

# 带等待时间约束并行机调度问题的 Copula 分布估计算法

曹政才,林诚然,黄 冉

(北京化工大学信息科学与技术学院,北京 100029)

**摘 要:** 本文针对一类带等待时间约束的不相关并行机调度问题,提出了一种基于 Copula 函数的分布估计算法. 该算法以同类订单工件数与总工件数的比值为变量,对每台机器构造了一个 Copula 函数,进而建立了优势种群的概率模型. 基于概率模型通过采样生成子代个体编码向量组,保留了父代种群的相对位置信息. 从理论上分析了所提出算法的时间复杂度,其随工件个数的增加呈对数增长. 通过基于实例的数值仿真以及与已有算法的比较验证了所提算法的有效性和鲁棒性.

**关键词:** 并行机调度; 等待时间约束; Copula 理论; 分布估计算法; 对数时间复杂度

**中图分类号:** TP273      **文献标识码:** A      **文章编号:** 0372-2112 (2017)12-2949-08

**电子学报 URL:** <http://www.ejournal.org.cn>      **DOI:** 10.3969/j.issn.0372-2112.2017.12.017

## An Estimation of Distribution Algorithm Based on Copula for Parallel Machine Scheduling with Constrained Waiting Time

CAO Zheng-cai, LIN Cheng-ran, HUANG Ran

(College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China)

**Abstract:** This paper proposes an estimation of distribution algorithm based on Copula for solving parallel machine scheduling problem with constrained waiting time. By considering the ratios of each class of orders to total lots as variables, a Copula function is constructed for each machine, and then the probability model of the dominant population is established. This algorithm generates individual coding vector group by sampling based on the probability model, and preserves the relative location information of the parent population. The time complexity of the proposed algorithm is analyzed, which increases logarithmically as the number of lots raises. Simulation results based on some instances and comparisons with some existing algorithms demonstrate the effectiveness and robustness of the proposed algorithm.

**Key words:** parallel machines scheduling; constrained waiting time; copula; estimation of distribution (EDA)

### 1 引言

在工业生产过程中,每台机器可加工的工件种类常受工艺约束,各机器加工能力也不尽相同<sup>[1,2]</sup>. 特别地,某些工件在机器缓冲区内还受等待加工时间的限制. 若等待时间过长,产品次品率将会大幅提高. 因此,对带等待时间约束的不相关并行机调度问题进行研究具有重要的理论意义和经济价值<sup>[3,4]</sup>.

早期求解并行机调度问题的方法主要是启发式方法和精确算法. 启发式方法可快速求解,但对调度环境

和调度目标的依赖性较强,通用性差,且难以保证解的质量<sup>[5]</sup>. 精确算法在理论上能得到最优解,但其计算时间过长,难以满足实际生产线的需求,通常只适用于小规模问题. 近年来,智能计算方法被广泛应用于求解并行机调度问题<sup>[6,7]</sup>,如遗传算法(GA)、粒子群算法(PSO)、模拟退火算法(SA)、差分进化算法(DE)<sup>[8,9]</sup>等.

分布估计算法(EDA)是一种新颖的群体进化算法,其利用统计学方法从全局建立解的概率模型,具有全局收敛、适用于非线性、高维度问题求解的优点,近年

收稿日期:2016-11-03;修回日期:2017-01-13;责任编辑:梅志强

基金项目:国家自然科学基金(No. 51375038,61403018);高等学校博士学科点专项科研基金博导类资助课题(No. 20130010110009);北京市自然科学基金(No. 4162046)

来已成为了国内外学者的研究热点<sup>[10]</sup>,在神经网络设计、特征选择、模式匹配、车间调度<sup>[11,12]</sup>等诸多领域得到了广泛应用.例如,文献[13]将 EDA 与 GA 相结合,给出一种求解时间约束不相关并行机问题的混合 EDA 算法;文献[14]将 Copula 函数应用于 EDA 算法,有效地改善了 EDA 算法解的质量.尽管 Copula EDA 等智能算法凭借优异的寻优能力取得较好的效果,但该类算法普遍存在时间消耗偏高的缺点,当问题规模变大后,难以满足实时调度的要求.

本文针对一类具有等待时间约束的不相关并行机调度问题,以最小化完工周期和超限等待时间为目标,研究一种基于 Copula 理论的多种群混合分布估计算法(Multi Subpopulation with Hybrid Estimation of Distribution Algorithm based on Copula, MHEDAC).该算法的优势在于不仅可以改善解的质量,同时可降低算法的时间复杂度,其随工件个数增长而对数增长.

