

基于 SA-PSO 算法的发动机缸体机加工线平衡研究*

徐立云 徐昌飞 邓伟 李爱平

(同济大学现代制造技术研究所, 上海 201804)

摘要: 考虑到粒子群算法的高搜索效率和模拟退火算法的局部搜索能力, 提出了采用粒子群和模拟退火的混合算法求解机加工工艺线平衡问题的方法。针对机加工过程实际约束条件的特点, 提出了一种改进的编码和解码方式, 使得离散的编码序列与连续的微粒位置及速度迭代进化对应起来, 并给出粒子约束修正的方法, 使得粒子更新后总能满足约束条件, 加快了算法的收敛速度。最后以某柴油发动机缸体机缸体加工生产线为例, 验证了该方法的有效性。

关键词: 发动机缸体 生产线平衡 节拍 模拟退火算法 混合粒子群算法

中图分类号: TH162; TK413.1 **文献标识码:** A **文章编号:** 1000-1298(2014)02-0016-06

引言

生产线工艺平衡问题是制造领域的一个重要问题, 直接关系设施利用率和生产效率。制造业的生产线基本上都是在进行作业细分化之后的多工序、离散的流水生产线, 其特点是各工序分工作业、操作内容不变、作业效率较高。然而在经过了这样的作业细分化之后, 各工序作业时间往往存在一定的差异, 结果造成工序间作业负荷不均衡的现象, 负荷高的工序不能按时完成生产任务, 负荷低的工序却经常停工待料, 造成工时损失, 还造成大量的工件堆积滞留, 甚至造成生产线的中断, 阻碍生产线的顺畅流通。为了解决上述问题就必须对作业时间进行测定, 对作业顺序进行研究, 再对各工序的作业时间进行均衡化, 以使生产线顺畅运行, 实现生产的平衡。

对于生产线平衡问题来说, 许多专家学者都进行了较多的研究^[1-10]。本文基于粒子群算法及模拟退火算法的混合算法求解生产线上缸体加工工艺平衡问题, 对算法的编码、解码方式以及粒子更新后的约束处理进行改进, 并且结合具体的实例进行验证。

1 机加工工艺平衡问题描述

机加工工艺平衡问题是指通过将零件的一些加工任务(操作)在满足优先关系约束、夹具装夹约束, 并综合考虑刀具的走刀时间、换刀时间、机床之

间物流输送时间的情况下分配到工作站中, 使加工线设置的工作站数和生产节拍合理, 同时各工作站的生产负荷尽量均衡, 以减少机床数量及工人忙闲不均等现象。

设 $N_i = (N_1, N_2, \dots, N_n)$ 是一个零件加工所需要的 n 步操作的集合, $S = \{S_1(N_1, N_2, \dots, N_i); S_2(N_{i+1}, N_{i+2}, \dots, N_j); \dots; S_k(N_m, N_{m+1}, \dots, N_n)\}$, 表示对应 k 个工作站上操作的集合。理论上 n 步操作有 $n!$ 种排列, 对于每一种排列, 又有 C_{n-1}^k 种方法将操作分配到工作站。企业在分配这些操作任务时, 通常的做法是先根据机床的主轴转速和进给, 计算出各操作加工所需要的时间, 然后根据工艺师的经验进行初步的分配, 再通过不断的加工试制, 对各工作站的操作进行调整, 最后得到一个比较满意的平衡结果。

然而, 对于发动机缸体这样复杂的零件, 其特征的数目和种类繁多, 对应的操作数目也多, 通过人工分配的方法往往费时费力, 难以得到较优的平衡方案。因此, 为缸体机加工工艺平衡寻求一种有效求解优化的方法是非常必要的。

2 数学模型

机加工工艺平衡即是将所有的加工操作进行组合排序, 然后再根据工作站数进行划分, 且使得划分后的生产节拍最优, 负荷最均衡。

将零件特征机加工过程抽象为数学模型时可如

下描述:①给定特征的操作 $N_i (i = 1, 2, \dots, n)$, n 为操作的总数, $N_i \rightarrow N_j$ 表示操作 N_i 须在操作 N_j 之前完成。② $t_j (j = 1, 2, \dots, n)$ 为操作的加工时间。③工作站 $S_p (p = 1, 2, \dots, k)$, k 为工作站的总数。④各工作站作业时间 $T_q (q = 1, 2, \dots, k)$ 。⑤ $S_{ij} = 1$ 表示第 i 个操作被分配到第 j 个工作站, 否则 $S_{ij} = 0$ 。

