

一种多目标置换流水车间调度问题的优化算法^①

何启巍, 张国军, 朱海平, 刘 敏

(华中科技大学 机械科学与工程学院, 武汉 430074)

摘 要: 针对最大完工时间最小和总流经时间最小的多目标置换流水车间调度问题(permutation flow shop scheduling problem, PFSP), 提出一种粒子群优化算法与变邻域搜索算法结合的混合粒子群优化(hybrid particle swarm optimization algorithm, HPSO)算法, 并使算法在集中搜索和分散搜索之间达到合理的平衡. 在该混合算法中, 采用 NEH 启发式算法进行种群初始化, 以提高初始解质量; 运用随机键表示法设计基于升序排列规则(rank-order-value, ROV), 将连续 PSO 算法应用于置换流水车间调度问题; 引入外部档案集存储 Pareto 解, 并采用强支配关系和聚集距离相结合的混合策略保证解集的分布性; 采用 Sigma 法和基于聚集距离的轮盘赌法进行全局最优解的选择; 提出变邻域搜索算法, 对外部集中的 Pareto 解作进一步地局部搜索. 最后, 运用提出的混合算法求解 Taillard 基准测试集, 并将测试结果与 SPEA2 算法进行比较, 验证该调度算法的有效性.

关键词: 粒子群优化算法; 变邻域搜索; 多目标; 置换流水车间调度

A Hybrid Particle Swarm Optimization Algorithm for Multi-objective Permutation Flow Shop Scheduling Problem

HE Qi-Wei, ZHANG Guo-Jun, ZHU Hai-Ping, LIU Min

(Mechanical Science and Engineering College, Huazhong University of Science and Technology, Wuhan 230039, China)

Abstract: This paper proposes a hybrid particle swarm optimization algorithm for the minimization of makespan and total flowtime in permutation flow shop scheduling problems, which combines particle swarm optimization algorithm with variable neighborhood search algorithm. The initial population is generated by the NEH constructive heuristic to enhance the quality of the initial solutions. A heuristic rule called the ranked order value (ROV) borrowed from the random key representation is developed, which apply the continuous particle swarm optimization algorithm to all classes of sequencing problems. The strategy of constructing external data set based on combining strong predominance ranking and crowding distance ranking was introduced. The global best solution was updated based on the strategy of combining Sigma method and roulette method. VNS was applied to enhance the local search for the pareto solutions. Finally, the proposed algorithm is tested on a set of standard instances taken from the literature provided by Taillard and compared with SPEA2. The computation results validate the effectiveness of the proposed algorithm.

Key words: particle swarm optimization; variable neighborhood search; multi-objective; permutation flow shop scheduling

1 引言

置换流水车间调度问题是目前研究最广泛的一类典型调度问题, 对该问题的研究具有很重要的意义. 国内外学者过去几十年里在这方面做了大量研究, 其中绝大多数的研究都是集中在求解单目标的置换流水

车间调度问题. 然而, 在实际生产制造的过程中, 往往面临多目标决策的问题, 因此对于求解多目标置换流水车间调度问题算法的研究具有重要的理论意义和工程实用价值.

近年来, 基于智能算法的多目标调度研究逐渐得

^① 收稿时间:2013-01-22;收到修改稿时间:2013-04-25

到了学术界和工程界的广泛重视,特别是《欧洲运筹学》等著名国际期刊相继推出相应的专刊,更是推动了多目标智能算法在车间调度问题上的应用。

Ishibuchi 和 Murata^[1]将遗传算法和局部搜索策略相结合,采用基于随机加权的适配值函数进行解的评价和指导遗传操作,提出了一种多目标遗传局部搜索(multi-objective genetic local search, MOGLS)算法。类似地, Jaszkievicz^[2]采用基于随机权的适配值函数,选择父代个体和指导局部搜索,进而给出了一种多目标混合遗传算法。另外, Loukil 等^[3]提出了一类多目标模拟退火算法,通过产生一定数量的随机权向量来构造评价函数集合,进而基于每个评价函数在模拟退火策略的指导下进行局部搜索。

区别于上述方法, Arroyo 和 Armentano^[4]采用 Pareto 支配的概念对当前种群进行层级划分,并基于此对种群中的个体赋予合适的适配值,同时对种群实施并行的多目标局部搜索,以加强对目标空间中解密度较低地区的搜索能力。Li 和 Wang 最近提出了一种基于量子计算的混合遗传算法,同样采用了快速排序和适配值赋值的评价方法,同时算法引入了两种消除种群个体冗余性的环节。

粒子群优化(Particle Swarm Optimization, PSO)算法作为一种新型的基于群体的优化算法,最早由 Kennedy 和 Eberhart^[5]于 1995 年提出。PSO 算法是通过种群内粒子之间的合作与竞争产生的群体智能优化算法。与遗传算法比较, PSO 算法保留了基于种群的全局搜索策略,搜索模型简单,还有收敛速度快和鲁棒性高等特点。PSO 算法早期用于无约束连续函数的优化,并在很多问题上都取得了成功的应用,例如,电压控制、神经网络训练。Tasgetiren 等^[6]又成功将 PSO 算法用于求解单目标的置换流水车间调度问题,推动了 PSO 算法在离散组合优化问题上的应用。

然而,当前粒子群优化算法在多目标调度问题上的研究还是寥寥无几,代表性的工作是 Arroyo 和 Armentano^[7]最近提出的一种多目标 PSO 算法,作者通过对平均完成时间和加权平均拖后时间进行加权来处理评价环节。

本文在已有研究基础上,结合粒子群优化算法和变邻域搜索算法各自的优点,设计了一种混合粒子群优化算法,并求解双目标置换流水车间调度问题。该混合算法采用基于升序排列规则(ROV)的连续 PSO 算