## 2 问题描述

设  $I = \{1, 2, \dots, i\}$  为待加工工件集合,  $M = \{1, 2, \dots, m\}$  为并行机器集合,每个工件仅需经一道工序即可完成加工.在加工过程中存在如下 5 类约束:(1)等待时间约束:工件在设备缓冲区内的等待时间不应超过一个预设的值.例如在半导体生产线中,多晶硅氮化层在淀积工序之前,若等待时间过长容易被氧化<sup>[15]</sup>,因此在多晶硅氮化层生成后必须设置一个有限的等待时间,否则会导致次品率增加.(2)工艺约束:每台机器可加工的工件种类不确定,至少有一类工件可以在多台机器上进行加工,且同一工件在不同的机器上加工所需的时间各不相同.一台机器同一时刻只能加工一件工件,加工一旦开始不能中断,不同工件的加工没有先后约束.(3)设备故障约束:假定机器发生故障的概率服从泊松分布,且故障修复时间为一个预先确定的数.(4)整定时间:在不同类的工艺菜单间进行切换时,机器需要一个预先确定的整定时间.(5)到达时间:各类的工件进入并行机群进行加工的时间为到达时间,它是一个预先给定的数,可能各不相同.

设机器  $k$  上共有  $v(v \leq i)$  个工件进行加工,第  $r(r \leq v)$  个加工的工件记为  $a(k, r)$ ,  $t_{a(k,r)}$  为工件  $a(k, r)$  在机器  $k$  上的加工时间,  $t_a$  是工件  $a$  的等待时间约束,则工件  $a(k, r)$  的超限等待时间  $T_{a(k,r)}$  为:

$$T_{a(k,r)} = \begin{cases} D_{a(k,r)}, & D_{a(k,r)} > 0 \\ 0, & D_{a(k,r)} \leq 0 \end{cases} \quad (1)$$

其中,  $D_{a(k,r)} = \sum_{u=1}^{r-1} t_{a(k,u)} + T_{\text{setup}} - t_a$ ,  $T_{\text{setup}}$  为加工该工件

前的该机器所花费的整定时间之和.则所有工件的总超限等待时间为:

$$T_{t1} = \sum_{k=1}^m \left( \sum_{r=2}^v \left( \sum_{u=1}^{r-1} T_{a(k,u)} \right) \right) \quad (2)$$

设备  $k$  的完工时间为:

$$T_k = \sum_{u=1}^v t_{a(k,u)} \quad (3)$$

最小完工周期为:

$$T_{t2} = \max(T_k), k = 1, 2, \dots, m \quad (4)$$

结合式(2)和式(4),本文优化目标为:

$$Z = \min(T_{t1} + W \cdot T_{t2}) \quad (5)$$

其中  $W$  为归一化系数.

## 3 MHEDAC 及时间复杂度分析

本小节介绍 MHEDAC 算法求解带等待时间约束的不相关并行机调度问题. MHEDAC 核心在于构建 Copula 函数,由 Copula 函数建立优势种群的概率模型.该算法优势在于不仅可改善解的质量,同时可降低时间复杂度.算法结构如图 1 所示.

图 1 中粗框部分的主要作用在于在保证算法的收敛速度的前提下避免陷入局部最优.但若选择个体数量过多可能无法保证种群进化方向.

### 3.1 编码和解码

根据并行机调度问题特点,对机器分配和工件排序分别进行编码.

如图 2 所示,其中第 1 部分采用文献[16]提出的轮盘赌方式的整数编码,保证编码自动满足工艺约束,避免产生不可行解.第 2 部分采用 0-1 的小数编码,数字越小,工件加工优先级越高.

解码方式以表 1 的问题实例为例说明,该实例共有 5 类订单,10 台机器与 82 个工件.第一类订单包含 13 个工件,第二类订单包含 5 个工件,类推.同类订单工件属性相同.设某个体的编码第 1 位为 5,该编码位置在 1~13 之间,对应第 1 类订单的工件,共有 6 台可行机可加工该工件,分别是机器 3,4,5,6,8,9.将 1 按照均等概率分成 6 份,每份 0.167.用编码值 5 除以机器数 + 1,得到  $5/11 = 0.455$ ,落在第 3 份(0.334 与 0.501 之间),故将工件 1 分配给机器 5.

### 3.2 种群初始化

种群初始解对算法性能有一定影响,为保证解空间均匀性,本文使用均匀分布随机产生初始解.

### 3.3 构建 Copula 函数

对于每一台机器,通过 Copula 函数获取其各类订单工件数所占该机器加工的总工件数比例的概率模型.再额外构建一个 Copula 函数,构建各机器加工的工件数占所有工件数比例的概率模型.

