

流水车间成组工件调度问题的多目标优化算法

杨开兵^{1*}, 刘晓冰²

(1. 大连工业大学 信息科学与工程学院, 辽宁 大连 116034; 2. 大连理工大学 CIMS 中心, 辽宁 大连 116024)

(* 通信作者电子邮箱 kaibingy@126.com)

摘要:针对优化目标是最小化全部提前/拖期和机器调整次数的多目标流水车间成组工件调度问题,提出了一种改进的变权重进化算法结合延迟调整算法的联合优化方法。首先采用改进的变权重进化算法对加工排序进行寻优;其次,在给定调度序列的情况下采用延迟调整算法对加工时刻进行优化。仿真实验表明,所设计的算法能够有效地求解该类问题。

关键词:提前/拖期;多目标优化;进化算法;工件组调整;流水车间

中图分类号: TP301 **文献标志码:** A

Multi-objective optimization algorithm for flow shop scheduling with family setup times

YANG Kai-bing^{1*}, LIU Xiao-bing²

(1. College of Information Science and Engineering, Dalian Polytechnic University, Dalian Liaoning 116034, China;

2. CIMS Center, Dalian University of Technology, Dalian Liaoning 116024, China)

Abstract: The objective optimization was to minimize total earliness/tardiness and number of setups at machine. A jointed algorithm to solve problems based on Control Weight Evolutionary Algorithm (CWEA) and optimization algorithm was presented. Firstly, the CWEA was used to determine scheduling sequence preference. Secondly, a kind of optimization algorithm was put forward to adjust the starting time for determined scheduler. The simulation results show that the effectiveness of the proposed algorithm in solving the problem.

Key words: earliness/tardiness; multi-objective optimization; evolutionary algorithm; family setup; flow shop

0 引言

流水车间调度问题是一类重要的生产调度问题,传统的流水车间调度问题假设调整时间相对于加工时间是可忽略的,或者将调整时间包含在加工时间中。然而,在实际制造环境中,当调整时间占有相当比例,且存在着多种类别的工件时,不但要考虑可分离的调整时间,还需考虑任务间的相似性。如机械行业的零部件加工过程中,待加工的工件往往有多种类型,每种类型的工件包含若干数量,同类工件连续加工时,其后的工件无需调整时间,不同类型的工件连续加工时,其后的工件需要调整时间,此类问题可归结为流水车间成组工件调度问题。

总提前/拖期最小是准时制生产方式下重要的优化目标,在考虑生产率情况下,机器的调整次数也是需要优化的目标,而这两个目标之间并非正则的,即一个目标的减少有可能使另一个目标增加,因此,本文选取这两项调度准则作为同时优化的目标。目前,对提前/拖期调度的研究主要集中于单目标优化^[1-2],而实际的调度问题常常需要考虑多个相互冲突的目标。研究^[3]表明,对于有工件组调整时间的单目标调度问题,即使只有两个工件组也是 NP 难题,对于有多个工件组和较大规模工件数的多目标优化问题,更是如此。

20 世纪 80 年代作为关键智能计算技术之一的进化计算理论在多目标优化领域开始得到应用,并逐渐形成了众多的多目标进化算法,如非劣分类遗传算法(Nondominated Sorting Genetic Algorithm, NSGA-II)^[4]、强度 Pareto 进化算法

(Strength Pareto Evolutionary Algorithm, SPEA)^[5]、Pareto 存档进化策略(Pareto Archived Evolution Strategy, PAES)^[6]、Pareto 适应度遗传算法(Pareto Fitness Genetic Algorithm, PFGA)^[7]等。近年来,将遗传算法与局部搜索结合用于多目标组合优化问题,受到了许多研究者的关注^[8-9],同时,遗传算法与各种混合算法的融合在多目标流水车间调度中的应用也取得了一些成功^[10-12]。本文设计了一种改进的变权重进化算法并结合工件的延迟调整算法,用来求解多目标流水车间成组工件调度问题,取得了满意的效果。与现有文献中随机生成权重方法不同,本文的权重在每次迭代中围绕着预先定义的方向向量变化,使算法在每一阶段着力搜索 Pareto 前沿的特殊区域,而不同的方向向量将引导搜索朝向 Pareto 前沿的不同区域。因此,通过多阶段进化迭代过程,可使搜索范围覆盖整个非劣最优区域,同时又可保证最优解的多样性。