法进行全局搜索,应用基于关键路径的变邻域搜索算法对全局优化的粒子进行局部搜索,使算法在分散搜索和集中搜索达到合理的平衡。并基于 Pareto 支配的概念,设计了一种有效的非劣解集的更新策略。对 Taillard 提出的基准测试集进行仿真实验,与其他多目标算法比较,证实算法的有效性。

2 问题描述

2.1 多目标优化问题的描述

多目标优化问题(multi-objective optimization problem), MOP 问题可以描述为^[8]: 寻找一组既满足约束条件又使总目标函数最优化的决策变量的取值,总目标函数的元素是子目标函数。若 MOP 问题的目标函数为:

$$\min f(x) = (f_1(x), f_2(x), \dots, f_N(x)), x \in X \quad (1)$$

其中, f_1, f_2, \dots, f_N 为 N 个目标函数, x 表示决策解, X 表示解空间

考虑两个解 x_1 和 x_2 , 若满足

$$\begin{cases} \forall j \in \{1, 2, \dots, N\}, f_j(x_1) \leq f_j(x_2) \\ \exists k \in \{1, 2, \dots, N\}, f_k(x_1) < f_k(x_2) \end{cases} \quad (2)$$

则称解 x_1 支配解 x_2 , 记作 $x_1 \succ x_2$ 。

给定一个解 x^* , 若在解空间 X 中不存在支配 x^* 的解, 则称 x^* 为 Pareto 最优解, 或非支配解集(non-dominated solution)。所有非支配解的集合构成多目标意义下的最优解集, 这些解在目标空间中构成问题的 Pareto 前沿(Pareto Front)。

2.2 置换流水车间调度问题描述

置换流水车间调度可以描述为^[9]: n 个工件要在 m 台机器上进行加工, 每个工件的加工顺序相同, 每台机器加工的工件顺序也相同, 各工件在各机器上的加工时间已知, 要求得到一个加工方案使得某一调度目标最优。调度目标一般有: 使工件最大延期量最小; 使工件的总流经时间(total flow time, TFT)最短; 使所有工件最大完工时间(Makespan)最短; 工件的平均等待时间最短等。本文选取最大完工时间最短和总流经时间最短为双目标, 对该问题通常做如下的假设:

- 1) 一个工件在同一时刻只能在一台机器上加工;
- 2) 一台机器在同一时刻只能加工一个工件;
- 3) 工件一旦在某台机器上进行加工就不能停止;
- 4) 每台机器上工件的加工顺序相同。

置换流水车间调度问题的数学描述如下. 令 t_{ij} 为工件 i 在机器 j 上的加工时间, 不计同一机器上加工完工件 i 后马上加工工件 j 所需的准备时间, $C(\pi_i, j)$ 为工件 π_i 的在机器 j 加工完毕时间, 不失一般性, 假设各工件按机器 1 至 m 的顺序进行加工, 令 $\pi = \{\pi_1, \pi_2, \dots, \pi_n\}$ 为所有工件的一个排序

$$\begin{cases} C(\pi_1, 1) = t_{\pi_1, 1} \\ C(\pi_i, 1) = C(\pi_{i-1}, 1) + t_{\pi_i, 1}, i = 2, \dots, n \\ C(\pi_i, j) = C(\pi_i, j-1) + t_{\pi_i, j}, j = 2, \dots, m \\ C(\pi_i, j) = \max\{C(\pi_{i-1}, j), C(\pi_i, j-1)\} + t_{\pi_i, j} \\ i = 2, \dots, n \quad j = 2, \dots, m \end{cases} \quad (3)$$

$$makespan = C_{\max}(\pi) = C(\pi_n, m) \quad (4)$$

$$TFT = \sum_{i=1}^n F(\pi_i) = \sum_{i=1}^n C(\pi_i, m) \quad (5)$$

式(4)和式(5)分别为最大完工时间和总流经时间的计算公式.

3 求解多目标置换流水车间调度问题的混合粒子群优化算法

3.1 标准粒子群优化算法(PSO)

基本粒子群算法采用速度-位置模型进行搜索, 待优化问题的每个候选解称为一个“粒子”, 每个粒子都有自己的位置和速度, 还有一个由被优化函数决定的适应值. 各个粒子记忆、追随当前的最优粒子, 在解空间中搜索. 每次迭代的过程不是完全随机的, 如果找到较好解, 将会以此为依据来寻找下一个解. 令 PSO 算法初始化为一群随机粒子, 在每一次迭代中, 粒子通过跟踪两个“极值”来更新自己: 第一个就是粒子本身所找到的最好解, 叫做个体极值点(用 $pbest$ 表示其位置), 全局版 PSO 中的另一个极值点是整个种群目前找到的最好解, 称为全局极值点(用 $gbest$ 表示其位置), 而局部版 PSO 不用整个种群而是用其中一部分作为粒子的邻居, 所有邻居中的最好解就是局部极值点(用 $lbest$ 表示其位置). 在找到这两个最好解后, 粒子根据如下的式(6)和式(7)来更新自己的速度和位置. 粒子 i 的信息可以用 n 维向量表示, 位置表示为 $X_i = [x_{i1}, x_{i2}, \dots, x_{in}]$, 速度为 $V_i = [v_{i1}, v_{i2}, \dots, v_{in}]$, 其他向量类似. 则速度和位置更新方程为:

$$v_{ij}^{t+1} = w^t \times v_{ij}^t + c1r1(P_{ij}^t - X_i^t) + c2r2(P_{gi}^t - X_i^t) \quad (6)$$

$$X_{ij}^{t+1} = X_{ij}^t + v_{ij}^{t+1} \quad (7)$$

其中: $w^t = w^{t-1} \times \beta$ 是惯性系数, β 为线性递减因子, 其主要作用是产生扰动, 以防止算法的早熟收敛; $c1$ 和 $c2$ 为学习因子, 分别调节向个体最好粒子和全局最好粒子方向飞行的最大步长, 若太小, 则粒子可能远离目标区域, 若太大则会导致突然向目标区域飞去, 或飞过目标区域, 合适的 $c1$ 和 $c2$ 可在加快收敛速度的同时还能不易陷入局部最优, 通常令 $c1=c2=2$; $r1$ 和 $r2$ 是 $[0, 1]$ 之间均匀产生的随机数.

