{t'}, PN_{r'}^{t'})$ 加入待删除集合,然后对其余 $r \in R, t = 1, 2, \dots, T$ ,分别计算相似度 $\psi$ ,选择相似度最大的 $(E - 1)$ 条记录进行移除.

### 2.2.4 修复规则

将已删除的记录 $(P_r^t, PN_r^t)$ 拆分成若干个送料任务 $(s, k)$ ,并采用修复规则重新分配给送料机器人,即对相关机器人确定 $(Q_r^t, QN_r^t)$ ,形成新的调度方案.本文共采用3种修复规则.

1) 贪婪修复规则GR.此规则将已删除的送料任务依次添加到使送料机器人行走时间增加最少的计划中.因此,对于每一个送料任务 $(s, k)$ ,计算将其插入后增加的行走时长 $IT_r^t$ ,并根据式(12)确定插入的机器人 $r$ 和相应节拍 $t$ ,形成 $(Q_r^t, QN_r^t)$ .此外,为平衡机器人的总运作时间,当添加到多组 $(r, t)$ 所得 $IT_r^t$ 相同时,则优先插入到当前解中总运行时间最短的机器人,其中

$$(r, t) = \arg \min_{r \in R, t=1,2,\dots,T} (IT_r^t). \quad (12)$$

2) 遗憾修复规则RR.本规则在兼顾GR规则的同时,权衡多个送料机器人的信息.如文献[12]提及“regret- $k$ ”规则,当 $k = 1$ 时,即为GR规则;当 $k > 1$ 时,则为RR规则.本文采用 $k = |R|$ .

对于每个待添加任务 $(s, k), \forall r \in R$ ,计算并确定最小值 $*IT_r = \min_{t=1,2,\dots,T} \{IT_r^t\}$ 和对应的最优插入节拍 $t$ ,再由 $*IT = \min_{r \in R} \{*IT_r\}$ 确定增加的最小行走时间,并根据下式求出待插入的送料机器人:

$$r = \arg \max \left\{ \sum_{r \in R \setminus r^*} (*IT_r - *IT) \right\}, \quad (13)$$

其中 $r \in R \setminus r^*$ 表示不包括 $*IT$ 对应的送料机器人.

3) 拆分修复规则DR.DR规则以最大化送料机器人的满载率、减少机器人数量为目标,允许对送料任务进行拆分配送.这对应于2.1节中的情景1,如将送料任务 $(s, k)$ 拆分成 $n$ 份,且 $k_1 + k_2 + \dots + k_n = k$ .

对于每个待添加的送料任务 $(s, k)$ ,首先确定可

行的待插入计划,即 $\forall r \in R, t = 1, 2, \dots, T$ ,检查计划 $(r, t)$ 中机器人的剩余配送能力 $RC_r$ ,并验证约束条件;然后依次将 $(s, k)$ 添加到使 $IT_r^t$ 增加最少的可行计划,且每次添加的配送量尽可能大,即尽量提高送料机器人的有效利用率.

### 2.2.5 自适应选择

为兼顾不同破坏或修复规则的特点,将MPAS算法的整个迭代过程划分为 $M$ 个阶段,每个阶段进行 $G$ 次迭代.在每个阶段 $m(m = 1, 2, \dots, M)$ ,为每个规则 $h(h = 1 \sim 6)$ 设置积分属性 $\pi_h^g(g = 1, 2, \dots, G)$ ,并在子阶段的迭代过程中更新积分

$$\pi_h^{g+1} = \pi_h^g + \sigma. \quad (14)$$

其中:每开始一个新阶段,对积分清零,即 $\pi_h^0 = 0$ . $\sigma$ 为单次迭代得分,其取值可分成4种情况:1)新解为当前最优解,则 $\sigma = 20$ ;2)新解非最优解,但优于当前解,则 $\sigma = 10$ ;3)新解比现有解差,但仍被接受,则 $\sigma = 5$ ;4)若以上情况均不符合,则 $\sigma = 0$ .若 $X^{\text{new}}$ 代表产生的新解, $X^*$ 为当前的最优解,则当 $f(X^{\text{new}}) < (1 + 0.01) \cdot f(X^*)$ 时,接受 $X^{\text{new}}$ 并存放入精英库 $X_{\text{pool}}$ ,并计算相应积分 $\sigma$ .这种根据一定间隙值确定是否更新解的方式在一定程度上保证了解的优化趋势,避免搜索陷入局部最优.