生产线的瓶颈节拍为

$$J_1 = \max(T_q) \quad (1)$$

生产线的负荷均衡度

$$J_2 = \sqrt{\frac{1}{k-1} \sum_{i=1}^k (t_i - J_1)^2} \quad (2)$$

每一个操作必须且只能分配到一个工作站

$$\sum_{j=1}^k S_{ij} = 1 \quad (3)$$

操作分配时需要满足的优先关系(i 在 j 前)

$$\sum_{q=1}^m S_{jq} + \sum_{q=m+1}^k S_{iq} \leq 1 \quad (m \geq 1) \quad (4)$$

故可将目标函数定义为: $\min(J_1)$ 。对于不同分配方案, 当 J_1 相同时: $\min(J_2)$ 。因此, 机加工工艺平衡的数学模型可描述为: 对于零件特征的所有加工操作, 寻找一个满足约束条件的加工序列 $N = (N_1, N_2, \dots, N_n)$, 并对该加工序列按照工作站数目进行划分, 使得 J_1 最小的同时 J_2 也最小。

3 标准粒子群算法

粒子群优化算法 (Particle swarm optimization, PSO) 最早是在 1995 年由 Kennedy 和 Eberhart 共同提出的^[11], 算法中每个粒子就是解空间中的一个解。假设在一个 D 维的目标搜索空间中, 有 n 个粒子组成一个群体, 其中第 i 个粒子的位置表示为向量 $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{id})$, 其速度也是一个 D 维向量, 记 $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{id})$ 。第 i 个粒子迄今为止搜索到的最优位置为 $\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{id})$, 整个粒子群搜索到的最优位置为 $\mathbf{p}_g = (p_{g1}, p_{g2}, \dots, p_{gd})$, 在每次迭代中, 粒子更新速度和位置^[12-13]为

$$v_{id}^{k+1} = wv_{id}^k + c_1 r_1 (p_{id} - x_{id}^k) + c_2 r_2 (p_{gd} - x_{id}^k) \quad (5)$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \quad (i = 1, 2, \dots, n; d = 1, 2, \dots, D)$$

(6)

式中: k 迭代次数; w 惯性权重, 影响粒子全局搜索能力和局部搜索能力的平衡; c_1, c_2 学习因子, 分别调节 p_{id} 和 p_{gd} 方向飞行的步长, 使粒子具有自我总结和向群体中优秀个体学习的能力, 从而向自己的历史最优点以及群体内历史最优点靠近; r_1, r_2 是介于 $[0, 1]$ 之间的随机数。

4 混合粒子群算法

PSO 算法简洁而容易实现, 多年来该算法在理

论研究和实际应用上都取得了大量的研究成果^[14-19]。PSO 算法早期收敛速度快, 但后期会受到随机振荡影响, 使其在全局最优值附近需要较长的搜索时间, 收敛速度慢, 极易陷入局部极小值, 使得精度降低且易发散。为此, 本文将模拟退火思想引入 PSO 算法, 建立基于模拟退火的 PSO 算法模型, 使每个粒子的速度和位置更新过程中加入模拟退火机制, 对粒子群的适应度按 Metropolis 准则接受优化的同时概率接受恶化解, 算法从局部极佳区域中跳出, 自适应调整退火温度, 随着温度逐渐下降, 粒子逐渐形成低能量基态, 收敛至全局最优解。

4.1 模拟退火算法包含的基本过程^[20]

(1) 给定初始温度及初始点, 退火速度为 K , 最低温度 T_0 。

(2) 计算该点的函数适应度 $E = f(x)$ 。

(3) 按照由生成函数 $g(\Delta x, T)$ 确定的概率选择 Δx , 令新点 X_n 等于 $x + \Delta x$ 。

(4) 计算新的目标函数 $E_n = f(X_n)$ 。

(5) 按照由接收函数 $h(\Delta E, T)$ 确定的概率 X 将设为 X_n , E 设为 E_n , 其中 $\Delta E = E_n - E$ 。