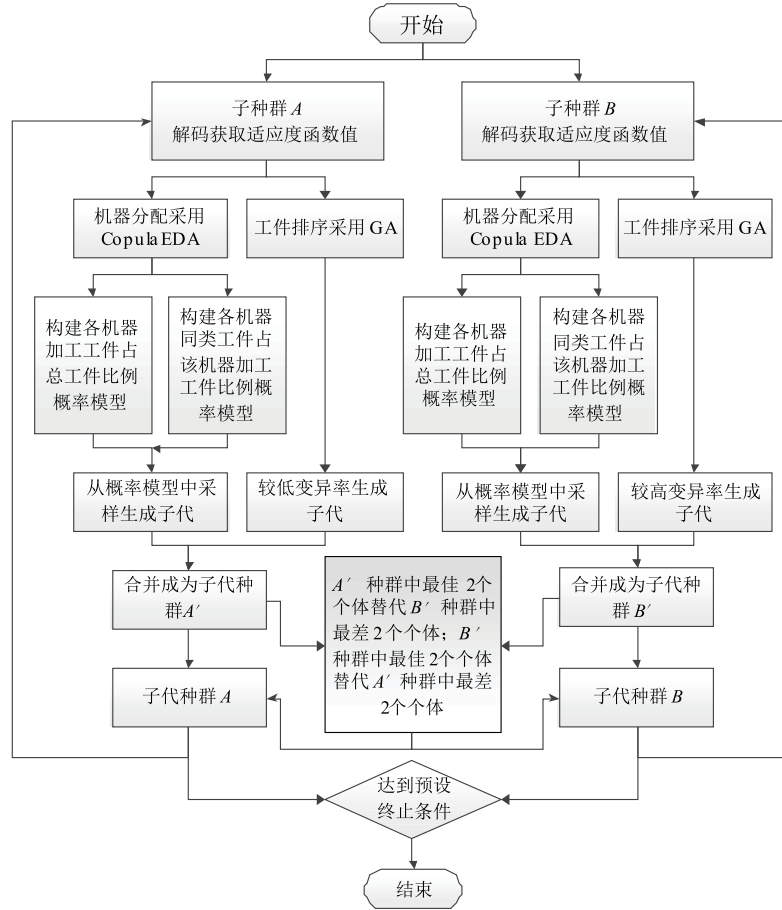


图1 MHEDAC算法模型结构

表 1 82 × 10 规模问题实例

订单号	数量 (卡)	到达时间 (小时)	等待时间约束 (小时)	工艺菜单 (在各设备上的加工时间)
1	13	0.3	5	M3(1.5)
				M4(0.8)
				M5(2.6)
				M6(1.3)
				M8(0.7)
2	5	0	15	M1(0.3)
				M2(1.6)
				M4(1.9)
3	25	1.8	36	M3(0.9)
				M6(1.4)
				M10(2.1)
4	21	0.2	55	M7(3.2)
				M8(1.7)
5	18	3	45	M10(2.5)

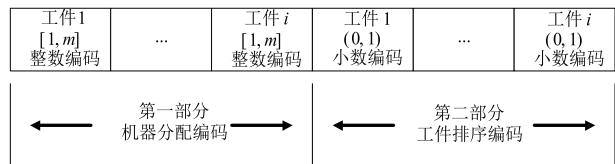


图2 MHEDAC编码方式

具体做法如下:设待解决问题为  $m$  台机器,  $i$  个工件,  $J$  类订单. 种群数量为  $2n$ , 优势种群采用截断选择法, 即适应度函数值较小的  $n$  个个体构成优势种群. 不失一般性, 仅讨论对机器  $k$  构建 Copula 函数. 设机器  $k$  上共有  $j(j \leq J, j \leq i)$  类订单与  $I_k(I_k \leq i)$  个工件可进行加工.

**步骤 1 确定各变量的边缘分布函数**

对于  $n$  个优势群体中第  $b$  个 ( $b \leq n$ ) 个体所表示的, 在机器  $k$  上加工的第  $v$  类 ( $v \leq j$ ) 订单, 统计该类订单在机器  $k$  上加工的工件个数  $v_{k(b)}$ , 记  $P_{v(b)} = v_{k(b)} / I_k$ . 对  $n$  个优势群体, 可以求出  $n$  个同类订单工件所占比例值:  $P_v = \{P_{v(1)}, P_{v(2)}, \dots, P_{v(n)}\}$ , 对这  $n$  个值按由小到大的顺序进行排列, 即可获得机器  $k$  上第  $v$  类订单工件数所占该机器总工件数比例的顺序统计量及其分布:

$$x_v = \{x_{v(1)}, x_{v(2)}, \dots, x_{v(n)}\} \quad (6)$$

$$h_v(s_v) = \begin{cases} 0, & s_v < x_{v(1)} \\ \frac{w}{n}, & x_{v(w)} < s_v \leq x_{v(w+1)} \\ 1, & s_v > x_{v(n)} \end{cases} \quad (7)$$

$s_v$  为第  $v$  类订单的随机变量.  $h_v$  为第  $v$  类订单的边缘分布函数, 是一个单调非减左连续阶梯函数.

