

doi: 10.12068/j.issn.1005-3026.2020.02.019

# 考虑能耗的混流装配线物料配送多目标调度方法

周炳海, 费芊然

(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的能源利用效率, 考虑采用“转运”概念的送料机器人和线边集成超市配送模式, 构建了存在换电情形的物料供应模型. 结合送料机器人的能耗特点, 以最小化送料机器人的使用数量和配送能耗为优化目标, 建立了数学模型. 在此基础上提出了变邻域搜索策略的改进型离散差分进化算法(VNS-MDDE), 用以解决多目标优化问题; 该算法以最近邻启发式方法构建初始解, 并引入变邻域策略进行局部搜索以提高解的质量. 最后通过仿真实验验证了算法的可行性和有效性.

**关 键 词:** 混流装配线; 送料机器人; 物料配送; 线边集成超市; 能源消耗; 多目标进化算法

中图分类号: TP 29      文献标志码: A      文章编号: 1005-3026(2020)02-0258-07

## Multi-objective Scheduling Algorithm for Mixed-Model Assembly Line Considering Energy Consumption

ZHOU Bing-hai, FEI Qian-ran

(School of Mechanical Engineering, Tongji University, Shanghai 201804, China. Corresponding author: ZHOU Bing-hai, E-mail: zbh719@sohu.com)

**Abstract:** To improve the energy efficiency of parts feeding in mixed-model assembly line, mobile robots with “transfer” strategy and line-integrated supermarkets are considered, and parts feeding models with recharge situation are introduced. Combined with the energy consumption characteristics of robots, mathematical programming formulations are given with the aim to minimize the robots group scale and total energy consumption. On this basis, a modified multi-objective discrete differential evolutionary algorithm with variable neighborhood strategy is developed, where a nearest-neighbor heuristics constructs the initial solution and a variable neighborhood strategy as a local search procedure improves the quality of solutions generated. Finally, the feasibility and effectiveness of the as-proposed algorithm is validated by simulation results.

**Key words:** mixed-model assembly line; mobile robot; part feeding; line-integrated supermarket; energy consumption; multi-objective evolutionary algorithm

目前关于混流装配线物料配送问题的研究, 多以最小化碳排放或耗电量为目标, 考虑能耗的研究较少. 如 Xiao 等<sup>[1]</sup>提出模拟退火算法, 解决以最小化碳排放为目标的车辆路径问题(vehicle routing problem, VRP); Ilgin 等<sup>[2]</sup>则以最小化车辆总行驶距离和耗电量为目标解决 VRP 问题.

混流装配线边的零件进料问题为 NP-hard 问题<sup>[3]</sup>, 因此精确算法只适合求解小规模零件配送问题; 而求解大规模的零件配送问题常用元启发式算法, 如 Fathi 等<sup>[4]</sup>提出一种改进粒子群优

化算法解决以多载量小车运输行程和运载量为优化目标的装配线零件进料问题, Nouri 等<sup>[5]</sup>则考虑多个机器人的车间机器人路径规划问题, 并提出基于邻域的遗传算法来优化作业完成时间.

在分析上述文献基础上, 本文提出了考虑能耗的零件配送问题, 引入“转运”策略以减少能耗, 即允许送料机器人对配送物料进行拆分、交换等操作, 并构建变邻域搜索策略的改进型离散差分进化算法对送料机器人进行调度, 以期最小化配送系统的送料机器人团队规模及总能耗.

# 1 数学建模

## 1.1 问题描述

在线边超市配送系统<sup>[6]</sup>中,装配所需的物料统一存放在装配工位旁的线边超市中,送料机器人根据生产计划对各目标工位的 JIS (just-in-sequence) 料箱进行配送.机器人在配送过程中能量耗尽时需进行充电,且在此期间无法执行配送任务,因此需要重新分配补料任务,极大地增加了调度的复杂度.

送料机器人的能耗主要与其载重及行驶距离有关<sup>[7]</sup>,为了减少配送过程中的机器人能耗,本文提出图 1 所示的“转运”策略,即在配送过程中,允许送料机器人将其配送任务内的物料暂存在线边超市或工位处,交由其他送料机器人进行配送.该策略可以有效地缩短送料机器人配送任务中的行驶距离并优化载重分配,从而减少能耗.许多学者已经建立了若干能耗模型,用以确定单位行驶距离的能耗与机器人载重量之间的关系.基于 Qiu 等<sup>[8]</sup>的研究,本文采用式(1)计算机器人  $r$  在工位  $s$  和  $s'$  之间行驶的能耗:

$$E_{ss'} = l_{ss'} f(Q_k^*) = l_{ss'} f(Q_{k0} + Q_k). \quad (1)$$

式中: $l_{ss'}$ 表示工位  $s$  与  $s'$  之间的距离;函数  $f(Q_k^*)$  表示与机器人载重相关的能耗; $Q_{k0}$ 表示机器人未载重时质量; $Q_k$ 表示机器人运输时的装载质量.