3.2 求解 PFSP 的粒子群优化算法

3.2.1 解的表示和 ROV 规则

对于 PFSP 问题, 文献中最常用的编码方式就是直接采用工件的排序. 由于连续 PSO 算法中微粒的位置为连续值矢量, 为了实现微粒位置矢量到工件排序的映射关系, 借用随机键编码, 王凌等^[10]提出了 ROV 规则, 将粒子的连续位置矢量 $X_i = [x_{i1}, x_{i2}, \dots, x_{in}]$ 转换为离散的加工排序 $\pi = \{\pi_1, \pi_2, \dots, \pi_n\}$, 即机器上各工件的加工顺序.

ROV 规则具体描述如下. 对于一个微粒的位置矢量, 首先将值最小的分量位置赋予 ROV 值为 1, 其次将值第二小的分量位置赋予 ROV 值为 2, 依次类推, 直到将所有的分量位置都赋予一个唯一的 ROV 值, 从而基于 ROV 值则可构造出一个工件排序.

表 1 中用一个简单的例子来表示了 ROV 规则的构造过程. 考虑 7 个工件的置换流水车间调度, 粒子的位置矢量则为 6 维, 设位置矢量 $X_i = [0.49, 2.90, 1.58, 0.82, 2.81, 0.35, 1.28]$, 比较可知 x_{i6} 为最小, 所以将 x_{i6} 对应的分量位置 ROV 值赋为 1, 接下来将 x_{i1} 对应的分量位置 ROV 值赋为 2, 依次类推分别赋予 x_{i4} 、 x_{i7} 、 x_{i3} 、 x_{i5} 和 x_{i2} 对应的分量位置 ROV 值为 3、4、5、6 和 7, 从而得到工件的加工次序, 即 $\pi = \{2, 7, 5, 3, 6, 1, 4\}$.

表 1 粒子位置矢量对应的 ROV 值

分量位置	1	2	3	4	5	6	7
位置分量值	0.49	2.90	1.58	0.82	2.81	0.35	1.28
ROV 值	2	7	5	3	6	1	4

3.2.2 种群初始化

初始种群应该具有一定的分布性, 能够以较大的概率覆盖整个解空间. 此外为了提高种群的搜索效率, 避免盲目搜索, 初始种群中也应该包括部分质量较高的解. 因此初始解的产生方式基本上可分为两种, 一是在一连续区间内随机产生; 二是用构造性启发式方法产生. 在这里我们选用性能最好的 NEH 启发式算法, NEH 启发式算法^[11]是由 Nawaz, Enscore 和 Ham 共同

提出的算法步骤如下:

步骤 1: 按工件在机器上的总加工时间递减的顺序排列 n 个工件.

步骤 2: 取前两个工件调度, 使部分总完工时间达到最小.

步骤 3: 从 $k=3, \dots, n$, 把第 k 个工件插入到 k 个可能的位置, 求得最小的部分总完工时间.

NEH 启发式方式产生的解是工件序列, 必须转换为一定区间内的位置矢量, 在此, 按如下方式实现转换.

$$x_{NEH,j} = x_{min,j} + \frac{x_{max,j} - x_{min,j}}{n} \times (s_{NEH,j} - 1 + r3), \quad (8)$$

$$j = 1, 2, \dots, n$$

其中, $x_{NEH,j}$ 为粒子在第 j 维的位置值, $s_{NEH,j}$ 是通过 NEH 方法得到解的第 j 维工件序号, $x_{max,j}$ 和 $x_{min,j}$ 分别为连续空间上粒子位置矢量的上界值和下界值, $r3$ 代表 0 至 1 间的均匀产生的随机数.

随机产生的方式为, $x_{ij}^0 = x_{min} + (x_{max} - x_{min}) \times r1$, $v_{ij}^0 = v_{min} + (v_{max} - v_{min}) \times r2$, 其中位置矢量值在连续区间 $[x_{min}, x_{max}]$ 间变化, 速度矢量值在连续区间 $[v_{min}, v_{max}]$ 变化, $r1$ 和 $r2$ 均为 0 至 1 之间均匀产生的随机数.

3.3 多目标进化策略

多目标进化算法需保证 Pareto 前沿的收敛性和多样性特征, 关键是设置合理的 Pareto 集多样性维持策略和粒子群全局最优值更新操作. 本文采用基于 Pareto 关系的快速排序法来构造非支配集, 用外部集合(external set, ES)存贮, 引入强支配关系和聚集距离相结合的存档策略, 以使 Pareto 前沿具有良好的分布性. 在全局最优解的选择上, 也采取两阶段领导策略, 前期采用收敛速度极快的 Sigma 方法领导, 后期采用基于聚集距离的轮盘赌方法领导以进行更深入的搜索.

3.3.1 外部集合 ES 的生成

初始种群产生后, 计算各粒子的目标值, 利用快速排序法, 根据目标函数值对粒子进行支配关系排序, 将非支配解存入外部集合 ES 中.

快速排序的思想是每一次循环都从种群中选择一个个体 x (通常选择第一个个体), 种群中其它个体依次与 x 进行比较. 通过一趟比较将种群分割成两部分: 种群的后半部分是被 x 支配的个体, 前半部分是支配 x 或者与 x 不相关的个体. 若 x 不被其它任何一个个体支配, 则将 x 并入到非支配集, 接着再对前半部分重复

上述过程直到前半部分为空.

