

文章编号: 1001-0920(2009)01-0066-05

## 基于偏好信息的多目标微粒群优化算法研究

余进, 何正友, 钱清泉

(西南交通大学 电气工程学院, 成都 610031)

**摘要:** 在实际决策过程中, 决策者可能并不需要完全获悉所有的决策方案, 而是只对一些特定方案产生兴趣, 对此, 提出指定目标间重要关系和给定目标空间参考点情况下的多目标微粒群优化算法. 以格栅作为解的多样性保持策略, 对于给定目标间重要关系的偏好信息, 可以获得特定区域的多个解; 对于给定参考点的偏好信息, 可以同时获得多个特定区域中的多个解, 有利于决策者进行更有效的决策. 通过对典型测试问题的仿真实验, 验证了本算法的正确性和有效性.

**关键词:** 偏好信息; 多目标微粒群优化算法; 优化; Pareto 前沿

中图分类号: O221.6

文献标识码: A

## Study on multiobjective particle swarm optimization algorithm based on preference

YU Jin, HE Zheng-you, QIAN Qing-quan

(School of Electrical Engineering, Southwest Jiaotong University, Chengdu 610031, China. Correspondent: YU Jin, E-mail: jinyu\_yj@yahoo.com.cn)

**Abstract:** During practical making decision, the maker need not know all the solutions to the problem, but is interested in certain solutions. To meet this requirement, multi-objective particle swarm optimization algorithm based on importance relationship among the objectives and reference points in objective space is proposed, which employs grid strategy to keep solutions diversity. More than one solutions located in certain area can be got when importance relationship among the objectives is specified. And more than one solutions located in more than one certain areas can be also obtained when preference based upon reference points is indicated, which is beneficial to the decision maker making efficient and reliable decisions. Simulation results of a series of classical benchmark problems show the correctness and effectiveness of the algorithm.

**Key words:** Preference; MOPSO; Optimization; Pareto front

### 1 引言

现实生活中的决策问题, 大多是多目标的优化问题, 这类问题不存在绝对意义上的最优解, 需要决策者在各目标间进行均衡. 为了更好、更有效地决策, 人们提出了多种多目标优化算法. 其中, 微粒群算法由于其良好的收敛性、简便的计算形式和参数设置, 受到了广泛的关注, 并已成功应用于半导体综合设计、电力调度、作业调度和神经网络训练等领域<sup>[1,2]</sup>. Xiaohui<sup>[3]</sup>, Jonathan<sup>[4]</sup> 和 Coello<sup>[5]</sup> 等也陆续提出了相应的多目标微粒群优化算法 (MOPSO). 此后又相继涌现出多种新的和改进的算法.

MOPSO 可以在一次运行中产生多个“最优”决

策方案, 增加了决策的灵活性, 有利于决策者高效地决策, 但随之带来的是计算复杂度的增加. 随着求解问题目标的增大, 要想获得一定密度的 Pareto 前沿, 计算的代价将按指数级增加. 而现实生活中, 决策总是在有限的方案中做出选择, 只需要得到一部分 Pareto 最优解的子集即可. 在搜索过程中, 通过融合决策者的偏好信息, 引导最优解向特定区域运动, 而迭代中产生的其他绝大多数解则可忽略. 这样, MOPSO 在运行中只需处理少数的解, 从而极大地改善了算法效率, 与一些传统方法相比, 具有很大的优势. 传统的交互式多目标决策方法中, 需要决策者不断地给出一些参考方向、参考点或者其他有助

收稿日期: 2007-11-26; 修回日期: 2008-04-02.

基金项目: 国家自然科学基金项目 (50878188).

作者简介: 余进 (1973—), 男, 陕西南郑人, 博士生, 从事列车运行过程控制优化等研究; 钱清泉 (1936—), 男, 江苏丹阳人, 中国工程院院士, 教授, 博士生导师, 从事轨道交通电气化自动化等研究.

于问题求解的信息<sup>[6]</sup>, 据此获得 Pareto 前沿的解集. 该方法根据提供的信息, 通常只能得到一个解, 而单个的解不能提供所期望解周围其他解的特征. 决策者通过提供相应的信息, 并不仅仅是为了获得单个的解, 而是希望通过这些信息得到他们所期望的最优解附近那些解的特征信息. 只有这样, 决策者才能最终选择出高效的解决方案.