(6) 若 $\Delta E \leq 0$, 则接受新点作为下一次模拟的初始点。

(7) 若 $\Delta E > 0$, 则计算新接受概率: 若 $\exp(-\Delta E/T) > \varepsilon$, ε 为 $[0, 1]$ 之间的随机数, 则接受新位置, 否则拒绝, 维持先前点的值。

(8) 增加迭代次数 K , 如果 K 达到最大迭代次数, 停止迭代, 否则返回步骤(2)。

4.2 粒子的编码与解码

对于机加工生产线平衡问题, 目标函数相对于离散设计变量具有跳跃性, 单纯的实数编码无法将离散编码序列和连续的微粒位置及速度迭代进化对应起来。本文提出采用随机数与实数相结合的方式对粒子进行编码。

粒子的编码由两部分组成, 左边部分 $a, b, c \dots$ 由 $(0, 1)$ 之间的随机数组成, 其个数等于操作的总数 n , 右边部分 $A, B, C \dots$ 由 $(0, n)$ 之间的实数组成

$$\underbrace{a, b, c \dots}_{n \text{ 个}} \underbrace{A, B, C \dots}_{n \text{ 个}}$$

其个数等于工作站的总数, 且其总和 $A + B + \dots$ 等于操作总数 n 。采用如此的编码方式可将离散编码序列和连续的微粒位置及速度迭代进化进行一一对应, 有助于粒子速度与位置的更新。

解码时, 首先将随机序列 $a, b, c \dots$ 按 $A, B \dots$ 的个数进行划分, 前 A 个随机数表示分配在第 1 个工作站, 第 $A + 1$ 个至 $A + B$ 表示分配在第 2 个工作站, 以此类推。再将 $a, b, c \dots$ 按从小到大进行排序, 最

小的表示1号操作,最大的表示 n 号操作,从而将随机数还原为操作编号。

4.3 SA-PSO 算法实现步骤

(1)初始化参数:惯性权重 w ,加速常数 $c_1、c_2$,退火起、止温度 T_0 和 T ,退火速度 K 。

(2)随机产生 n 个粒子的种群,即随机产生 n 个始解 $X_i(k)$ ($i=1,2,\dots,n$)和 n 个初始速度 $V_j(j=1,2,\dots,n)$ 。

(3)计算每个粒子的适应度 $f(x_i(k))$ ($i=1,2,\dots,n$)。

(4)对每个粒子,将其适应度 $f(x_i(k))$ 与个体极 $pbest_i(k)$ 比较,取优更新为全局极值 $pbest_i(k)$ 。

(5)对每个粒子,比较适应度 $f(x_i(k))$ 与全局极值 $gbest_i(k)$,取优更新为全局极值 $gbest_i(k)$ 。

(6)根据式(5)、(6)更新每个粒子的位置 $X_i(k+1)$ 和飞行速度 $V_i(k+1)$,并把速度限制在 V_{max} 内。

(7)计算每个粒子更新后的适应度 $f(x_i(k+1))$ 。

(8)计算两个位置所引起的适应度的变化量 ΔE ,若 $\Delta E < 0$,则接受新位置;若 $\exp(-\Delta E/T) > \varepsilon$, ε 为 $[0,1]$ 之间的随机数,就接受新位置,否则拒绝。若接受新位置,降温 $T \leftarrow kT$,否则不降温,返回步骤(3)。

5 应用实例

5.1 实例描述

以图1所示的某公司柴油发动机缸体模型为例。该发动机主要应用在皮卡、拖拉机等农用车辆中。通过分析得到缸体主要特征:顶面、底面、前端面、后端面、左面、右面等面特征,以及油道孔、凸轮轴孔、曲轴孔、缸孔等孔特征。

缸体零件的材料为HT250,加工所用机床型号为某型号五轴加工中心,每个特征的加工时间根据主轴转速和进给量计算得出。将几何特征相同的特征聚类为一个特征,用同一代号表示,经聚类后,该缸体零件有34个主要制造特征,相关特征的加工操作和加工时间由表1给出。