### 步骤 2 确定 Copula 函数

由式(7)将优势种群每个个体映射为 1 个  $j$  维向量:  $\mathbf{H} = (h_1(s_1), h_2(s_2), \dots, h_j(s_j))$ . 其中,  $h_r(s_r) \in [0, 1/n, \dots, 1]$ ,  $r = 1, \dots, j$  再将区间  $[0, 1]$  分成  $k$  等份, 记做  $y_1, y_2, \dots, y_k$ , 其中,  $y_q = ((q-1)\delta, q\delta]$ ,  $\delta = 1/k$ ,  $q = 1, 2, \dots, k$ , 这里的  $k$  按照式(8)选取:

$$k = \lceil \frac{i}{m} \rceil \quad (8)$$

即  $k$  为工件数比上机器数向上取整. 并将  $j$  维单位空间  $[0, 1]^j$  被分解成  $k^j$  个小空间:  $\Omega_q = \Omega_{q1} \times \Omega_{q2} \times \dots \times \Omega_{qj}$ ,  $\mathbf{q} = (q_1, q_2, \dots, q_j) \in \{1, 2, \dots, k\}^j$ . 令 Copula 的密度函数  $c(u)$  为空间中  $n$  个点落在对应子空间中的个数的函数:

$$c(\mathbf{u}) = f_{\mathbf{q}} = B_{\mathbf{q}} / N \delta^j, \mathbf{u} \in [0, 1]^j \quad (9)$$

$B_{\mathbf{q}}$  表示落在子空间  $\Omega_{\mathbf{q}}$  上的点的个数. 这里,  $u_d \in y_q \setminus \{0\}$ ,  $q = \lceil u_d k \rceil$ ,  $u_d = 0$ ,  $q = 1$ , 同时定义:

$$g(u) = \max\{1, \lceil uk \rceil\} \quad (10)$$

则边缘密度函数为:

$$c_v = (u_1, u_2, \dots, u_v) = \delta^{j-v} \cdot f_{g(u_1), \dots, g(u_v)}^{(v)} \quad (11)$$

其中,  $f^{(v)}$  是  $g(u_r)$  的函数,  $r = 1, 2, \dots, v$ .

$$f_{g(u_1), \dots, g(u_v)}^{(v)} = \sum_{(q_1, \dots, q_j) = (1, \dots, 1)}^{k \times \dots \times k} f_{\mathbf{q}}, q_1, \dots, q_v \in \{1, \dots, k\}^j \quad (12)$$

### 步骤 3 确定条件分布函数与联合分布函数

$$C_2(u_2 | u_1) = \delta^{D-1} \sum_{q_2=1}^{g(u_2)-1} f_{g(u_1), q_2}^{(2)} + (u_2 - (g(u_2) - 1)\delta) \delta^{D-2} f_{g(u_1), g(u_2)}^{(2)} \quad (13)$$

推广到一般:

$$C_v(u_v | u_1, u_2, \dots, u_{v-1}) = \frac{\delta \cdot \sum_{q_v=1}^{g(u_v)-1} f_{g(u_1), \dots, g(u_{v-1}), q_v}^{(v)}}{\delta \cdot f_{g(u_1), \dots, g(u_{v-1})}^{(v-1)}} + \frac{[u_v - (g(u_v) - 1) \cdot \delta] \cdot f_{g(u_1), \dots, g(u_v)}^{(v)}}{\delta \cdot f_{g(u_1), \dots, g(u_{v-1})}^{(v-1)}} \quad (14)$$

其中,  $v = 3, \dots, j$ . 由上述条件分布函数可得优势种群的联合分布函数:

$$C(u_1, u_2, \dots, u_j) = C_1(u_1) \times C_2(u_2 | u_1) \times \dots \times C_j(u_j | u_1, \dots, u_{j-1}) \quad (15)$$

此即为优势种群的概率模型, 其余机器的概率模型与

构造方式类似. 另外, 额外构建一个各机器加工总工件数占所有工件数比值的概率模型, 其中  $k$  的意义与前文相同, 这里采用机器总数  $m$  而不是订单总数  $j$  作为划分子空间的维度.

### 3.4 从联合分布函数中采样产生新种群

新种群中个体的产生分为机器选择与工件排序两部分.

机器选择部分编码采用 Copula EDA 算法. 这些编码共同构成一个编码向量, 每一个子代个体中机器分配部分编码只由一个编码向量确定. 编码向量由相关的随机数向量得到. 生成随机数向量的具体做法是, 首先对各机器的概率模型按照均匀分布产生 1 个在  $[0, 1]$  之间的随机数  $u$ , 按照各边缘分布函数  $h$  产生一个服从该类工件所占比例的边缘分布的随机数  $u_1$ . 然后按式(16)产生该机器后面的随机数  $u_v, v = 2, \dots, j$ ,

$$u_v = \frac{u \cdot f_{g(u_1), \dots, g(u_{v-1})}^{(v-1)} - \sum_{q_v=1}^{g(u_v)-1} f_{g(u_1), \dots, g(u_{v-1}), q_v}^{(v)}}{f_{g(u_1), \dots, g(u_v)}^{(v)}} \cdot \delta + (g(u_v) - 1) \cdot \delta \quad (16)$$

其中,  $g(u_v) \in \{1, 2, \dots, k\}$  是满足式(17)是最小整数:

$$u \cdot f_{g(u_1), \dots, g(u_{v-1})}^{(v-1)} \leq \sum_{q_v=1}^{g(u_v)} f_{g(u_1), \dots, g(u_{v-1}), q_v}^{(v)} \quad (17)$$

由以上两式, 可以分别得到各机器加工各类订单比例概率模型的随机向量, 以及各机器加工工件占所有工件数比例概率模型的随机向量. 将这些随机向量分别根据式(7)由各自边缘分布函数  $h$  的反函数求出比例值  $P_{v(k)}, v = (1, \dots, j)$ .  $k$  为机器号, 特别地, 当  $k = 0$  时,  $P_{v(0)}$  表示各机器加工工件占所有工件数比例值, 此时  $v = 1, \dots, j$ . 机器  $k$  上第  $v$  类工件占该机器上加工总工件数的比例为  $P_{v(k)}$ , 将其乘上对应的机器  $k$  加工工件数占所有工件数比例  $P_{k(0)}$ , 即得到机器  $k$  上第  $v$  类工件占所有工件数的比例. 按照该值对子代编码进行赋值.