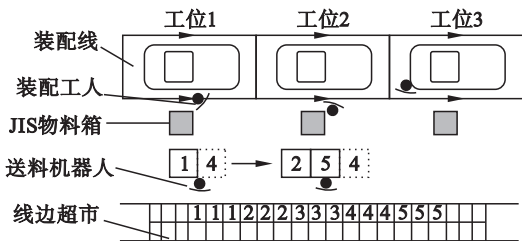


图 1 线边集成超市物料配送单元示意图  
Fig. 1 Schematic of part feeding unit in a line-integrated supermarket

为了解决考虑能耗的机器人物料配送调度问题,做出如下具体假设:① 机器人装卸料的能耗忽略不计;② 机器人往来物料超市的工位间的速度为定值;③ 机器人的充电时间为定值,机器人返回后最早可于下一节拍开始工作;④ 同一机器人完成的不同补货任务的补货时间不能重叠;⑤ 不同机器人在同个工位进行的不同补货任务的时间不能重叠;⑥ 物料自物料超市出发可经由多个机器人配送到目标工位;⑦ 物料的转运操作只能在物料超市和各工位处发生;⑧ 机器人在一个节拍

中最多补料一趟;⑨ 机器人的载重量和工位库存容量均用标准化箱的数量表示.

## 1.2 数学模型

1) 参数及其定义:

$S$  装配线的工位集合,且工位  $s = 1, 2, \dots, |S|$ ;

$R$  送料机器人集合,且机器人  $r = 1, 2, \dots, |R|$ ;

$T$  调度期间的生产节拍总数;

$d_{st}$  工位  $s$  在节拍  $t$  的物料需求量;

$B_s$  工位  $s$  处 JIS 物料箱的容量;

$C_r$  送料机器人  $r$  的物料配送能力;

$v$  送料机器人的配送速度;

$l_{ss'}$  工位  $s$  与  $s'$  之间的距离,且  $s, s' \in S \cup \{0\}$ ,其中  $\{0\}$  表示线边超市;

$t_r$  送料机器人  $r$  卸载单位物料的时间;

$m$  送料机器人未载重时的质量;

$A_{st}^r$  送料机器人  $r$  在节拍  $t$  时到达工位  $s$  时的电量;

2) 决策变量:

$k_n$  送料机器人在工位  $n$  时对应的配送物料量;

$P_r^t$  送料机器人  $r$  在节拍  $t$  经过的工位集合;

$PN_r^t$   $P_r^t$  对应工位的物料量集合,  $\{k_1, \dots, k_n\} \in PN_r^t$ ;

$PM_r^t$   $P_r^t$  对应工位的装载物料的总质量;

$x_s^{tr}$   $x_{ss'}^{tr} \in \{0, 1\}$ ,若送料机器人  $r$  在节拍  $t$  时服务工位  $s$  则为 1,否则为 0,且  $s \in S \cup \{0\}$ ;

$y_{ss'}^r$   $y_{ss'}^r \in \{0, 1\}$ ,若送料机器人  $r$  服务完工位  $s$  后前往工位  $s'$  则为 1,否则为 0,且  $s \in S \cup \{0\}$ ;

$z_t^r$   $z_t^r \in \{0, 1\}$ ,若  $A_0^r \geq \xi$ ,即机器人需充电则  $z_t^r$  为 1,否则为 0,  $A_0^r$  表示机器人  $r$  在周期  $t$  到达物料超市时的电量;

$\xi$  机器人最低电量阈值.

3) 数学模型:

$$\min f = \{f_1, f_2\}; \quad (2)$$

$$f_1 = |R|,$$

$$f_2 = \sum_{t=1}^T \sum_{r=1}^{|R|} \sum_{s \in P_r^t} \sum_{s' \in P_r^t} z_t^r l_{ss'} y_{ss'}^r f(PM_r^t + m).$$

s. t.