装夹约束:缸体加工采用4个工作站,前2个工作站各完成两道工序的加工,后2个工作站各完成一道工序的加工。根据夹具对缸体的装夹姿态,对应工作站只能加工一些特定面上的特征,见表2。

加工优先顺序的约束:先面后孔,先粗后精,同一特征的前一步操作必须先于其后一步操作,交叉孔优先加工长孔(26号特征要先于9号特征加工),定位基准要优先加工(33号特征要在工序4之前加工好)。

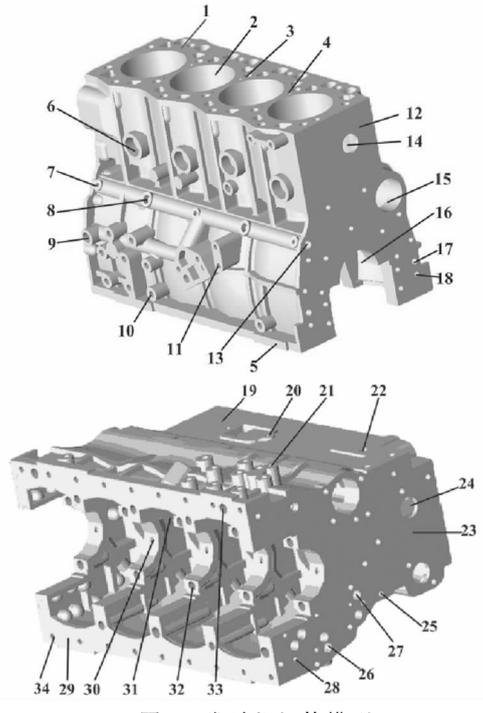


图1 发动机缸体模型

Fig. 1 Engine cylinder block model

表1 操作编号及加工时间

Tab. 1 Operation number and cycle time

特征编号	特征名称	操作编号及名称	操作时间/s
1	顶面	f_{11} 粗铣	81.95
		f_{12} 精铣	169.17
2	缸孔4- $\phi 99$	f_{21} 粗镗	74.0
		f_{22} 半精镗	63.96
		f_{23} 精镗	82.68
3	18-M12*1.75	f_{31} 钻	93.96
		f_{32} 攻丝	48.24
4	顶面斜油孔7- $\phi 5$	f_{41} 预钻	12.04
		f_{42} 钻	234.5
5	右面	f_{51} 粗铣	137.79
6	水闷孔4- $\phi 30$	f_{61} 钻	12.52
		f_{62} 精镗	15.56
7	3-NPT1/16	f_{71} 钻	3.81
		f_{72} 攻丝	5.67
8	2-NPT1/4	f_{81} 钻	4.19
		f_{82} 攻丝	6.72
9	M16	f_{91} 钻	5.32
		f_{92} 攻丝	4.58
10	10-M10*1.5	f_{101} 钻	29.3
		f_{102} 攻丝	25.6
11	8-M8*1.25	f_{111} 钻	15.02
		f_{112} 攻丝	15.76
12	后端面	f_{121} 粗铣	81.37
		f_{122} 精铣	167.96
13	主油道孔	f_{131} 预钻	3.63
		f_{132} 枪钻	73.64
14	$\phi 30$	f_{141} 钻	1.36
		f_{142} 粗镗	1.39
		f_{143} 精镗	0.72