1 问题描述

本文问题可描述为:有 n 个工件依次经过机器 1,机器 2, ..., 机器 M 进行加工,工件在每台机器上的加工顺序相同;每台机器在同一时间只能加工一个工件,而每个工件在同一时刻只能由一台机器进行加工;据加工的相似性,这 n 个工件分为若干种类型,同种类型的工件加工时机器不需调整,而机器顺序加工不同类型的工件时需要进行适当的调整。假定零时刻所有工件均已到达、不允许抢先加工,每台机器在顺序加工同种类型的工件时不发生空闲,机器在整个加工过程中无故障,要求确定各工件的加工顺序和开始时刻,使调度目标达到

收稿日期:2012-06-21;修回日期:2012-09-06。 基金项目:国家自然科学基金资助项目(70572098)。

作者简介:杨开兵(1967-),女,辽宁大连人,讲师,博士,主要研究方向:多目标优化、进化算法; 刘晓冰(1956-),男,吉林长春人,教授,博士生导师,主要研究方向:计算机集成制造、企业信息化。

最优,这里选取总提前/拖后时间和机器调整次数达到最小。

设 $N = \{1, 2, \dots, n\}$ 为工件集合, π 为工件的一个加工序列, N_{ST} 为 π 的机器调整次数, d_j 为工件 j 的交货期, C_j 为工件 j 的完成时间, E_j 和 T_j 分别为工件 $j \in N$ 的提前和拖后时间, $E_j = \max[d_j - C_j, 0]$, $T_j = \max[C_j - d_j, 0]$, 则优化目标为:

$$\min f_1(\pi) = \sum_{j \in N} (E_j + T_j)$$

$$\min f_2(\pi) = N_{ST}$$

根据实际生产作业,假定:

- 1) 同类工件可拆分,工件在每台机器上的顺序是相同的;
- 2) 顺序加工的同类工件在每台机器上必须连续加工,其间机器不发生空闲;
- 3) 每台机器的首个工件无须调整时间;
- 4) 一台机器同一时间只能加工一个工件,一个工件同一时间只能被一台机器加工;
- 5) 所有工件零时刻到达,不允许抢先加工,工件一旦加工不允许中断;
- 6) 机器零时刻可用。

2 多目标优化

对于提前/拖期调度问题,不但要考虑工件的加工顺序,还要考虑工件的开始加工时刻,需要优化多个变量。为此,本文设计了一种改进的进化算法优化工件的加工顺序,然后在加工顺序已定的情况下,优化工件的开工时刻。

2.1 可控权重进化算法

2.1.1 编码及遗传操作

解的编码采用基于工件加工顺序的自然数编码,表示为 $X = (x_1, x_2, \dots, x_n)$, 其中 $x_i \in \{1, 2, \dots, n\}$ 为工件代号。遗传操作包括交叉操作和变异操作,交叉操作采用两点交叉方式,即在一个父代个体中随机选择两个交叉点,两个交叉点将该个体分为三个部分,交叉点两边的工件由父代传给子代,其余的工件从另一父代个体中获得,但是该工件必须保证与已有的工件不同,且顺序与第二个父代相同。变异操作采用插入变异方式,即在实施变异的个体中随机选择一个工件,将其插入到随机选择的另一个工件之前。