快速排序的伪代码如图 1 所示:

```

sort(pop)
{Q=pop; (pop 为种群中的所有粒子)
while(Q 不为空集)do
{x∈Q, 令 Q=Q-{x};
x is nondominated; (开始时, 认为 x 为
非支配解)
for each y∈Q
{if (x dominated y) then
Q=Q-{y};
else if (y dominated x) then
x is nondominated=False; }
if (x is not nondominated) then
ES=ES ∪ {x}; }
}

```

图 1 快速排序的伪代码

在每一代粒子更新完成后, 计算单个粒子在各个目标上的适应度值, 更新外部集合 ES. 如图 2 所示, 其更新原则为: (1)若该粒子支配外部集合 ES 中的某些粒子, 则删除被支配的粒子, 将该粒子加入外部集合 ES; (2)若外部集合 ES 中有粒子支配该粒子, 则忽略; (3)若该粒子与外部集合 ES 中的各个粒子互不支配, 则将该粒子加入外部集合 ES 中.

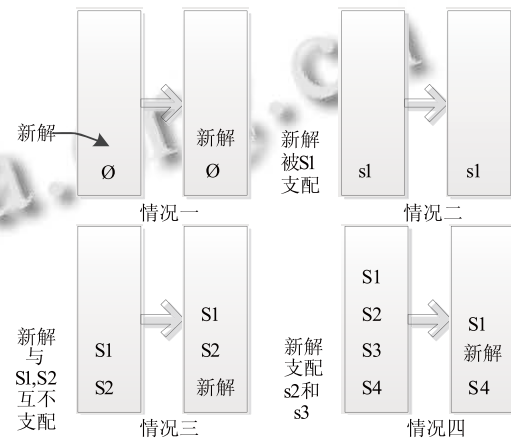


图 2 外部集合更新原则

3.3.2 外部集合 ES 的维护

对于复杂的多目标优化问题, 若保留进化过程中出现的所有的 Pareto 最优解, 会导致外部集合中出现大量的相似解, 不仅影响解的分布性能, 同时也会增加求解的内存开销和时间复杂性. 为保持外部集合的分布性, 本文采用强支配关系和聚集距离相结合的存

档策略对外部集合进行维护。

引入强支配关系可使各个粒子保持一定距离,有效控制粒子的聚合程度,有助于保持粒子群的多样性。强支配关系的定义为:1)如果粒子 $A \succ B$, 则认为 $A \succ_{\text{强}} B$; 2)对于 $A \sim B$, \forall 目标函数 m , 如果有 $|f_m^A - f_m^B| < \varepsilon_m$, ($m=1, 2, \dots, k$)为正常数, 则随机选择 A 或 B; 如果 A 被选中, 则认为 $A \succ_{\text{强}} B$, 反之亦然。

如果 ε_m 取的过小, 则强支配关系的作用就会变小, 当 $\varepsilon_m = 0$ 时, 强支配关系就退化为支配关系; 如果取值过大, 强支配关系变强, 会使粒子变得过于稀疏。可见 ε_m 的取值十分重要。本文利用经验公式取值。

对于只有两个目标的情况: $\varepsilon_m = \frac{f_m^{\max} - f_m^{\min}}{c}$, 其

中 C 为强支配系数, $C > 80$ 。

聚集距离是由 Deb 等^[12]在 2002 年提出并用于 NSGA-II 保持种群多样性的算子, 可以用来表示解的疏密程度。聚集密度小的个体其聚集距离反而大, 一个个体的聚集距离可以通过与其相邻的两个个体在每个子目标上的距离差之和来求取。计算聚集距离的伪代码如下:

```

N=|ES|;
//N 为群体大小
For each l, ES[i].distance=0;
//初始化粒子的聚集距离
  For each objective m
//针对每一个子目标执行如下操作
    {ES=sort(ES, m);
//对子目标 m 进行排序
    For i=2 to (N-1)
      ES[i].distance= ES[i].distance+( ES[i+1].m-
      ES[i-1].m);
//边界点之外的解
    }
  ES[0].distance=ES[N].distance=∞
//给边界点聚集距离赋值无穷大

```

图 3 计算聚集距离的伪代码

3.3.3 全局最优值的选取

由于全局最优解对粒子的导向作用非常明显, 因此如何选取较好的全局最优解 $gbest$ 来引导粒子的飞行, 对算法具有非常重要的作用, 关系到算法的收敛速度、解的多样性等。采取两阶段领导策略, 前期采用收敛速度极快的 Sigma 方法领导, 后期采用基于聚集距离的轮盘赌方法领导以进行更深入的搜索。

Mostaghim 等^[13]提出了基于 Sigma 值的领导选择

方法, 其基本思想是: 赋予群和外部档案中每个粒子一个 Sigma 值, 可定义粒子的 Sigma 值。两目标情况下的 Sigma(δ 表示)的计算公式为:

$$\delta = \frac{f_1^2 - f_2^2}{f_1^2 + f_2^2} \quad (9)$$

其中 f_1, f_2 分别为粒子的两个目标的函数值。

为防止由于目标函数值相差大而造成 δ 的值总是接近于 1 或 -1 的情况, 对目标函数进行归一化处理, 即

$$f_1' = \frac{f_1 - f_1^{\min}}{f_1^{\max} - f_1^{\min}} \quad (10)$$

$$f_2' = \frac{f_2 - f_2^{\min}}{f_2^{\max} - f_2^{\min}} \quad (11)$$

其中 $f_1^{\max}, f_1^{\min}, f_2^{\max}, f_2^{\min}$ 分别为粒子在两个目标上的最大值和最小值。从而得到新的 δ 的计算公式:

$$\delta = \frac{f_1'^2 - f_2'^2}{f_1'^2 + f_2'^2} \quad (12)$$

Sigma 方法求解全局极值的步骤如下:

- 1) 分别求解种群中每个粒子的 δ 值、外部集 ES 中各非支配解集的 δ 值;
- 2) 找出距离种群第 i 个粒子 δ 值最近的外部集中的粒子 j ;
- 3) 将外部集合中第 j 个粒子的解作为种群第 i 个粒子的 $gbest(i)$ 。

在算法的前期采用 Sigma 方法从外部集合中寻找粒子 i 的全局极值点, 这样能够促使算法很快收敛到全局非劣最优面。当算法运行到一定代数时, 作为全局极值的粒子已经在非劣最优面上, 这时需要考虑选择处于 Pareto 前沿中分散区域的个体, 引导粒子群向分散区域进化。因此, 在进化后期用以下策略对全局最优个体进行更新:

- 1) 若 Pareto 中所有个体的拥挤距离都为无穷大, 即仅包括数量较少的边界个体, 则随机选择一个作为 $gbest$ 。
- 2) 若 Pareto 中含有拥挤距离不为无穷大的个体, 则使用轮盘法选择, 即以较大概率选择拥挤距离较大的个体为 $Gbest$ 。计算公式为

$$P(i) = \frac{ES[i].distance}{\sum_{i=1}^{psize} ES[i].distance} \quad (13)$$

其中 $P(i)$ 为外部集合中第 i 个个体被选中的概率,

$ES[i]_{distance}$ 为第 i 个个体的聚集距离, $psize$ 为当前外部集合的粒子个数. 需要注意的是, 个体聚集距离含有无穷大会造成轮盘法选择失效, 因此公式中的 $psize$ 不计边界点.

3.4 变邻域搜索

Mladenovic 和 Hansen^[14]提出了一种变邻域搜索算法, 其在很多问题中取得了很好的应用. Zolobas^[15]将遗传算法与变邻域搜索算法结合, Bassem^[16]将分布评估算法与变邻域搜索算法结合. 变邻域搜索算法的应用大大加强了这些算法的集中搜索能力, 因此对 Pareto 解采用变邻域搜索算法进行有效的邻域搜索.

变邻域搜索算法的执行流程如图 4 所示:

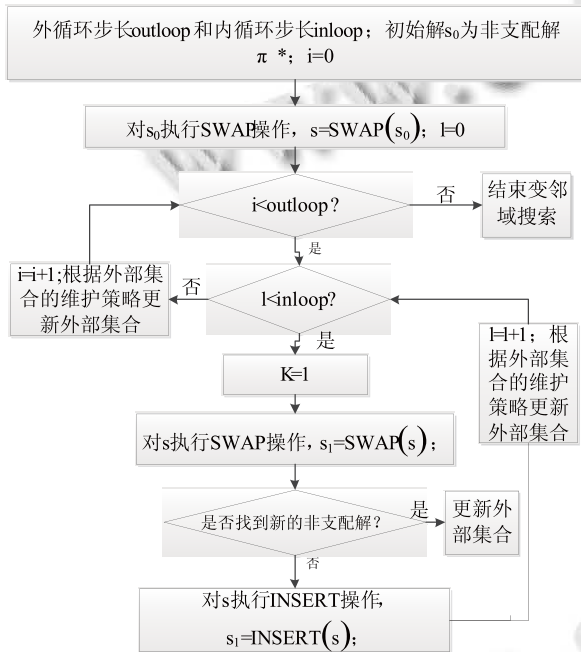


图 4 变邻域搜索算法流程

3.5 算法流程

本文采用一种多目标粒子群优化算法对 PFSP 进行分散搜索, 提出了其 Pareto 解集的构成和维护策略, 并提出了一种两阶段的全局最优解的选择机制, 有效地领导粒子向非支配解前沿收敛. 提出结合变邻域搜索算法, 对多目标粒子群算法求的非支配解进行集中地邻域搜索, 以搜索到更多的 Pareto 解. 混合算法的具体算法流程如下.

1) 初始化算法参数: 进化种群大小 $popsiz$, 外部集合 ES 大小 $psize$, 惯性系数 w 、认知系数 $c1$ 和社会系数 $c2$, 强支配系数 C .

2) 初始化粒子种群:

① 利用 NEH 生成 10%个工件加工序列, 计算调度目标, 并据公式(5)转换为一个粒子的位置矢量;

② 随机产生余下的 90%个粒子的位置矢量, 根据 ROV 规则得出其对应的工件加工序列, 根据加工序列计算各粒子的两个调度目标;

③ 随机初始化种群中所有粒子的速度矢量;

④ 令各粒子的局部最优为当前位置, 并对初始化种群执行快速排序, 将非支配解加入外部集合 ES 中;

⑤ 根据 3.3.3 更新全体极值.

3) 循环步骤 4)–6)直到满足停止条件.

4) 对所有粒子执行下列操作:

① 采用式(6)和(7)更新所有粒子的速度和位置;

② 根据 ROV 规则, 确定各粒子位置矢量所对应的工件加工序列, 并计算各粒子两个调度目标;

③ 根据 3.3.1 和 3.3.2 更新外部集合 ES , 并更新各粒子的个体极值.

5) 对 ES 外部集合中的 Pareto 解执行变邻域搜索算法, 更新外部集合 ES .

6) 根据 3.3.3 更新全体极值.

7) 输出外部集合 ES .

混合粒子群算法的具体算法流程如图 5 所示.