续表 1

特征编号	特征名称	操作编号及名称	操作时间 /s
15	凸轮轴孔	f ₁₅₁ 粗镗	51.66
		f ₁₅₂ 精镗	130.62
16	曲轴孔	f ₁₆₁ 粗镗	128.72
		f ₁₆₂ 半精镗	94.61
		f ₁₆₃ 精镗	433.74
17	2-φ10	f ₁₇₁ 钻	4.52
		f ₁₇₂ 精镗	4.80
18	17-M8 * 1.25	f ₁₈₁ 钻	34.36
		f ₁₈₂ 攻丝	35.88
19	左面	f ₁₉₁ 粗铣	60.01
		f ₁₉₂ 精铣	75.52
20	2-M8 * 1.25	f ₂₀₁ 钻	4.87
		f ₂₀₂ 攻丝	3.87
21	8-M10 * 1.5	f ₂₁₁ 钻	21.06
		f ₂₁₂ 攻丝	18.46
22	4-M12 * 1.5	f ₂₂₁ 钻	12.41
		f ₂₂₂ 攻丝	10.72
23	前端面	f ₂₃₁ 粗铣	81.37
		f ₂₃₂ 精铣	167.96
24	2-φ30	f ₂₄₁ U 钻	2.72
		f ₂₄₂ 粗镗	2.78
		f ₂₄₃ 精镗	1.44
25	19-M8 * 1.25	f ₂₅₁ 钻	31.37
		f ₂₅₁ 攻丝	36.78
26	2-φ14	f ₂₆₁ 预钻	4.04
		f ₂₆₂ 钻	158.23
27	5-φ14	f ₂₇₁ 钻	10.91
		f ₂₇₂ 镗	15.21
28	5-M6	f ₂₈₁ 钻	12.21
		f ₂₈₂ 攻丝	6.27
29	底面 60	f ₂₉₁ 粗铣	179.7
		f ₂₉₂ 精铣	370.2
30	5-φ6 斜油孔	f ₃₀₁ 预钻	8.01
		f ₃₀₂ 钻	157.08
		f ₃₁₁ 钻	28.48
31	挺杆孔 8-φ14	f ₃₁₂ 扩	58.18
		f ₃₁₃ 精镗	76.24
32	10-M14 * 2	f ₃₂₁ 钻	48.35
		f ₃₂₂ 攻丝	29.62
33	工艺销孔 2-φ14	f ₃₃₁ 钻	5.66
		f ₃₃₂ 半精镗	14.55
		f ₃₃₃ 精镗	13.51
34	18-M8	f ₃₄₁ 钻	29.37
		f ₃₄₂ 攻丝	34.78

表 2 各工作站的装夹约束

Tab.2 Clamping constraint of each workstation

工作站编号	可加工的面特征
1	右面、后端面、前端面、底面
2	顶面、右面、左面、底面
3	前端面、后端面、底面
4	顶面、前端面、后端面

依据该加工中心的实际情况,每次换刀时间为 4.2 s,本案例中将换刀时间计算在操作时间之内。以 2 号特征的 f₂₁ 粗镗为例,其操作时间用 90.8 s (74.0 + 4 × 4.2) 来进行代替。各加工中心为串行布局,之间直线距离都为 1.5 m,采用人工和机械手结合的物流搬运方式,搬运时间为 90 s。

5.2 算法的求解

5.2.1 粒子的编码与解码

随机产生一组 1 行 73 列的 (0,1) 之间的随机数来表示微粒 X_i 的位置,将随机数的大小进行比较,最大的数表示 73 号操作 (f₃₄₂),最小的数表示 1 号操作 (f₁₁),得到对应的操作序列为

$$\underbrace{67, 41, 56, \dots, 45, 22, 37}_{73\text{个}}$$

由于要将上述操作序列中的所有操作分配到 4 个工作站中,故随机生成 4 个和为 73 的整数,如 [13, 23, 17, 20]。解码时将操作序列的前 13 个操作分配到第 1 个工作站,接下来的 23 个操作 (即 14 到 36) 分配到第 2 个工作站,以此类推。解码后 4 个工作站分配的操作结果如下

$$\underbrace{67, 41, \dots}_{13\text{个}}, \underbrace{36, 52, \dots}_{23\text{个}}, \underbrace{60, 43, \dots}_{17\text{个}}, \underbrace{\dots, 22, 37}_{20\text{个}}$$

5.2.2 粒子速度与位置更新

(1) 速度更新

首先生成介于 (0,1) 之间的随机数粒子的初始位置和速度,粒子群迭代过程中, (p_{gd} - x_{id}^k) 表示全局最优粒子的位置对应的随机数减去当前粒子的位置对应的随机数,再分别与 c₁r₁、c₂r₂, 相乘得到随机数值,然后根据式(5)更新粒子的速度 v_{id}^{k+1}。