$$d_{st} \leq \sum_{r \in R} \sum_{k_n \in PN_r^t} k_n \cdot x_s^{tr} \leq d_{st} + B_s;$$

$$\forall s \in S; t = 1, 2, \dots, T. \quad (3)$$

$$\sum_{s \in P_r^t} \sum_{k_n \in PN_r^t} k_n \leq C_r;$$

$$\forall r \in R; t = 1, 2, \dots, T. \quad (4)$$

$$\sum_{s' \in P_r^t} y_{0s'}^r = \sum_{s \in P_r^t} y_{s0}^r = 1; \quad (5)$$

$$\forall r \in R; t = 1, 2, \dots, T. \quad (5)$$

$$0 \leq \sum_{t=1}^T x_0^{tr} - \sum_{t'=1}^T x_0^{t'r} \leq 1; \forall r \in R. \quad (6)$$

$$\sum_{s \in S} x_s^{tr} \leq 1; \forall r \in R; t = 1, 2, \dots, T. \quad (7)$$

$$t \cdot \sum_{r \in R} x_s^{tr} + t_r \cdot \sum_{k_n' \in \text{PN}_r^t} k_n \cdot x_s^{tr} \leq t' \cdot \sum_{r \in R} x_s^{t'r}; \quad (8)$$

$$\forall s \in S; 1 \leq t < t' \leq T. \quad (8)$$

$$x_s^{tr} \cdot (t + t_r \cdot \sum_{k_n' \in \text{PN}_r^t} k_n') + \frac{l_{ss'}}{v} - x_{s'}^{t'r} \cdot t' \leq m(1 - y_{ss'}^r); \quad (9)$$

$$\forall r \in R; t, t' = 1, \dots, T; t \neq t'; s, s' \in P_r^t. \quad (9)$$

$$A_{s'}^r = A_s^r - y_{ss'}^r \cdot l_{ss'} f(\text{PM}_r^t + m); \quad (10)$$

$$\forall r \in R; t = 1, 2, \dots, T; s, s' \in P_r^t. \quad (10)$$

$$A_s^r \geq \xi; \quad (11)$$

$$\forall r \in R; t = 1, 2, \dots, T; s, s' \in P_r^t \cup \{0\}. \quad (11)$$

$$x_s^{tr} \in \{0, 1\}; \forall s \in S; r \in R; t = 1, \dots, T. \quad (12)$$

$$y_{ss'}^r \in \{0, 1\}; \forall s, s' \in S; r \in R. \quad (13)$$

$$z^{tr} \in \{0, 1\}; \forall r \in R; t = 1, \dots, T. \quad (14)$$

式(2)为目标函数,包含两个目标,分别为最小化机器人团队规模和最小化机器人总能耗.约束(3)和(4)分别表示工位容量限制和机器人配送能力限制;约束(5)和(6)保证每个机器人都从线边超市出发并返回;约束(7)确保任意机器人在某一时间仅在一个工位补料;约束(8)可避免不同机器人在同一时刻对同一工位进行补料的情况;约束(9)保证了机器人连续两次补料任务的时间差;约束(10)表示机器人到达工位时的电量;约束(11)确保机器人每次配送时的电量不低于安全电量;约束(12)~(14)为二进制变量.

### 1.3 问题性质

性质 1 给定参数  $S, T, C_r, d_{st} (d_{s0} = 0)$ , 且  $\forall t = 1, 2, \dots, T, \Psi = \max \{ \sum_{s \in S} d_{st} \}$ , 则全局目标  $f_1(|R|)$ , 机器人数量  $|R|$  的下界值为  $\lceil \Psi / C_r \rceil$  ( $\lceil \cdot \rceil$  表示向上取整).

证明  $\forall t = 1, 2, \dots, T, \Psi = \max \{ \sum_{s \in S} d_{st} \}$  表示所有工位在工作节拍  $t$  内物料总需求量的最大值. 为保证机器人在任意节拍  $t$  内都能满足装配线的物料需求、不发生缺货的情况, 结合假设条件和约束条件, 可得该装配区域最少需要  $\lceil \Psi / C_r \rceil$  个

机器人, 故性质 1 得证.

性质 2 已知参数  $S, T, C_r, l_{ss'}$ , 对于  $\forall t = 1, 2, \dots, T$ , 安全电量  $\xi$  的阈值为  $\xi = \sum_{s=0}^{|S|} l_{s(s+1)} \cdot f(C_r + m)$ .

证明  $\forall t = 1, 2, \dots, T$ , 为避免送料机器人在每个节拍  $t$  的运输过程中发生电量耗尽导致无法工作的情况, 结合假设条件和约束条件可得, 送料机器人在节拍  $t$  运输开始时的工作电量不得低于  $\sum_{s=0}^{|S|} l_{s(s+1)} \cdot f(\text{PM}_r^t + m)$ , 故性质 2 得证.