仍以表 1 问题为例说明赋值方法. 设得到的在第 3 台机器上加工第 1 类订单的工件数有 2 个, 第 4 台机器上加工的第 1 类订单的工件数有 3 个, 则对编码中对应第 1 类工件的 1~13 位编码中的 1、2 位赋值 1, 解码后对应于将该工件分配给机器 3 加工; 编码第 3、4、5 位赋值 3, 以此类推. 使编码的 1~13 列, 14~18 列, 19~44 列, 45~65 列, 66~82 列, 这 5 类订单所占编码对应位置分别按升序排列, 保证生成子种群机器选择与工件排序这两段编码的相对对应位置不变.

工件排序部分编码. 将子种群  $A$  和子种群  $B$  分别采用交叉概率 0.8, 变异概率 0.1 与交叉概率 0.5, 变异概率 0.8. GA 采用轮盘赌方式为选择算子, 两点交叉法为交叉算子, 变异方式为在随机交换基因在染色

中位置的变换变异. 较高交叉概率较低变异概率的种群为保证种群收敛速度, 另一种群则避免陷入局部最小<sup>[13]</sup>. 最后, 将 EDA 生成的前半段编码与 GA 生成的后半段编码组成子代种群. 种群 A 与种群 B 均采用上述方式生成分别子代种群, 同时, 两个种群各自采用最佳个体保留法, 即种群中最佳个体不经过概率模型采样或交叉变异操作, 直接作为该种群子代个体. 在种群 A 与种群 B 分别迭代生成子种群后, 将每个子种群中适应度函数值最佳和次佳的两个解分别替代掉另一种群中适应度函数值最差的两个解. 作为两个种群间合作进化的方式. 最后, 由生成的子代种群继续获取优势种群, 构建概率模型. 迭代至预设代数或预设时间.

### 3.5 时间复杂度分析

本节将分析从构建 Copula 函数到生成子代编码这一过程的时间复杂度.

文献[17]已经证明, 当优势群体个数  $n$  能被细分密度  $k$  整除时, 按照式(7) ~ 式(15)的方式构建一个 Copula 函数本身也是一个 Copula 函数. 构建 Copula 函数的时间复杂度为  $O(j^2 n \log n)$ , 式中  $n$  为优势种群数,  $j$  为订单类别数, 与工件个数无关.

具体分析如下, 构建边缘分布需要对至多  $j$  个维度, 每一维上  $n$  个优势个体进行时间复杂度为  $O(n \log n)$  的排序, 排序时间复杂度为  $O(j n \log n)$ .

另一方面, 由于  $f(v)$  可用稀疏矩阵表示, 只有  $n$  个函数值不为 0, 故不需要遍历  $k^v$  个自变量值即可构建概率模型. 该部分的时间复杂度仅为  $O(jn)$ . 但对稀疏矩阵的操作同时需要读取索引信息, 故构建描述机器上同类工件比例的 Copula 函数的时间复杂度为  $O(j^2 n \log n)$ . 同理, 构建描述各机器加工工件数占所有工件数比例的 Copula 函数时间复杂度为  $O(m^2 n \log n)$ .

根据联合分布函数  $C(u)$ , 按照式(16)和式(17)产生随机向量时. 每个数的产生需要对每个维度  $v$  分别计算一次, 每次计算需遍历细分密度  $k$ . 故时间复杂度为  $O(j^2 k)$ .

该部分时间复杂度可以由式(19) ~ (21)的方式<sup>[18]</sup>, 通过构建一个空间复杂度不超过  $O(n)$  的辅助矩阵, 由  $O(j^2 k)$  降低到  $O(j^2 \log k)$ :

$$\text{令 } s_{q_1, \dots, q_v}^{(v)} = \sum_{q_v=1}^{g(u_v)-1} \int_{g(u_1), \dots, g(u_{v-1}), q_v}^{(v)} \quad (18)$$

其中,  $(q_1, \dots, q_{v-1}) \in \{1, \dots, k\}^{v-1}$   
 $q_v = 0, \dots, k, v = 2, \dots, j$

由此, 当问题规模增大时, 本文所提的算法的时间复杂度不会显著增长. 若将表 1 问题中工件数改为原问题 5 倍, 其他条件均不变, 则  $k$  从 8 变为 40. 时间

复杂度仅从  $\log_2 8 \cdot j^2 = 3 \cdot j^2$  变为  $\log_2 40 \cdot j^2 = 5.3 \cdot j^2$ . 若不以比例值而直接将每个工件的编码作为边缘分布函数的变量, 其时间复杂度为  $O(i^2 \log k')$ ,  $k'$  为可选取的细分密度, 为保证达到与本文算法相同的概率模型精度, 应选取为机器数 10.  $i$  仍为工件数. 该时间复杂度在表 1 的问题规模下将超过 MHEDAC 时间复杂度 50 倍.