2.1.2 多阶段搜索策略

可控权重进化算法 (Control Weight Evolutionary Algorithm, CWEA) 采用多阶段搜索策略,算法执行几个搜索阶段,每个阶段搜索解空间的特定区域,并且迭代若干次,以加强该区域的搜索。这里采用 5 个搜索阶段,每一阶段的最大迭代次数为 G ,共进行 $5G$ 次迭代。每一阶段预先赋予一个方向向量 λ^r 引导群体朝着不同的 Pareto 最优区域进化。为保证得到的 Pareto 最优解具有良好的分散性,方向向量取为 $\lambda^1 = (1, 0)$, $\lambda^2 = (0, 1)$, $\lambda^3 = (1/4, 3/4)$, $\lambda^4 = (3/4, 1/4)$, $\lambda^5 = (1/2, 1/2)$ 。

2.1.3 计算解的适应度

解的适应度采用多个目标线性加权的形式,因为是极小化问题,所以解的适应度为 $fitness(X) = - \sum_{k=1}^2 w_k f_k(X)$, 其中 $w_k (k = 1, 2)$ 为各目标的权重, w_k 决定着每次迭代的搜索方向。在阶段 $r (r = 1, 2, \dots, 5)$ 的第 i 次迭代 ($i = 1, 2, \dots, G$) 中,权重 $w = (w_1, w_2)$ 围绕着方向向量 $\lambda^r = (\lambda_1^r, \lambda_2^r)$ 作随机变化,具体地,令 $w = (\lambda_1^r + \delta_1, \lambda_2^r + \delta_2)$, 而 δ_1, δ_2 是区间 $[-1/8,$

$1/8]$ 中的随机数。若 $\lambda_1^r + \delta_1, \lambda_2^r + \delta_2$ 中出现负数,则令之为零,然后标准化,使 $\sum_{k=1}^2 w_k = 1$ 。

2.2 确定工件的开工时刻

在工件的加工排序中,连续加工的同种类型的工件序列,本文称为作业组,一个可行的调度方案是由一系列作业组构成的,同一个作业组的工件都是同种类型的工件,相邻两个作业组属于不同的工件类型,因而存在着调整时间。作业组延迟调整算法的基本思想是先按尽早开工方式确定每个作业组的开工时间,然后从最后一个作业组开始,逐个作业组进行调整:计算该作业组迟后开工单位时间时目标函数 $f_1(\pi)$ 的改变量 Δf , 如果 $\Delta f < 0$, 则将作业组迟后开工,直到不再有 $\Delta f < 0$ 为止。

设一个加工序列中有 F 个作业组,按先后顺序依次表示为集合 $1, 2, \dots, F, N_f$ 表示作业组 f 中的工件数, $f = 1, 2, \dots, F$, 则 $\sum_{f=1}^F N_f = n$ 。又设 I_f 为作业组 $f-1$ 与 f 之间在机器 M 上的空闲时间,如果 $I_{f+1} = 0$, 则作业组 f 迟后开工会改变作业组 $f+1$ 的完成时间。设作业组 f 迟后开工一个时间单位,完成时间受到影响的工件组为 $f+1, f+2, \dots, f+r$, 则 $I_{f+1} = I_{f+2} = \dots = I_{f+r} = 0, I_{f+r+1} > 0$ 。又设作业组 f 中工件 j 的提前时间为 $E_{f,j}$, 拖后时间为 $T_{f,j}$, 作业组 f 中提前交货的工件数为 NE_f , 拖后交货以及准时交货的工件数为 NT_f , 当 $\sum_{g=f}^{f+r} NE_g > \sum_{g=f}^{f+r} NT_g$ 时, 作业组 f 迟后开工至多 $\min\{\min_j\{E_{f,j}, E_{f+1,j}, \dots, E_{f+r,j}\}, I_{f+r+1}\}$ 个时间单位,可使目标函数 $f_1(\pi)$ 值单调减小。算法步骤如下:

步骤 1 对于已定的工件排序,首先确定每一作业组的最早开工时刻,然后计算各作业组的提前时间 $E_{f,j}$, 拖后时间 $T_{f,j}$ 及机器的空闲时间 $I_f (f = 1, 2, \dots, F)$;