(2) 位置更新

x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} 表示粒子当前位置对应的随机数加上粒子速度更新后对应的随机数,若

$$x_{id}^k = \underbrace{0.9595, 0.6557, \dots, 0.6787, 0.7577}_{73\text{个}}$$

$$v_{id}^{k+1} = \underbrace{0.7431, 0.3922, \dots, 0.0318, 0.2769}_{73\text{个}}$$

则 x_{id}^{k+1} = 1.7026, 1.0479, ..., 0.7105, 1.0346_{73个}

从而更新粒子的位置 x_{id}^{k+1}。

5.2.3 约束关系处理

由于初始种群的粒子是随机生成的,解码后的随机操作序列一般都不满足约束条件,即使初始种群粒子满足约束条件,但经过位置更新以后,也往往不能满足约束条件,如不对生成的粒子进行修正,则需要多次重复直到生成符合约束的粒子,则会使算法的收敛速度大大降低。本文提出一种对粒子约束修正的方法,以使粒子的生成及经过更新后都能满足约束条件。

设有 N 步操作,解码后对应的操作序列为 [N₁,

$N_2, \dots, N_n]$, 对应建立一个 n 行 n 列的约束矩阵 $Y = [N_a, N_b, \dots; \dots; N_i, N_j, \dots]$; 矩阵的第 1 行表示第 1 个操作对应的约束条件, 其中 N_a, N_b 表示一定要在 N_1 之前, 没有约束的则以 0 补齐, 其他行则以此类推。具体的约束添加过程如图 2 所示。

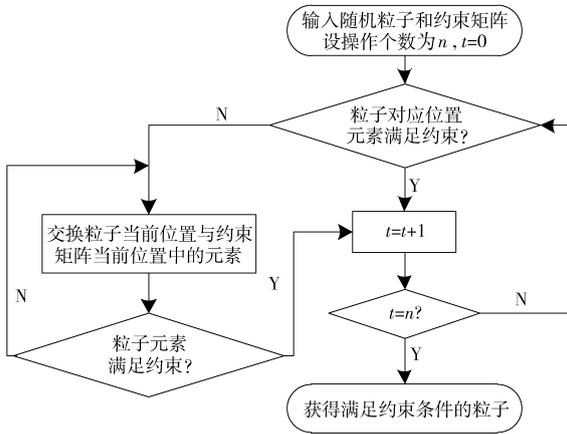


图 2 约束添加流程图

Fig. 2 Flow chart of adding constraints

根据上述算法, 运用 Matlab 软件编写相关程序, 算法中各参数设置如下: 取粒子数 $m = 40$, 最大迭代次数 $M = 250$, 学习因子 c_1 和 c_2 为 2, 惯性权重 $w = 0.8$, 退火起、止温度 T_0 和 T 分别为 1000°C 和 10°C , 退火速度 $K = 0.9$, 采用 SA-PSO 算法和标准 PSO 算法的目标函数 J_1 和 J_2 的收敛情况如