## 4 数值仿真与算法比较

采用来自某实际生产线的数据验证文中提出的调度算法的有效性.

### 4.1 仿真参数设置

为模拟实际生产线的复杂情况, 本文考虑表 2 所示的 5 个场景. 比较算法的参数采用文献[18]所提的经典优化参数: GA 算法采用轮盘赌, 两点交叉, 变异方式为随机交换基因在染色中位置的变换变异. 交叉概率为 0.9, 变异概率为 0.1. EDA 算法中遗忘因子为 0.5. 对表 1 问题各算法均迭代 500 次, 对其他两个大规模问题则限制算法运行时间为 3 分钟.

### 4.2 EDA、Copula EDA 和 MHEDAC 运行时间与解的质量的比较

EDA、Copula EDA 与 MHEDAC 对表 1 实例在 5 个场景下的仿真结果及算法运行时间如表 3 所示.

表 2 仿真场景设置

场景号	描述
1	普通场景: 正常的生产情况.
2	设备故障: 各设备按照预设的概率随机产生故障, 在确定的时间进行维修, 此段时间内机器不可使用.
3	设备生产时间: 工件在设备上的加工时间可能存在波动, 将所有工件在各设备上加工的时间设为正常值的 1.25 倍.
4	工件等待时间: 由于生产线实际条件的变化, 工件在缓冲区内的超限等待时间可能发生变化, 将所有工件最大等待时间设置为正常值的 80%.
5	综合场景: 综合考虑场景 2~4 的情况.

由表 3 可知, 引入 Copula 函数后, 解的质量有了较大的改善. 这是由于采用了联合分布, 按照一个相关向量产生整个子代设备分配部分编码, 不仅保存了父代染色体中基因的绝对位置, 还保存了父代染色体中的相对位置信息. 也可认为不仅保留了每个工件各自的加工路径信息, 还保留了各个工件在其它工件的加工路径已知的情况下, 应该在哪个机器上加工的信息. 即各台机器上加工各类工件的比例信息.

表 3 Copula EDA 与 EDA 比较

场景号	最优解均值(标准差)			算法平均运行时间(s)		
	EDA	Copula EDA	ML EDAC	EDA	Copula EDA	ML EDAC
1	208.515 (3.074)	203.441 (1.342)	203.787 (1.968)	36	139	37
2	223.601 (14.813)	209.642 (1.241)	209.840 (3.771)	42	134	43
3	219.8778 (6.100)	208.6143 (1.629)	208.9018 (2.742)	39	138	41
4	221.857 (10.857)	214.500 (1.334)	213.528 (0.528)	38	132	40
5	254.471 (5.134)	250.385 (2.642)	251.242 (3.801)	43	141	42

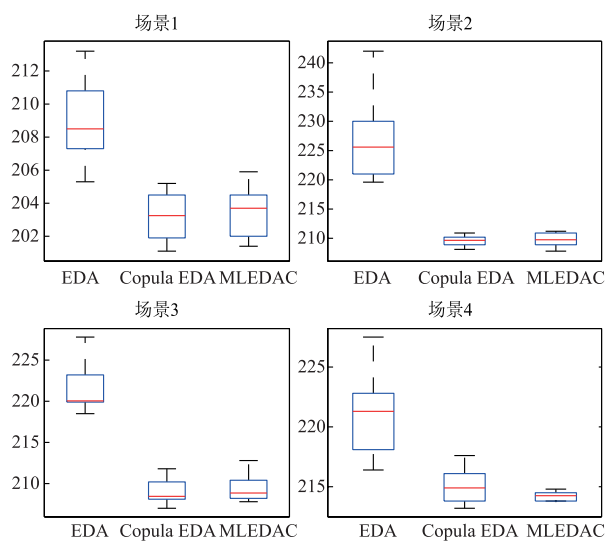


图3 Copula EDA与普通EDA结果盒状图

由表3、图3可以看到引入Copula函数后EDA算法的标准差得到了较明显的改进.但算法运行时间有较大增加.而MHEDAC在保证解的质量变化不大的前提下,极大地改善了算法运行效率.并且随着问题规模的增大,MHEDAC的求解效率将表现出更大优势.

此外,尽管在算法比较的过程中,并未对所提算法参数进行特殊调整,MHEDAC仍获得了满意的结果,这表明算法具有较强的鲁棒性.

### 4.3 MHEDAC、GA和EDA不同问题规模下的比较

以3种不同问题规模的实例比较MHEDAC、GA与EDA性能.