性质 3  $\forall t = 1, 2, \dots, T$ , 如果机器人 1 和机器人 2 分别对应  $P_1^t = \{a, b\}, P_2^t = \{c, d\}, a, b, c, d \in S$ , 且有  $a < c < b$ , 在机器人载重条件允许的情况下宜进行转运操作.

证明 若  $P_1^t = \{a, b\}, P_2^t = \{c, d\}$ , 且  $a < c < b$ , 则说明机器人 1 和机器人 2 的行驶路径有重合之处. 假设  $\Delta$  表示  $s$  和  $s+1 (s \in S \cup \{0\})$  之间的距离,  $f(\text{PM}_r^t + m) = \text{PM}_r^t + m, \lambda_a, \lambda_b, \lambda_c, \lambda_d$  分别表示配送到工位  $a, b, c, d$  的物料量, 且  $\lambda_a + \lambda_b \leq C_r, \lambda_c + \lambda_d \leq C_r, \lambda_b + \lambda_d \leq C_r$ . 若不采取转运操作, 则能耗  $E_n = \Delta[\lambda_a + (1+b-a)\lambda_b + \lambda_c + (1+d-c)\lambda_d + (4+b+d-c-a)m]$ ; 若在工位  $c$  处采取转运操作, 则能耗  $E_t = \Delta[\lambda_a + (1+b-a)\lambda_b + \lambda_c + (1+d-c)\lambda_d + (4+d-a)m]$ . 综上, 在进行转运操作后  $E_t < E_n$ , 即进行转运操作更有利于节省能耗. 同理可得在其他工位处进行转运操作的能耗.

## 2 变邻域搜索策略的改进型离散差分进化算法(VNS-MDDE)

由于混流装配线边的零件配送问题为 NP-hard 问题<sup>[3]</sup>, 采用启发式算法可以更有效地在可接受时间内解决问题. 多目标差分进化算法是基于群体智能理论的优化算法, 具有较强的鲁棒性和全局寻优能力. 本文构建变邻域搜索策略的改进型离散差分进化算法 VNS-MDDE (multi-objective discrete differential evolution algorithm with variable neighborhood search), 在多目标差分进化算法的基础上引入变邻域搜索策略进行局部优化.

### 2.1 编码

VNS-MDDE 算法采用混合双层编码方式, 如图 2 所示, 编码的长度为优化问题中装配线工位的总数  $|S|$  的两倍, 其中后  $|S|$  个工位为转运工

位. 第一层编码表示机器人服务工位的详细信息, 第二层表示机器人的配送物料量.



图2 编码方式

Fig. 2 Encoding mode

## 2.2 种群初始化

选择合适的初始种群能加快算法的收敛并影响解的优劣. 根据上述编码形式, 本文采用最近邻启发式(NNH)算法构建初始可行解, 步骤如下:

① 令  $t = 1$ .

② 选取机器人  $r = 1$ , 从超市出发随机服务某个工位  $s$ . 如果服务该工位可行时, 即满足约束(3) ~ (14), 则前往工位  $s = s + 1$ , 否则该机器人返回超市, 下一个机器人从超市出发 ( $r = r + 1$ ). 直到所有工位  $s$  的物料需求均被满足为止.

③  $t \leftarrow t + 1$ .

④ 重复步骤②和③, 直至  $t = T$  为止.

重复上述步骤② ~ ⑤共  $N$  次, 生成初始种群.

## 2.3 变异运算

本文算法采用的是差分进化算法(differential evolution algorithm, 简称 DE)中标准的变异策略 DE/rand/1<sup>[9]</sup>. 根据一般的 DE 变异算子的结构, 在第  $G$  次迭代时, 在种群中随机选择 3 个个体, 通过把 2 个个体之间的加权向量差加到第 3 个个体上来产生新的变异个体  $V_i(G)$ . 本文采用如下针对整数规划和混合整数规划的变异算子来生成变异个体:

$$V_i(G) = X_a(G-1) \oplus F \otimes (X_b(G-1) - X_c(G-1)). \quad (15)$$

式中:  $a, b, c \in \{1, \dots, N\}$  是随机选择的, 且  $a \neq b \neq c \neq i$ ;  $F \in [0, 1]$  是控制偏差变量放大的变异常数因子. 等式(15)的等号右侧由两部分运算组成. 第一部分表示两个个体  $X_b$  和  $X_c$  之间的加权差, 用下列等式表示:

$$\Delta_i(G) = F \otimes (X_b(G-1) - X_c(G-1)) \Leftrightarrow \delta_{ij}(G) = \begin{cases} (x_{bj}(G) - x_{cj}(G), \text{rand}(j) \leq F; \\ 0, \text{其他}. \end{cases} \quad (16)$$