人们借鉴各种进化算法思想, 提出了许多有效的方法. Deb<sup>[7]</sup> 通过将目标规划和非劣排序遗传算法(NSGA)相结合, 提出一种改进的 NSGA 算法. 该方法并不刻意寻求指定多目标优化问题的 Pareto 最优解, 而是以找到满足给定目标解为主要出发点, 无论给定目标点是否位于可行区域, 该方法都能得到相应的解集. Cvetkovic<sup>[8]</sup> 采用权重和方法得到多目标优化问题特定区域的解, 对每一目标赋予权重  $w_i$  和优胜系数  $\alpha_i$ , 并按下式定义向量间的优胜关系:

$$x \succ y \Leftrightarrow \begin{cases} \exists i, f_i(x) < f_i(y) \\ \forall i, f_i(x) \leq f_i(y) \end{cases} \quad (1)$$

然而该方法仅考虑到一个解优于另一解, 却没有反映出这种“优”的程度. Jin<sup>[9]</sup> 将模糊优胜关系转化为相应的权重间隔, 采用动态权重和进化算法, 将多目标优化转化为单目标优化问题, 以此获得所关心区域的解集. Branke<sup>[10]</sup> 提出将用户的“偏好”溶入优胜关系的定义中, 该方法允许用户指定每对目标可以接受的最大“均衡量”, 即一个目标获得单位量程度改进的同时, 允许另一目标出现最大的劣化程度. 研究表明, 该方法在两个目标情况下可以获得较好的结果, 但在目标较多时, 建立每对目标间的优胜关系则比较困难.

以上这些方法各有特点, 都能获得 Pareto 最优解集中反映用户偏好关系的子集, 但不能用于解决同时给定多个不同区域偏好关系的问题; 另外, 由于它们在一次运行中只产生一个解, 不能反映出决策者所关心区域内各决策目标的变化关系. Deb<sup>[11]</sup> 通过对 NSGA2 小生境策略进行改进, 可同时获得能反映多个偏好关系的 Pareto 最优解集, 该方法对 Pareto 前沿形状不敏感, 而且在目标较多的情况下也能获得较好的结果.

本文考虑以下两类偏好信息: 1) 指定目标间的重要关系; 2) 通过给定参考点指定决策者所关心的区域. 将偏好信息融入 Pareto 优胜关系, 可以引导解的搜索过程向着特定区域运动, 对于采用精英保持策略的 MOPSO, 仅需考虑位于特定区域的解, 从而提高了算法的效率.

对于给定目标间重要关系的偏好信息, 在 Pareto 前沿上以锦标赛方法选择一定数量解作为精

英保留, 重要的目标赋予最高优先级; 对于给定参考点的偏好信息, 结合目标规划的思想 and MOPSO 的优点, 可同时指定多个参考点, 无论是否位于可行域, 都可获得相应 Pareto 最优解集; Pareto 前沿无论是凸或凹、连续或非连续, 都可得到满意结果; 在超高维且目标较多的情况下, 也可获得较好的结果.

## 2 多目标微粒群优化算法简介

微粒群优化算法是受鱼、鸟群觅食等自然现象的启发, 由 Kennedy 和 Eberhart 提出的<sup>[12]</sup>. 对于每个微粒, 由其当前位置向量  $x_n$  和运行速度  $v_n$  唯一表示, 通过个体与群体之间相互作用, 维持一个个体最佳位置和全局最佳位置, 并以此引导所有个体向最佳位置运动.

$$v_{nk}^{t+1} = v_{nk}^t + c_1 * \text{rand}() * (\text{pLocal}_n^t - x_{nk}^t) + c_2 * \text{rand}() * (\text{pBest}^t - x_{nk}^t), \quad (2)$$

$$x_n^{t+1} = x_n^t + v_n^t. \quad (3)$$

其中:  $v_{nk}^t$  为第  $t$  代微粒  $n$  在  $k$  维的速度;  $x_{nk}^t$  为第  $t$  代微粒  $n$  在  $k$  维的位置;  $w$  为惯性权重, 取 0.4;  $c_1$  和  $c_2$  分别为个体意识学习因子和群意识学习因子;  $\text{pLocal}_n^t$  为第  $t$  代第  $n$  个微粒最佳位;  $\text{pBest}^t$  为第  $t$  代全局最佳位置. 通过个体与群体间的信息交换, 个体不断地“跟踪”全局最优解, 直到迭代结束. 因此, 全局最佳位置  $\text{pBest}$  的选择至关重要.