步骤 2 如果作业组 F 中提前工件数 NE_F 大于拖后以及准时的工件数 NT_F , 则将延迟加工 $\min\{E_{F,j}\}$ 个时间单位,并修正 $E_{F,j}, T_{F,j}, j \in F$ 以及 I_F , 置 $f = F - 1$;

步骤 3 寻找使 $I_{f+r+1} > 0$ 的最小整数 $r (r \geq 0)$, 直到 $f + r + 1 = F$, 若无满足条件的 r , 则令 $r = F - f$;

步骤 4 计算作业组 $f+1, f+2, \dots, f+r$ 中提前、拖后及准时交货的工件数。当 $\sum_{g=f}^{f+r} NE_g > \sum_{g=f}^{f+r} NT_g$ 时, 作业组 f 后移 $\min\{\min_j\{E_{f,j}, E_{f+1,j}, \dots, E_{f+r,j}\}, I_{f+r+1}\}$ 个时间单位,并修正 $E_{g,j}, T_{g,j}, j \in g, g = f, f+1, \dots, f+r$ 以及 I_{f+r+1} , 转步骤 3;

步骤 5 置 $f = f - 1$, 如果 $f \geq 1$ 转步骤 3;

步骤 6 算法结束,确定各工件的开工时刻。

2.3 混合邻域搜索

对于工件有类别属性,且需要考虑调整时间的调度问题,利用问题本身的知识设计邻域能够有效地提高算法的收敛性能^[13]。这里采用两种邻域 $N_1(\cdot)$ 和 $N_2(\cdot)$ 进行局部搜索。

$N_1(\cdot)$ 表示作业组间交换,即在当前作业组序列中随机选择一个作业组与其他作业组交换加工顺序形成新的作业组序列。

$N_2(\cdot)$ 表示作业组内交换,即在当前作业组序列中随机选择一个工件,与同一作业组内的其他工件交换位置形成新的作业组序列。

局部搜索采用两阶段混合邻域搜索策略:第一阶段 (LS1) 采用组间交换邻域结构,在当前解的邻域内搜索使目

标函数有较大改善的解;第二阶段(LS2)采用组内交换邻域结构,对第一阶段获得的解进一步改善。算法的流程如下:

- 1) LS1 初始化。确定算法的初始解 x_0, k_{\max} 为搜索的最大步长,令算法当前解 $x = x_0$ 。
- 2) 令 $k = 1$ 。
- 3) 随机产生 $x' \in N_1(x)$ 。
- 4) 如果 x' 优于 x , 令 $x = x'$, 并令 $k = 1$; 否则令 $k = k + 1$ 。
- 5) 如果 $k > k_{\max}$, 转到步骤 6); 否则转到步骤 3)。
- 6) LS2 初始化。令当前解 $y = x$ 。
- 7) 令 $k = 1$ 。
- 8) 随机产生 $y' \in N_2(y)$ 。
- 9) 如果 y' 优于 y , 令 $y = y'$ 并令 $k = 1$; 否则令 $k = k + 1$ 。
- 10) 如果 $k > k_{\max}$, 则输出 y ; 否则转到步骤 8)。

2.4 CWEA 实现步骤

- 1) 设定参数: 每一阶段的进化代数 G , 群体规模 N_{pop} , 交叉概率 P_c , 变异概率 P_m , 最大精英解的数目 N_{elite} , 局部搜索步长 k_{\max} 。
- 2) 定义每一阶段的方向权重 $\lambda^1, \lambda^2, \dots, \lambda^5$, 初始化外部集 $P_{nd} = \emptyset$ 。
- 3) 执行第 r 阶段的搜索:
 - ① 随机产生 N_{pop} 个的初始个体, 计算每一解个体的目标值;
 - ② 按 2.1 节的方法计算当前群体 P_t 中所有解个体的适应度;
 - ③ 遗传操作, 即从 P_t 中以赌轮方式选择两个不同的父代个体进行交叉, 得到子代个体, 每个子代个体进行变异, 共产生 $N_{\text{pop}} - N_{\text{elite}}$ 个个体形成子代群体 P_{t+1} ;
 - ④ 从集合 P_{nd} 中随机地选出 N_{elite} 个个体, 加入到子代群体 P_{t+1} 中;
 - ⑤ 应用 2.3 节的方法对群体 P_{t+1} 中的每一个体进行局部搜索, 得到子代群体 P_{t+1}^* ;
 - ⑥ 修正集合 P_{nd} : 将当前群体 P_t 中的非劣解加入到 P_{nd}