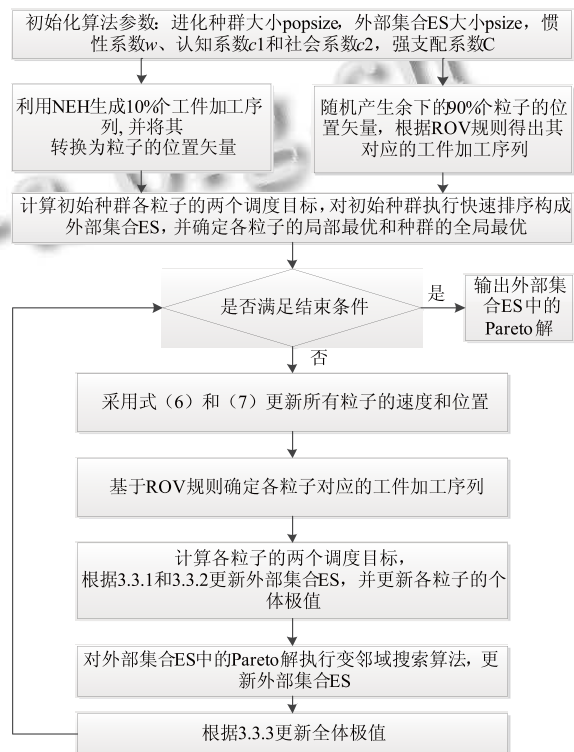


图 5 混合粒子群优化算法具体流程

4 实例仿真与结果分析

为了测试提出的混合粒子群优化算法, 本文数据基于 Taillard^[17]在 1993 年提出的 120 个基准测试问题, 用本文提出的多目标混合离子群优化算法进行求解, 并在同样的硬件条件下用著名的强支配进化算法(SPEA2)进行求解, 将两种算法求得的结果进行比较. 算法运用 Visual C++ 6.0 编程实现, 计算机 CPU 是 Intel Celeron M 520, 主频为 1.6G, 物理内存 512MB, 操作系统 Windows XP. 首先, 对强支配进化算法作简单介绍.

4.1 强支配进化算法

Zitzler 和 Thiele^[18]于 1999 年提出了基于 Pareto 解的强支配进化算法(strong pareto evolutionary algorithm, SPEA), 2001 年针对其存在的不足, 对 SPEA 做了改进, 提出了 SPEA2^[19]. 在 SPEA2 算法中, 种群和精英集中每一个粒子都被赋予一个强度值, 强度值包括了支配关系信息和密度信息. 基于强度值, 每一个粒子的排序值就由支配当前这个粒子的个体的强度值之和来确定. 同时, 算法采用小生境策略确定每一个粒子的密度信息. 最终的适应度值就为粒子的排序值和密度值之和. 最后, 采用竞标赛策略来保证精英集合的规模.

4.2 评判指标

1) 非支配解的数目(number of pareto solution, NPS)

比较不同算法所能找到非支配解的个数, 并可以与完整的 Pareto 前沿进行较.

2) 均匀性指标(spacing metric, SM)

通过计算解集中每个个体与邻居个体的距离变化来评价解集在目标空间的分布情况, 其评价函数定义如下:

$$S = \sqrt{\frac{\sum_{i=1}^N (\bar{d} - d_i)^2}{N-1}} \quad (14)$$

其中, $d_i = \min(\sum_{m=1}^k |f_m^i - f_m^j|)$, $i, j = 1, 2, \dots, N, i \neq j$; N

为解集中个体的数目; \bar{d} 为所有 d_i 的平均值; S 的值越小说明解集分布越均匀.

3) 多样性指标(diversification metric, DM)

这个指标用来测试解集的延展性, 它通过个体与其它个体的最大欧式距离来评价. 其评价函数定义如下:

$$D = \sqrt{\max \sum_{i=1}^N (\|x_i - y_i\|)} \quad (15)$$

其中 $\|x_i - y_i\|$ 为非支配解 x_i 和 y_i 之间的欧式距离.

4) 相对增长比指标(relative percentage increase, RPI)

这个指标用来评判解集中的解相对理想点的目标值相对增长比例, 将算法中以单一目标最优值组成的解为理想点, RPI 则为解集中每一个解相对理想点目标值增长比例的平均值. 设理想解的 *makespan* 值为 $\min(MS)$, TFT 值为 $\min(TFT)$. 第 i 个 Pareto 的 *makespan* 值表示为 $MS(S(i))$, TFT 值表示为 $TFT(S(i))$; 则相对增长比指标的函数定义如下:

$$RPI = \frac{\sum [w_1 * (MS(S(i)) - \min(MS)) + w_2 * (TFT(S(i)) - \min(TFT))]}{N} \quad (16)$$

其中 N 为 Pareto 解集中解的个数.

4.3 实验参数设置

本文算法实验参数设置如下: $c1=c2=2.0$, 惯性系数 w 初始值设为 0.9, $\beta = 0.975$, 最小不能小于 0.4, 粒子的最小位置值 $x_{\min} = 0$, 粒子的最大位置值 $x_{\max} = 4.0$, 粒子的最小速度值 $v_{\min} = -4.0$, 粒子的最大速度值 $v_{\max} = 4.0$, 最大迭代次数设为 500, 种群规模设为 40, 外部集合规模设为 60, 强支配系数 C 为 90 每个实例独立连续运行 10 次.

SPEA 算法实验参数设置如下: 初始化方法如本文算法一样, 采用 NEH 和随机初始化结合, 采用竞标赛选择策略, 交叉算子采用 OX 交叉, 变异方式为互换变异, 交叉概率设为 0.8, 变异概率设为 0.8, 递减系数 0.9, 最小不小于 0.4. 种群规模设为 40, 最大迭代次数为 500.

4.4 实验结果比较与分析

本文选取 TA 测试集问题中的 20 个工件系列的 30 个问题进行了测试和比较实例的数据通过 Taillard 网站 <http://mistic.heig-vd.ch/taillard/> 获得. 本文的混合多目标粒子群算法用 MPSO-VNS 表示, 以 NPS, SM, DM, RPI 四个指标对两种算法进行比较, 实验比较结果如表 2 所示.