每完成一个阶段 $m$ 的迭代过程,更新各规则的权重 $\omega_h^m(m = 2, 3, \dots, M)$ :

$$\omega_h^m = (1 - \lambda) \cdot \omega_h^{m-1} + \lambda \cdot \pi_h^G / \theta_h^G. \quad (15)$$

其中: $\omega_h^1 = 1; \theta_h^G$ 为上一阶段中规则 $h$ 被调用的次数;参数 $\lambda$ 控制了合成 $\omega_h^m$ 的历史取值和最新迭代结果的比例,且 $\lambda \in [0, 1]$ .当对各规则更新权重 $\omega_h^m$ 后,在下一阶段的子迭代过程中,采用轮盘赌算法分别选择破坏和修复规则进行优化.

### 2.2.6 扰动策略

通过自适应选择破坏和修复规则进行邻域搜索时,若连续迭代 $\eta$ ,当前解仍未改善或未有新解收入精英库,则继续在当前解的基础上进行邻域搜索也难以提高解质量,甚至陷入局部最优.因此,本文在每个阶段引入扰动策略以增加解的多样性.

被扰动解从精英库中随机选择,其作为扰动过程的输入,对多个机器人之间的送料计划进行调整,具体过程如下,其中 $WT_r$ 为机器人 $r$ 的总行走时间.

Step 1:根据 $r_a, r_b = \arg \min_{r \in R} \{WT_r\}$ 选择待调整的机器人 $r_a, r_b$ ,且 $WT_{r_a} < WT_{r_b}$ ;

Step 2: $\forall t = 1, 2, \dots, T$ ,将计划 $P_{r_a}^t$ 拆分、转运或合并到计划 $P_{r_b}^t$ ,同时验证约束条件(3)~(10);

Step 3: 若调整之后,  $\forall t = 1, 2, \dots, T, P_{r_a}^t = \emptyset$ , 则重复 Step 1 和 Step 2, 否则转 Step 4;

Step 4: 选择待调整机器人,  $r_a = \arg \min_{r \in R} \{WT_r\}$ ,  $r_b = \arg \max_{r \in R} \{WT_r\}$ ;

Step 5:  $\forall t = 1, 2, \dots, T$ , 按照  $\min(WT_{r_a} + WT_{r_b})$  重新确定  $Q_{r_a}^t, Q_{r_b}^t$ ;

Step 6: 若调整后新解的目标函数值  $f(X^{\text{new}})$  减小, 则重复 Step 5, 否则终止扰动过程.

### 3 实验分析

本文所有算法和实验均在内存为4 GB、处理器为1.4 GHz Intel Core i5的计算机上进行, 采用 Matlab (2014b)编程实现.

#### 3.1 参数设置

混流装配线及线边集成超市实验的相关参数是在文献[11]的实验数据基础上生成的. 为简化实验过程, 假设各工位均有一定初始库存, 且  $\forall s \in S, t = 1, 2, \dots, T$ , 所需物料采用  $[\text{rnd}^{\text{uni}}(0, 4)]$ , 即(0, 4)的均匀分布,  $[\cdot]$ 表示就近取整. 此外,  $B_s^{\text{max}} = [\text{rnd}^{\text{uni}}(10, 15)]$ ,  $C_r^{\text{max}} = 10$ ,  $ct = 60 \text{ s}$ ,  $\Delta_{ss'} = 6 \text{ s}$ ,  $l^r$ 、 $u^r$ 与物料个数呈线性关系, 装货卸载单个物料时间为5 s. 为增加解的可能性, 不限制  $O_r$ 取值, 即等同于调度时长. 单位成本  $\Gamma = 10^5$ ,  $\xi = 1$ .  $|S|$ 分别取10、20、50,  $T$ 分别取30、60、120, 每种组合运行20次, 并进行统计分析.

MPAS 涉及3个重要参数, 即删除记录数  $E$ 、权重因子  $\lambda$  和扰动界限  $\eta$ . 为确保 MPAS 算法的适用性和高效性, 采用田口实验确定参数取值. 其中:  $E = \varepsilon \cdot |S| \cdot T$ , 故对  $\varepsilon$  分别取0.05、0.10、0.15、0.20;  $\lambda$  分别取0.1、0.2、0.3和0.4;  $\eta$  的值分别为0.2、0.4、0.6和0.8, 即三因素四水平实验. 实验中问题规模  $|S| = 10, T = 30$ , 实验结果如表1所示.