中, 再从 P_{nd} 中删去被占优的个体;

- ⑦ 如果已经迭代 G 次则转 ⑧; 否则 $P_t = P_{t+1}^*$, 转 ②;
- ⑧ 如果已经执行 5 个阶段, 则转 4); 否则转 3)。
- 4) 输出外部集 P_{nd} 。

3 仿真实验

为检验本文的变权重进化算法(CWEA)的优化性能, 采用多目标遗传局部搜索(Multi-Objective Genetic Local Search, MOGLS)^[10], NSGA-II^[4] 和多目标粒子群优化算法(Multi-Objective Particle Swarm Optimization, MOPSO)^[14] 作为参照算法, 采用文献[15]的调度实例进行仿真实验。每一调度实例表示成 $ABCfm$, 其中 ABC 表示问题的类型, f 表示工件的类别数, m 表示机器台数, 工件总数 n 为 f^2 。如果调整时间比加工时间小, 以 SSU 表示; 如果调整时间与加工时间相当, 以 MSU 表示; 如果调整时间比加工时间大, 以 LSU 表示。每一问题又随机生成 10 个算例。本文按照文献[16]的方法产生每个工件的交货期。各算法在 CPU 主频 3.2 GHz、内存 512 MB 的个人电脑上进行, 参数的设置与文献[10]相同, 群体规模 $N_{\text{pop}} = 30$, 交叉概率 = 0.9, 变异概率 = 0.3, 精英解的数目 $N_{\text{elite}} = 3$, 局部搜索步长 $k_{\max} = 2$, 终止条件均设为评价 100 000 个解。每种算法在相同的初始群体下独立运行 10 次, 将各次得到的近似最优解集合并后, 获得各问题的最终解集, 作为参考集。

获得的解的质量采用文献[17]提出的基于与参考集的距离指标 D_{1R} 来评价, D_{1R} 可同时考察解集的收敛性和多样性。设 S^* 是参考解集, S_j 是评价的解集, 则 D_{1R} 定义如下:

$$D_{1R}(S_j) = \frac{1}{|S^*|} \sum_{y \in S^*} \min \{d_{xy} \mid x \in S_j\}$$