由实验比较结果可得, 本文提出的多目标混合粒子群优化算法相比 SPEA2 算法在大多数实例上的实验结果在各指标上都好一些, 能够获得更多的非支配解的数目, 具有良好的分布性和延展性, 相对增长比例指标也有一定程度上的改进, 因而证实了算法的有效性.

两种算法求解 ta021 问题所产生的 Pareto 解集的比较如图 6 所示, 可以看出 MPSO-VNS 获得的 Pareto 解集具有相当好的分布性和收敛性.

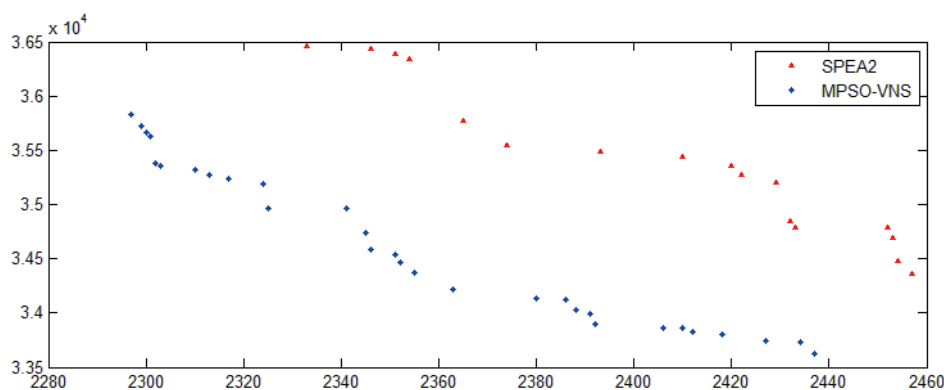


图 6 两种算法求解 ta021 问题的结果比较

表 2 实验比较结果

问题	规模		NPS		SM		DM		RPI	
	n	m	SPEA2	MPSO-VNS	SPEA2	MPSO-VNS	SPEA2	MPSO-VNS	SPEA2	MPSO-VNS
Ta001	20	5	8	4	25.14	33.00	32.25	8.27	0.0221	0.0136
Ta002	20	5	5	8	8.29	1.85	25.64	23.72	0.0147	0.0138
Ta003	20	5	6	15	65.31	60.10	20.41	23.51	0.0400	0.0260
Ta004	20	5	9	16	25.26	11.89	29.46	29.94	0.0239	0.0171
Ta005	20	5	9	20	28.78	13.67	19.78	25.53	0.0376	0.0331
Ta006	20	5	7	25	9.07	3.49	34.56	40.19	0.0274	0.0431
Ta007	20	5	13	12	13.28	6.41	28.33	22.87	0.0253	0.0150
Ta008	20	5	15	18	22.93	5.32	28.07	23.97	0.0298	0.0205
Ta009	20	5	6	18	20.32	1.79	15.30	26.27	0.0168	0.0244
Ta010	20	5	16	15	7.12	5.60	28.74	26.89	0.0286	0.0264
Ta011	20	10	7	14	19.48	9.79	31.79	28.80	0.0242	0.0274
Ta012	20	10	7	25	18.20	11.98	20.53	33.32	0.0190	0.0346
Ta013	20	10	14	13	13.74	12.53	29.98	28.44	0.0257	0.0261
Ta014	20	10	13	21	10.83	10.40	26.03	30.96	0.0262	0.0320
Ta015	20	10	4	13	15.01	5.51	9.19	25.29	0.0123	0.0191
Ta016	20	10	15	31	17.85	10.67	35.64	40.13	0.0403	0.0454
Ta017	20	10	9	16	13.90	8.13	28.72	29.39	0.0258	0.0262
Ta018	20	10	5	15	5.22	15.80	24.59	36.60	0.0115	0.0200
Ta019	20	10	16	21	13.00	5.30	19.30	21.90	0.0220	0.0239
Ta020	20	10	18	23	9.35	11.95	22.58	37.57	0.0243	0.0320
Ta021	20	20	17	30	9.73	3.95	45.83	47.04	0.0303	0.0277
Ta022	20	20	14	26	10.95	22.57	43.88	36.68	0.0214	0.0280
Ta023	20	20	22	40	15.83	10.36	54.04	53.29	0.0363	0.0379
Ta024	20	20	10	19	33.68	11.15	27.56	34.14	0.0135	0.0171
Ta025	20	20	16	25	10.75	15.76	47.48	39.15	0.0310	0.0383
Ta026	20	20	13	21	19.16	5.04	27.98	36.01	0.0202	0.0196
Ta027	20	20	19	22	8.47	6.58	41.11	32.45	0.0196	0.0218
Ta028	20	20	22	49	12.64	6.28	39.99	48.64	0.0340	0.0365
Ta029	20	20	14	33	12.20	31.39	27.13	31.49	0.0239	0.0322
Ta030	20	20	13	22	23.91	10.96	22.08	36.01	0.0180	0.0244

5 结语

本文针对多目标 PFSP 问题提出了一种结合粒子群优化算法和变邻域搜索算法的多目标混合优化算法。算法采用 NEH 启发式算法初始化种群, 大大提高了初始解的质量; 基于 ROV 规则, 运用连续 PSO 算法进行有效的全局搜索; 引入外部集合管理非支配解, 采用强支配关系和聚集距离结合的策略对外部集合进行有效的维护; 采用两阶段全局最优解的选择机制, 合理

引导了种群的收敛; 采用变邻域搜索算法对外部集中的 Pareto 解集进行集中搜索, 加强了算法的集中搜索能力。算法将 PSO 算法的全局搜索能力和变邻域搜索算法的局部搜索能力相结合, 并使分散搜索和集中搜索达到有效的平衡, 大大增加了算法的搜索能力。运用混合算法求解 Taillard 基准问题, 并将测试结果与 SPEA2 算法比较, 本文算法都取得了很好的效果, 验证了该算法的有效性。