图 3 所示, 从图中可以看出, 采用混合算法的目标函数收敛速度明显加快。最终得到各工作站最优的操作分配、加工时间、节拍、负荷均衡度如表 3 所示。

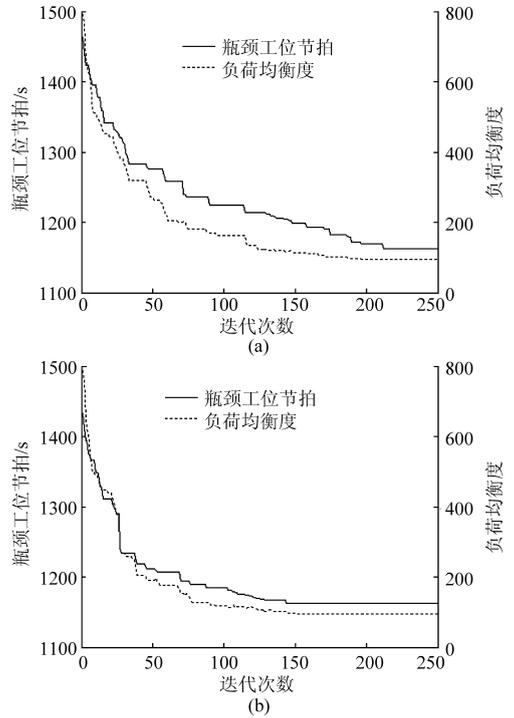


图 3 算法结果

Fig. 3 Result of algorithm

(a) SA-PSO (b) 标准 PSO

表 3 生产线工艺平衡结果

Tab. 3 Result of production line process balancing

工作站编号	各工作站操作	作业时间/s	瓶颈节拍/s	负荷均衡度
1	$f_{51} f_{71} f_{61} f_{42} f_{291} f_{91} f_{81} f_{82} f_{231} f_{251} f_{252} f_{101} f_{111} f_{121} f_{131} f_{141} f_{151} f_{132} f_{161} f_{162} f_{261} f_{271} f_{281}$	1 108.3		
2	$f_{11} f_{21} f_{22} f_{301} f_{31} f_{32} f_{41} f_{42} f_{311} f_{302} f_{62} f_{92} f_{192} f_{201} f_{221} f_{222} f_{102} f_{112} f_{191} f_{202} f_{211} f_{212}$	1 163.0	1 163.0	93.7
3	$f_{122} f_{241} f_{242} f_{243} f_{262} f_{272} f_{282} f_{292} f_{312} f_{313} f_{321} f_{322} f_{331} f_{332} f_{333} f_{341} f_{342}$	1 106.5		
4	$f_{12} f_{23} f_{152} f_{181} f_{142} f_{143} f_{163} f_{171} f_{172} f_{182} f_{232}$	1 112.0		

6 结束语

针对发动机缸体结构的复杂性和加工特征数量大等特点, 以缸体加工生产线上工作站的节拍最短、负荷最均衡为优化目标建立数学模型, 采用 SA-

PSO 混合算法进行求解, 并对提出一种编码、解码及粒子约束处理方法, 加快了算法的收敛速度, 并且有效地避免陷入局部最优, 最后以某缸体生产为例验证了该方法的有效性。

参 考 文 献

- 1 Nima Hamta, Fatemi Ghomi S M T, Jolai F, et al. A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect [J]. International Journal of Production Economics, 2013, 141(1): 99-111.
- 2 Dimitris I Petropoulos, Andreas C Nearchou. A particle swarm optimization algorithm for balancing assembly lines [J]. Assembly Automation, 2011, 31(2): 118-129.
- 3 皮兴忠, 范秀敏, 严隽琪. 基于可行作业序列的遗传算法求解第二类装配线平衡问题 [J]. 上海交通大学学报, 2005, 39(7): 1123-1127.
Pi Xingzhong, Fan Xiumin, Yan Junqi. Applying the genetic algorithm based on feasible task sequence to ALB-2 [J]. Journal of Shanghai Jiaotong University, 2005, 39(7): 1123-1127. (in Chinese)
- 4 窦建平, 苏春, 李俊. 求解第 I 类装配线平衡问题的离散粒子群优化算法 [J]. 计算机集成制造系统, 2012, 18(5): 1021-

1 030.

Dou Jianping, Su Chun, Li Jun. Discrete particle swarm optimization algorithms for assembly line balancing problems of type I [J]. Computer Integrated Manufacturing Systems, 2012, 18(5): 1 021 - 1 030. (in Chinese)