由表4、5、6的仿真结果,当问题规模较小时,MHEDAC与GA、EDA等算法相比,除标准差明显较小外,解

的质量和求解时间的差距不大.但是随着问题规模的增大,MHEDAC表现出算法初期迅速根据全局信息找到最优解所在区域,以及算法每一代迭代速度快的特点.对较大规模问题,MHEDAC解的质量与标准差都较GA等传统算法有较大优势.GA在3分钟时求得的最佳适应度函数值,MHEDAC通常仅需不超过20秒即可获得同样适应度的解.在 $729 \times 15$ 和 $571 \times 15$ 的问题规模,限制运行时间的情况下,MHEDAC表现出适应度函数值比之GA约20%~30%的改进量,比之EDA约10%~25%的改进量.

表 4  $82 \times 10$  问题规模下5种情况仿真结果

场景号	最优解均值(标准差)			MHEDAC改进量(标准差改进量)	
	EDA	GA	ML EDAC	GA	EDA
1	208.515 (3.074)	207.420 (4.351)	203.787 (1.968)	1.01% (121%)	1.02% (64%)
2	223.601 (14.813)	218.642 (2.651)	209.840 (3.771)	4.12% (-30%)	6.58% (292%)
3	219.8778 (6.100)	216.6143 (6.629)	208.9018 (2.742)	3.69% (140%)	5.27% (122%)
4	221.857 (10.857)	219.500 (1.344)	213.528 (0.528)	2.81% (154%)	3.89% (1905%)
5	254.471 (5.134)	255.385 (4.642)	251.242 (3.801)	1.64% (35%)	1.38% (22%)

表 5  $729 \times 15$  问题规模下5种情况仿真结果

场景号	最优解均值(标准差)			MHEDAC改进量(标准差改进量)	
	EDA	GA	ML EDAC	GA	EDA
1	91858 (11033)	89133 (9111)	78124 (7788)	17.58% (17%)	14.09% (42%)
2	97724 (14011)	104592 (11567)	82840 (8133)	26.22% (42%)	17.91% (72%)
3	120277 (19251)	117553 (10687)	90846 (6742)	29.39% (59%)	32.39% (185%)
4	110016 (8221)	115547 (9353)	89438 (6528)	29.19% (43%)	23.00% (26%)
5	125729 (22682)	133644 (11159)	107616 (11801)	24.19% (-6%)	16.83% (92%)

表 6 571 × 15 问题规模下 5 种情况仿真结果

场景号	最优解均值 (标准差)			MHEDAC 改进量 (标准差改进量)	
	EDA	GA	ML EDAC	GA	EDA
1	72990 (9074)	73889 (9342)	60975 (3968)	21.18% (135%)	19.70% (128%)
2	81779 (18813)	83442 (8241)	65719 (6771)	26.97% (22%)	24.44% (177%)
3	94791 (6100)	94233 (6629)	71866 (5742)	31.12% (15%)	31.89% (6%)
4	90255 (10857)	91975 (15334)	67358 (5528)	36.55% (177%)	33.99% (96%)
5	95963 (5134)	105646 (10642)	87500 (8801)	20.73% (21%)	9.67% (-42%)

## 5 结束语

本文提出了一种新型分布估计算法,用于求解流程工业中带等待时间约束的不相关并行机调度问题.该算法以同类订单工件数与总工件数的比值为变量,建立了优势种群的概率模型,并基于概率模型通过采样生成子代个体编码向量组.通过基于实例的数值仿真以及与已有算法的比较验证了所提算法的有效性和鲁棒性,其表明该算法不仅改善了解的质量,还大大降低了时间复杂度.