其中  $\Delta_i(G) = \delta_{i1}(G), \dots, \delta_{i,T-1}(G)$  是临时变量,  $\text{rand}(j)$  是  $[0, 1]$  之间的随机数. 第二部分表示  $\Delta_i(G)$  和目标个体的加和, 用下列等式表示:

$$V_i(G) = X_a(G-1) \oplus \Delta_i(G) \Leftrightarrow v_{ij}(G) = x_{aj}(G-1) \oplus \delta_{ij}(G) = \begin{cases} 0, (x_{aj}(G-1) + \delta_{ij}(G) + n) \% n = 0; \\ (x_{aj}(G-1) + \delta_{ij}(G) + n) \% n, \text{其他}. \end{cases} \quad (18)$$

式中  $\%$  表示模运算符, 用来计算其左边的整数除以其右边的整数得到的余数.

## 2.4 交叉运算

交叉算子是指将变异个体的参数与另外预定的目标个体按一定规则混合来产生试验个体. 其中通过将变异个体和父代个体  $X_i(G-1)$  组合得到交叉个体  $U_i(G)$ , 交叉个体的形成过程如下:

① 如果  $\text{rand}(j) \leq W, j = 1, \dots, n$ , 则  $u'_{ij}(G) = v_{ij}(G)$ , 否则为空, 生成个体  $u'_i(G)$ ;

② 在  $U'_i(G)$  空缺的部分中填入  $X_i(G-1)$  中相应位置的机器人编号和物料配送量, 形成  $U_i(G)$ .

## 2.5 选择运算

在交叉运算之后, 选择算子将决定试验个体是否可以保存至下一次迭代. 本文将使用非支配排序方法将种群分成各个非支配层, 随后计算各层级每个个体的拥挤度<sup>[10]</sup>, 最后在  $U_i(G)$  和  $X_i(G-1)$  中选择优者成为  $X_i(G)$ .

## 2.6 局部优化

为了增强其局部搜索的能力, 本文在每次迭代过程中加入变邻域搜索策略(VNS), 对结构邻域内的解进行搜寻改进. VNS 的思路是在不同的邻域结构  $N_1, \dots, N_k$  之间进行搜索. 从第一个邻域结构开始进行局部搜索, 直到无法进一步地改进为止. 本文提出的邻域结构如下:

随机交换(random exchange, RE): 随机选择  $Z$  对机器人, 交换其服务工位的配送物料量.

随机转移(random remove, RR): 随机删除机器人  $r$  在周期  $t$  的配送任务, 将其配送物料随机分配给其他机器人配送.

贪婪转移(greedy shift, GE): 该邻域结构依据贪婪算法的思想, 将周期  $t$  内配送距离最长的机器人的配送任务删除, 并逐个分配到使机器人配送距离增加最少的配送计划中.

转运(optional transfer, OT): 根据性质 3 选择  $Z$  对机器人执行转运操作.

## 2.7 计算复杂度

VNS-MDDE 算法流程如图 3 所示. 当装配线工位数、机器人数量和生产节拍分别为  $|S|, |R|$



和  $T$  时,算法的时间复杂度为  $O(|S| \cdot |R| \cdot T)$ ;随着工位数、生产节拍数及待分配机器人数量的增加,计算复杂度将会大幅上升. VNS-MDDE 基本操作及最坏情况下的复杂度如下:

- ① 在一次迭代中从候选解中找出非支配解  $O(2N)$ ;
- ② 用非支配排序和拥挤度计算方法,从数量为  $2N$  的候选解中选取  $N$  个解:  $O(4N^2) + O(2N \times \text{lb}(2N))$ ;
- ③ 选取解在邻域结构中进行优化,即  $O(N \times Z)$ .
- 因此, VNS-MDDE 算法的复杂度约为  $O(4N^2)$ ,可见该算法的计算复杂度合理,可以有效地得出最优解.