- 5 郑巧仙,李元香,李明,等. 面向第II类装配线平衡问题的蚁群算法[J]. 计算机集成制造系统, 2012,18(5):999 - 1 005. Zheng Qiaoxian, Li Yuanxiang, Li Ming, et al. Ant colony optimization for type II assembly line balancing problem [J]. Computer Integrated Manufacturing Systems, 2012,18(5):999 - 1 005. (in Chinese)
- 6 扈静,刘明周,雷经发,等. 基于仿真的车间生产线动态平衡方法[J]. 农业机械学报,2009,40(3):194 - 198,207. Hu Jing, Liu Mingzhou, Lei Jingfa, et al. Method for dynamic balancing of production line in workshop [J]. Transactions of the Chinese Society for Agricultural Machinery, 2009,40(3):194 - 198,207. (in Chinese)
- 7 潘全科,王文宏,朱剑英. 基于粒子群优化和模拟退火的混合调度算法[J]. 中国机械工程,2006,17(10):1 044 - 1 047.
- 8 鲁建厦,蒋玲玲,李修琳. 基于混合粒子群算法求解装配线第二类平衡问题[J]. 中国机械工程,2010,21(4):420 - 424.
- 9 扈静,蒋增强,葛茂根,等. 基于改进遗传算法的混合装配生产线平衡问题研究[J]. 合肥工业大学学报,2010,33(7):1 006 - 1 009,1 019. Hu Jing, Jiang Zengqiang, Ge Maogen, et al. Study on balance problem of mixed-model assembly line based on improved genetic algorithm [J]. Journal of Hefei University of Technology, 2010,33(7):1 006 - 1 009,1 019. (in Chinese)
- 10 陈星宇. 基于改进遗传算法的装配生产线平衡技术研究[D]. 上海:上海交通大学,2011. Chen Xingyu. Study on assembly line production balance technique based on the improved genetic algorithm [D]. Shanghai: Shanghai Jiaotong University,2011. (in Chinese)
- 11 Eberhart R C, Kennedy J. A new optimizer using particle swarm theory[C]//Proceedings of the 6th International Symposium on Micro Machine and Human Science. Piscataway: IEEE Service Center,1995:39 - 43.
- 12 李丽,牛奔. 粒子群优化算法[M]. 北京:冶金工业出版社,2009.
- 13 魏秀业,潘宏侠. 粒子群优化及智能故障诊断[M]. 北京:国防工业出版社,2010.
- 14 王云,冯毅雄,谭建荣,等. 基于多目标粒子群算法的柔性作业车间调度优化方法[J]. 农业机械学报,2011,42(2):190 - 196. Wang Yun, Feng Yixiong, Tan Jianrong, et al. Optimization method of flexible job-shop scheduling based on multi-objective particle swarm optimization algorithm [J]. Transactions of the Chinese Society for Agricultural Machinery, 2011,42(2):190 - 196. (in Chinese)
- 15 张沙清,陈新度,陈庆新,等. 基于改进微粒群算法的模具多项目动态调度[J]. 计算机集成制造系统,2011,17(3):622 - 629. Zhang Shaqing, Chen Xindu, Chen Qingxin, et al. Dynamic scheduling for multiple mould and die projects based on improved particle swarm optimization[J]. Computer Integrated Manufacturing Systems, 2011,17(3):622 - 629. (in Chinese)
- 16 Nie P, Geng J, Qin Z G. Multi-path oriented particle swarm optimization automatic test case generation algorithm [J]. Computer Integrated Manufacturing Systems, 2012, 18(1): 216 - 223.
- 17 Coello C, Pulido G, Lechuga M. Handling multiple objectives with particle swarm optimization [J]. IEEE Transactions on Evolutionary Computation, 2004,8(3):256 - 279.
- 18 曾建潮,崔志华. 一种保证全局收敛的PSO算法[J]. 计算机研究与发展,2004,41(8):1 333 - 1 338.
- 19 钱锋. 粒子群算法及其工业应用[M]. 北京:科技出版社,2013.
- 20 汪定伟,汪俊伟,王洪峰,等. 智能优化方法[M]. 北京:高等教育出版社,2007.

Machining Line Balancing of Engine Cylinder Block Based on SA - PSO Algorithm

Xu Liyun Xu Changfei Deng Wei Li Aiping

(Institute of Advanced Manufacturing Technology, Tongji University, Shanghai 201804, China)

Abstract: Considering high efficient searching ability of PSO and local searching capability of SA, a hybrid PSO algorithm for solving the problem of machining processes balancing was proposed. Aiming at the characteristics of operations order and clamping constraints, an improved way of encoding and decoding was presented, which makes the coding sequence of discrete matches with the iterative evolution of continuous particle position and speed, at the same time, a method of dealing with such operational constraints was also stated, which makes the updated particles always satisfy the constraints, the convergence rate of the algorithm was improved greatly. Finally, a case of diesel engine cylinder block machining production line was illustrated to prove the validity of the proposed method.

Key words: Engine cylinder block Production line balancing Cycle time Simulated annealing algorithm Hybrid particle swarm algorithm