其中 d_{xy} 是解 x 和参考解 y 在 p 维标准化目标空间的距离。 $D_{1R}(S_j)$ 越小, 表明解集 S_j 越好。

表 1 是 4 种算法的评价指标 D_{1R} 和平均运行时间。

表 1 4 种算法评价指标 D_{1R} 和运行时间的比较

| 问题 | 评价指标 D_{1R} | | | | 运行时间/s | | | |
|---------|---------------|--------|---------|--------|--------|-------|---------|-------|
| | CWEA | MOGLS | NSGA-II | MOPSO | CWEA | MOGLS | NSGA-II | MOPSO |
| SSU55 | 0.04 | 0.14 | 0.14 | 0.43 | 18 | 19 | 30 | 27 |
| SSU56 | 0 | 0.23 | 0.11 | 0.19 | 24 | 25 | 37 | 29 |
| SSU65 | 0 | 0.18 | 0.03 | 3.23 | 25 | 26 | 39 | 41 |
| SSU66 | 0 | 0.21 | 0.14 | 1.10 | 29 | 38 | 46 | 47 |
| SSU88 | 0 | 0.1002 | 1.07 | 7.45 | 71 | 72 | 103 | 130 |
| SSU108 | 0 | 5.86 | 4.87 | 131.23 | 116 | 138 | 180 | 209 |
| SSU1010 | 0 | 3.52 | 1.56 | 26.07 | 150 | 169 | 203 | 262 |
| MSU55 | 0 | 0.12 | 0.08 | 0.20 | 20 | 22 | 32 | 26 |
| MSU56 | 0 | 0.04 | 0.15 | 0.35 | 23 | 25 | 35 | 29 |
| MSU65 | 0 | 0.19 | 0.06 | 1.38 | 27 | 28 | 33 | 40 |
| MSU66 | 0 | 0.22 | 0.10 | 0.54 | 30 | 32 | 46 | 47 |
| MSU88 | 0 | 0.28 | 0.27 | 4.27 | 83 | 83 | 111 | 125 |
| MSU108 | 0 | 1.23 | 0.72 | 12.96 | 128 | 138 | 146 | 217 |
| MSU1010 | 0 | 0.69 | 0.52 | 6.61 | 164 | 175 | 221 | 255 |
| LSU55 | 0 | 0.18 | 0.04 | 0.24 | 21 | 23 | 33 | 26 |
| LSU56 | 0 | 0.13 | 0.10 | 0.11 | 25 | 26 | 36 | 31 |
| LSU65 | 0 | 0.20 | 0.16 | 0.87 | 29 | 29 | 44 | 41 |
| LSU66 | 0 | 0.21 | 0.23 | 1.20 | 33 | 34 | 49 | 51 |
| LSU88 | 0.01 | 0.29 | 0.31 | 2.20 | 85 | 88 | 115 | 127 |
| LSU108 | 0.01 | 0.44 | 0.51 | 8.57 | 141 | 145 | 196 | 221 |
| LSU1010 | 0.02 | 0.32 | 0.36 | 4.76 | 169 | 178 | 230 | 253 |

从表 1 中可以看出, CWEA 的指标 D_{IR} 明显好于其他三种算法, 尤其是 MSU 问题。此外, CWEA 的运行时间较少, 说明 CWEA 较其他三种算法有较好的优化性能, 而且, 随着工件数的增加, 算法的优化性能发挥得越好。

图 1(a) 和(b)是以问题 MSU66 和 LSU65 为例, 描述了 4 种算法得到的 Pareto 前沿的比较。从图中可以看出, 在相同的算法终止条件下, 本文 CWEA 得到的近似 Pareto 解数量较多, 且均匀地分布在 Pareto 前沿上。

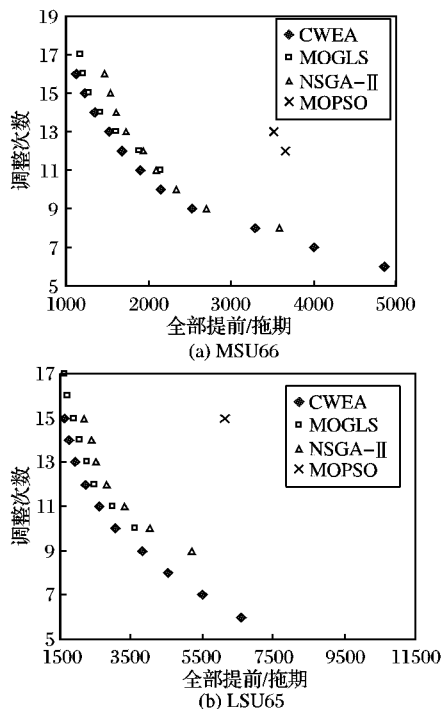


图 1 4 种算法得到的 Pareto 前沿

4 结语

本文针对最小化全部提前/拖期和调整时间的流水车间成组工件调度问题, 设计了可控权重进化算法对工件的加工顺序进行寻优, 采用延迟调整算法确定固定工件排序下的最优开工时刻。算法通过分阶段不同区域的搜索, 保持了解的多样性, 在每个阶段中嵌入基于问题知识的局部搜索过程, 提高了解的搜索效率。仿真实验验证了本文算法具有较好的优化效果。

参考文献:

- [1] 常俊林, 郭西进, 马小平. 并行机成组调度问题的启发式算法[J]. 计算机工程与应用, 2007, 43(4): 234-236.
- [2] SCHALLER J E, GUPTAN D. Scheduling on a single machine with family setups to minimize total tardiness[J]. International Journal of Production Economics, 2007, 105(2): 329-344.
- [3] CHEN ZHILONG. Scheduling with batch setup times and earliness tardiness penalties[J]. European Journal of Operational Research, 1997, 96(3): 518-537.
- [4] 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.
- [5] ZITZLER E, THIELE L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach[J]. IEEE Transactions on Evolutionary Computation, 1999, 3(4): 257-271.
- [6] KNOWLES J D, CORNE D W. Approximating the non-dominated front using the Pareto archived evolution strategy[J]. Evolutionary Computation, 2000, 8(2): 149-172.
- [7] ELAOU S, LOUKIL T, TEGHEM J. The Pareto fitness genetic algorithm: test function study[J]. European Journal of Operational Research, 2007, 177(3): 1703-1719.
- [8] JASZKIEWICZ A. Genetic local search for multi-objective combinatorial optimization[J]. European Journal of Operational Research, 2002, 137(1): 50-71.
- [9] JASZKIEWICZ A. On the performance of multiple-objective genetic local search on the 0/1 knapsack problem: a comparative experiment[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(4): 402-412.
- [10] ISHIBUCHI H, YOSHIDA T, MURATA T. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling[J]. IEEE Transactions on Evolutionary Computation, 2003, 7(2): 204-223.
- [11] ARROYO J E C, ARMENTANO V A. Genetic local search for multi-objective flowshop scheduling problems[J]. European Journal of Operational Research, 2005, 167(3): 717-738.
- [12] KARIMI N, ZANDIEH M, KARAMOOZ H R. Bi-objective group scheduling in hybrid flexible flowshop: a multi-phase approach[J]. Expert Systems with Applications, 2010, 37(6): 4024-4032.
- [13] 杨开兵. 基于进化计算的多目标流水车间批组调度问题研究[D]. 大连: 大连理工大学, 2011.
- [14] COELLO C A, PULIDO G T, LECHUNGA M S. Handling multiple objectives with particle swarm optimization[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(3): 256-279.
- [15] FRANCA P M, GUPTA N D, MENDES A S, *et al.* Evolutionary algorithm for scheduling a flowshop manufacturing cell with sequence dependent family setups[J]. Computers and Industrial Engineering, 2005, 48(3): 491-506.
- [16] RUIZ R, STUTZLE T. An iterated greedy heuristic for the sequence dependent setup time flowshop problem with makespan and weighted tardiness objectives[J]. European Journal of Operational Research, 2008, 187(3): 1143-1159.
- [17] KNOWLES J D, CORNE D W. On metrics for comparing nondominated sets[C]// Proceedings of the Congress on Evolutionary Computation. Piscataway: IEEE, 2002: 711-716.
- [10] GENG XIUBO, LIU TIEYAN, QIN TAO, *et al.* Query dependent ranking using K-nearest neighbour[C]// Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information and Retrieval. New York: ACM, 2008: 115-122.
- [11] 王扬, 黄亚楼, 谢茂强, 等. 多查询相关的排序支持向量机融合算法[J]. 计算机研究与发展, 2011, 48(4): 558-566.
- [12] 孙鹤立, 黄健斌, 冯博琴, 等. 查询依赖的有序多超平面排序学习模型[J]. 软件学报, 2011, 22(11): 2773-2781.
- [13] CAO YUNBO, XU JUN, LIU TIEYAN, *et al.* Adapting ranking SVM to document retrieval[C]// Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information and Retrieval. New York: ACM, 2006: 186-193.
- [14] CHANG C C, LIU C J. LIBSVM: A library for support vector machines[J]. ACM Transactions on Intelligent Systems and Technology, 2011, 2(3): 27.

(上接第 3334 页)