### 参考文献

- [1] An Y J, Kim Y D, Choi S W. Minimizing makespan in a two-machine flowshop with a limited waiting time constraint and sequence-dependent setup times[J]. *Computers & Operations Research*, 2016, 71: 127 - 136.
- [2] 曹政才, 彭亚珍, 李博, 等. 半导体生产线基于 DBR 和 ANFIS 相融合的动态调度方法研究[J]. *电子学报*, 2015, 43(10): 2082 - 2087.  
CAO Zhengcai, PENG Yazhen, LI Bo, et al. A DBR and ANFIS integrated dynamic scheduling algorithm for semiconductor wafer fabrication[J]. *Acta Electronica Sinica*, 2015, 43(10): 2082 - 2087. (in Chinese)
- [3] 曹政才, 赵会丹, 王永吉. 基于模糊 Petri 网推理的半导体生产线投料控制策略[J]. *电子学报*, 2011, 29(7): 1545 - 1550.  
CAO Zhengcai, Zhao Huidan, Wang Yongji. Releasing control policy for semiconductor wafer fabrication based on fuzzy Petri nets-reasoning [J]. *Acta Electronica Sinica*, 2011, 29(7): 1545 - 1550. (in Chinese)
- [4] 刘民, 吴澄. 并行多机调度问题的一种基于组合规则的遗传算法[J]. *电子学报*, 2000, 28(5): 52 - 54.  
Liu M, Cheng W U, Yang Y J. Genetic algorithm method based on combinatorial rules in identical parallel machine scheduling problem [J]. *Acta Electronica Sinica*, 2000, 28(5): 52 - 54. (in Chinese)
- [5] 尹文君, 刘民, 吴澄. 带工艺约束并行机调度问题的一种新的遗传算法[J]. *电子学报*, 2001, 29(11): 1482 - 1485.  
YIN Wen jun, LIU Min, WU Cheng. Parallel machine production scheduling problem with special procedure constraint by a new genetic. [J] *Acta Electronica Sinica*. 2001, 29(11): 1482 - 1485. (in Chinese)
- [6] CAO Zhengcai, ZHAO Huidan. ANFIS and SA based approach to prediction, scheduling, and performance evaluation for semiconductor wafer fabrication[J]. *Chinese Journal of Electronics*, 2013, 22(1): 25 - 30.
- [7] 张先超, 周泓. 变参数量子进化算法及其在求解置换流水车间调度问题中的应用[J]. *计算机集成制造系统*, 2016, 22(3): 774 - 781.  
ZHANG XianChao, ZHOU Hong. Variable parameters quantum-inspired evolutionary algorithm and its application in permutation flow-shop scheduling problem [J]. *Computer Integrated Manufacturing Systems*, 2016, 22(3): 774 - 781. (in Chinese)
- [8] CHEN, Xiao, Jing. Parallel test task scheduling with constraints based on hybrid particle swarm optimization and taboo search [J]. *Chinese Journal of Electronics*, 2012, 21(4): 615 - 618.
- [9] Wang K, Choi S H, Lu H. A hybrid estimation of distribution algorithm for simulation-based scheduling in a stochastic permutation flowshop [J]. *Computers & Industrial Engineering*, 2015, 90(C): 186 - 196.
- [10] 李作成, 钱斌, 胡蓉, 等. 求解一类异构并行机调度问题的分布估计算法[J]. *计算机集成制造系统*, 2013, 19(9): 2202 - 2212.  
LI Zuocheng, QIAN Bing, Hu Rong, et al. Estimation of distribution algorithm for a kind of heterogeneous parallel machine scheduling problem [J]. *Computer Integrated Manufacturing Systems*, 2013, 19(9): 2202 - 2212. (in Chinese)
- [11] 周树德, 孙增圻. 分布估计算法综述[J]. *自动化学报*, 2007, 33(2): 113 - 124.  
Zhou Shude, Sun Zengqi. A survey on estimation of distribution algorithms [J]. *Acta Automatica Sinica*, 2007, 33(2): 113 - 124. (in Chinese)
- [12] 王圣尧, 王凌, 许焯, 等. 求解混合流水车间调度问题的分布估计算法[J]. *自动化学报*, 2012, 38(3): 437 - 443.

- WANG ShengYao, WANG Ling, XU Ye. An estimation of distribution algorithm for solving hybrid flow-shop scheduling problem[J]. Acta Automatica Sinica. 2012, 38(3):438-443. (in Chinese)
- [13] Wang Hungkai, Chien Chenfu. An Algorithm of multi-Subpopulation parameters with hybrid estimation of distribution for semiconductor scheduling with constrained waiting time [J]. IEEE Transactions on semiconductor manufacturing. 2015, 28(3):353-366.
- [14] 王丽芳. Copula 分布估计算法[M]. 北京:机械工业出版社, 2012. 21-131.  
WANG Lifang. Copula Estimation Of Distribution Algorithm[M]. Beijing: China Machine Press 2012. 21-131. (in Chinese).
- [15] Mönch L, Fowler J W, Dauzère - Pérès S, et al. A survey of problems, solution techniques, and future challenges in scheduling semiconductor manufacturing operations. Journal of Scheduling, [J]. Journal of Scheduling, 2011, 14(6):583-599.
- [16] 刘志雄, 杨光祥. 基于轮盘赌概率分配编码方法的并行机调度优化[A]. 第二十九届中国控制会议论文集[C]. 北京:中国自动化学会, 2010.
- [17] Strelan J C, Nassaj F. Analysis and generation of random vectors with Copulas[A]. Proceedings of the 2007 Winter Simulation Conference[C]. Washington DC: IEEE, 2007. 488-496.
- [18] Grefenstette J J. Optimization of control parameters for genetic algorithms [J]. IEEE Transactions on Systems, Man and Cybernetics, 1986, 16(1):122-128.

## 作者简介



**曹政才** 男, 1974 年出生于黑龙江富裕县, 博士/教授/博导, 主要研究方向: 复杂生产过程优化调度, 系统工程和智能控制.

E-mail: giftczc@163.com



**林诚然** 男, 1993 年 6 月出生于贵州省, 硕士生, 主要研究方向: 复杂生产过程优化调度, 机器学习算法.

E-mail: 2015200731@grad.buct.edu.cn



**黄冉** 男, 1985 年 9 月出生于四川省, 博士, 主要研究领域包括: 切换控制、自适应控制, 随机控制、最优控制等. 本文通讯作者.

E-mail: huangran@mail.buct.edu.cn