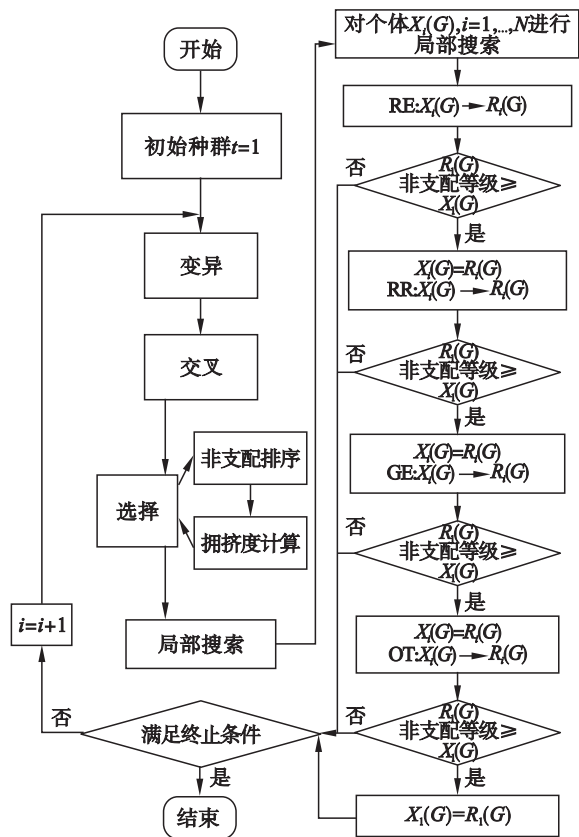


图 3 VNS-MDDE 算法流程图  
Fig. 3 Flow chart of VNS-MDDE algorithm

### 3 实验与分析

为了测试本文提出的 VNS-MDDE 算法的性能,在 Intel Core i5 - 7300U, 2. 71GHz CPU, 8GB RAM 内存的计算机上使用 MATLAB (2016a) 对该算法进行了实验.

#### 3. 1 实验参数

为了验证算法的有效性,基于文献[11]中汽车混流装配线零件进料问题及线边超市的相关实

验参数构建实验算例. 零件需求和工位初始库存均依据均匀分布随机生成,分别为  $[\text{rnd}^{\text{uni}}(0, 4)]$  和  $[\text{rnd}^{\text{uni}}(2, 4)]$ , 其中  $[\cdot]$  表示四舍五入取整. 每个机器人的最大载重  $F_{\max}$  为 10 箱,装配线的周期时间  $t$  和机器人在相邻工位及超市间的行走时间  $t_w$  分别为 90 s 和 5 s.

NSGA-II 和 IBEA 的结构和本文算法的结构相似,并具有权威性且应用广泛,因此将本文提出的 VNS-MDDE 算法与这两个标杆算法进行比较;此外,本文还将与最新的 MOEA/CT 算法<sup>[12]</sup> 进行比较,该算法在 MOEA 算法的基础上提出了坐标转换策略、外部档案更新策略及多样性维护策略,以提高帕雷托前沿解集的多样性和收敛性. 算法参数的设置如表 1 所示.

表 1 多目标算法参数				
Table 1 Parameters of multi-objective algorithms				
算法	参数	参数设置		
		小	中	大
VNS-MDDE	种群规模	20	50	100
	变异因子	0.4	0.45	0.5
	邻域参数 Z	2	5	10
NSGA-II	种群规模	20	50	100
	变异因子	0.4	0.45	0.5
	交叉因子	0.5	0.55	0.6
IBEA	种群规模	20	50	100
	变异因子	0.4	0.45	0.5
	交叉因子	0.5	0.55	0.6
MOEA/CT	种群规模	20	50	100
	变异因子	0.4	0.45	0.5
	交叉因子	0.5	0.55	0.6

#### 3. 2 性能参数

本文是多目标算法,用单一性能指标很难较好地评价多目标算法的性能,因此本文使用下列性能指标测试算法:

① 非支配解的数量:  $\lambda_A = |N_A|$ , 其中  $N_A$  表示算法 A 中找到的非支配解数量,非支配解的数量越多表示决策者选择范围越大.

② 非支配解最优度:  $\psi_A = |N_A^*|/N_A$ , 其中  $N_A^*$  表示解集  $N_A$  中不受其他算法解支配的解的数量.  $\psi_A$  越高,表示该算法越优.

③ 分布度指标:

$$\Delta = \frac{1}{D} \sqrt{\frac{1}{|N_A|} \sum_{x_i \in N_A} (D_i - \bar{D})^2}, \text{ 其中 } \bar{D} = \frac{1}{|N_A|} \sum_{x_i \in N_A} D_i, D_i \text{ 表示非支配解集中解 } x_i \text{ 与其在}$$

目标函数空间内最近邻解之间的欧氏距离,  $\Delta$  的值越小, 表示非支配解分布得越均匀, 多样性越高。

3.3 算法对比实验及结果分析

为了验证 VNS-MDDE 算法的性能, 将其与标杆算法 NSGA-II, IBEA 和 MOEA/CT 算法的

测试结果进行对比. 算例分为大、中、小三个规模, 针对每个算例, 各个算法均独立运行 15 次. 分别计算三个规模下每个算法测试的三个性能指标  $\lambda_A, \varphi_A, \Delta$  的平均值, 如表 2 所示。