表1 田口实验统计结果

水平	因素		
	$\varepsilon$	$\lambda$	$\eta$
1	658 988.5	684 093.5	684 422.0
2	734 407.0	634 294.5	684 288.5
3	709 666.5	784 335.0	709 300.5
4	759 170.0	759 509.0	784 221.0
极差	100 181.5	150 040.5	99 932.5
主次排序	2	1	3
最优取值	0.05	0.2	0.4

由表1可知: 参数  $\lambda$  对 MPAS 影响最大, 其控制了 MPAS 算法在每个阶段自适应地选取破坏和修复规

则时, 对各规则的历史和实时应用性能的权衡. 其次, 参数  $\varepsilon$  (或  $E$ ) 对 MPAS 的搜索结果具有重要影响, 其控制了每次邻域搜索时删除并重新插入的送料任务数量.  $\varepsilon$  越大, 则每次更改的送料任务越多, 理论上算法的多样性越好, 而实际上该值过大, 增加了采用修复规则重新插入构成新解的难度和可行性. 此外,  $\eta$  的影响相对较小, 但其在一定程度上左右了算法的收敛速度和解的质量, 故仍需合理设置取值, 以获得最合适的 MPAS.

#### 3.2 规则分析

##### 3.2.1 规则权重分析

为分析各规则对 MPAS 算法的影响, 对  $|S| = 10, T = 30$  规模进行重复实验, 收集并整理各破坏和修复规则的权重, 统计结果如图2所示.

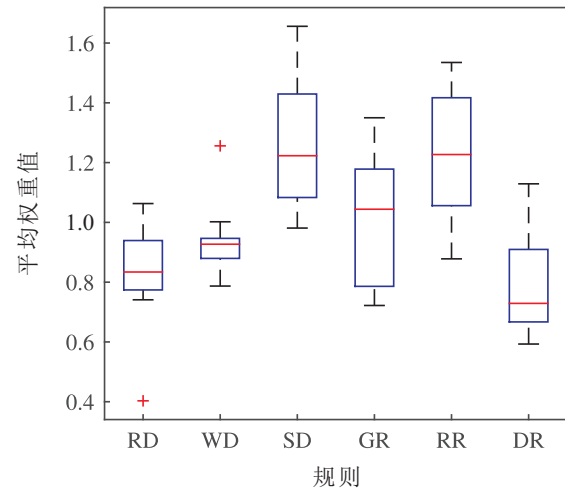


图2 破坏和修复规则权重统计箱线图

图2直观地展现了各破坏和修复规则的权重值分布情况. 在各破坏规则中, WD的权重值分布集中, RD次之, 而SD较分散, 但其值普遍较大; RD和WD均存在异常值, 剔除异常值后求权重的均值, 其结果如表2所示. 结合图2和表2可知, SD对邻域搜索过程中起主要作用, RD和WD的作用相当. 同理, 各修复规则的权重值均较分散, 且GR和RR的权重值普遍大于DR的权重值. 由表2进一步可得: RR、GR和DR的平均权重呈递减趋势, 即RR对提高解质量的作用较大; 而DR权重值较小, 可能是受配送量本身的大小限制, 即多次拆分后配送量为1, 不可再拆分, 故DR效果不显著, 但其拆分送料任务的属性符合协同送料的观念, 能够在一定程度上改变解的结构, 增加解的多样性.

表2 破坏和修复规则的权重值统计

规则	RD	WD	SD	GR	RR	DR
权重	0.881	0.899	1.272	0.996	1.213	0.786

3.2.2 规则应用分析

为进一步验证 MPAS 算法中各破坏和修复规则、自适应选择 AS 和扰动策略 PS 的作用, 本节将 MPAS

与 LNS、ALNS 和 MPAS-N 进行对比, 算法设置和实验结果如表 3 所示. 其中: 问题规模设置为  $|S| = 10$ ,  $T = 30$ , MPAS-N 表示 MPAS 不采用 DR 时的情况.