参考文献

- 1 Ishibuchi H, Murata T. A multi-objective genetic local search algorithm and its application to flowshop scheduling. *IEEE Trans Syst Man Cybern*,1998,28(2):392-403.
- 2 Ishibuchi H, Yoshida T, Murata T. Balance between genetic search and local search in memetic algorithms for multi-objective permutation flowshop scheduling. *IEEE Trans Evol Comput*,2003,7(2):204-223.
- 3 Jaskiewicz A. Genetic local search for multi-objective combinatorial optimization. *Eur.J.Oper.Res.*,2002,137(1):50-71.
- 4 Loukil T, Teghem J, Tuytens D. Solving multi-objective production scheduling problems using metaheuristics. *Eur. J.Oper. Res.*,2005,161(1):42-61.
- 5 Arroyo JEC, Armentano VA. Genetic local search for multi-objective flowshop scheduling problems. *Eur.J.Oper.Res.*, 2005,167(3):717-738.
- 6 Li BB, Wang L. A hybrid quantum-inspired genetic algorithm for multi-objective flow shop scheduling. *IEEE Trans.Syst, Man,Cybern.*,2007,37(3):576-591.
- 7 Rahim-Vahed AR, Mirghorbani SM. A multi-objective particle swarm for a flow shop scheduling problem. *J.Comb.*

(下转第 110 页)

参考文献

- 1 周琳娜,王东明.数字图像取证技术.北京:北京邮电大学出版社,2008.
- 2 Wang W, Farid H. Exposing digital forgeries in video by detecting double MPEG compression. Proc. of the 8th Workshop on Multimedia and Security. New York. 2006: 37-47.
- 3 Wang W, Farid H. Exposing digital forgeries in video by detecting double quantization. Proc. of the 11th ACM workshop on Multimedia and security. New York, USA 2009: 39-48.
- 4 Wang W, Farid H. Exposing digital forgeries in video by detecting duplication. Proc. of the 9th workshop on Multimedia and Security, New York, NY, USA. 2007: 35-42.
- 5 Lukas J, Fridrich J, Goljan M. Digital camera identification from sensor pattern noise. IEEE Transactions on Information Forensics and Security, 2006, 1(2): 205-214.
- 6 Shih T, Tsai J, Tang N, Chang SM, Lin Y. Video Forgery. Future Generation Information Technology, 2009: 7-11.
- 7 Shih T, Tsai J, Tang N, Chang SM, Lin Y. Video forgery and special effect production. Visual Informatics: Bridging Research and Practice, 2009: 35-37.
- 8 王俊文,刘光杰,张湛,王执铨,戴跃伟.基于模式噪声的数字视频篡改取证.东南大学学报(自然科学版),2008,38(增刊): 13-17.
- 9 秦运龙,孙广玲,张新鹏.利用运动矢量进行视频篡改检测.计算机研究与发展,2009,46(增刊):227-233.
- 10 张静,陈静,苏育挺.基于滤波检测的视频区域篡改检测算法.电子测量技术,2011,34(11):66-69.
- 11 张静,宋怡,苏育挺.基于时空联合匹配的视频篡改检测算法.电子测量技术,2011,34(11):58-61.
- 12 袁秀娟,黄添强,陈智文,吴铁浩,苏立超.基于纹理特征的数字视频篡改检测,计算机系统应用,2012,21(6):91-95.
- 13 秦运龙,孙广玲,张新鹏,刘焕,王朔中,魏为民.一种基于压缩域的视频序列 copy-move 篡改检测方法.中国发明专利,200910198209.2010-05-12.
- 14 黄添强,陈智文,苏立超,郑之,袁秀娟.利用内容连续性的数字视频篡改检测.南京大学学报,2011,47(5):493-503.
- (上接第 118 页)
- Optim., 2007, 13(1): 79-102.
- 8 郑金华.多目标进化算法及其应用.北京:科学出版社,2007.
- 9 王凌.车间调度及其遗传算法.北京:清华大学出版社,2003.
- 10 王凌.微粒群优化与调度算法.北京:清华大学出版社,2008.
- 11 Nawaz M, Enscore E, Ham I. A heuristic algorithm for the m-machine n-job flow shop sequencing problem. Omega, 1983, 11(1): 11-95.
- 12 Agrawal S, Deb K, et al. A fast and elitist multi-objective genetic algorithm, NSGA2, IEEE Transactions on Evolutionary Computation, 2002, 6(2): 182-197.
- 13 Mostaghim S, Teich J. Strategies for finding good local guides in multi-objective particle swarm optimization. Proc of the 2003 IEEE Swarm Intelligence Symposium. Indianapolis: IEEE Service Center, 2003: 26-33.
- 14 Mladenovic N, Hansen P. Variable neighborhood search. Computers and Operations Research, 1997, 24: 1097-1100.
- 15 Zobolas GI, Tarantilis CD, Ioannou G. Minimizing makespan in permutation flow shop scheduling problems using a hybrid metaheuristic algorithm. Computers & Operations Research, 2009, 36: 1249-1267.
- 16 Jarboui B, Eddaly M, Siarry P. An estimation of distribution algorithm for minimizing the total flowtime in permutation flowshop scheduling problems. Computers & Operations Research, 2009, 36: 2638-2646.
- 17 Taillard E. Benchmarks for basic scheduling problems. European Journal of Operational Research, 1993, 64: 278-285.
- 18 Zitler E, Thiele L. Multiobjective evolutionary algorithm: A comparative case study and the strength pareto approach. IEEE transactions on evolutionary computation, 3(4): 257-271.
- 19 Zitler E, Thiele L. SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. Eurogen 2001-Evolutionary methods for design, optimization and control with applications to industrial problems.