表 2 算法的实验结果  
Table 2 Experiment results of algorithms

组别	问题规模		$\lambda_A$				$\varphi_A$				$\Delta$			
	S	T	VNS-MDDE	MOEA/CT	NSGA-II	IBEA	VNS-MDDE	MOEA/CT	NSGA-II	IBEA	VNS-MDDE	MOEA/CT	NSGA-II	IBEA
1	10	30	12.51	20.12	8.72	9.22	0.82	0.24	0.11	0.16	0.65	0.80	0.98	0.82
2	10	60	14.36	16.73	6.29	5.87	0.74	0.28	0.12	0.13	0.75	0.87	0.97	0.99
3	10	90	15.56	16.99	8.78	9.52	0.94	0.30	0.07	0.19	0.71	0.82	0.98	0.99
4	10	120	17.55	19.63	9.34	9.68	0.84	0.35	0.16	0.10	0.71	0.75	0.99	0.98
5	20	30	18.90	22.30	10.15	11.20	0.99	0.25	0.13	0.11	0.69	0.85	0.98	0.91
6	20	60	20.19	24.64	12.33	13.56	0.72	0.23	0.10	0.13	0.86	0.95	0.95	0.98
7	20	90	21.25	25.78	12.57	12.67	0.74	0.24	0.10	0.10	0.82	0.87	0.97	0.93
8	20	120	25.19	30.31	15.56	15.47	0.75	0.39	0.16	0.11	0.75	0.83	0.93	0.98
9	40	30	29.12	35.06	16.67	17.89	0.84	0.20	0.16	0.18	0.84	0.90	0.98	0.99
10	40	60	37.57	40.74	20.67	20.98	0.85	0.36	0.18	0.13	0.83	0.93	0.99	0.96
11	40	90	40.07	44.23	25.36	27.50	0.73	0.30	0.12	0.15	0.83	0.90	0.97	0.97
12	40	120	46.95	49.53	30.05	33.94	0.77	0.24	0.14	0.17	0.75	0.87	0.98	0.96

根据表 2 的实验结果, 可以得出如下结论:

①根据表中  $\lambda_A$  指标值可知, VNS-MDDE 算法找出的非支配解数量远多于 NSGA-II 和 IBEA, 并且随着问题规模的增加, 它们之间的差距逐渐增大。

②根据表中  $\varphi_A$  指标值可知, VNS-MDDE 找出的解的质量远优于 NSGA-II 和 IBEA. 比如在小规模问题中, NSGA-II 和 IBEA 找出的非支配解集中大约有 93% 的解受到由 VNS-MDDE 得到的非支配解集中的解支配, 而仅有 5% 的解支配 VNS-MDDE 算法找到的非支配解集中的解. 这可以说明 VNS-MDDE 算法具有更优的搜索能力。

③根据表中  $\Delta$  指标值可以看出, VNS-MDDE 算法找出的非支配解的分布度值比其他算法更小, 表明 VNS-MDDE 能进行更好的全局搜索。

④ VNS-MDDE 算法和 MOEA/CT 算法在性能上相差不大, MOEA/CT 算法在数量和分布度方面略优于本文算法; 但是由于 VNS-MDDE 算法考虑了转运策略, 而其他算法并无体现, 因此其在寻优能力上表现最好。

除了算法运行结果对比, 各算法运行时间对比如图 4 所示, 当问题规模较小时 (如组别 1 ~

6), 4 种算法的运行时间都很短, 均低于 500 s, 其中, VNS-MDDE, NSGA-II 和 IBEA 的运行时间基本相等. 随着问题规模的扩大, 各算法运行时间均有较大的增幅, 尤其是 MOEA/CT 算法; 但总体而言, 各算法均在可接受的时间内解决机器人物料配送问题, 并获得较优解。

以上对比测试证明了 VNS-MDDE 算法的有效性, 表明该算法在处理大规模混流汽车装配线物料配送问题时能够提供满意解。

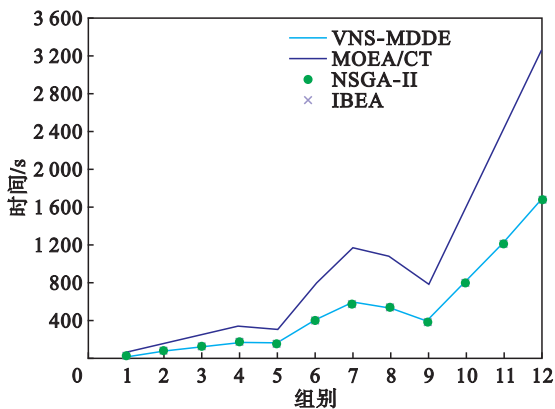


图 4 算法运行时间  
Fig. 4 Run time of algorithms

3.4 邻域结构分析

为分析各邻域结构对算法的影响,对 $|S|=20,T=30$ 规模的算例进行重复实验,统计非支配解数量,结果如图 5 所示.图 5 直观地展现了各邻域结构对算法结果的影响,可以看出,OT 的非支配解值的分布集中,RR 次之,RE 分布较散;邻域结构 GE 和 OT 均对非支配解具有优化作用,由此可知,转运可有效改善机器人物料配送的能源利用率.