表 3 LNS、ALNS、MPAS-N 和 MPAS 算法的实验结果

组别	算法	Destroy Rules			Repair Rules			AS	PS	$f$	$ R $
		RD	WD	SD	GR	RR	DR				
1	LNS	*			*					1 010 622.6	10.0
2	LNS	*				*				1 009 363.2	10.0
3	LNS		*		*					1 010 151.6	10.0
4	LNS		*			*				939 281.0	9.3
5	LNS			*	*					1 010 229.0	10.0
6	LINS			*						979 471.8	9.7
7	ALNS	*	*	*	*	*	*			808 611.4	8.0
8	MPAS-N	*	*	*	*	*		*		779 014.4	7.7
9	MPAS	*	*	*	*	*	*	*		489 541.4	4.8

实验组别 1~组别 6 为常见的 LNS 方法, 即整个运算过程中仅采用一种破坏和修复规则, 其目标函数值  $f$  和所需送料机器人数量  $|R|$  进一步验证了 3.2.1 节中对于各个规则的分析. 组别 7~组别 9 均运用了多种规则, 并进行自适应选择, 其结果明显优于 LNS, 这也验证了采用 AS 的优越性及算法研究从 LNS 到 ALNS 的必然趋势. 其中, MPAS 综合了本文创新的 DR 和 PS 过程, 其实验结果显著优于其他算法. 这也表明了, 虽然 DR 的权重值不如 RR 和 DR, 但对运算过程有重大影响, 其拆分送料任务, 充分利用机器人间的协同送料, 有效地减少了机器人数量, 降低了总成本, 同时说明了本文对送料机器人进行协同调度是十

分有意义的.

3.3 算法分析

为验证 MPAS 的算法性能及适用性, 将其与 ALNS 和 GRASP 算法<sup>[4]</sup>进行对比, 其中 GRASP 算法是将 ALNS 结合 PDP-T 问题构建的一种改进算法, 其主要采用贪婪构建转运操作的方式改善 ALNS, 并被验证运算效果良好. 本文按照问题规模共设置 9 组实验, 每组运行 20 次, 比较 ALNS、GRASP 和 MPAS 在总成本、送料机器人数量和运行时间等方面的差异.

表 4 所示为 ALNS、GRASP 和 MPAS 在不同问题规模下得到的平均送料机器人数量和目标函数值.

表 4 ALNS、GRASP 和 MPAS 算法的实验结果

组别	问题规模		ALNS		GRASP		MPAS	
	$ S $	$T$	$ R $	$f$	$ R $	$f$	$ R $	$f$
1	10	30	7.7	779 041.0	7.0	709 052.4	4.8	489 546.5
2	10	60	10.0	1 020 473.8	10.0	1 020 394.8	6.3	651 181.2
3	10	120	10.0	1 042 192.9	10.0	1 042 677.4	7.0	743 800.1
4	20	30	20.0	2 018 625.3	20.0	2 017 332.8	11.0	1 121 317.4
5	20	60	20.0	2 043 849.9	20.0	2 042 586.1	13.0	1 347 405.2
6	20	120	20.0	2 087 928.6	20.0	2 086 077.9	16.0	1 690 461.3
7	50	30	50.0	5 051 327.0	50.0	5 052 172.5	34.7	3 542 017.2
8	50	60	50.0	5 107 773.2	50.0	5 105 924.9	38.6	3 932 327.5
9	50	120	50.0	5 224 016.5	50.0	5 221 734.0	45.3	4 722 457.9

由表 4 可知: ALNS 和 GRASP 的结果始终相近, 即 GRASP 的贪婪构建转运操作的思想对本文问题的作用不大. 而 MPAS 计算所得的目标函数值均远小于 ALNS 和 GRASP, 尤其当规模较小时 (如组别

1~组别 4), 这种差距甚至高达约 50%, 虽然随着问题规模扩大, 差距逐渐缩小, 但当  $|S| = 50$ ,  $T = 120$  时, 仍能保持 10% 以上的差距. 显然, 这种差距主要来自送料机器人数量的差异, MPAS 能够充分调动机器

人进行协同送料,以减少其固定成本,从而降低了总运作成本.这也在一定程度上说明了DR、AS和PS过程对于提高MPAS寻优能力的重要作用.

在算法运行时间上,如图3所示:当问题规模较小时(如组别1~组别5),ALNS、GRASP和MPAS算法的运行时间相近,均低于500s;而随着问题规模扩大,所需运行时间增幅较大.其中,考虑转运操作,并对搜索过程进行扰动的MPAS比ALNS和GRASP的运行时间更长,但总体而言,能够在可接受时间内解决协同调度问题,且获得更优的解.