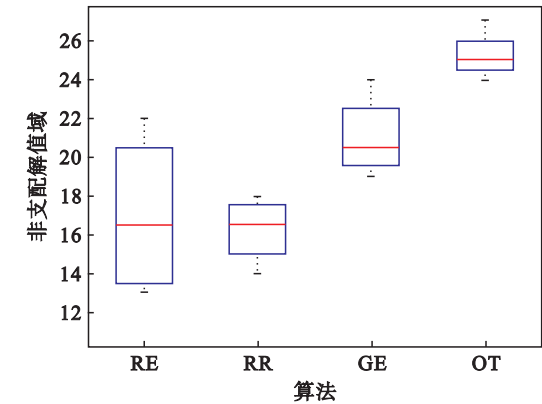


图 5 邻域结构统计箱型图  
Fig. 5 Box chart of neighborhood structure statistics

4 结 语

本文对混流装配线线边集成超市的物料配送问题进行了研究,并提出了转运策略以降低能耗.针对该组合优化问题,采用最近邻启发式算法构建初始解,并在改进型离散差分进化算法的基础上引入变邻域策略进行局部搜索,构建了变邻域搜索策略的改进型离散差分进化算法(VNS-MDDE).最后,通过与 NSGA-II, IBEA 和 MOEA/CT 算法的对比验证了本文算法的可行性和优越性,该算法可有效地减少送料机器人的使用数量并降低能耗.后续可以对随机交换、随机转移、贪婪转移,以及转运等邻域结构进行更深入的研究,以提高 VNS-MDDE 在求解大规模问题时的效率和效果.

参考文献:

[1] 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(10): 1419–1431.

[2] Ilgin M A, Gupta S M. Environmentally conscious manufacturing and product recovery (ECMPRO): a review of the state of the art [J]. *Journal of Environmental Management*, 2010, 91(3): 563–591.

[3] Kara I, Kara B, Yetis M. Energy minimizing vehicle routing problem [C]//International Conference on Combinatorial Optimization and Applications. Xi'an, China, 2007: 62–71.

[4] Fathi M, Rodriguez V, Fontes D B M M, et al. A modified particle swarm optimization algorithm to solve the part feeding problem at assembly lines [J]. *International Journal of Production Research*, 2016, 54(3): 878–893.

[5] Nouri H E, Driss O B, Ghedira K. Hybrid metaheuristics for scheduling of machines and transport robots in job shop environment [J]. *Applied Intelligence*, 2016, 45(3): 808–828.

[6] Nils B, Simon E, Lee J H. Scheduling the part supply of mixed-model assembly lines in line-integrated supermarkets [J]. *European Journal of Operational Research*, 2014, 239(3): 820–829.

[7] Zhou B Y, Shen C Y. Multi-objective optimization of material delivery for mixed model assembly lines with energy consideration [J]. *Journal of Cleaner Production* Troyes, 2018, 192(10): 293–305.

[8] Qiu L, Wang J, Chen W, et al. Heterogeneous AGV routing problem considering energy consumption [C]//2015 IEEE International Conference on Robotics and Biomimetics. Zhuhai, China, 2015: 1894–1899.

[9] Zhou S, Li X, Du N, et al. A multi-objective differential evolution algorithm for parallel batch processing machine scheduling considering electricity consumption cost [J]. *Computers and Operations Research*, 2018, 9: 55–68.

[10] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multi-objective genetic algorithm: NSGA-II [J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2): 182–197.

[11] 周炳海, 徐佳惠. 混流装配线物料配送机器人协同调度方法 [J]. *控制与决策*, 2018, 33(11): 1959–1966.  
(Zhou Bing-hai, Xu Jia-hui. Co-scheduling of mobile robots in mixed-model assembly lines [J]. *Control and Decision*, 2018, 33(11): 1959–1966.)

[12] Fang W, Zhang L Z, Yang S X, et al. A multi-objective evolutionary algorithm based on coordinate transformation [J]. *IEEE Transactions on Cybernetics*, 2019, 49(7): 2732–2743.