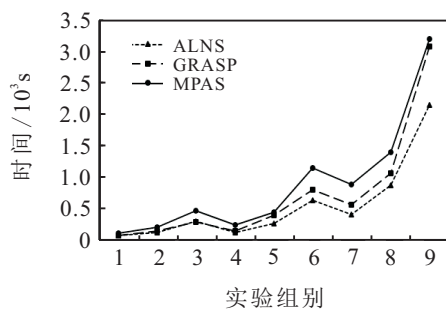


图3 算法运行时间对比

## 4 结论

本文响应绿色制造,对线边集成超市配送单元进行送料调度时考虑能源消耗,并基于充分利用送料机器人配送能力的考虑,提出了对送料机器人进行协同调度.针对该组合优化问题,采用聚类算法构建初始解,并在ALNS算法基础上引入了拆分修复规则、自适应选择、扰动策略等,构建多阶段自适应搜索算法MPAS.最后,通过与LNS、ALNS、GRASP等算法的对比实验,验证了MPAS的适用性和优越性,其可有效提高送料机器人的利用率,并减少机器人的数量和能耗.后续可以对破坏、修复规则和扰动策略等进行更深入的研究,以提高MPAS在求解大规模问题时的效率和效果.

## 参考文献(References)

- [1] Dang Q V, Nielsen I, Steger-Jensen K, et al. Scheduling a single mobile robot for part-feeding tasks of production lines[J]. *J of Intelligent Manufacturing*, 2014, 25(6): 1271-1287.
- [2] Dang Q V, Nielsen I E, Steger-Jensen K. Scheduling a single mobile robot incorporated into production environment[M]. *EcoProduction and Logistics*. Berlin: Springer, 2013: 185-201.
- [3] 周炳海, 彭涛. 混流装配生产线准时化物料补给调度方法[J]. *控制与决策*, 2017, 32(6): 976-982.

- (Zhou B H, Peng T. Scheduling methods of just-in-time material replenishment in mixed-model assembly lines[J]. *Control and Decision*, 2017, 32(6): 976-982.)
- [4] Qiu L, Wang J, Chen W, et al. Heterogeneous AGV routing problem considering energy consumption[C]. *IEEE Int Conf on Robotics and Biomimetics*. Zhuhai: IEEE, 2015: 1894-1899.
- [5] Kuo Y. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem[J]. *Computers & Industrial Engineering*, 2010, 59(1): 157-165.
- [6] Xiao Y, Zhao Q, Kaku I, et al. Development of a fuel consumption optimization model for the capacitated vehicle routing problem[J]. *Computers & Operations Research*, 2012, 39(7): 1419-1431.
- [7] Katrin N. My Colleague is A Robot[EB/OL]. (2017-06-08)[2016-07-07]. <http://w1.siemens.com.cn/POF/2011autumn/future/3206.aspx>.
- [8] Coltin B. Multi-agent Pickup and Delivery Planning with Transfers[D]. Pittsburgh: P Carnegie Mellon University, The Robotics Institute, 2014.
- [9] Zhou B, Xu J. An adaptive SVM-based real-time scheduling mechanism and simulation for multiple-load carriers in automobile assembly lines[J]. *Int J of Modeling, Simulation, and Scientific Computing*, 2017: 1750048.
- [10] Nielsen I, Do N A D, Banaszak Z A, et al. Material supply scheduling in a ubiquitous manufacturing system[J]. *Robotics and Computer-Integrated Manufacturing*, 2017, 45: 21-33.
- [11] Boysen N, Emde S. Scheduling the part supply of mixed-model assembly lines in line-integrated supermarkets[J]. *European J of Operational Research*, 2014, 239(3): 820-829.
- [12] Ropke S, Pisinger D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows[J]. *Transportation Science*, 2006, 40(4): 455-472.
- [13] 石建力, 张锦. 需求点随机的分批配送VRP模型与算法研究[J]. *控制与决策*, 2017, 32(2): 213-222. (Shi J L, Zhang J. Model and algorithm for split delivery vehicle routing problem with stochastic customers[J]. *Control and Decision*, 2017, 32(2): 213-222.)
- [14] Qu Y, Bard J F. A GRASP with adaptive large neighborhood search for pickup and delivery problems with transshipment[J]. *Computers & Operations Research*, 2012, 39(10): 2439-2456.

(责任编辑: 齐 霖)