采用外部种群档案的 MOPSO 算法结构如下所示:

### Begin

Initialize swarm

Initialize the external archive

Iteration = 0

**While** (Iteration < max\_Iteration)

**For** each particle

Select gBest

Update Position

Mutation

Evaluation

Update pBest

**End For**

Update the external archive

Iteration ++

**End While**

Report results in the external archive

**End**

保持解的多样性是衡量多目标优化算法性能的重要指标, 采用格栅方法可确保解的多样性, 同时在一定条件下可使解集以概率 1 收敛到优化问题的真

实最优解<sup>[13]</sup>. 本文采用了格栅方法保持解的多样性, 算法最终获得的解是优化问题最优解集的子集, 因此可以保证解集的多样性和概率收敛特性.

### 3 指定目标间重要关系偏好信息的 MOPSO 算法

多目标优化问题的最优解通常有多个, 甚至是无穷多个可行解, 这些解均匀分布于目标空间, 给决策带来了困难. 很多情况下, 决策者可能希望“某个目标比其他更重要些”或者“某些目标同等重要”, 这样便可在较小的范围内分析各个目标对决策效果的影响, 有利于更好地决策. 此处的目标间重要关系可由决策者事先指定. 对于此类决策问题, 可采用以下策略:

1) 若偏好信息为“某目标更重要”, 则在 Pareto 前沿上, 以该目标进行从小到大排序, 排在最前面的赋予“1”, 其后的赋予“2”, 以此类推, 序号越小的解被选择的可能性越大.

2) 若偏好信息为“某些目标同等重要”, 则对这些目标都进行从大到小排序, 排在最前面的赋予“1”, 其后的赋予“2”, 以此类推, 使得 Pareto 前沿上每一点对应的各目标值在相应列中都有唯一的序号. 将每一点各目标值对应的序列号求和, 该和值越大的解被选中的可能性越大. 这样可以保证选择的解的各目标可以获得较好的“均衡”.

3) 运用锦标赛选择方法, 在按照以上策略处理后的集合中选择非劣解作为精英保留.

4) 当外部群体档案已满时, 启动剪枝操作, 密度较大的区域中的部分解点将被剔除掉.

### 4 给定参考点偏好信息的 MOPSO 算法

文献[11]在传统目标规划方法基础上, 提出了基于参考点的多目标遗传算法, 采用了类似于  $\epsilon$ -dominance 方法保持解的多样性; Coello 在文献[5]中提出了一种性能优良的多目标微粒群优化算法, 该算法在一次运行中可得到 Pareto 前沿上所有解. 本文对该 MOPSO<sup>[5]</sup> 进行了一系列改进, 主要体现在以下几方面:

1) 对于获得的每个非劣解, 按下式<sup>[15]</sup> 计算其到每个参考点的归一化欧氏距:

$$d_{jk} = \sqrt{\sum_{i=1}^m \left( \frac{f_{ji}(x) - f_{ki}^0}{f_{ji}^{\max} - f_{ji}^{\min}} \right)^2}. \quad (4)$$

将非劣解按欧氏距降序排列, 排在最前面的赋予等级“1”, 以此类推. 等级越小的解越靠近参考点. 式(4)中:  $d_{jk}$  为第  $j$  个非劣解距第  $k$  个参考点的距离;  $m$  为问题的目标数;  $i$  为目标权重;  $f_{ji}(x)$  为第  $j$  个非劣解目标  $i$  的值;  $f_{ki}^0$  为第  $k$  个参考点目标  $i$  的值;

$f_{ji}^{\max}$  和  $f_{ji}^{\min}$  分别为第  $j$  个非劣解目标  $i$  的最大和最小函数值.

2) 距离所有参考点最近的非劣解赋予距离参数“1”, 距离略远的解赋予“2”, 以此类推.

3) 采用锦标赛方式, 在按照以上策略处理后的集合中选择非劣解存于外部群体档案中, 距离参数小的解被优先选中, 锦标赛规模为 2.

4) 当外部群体档案已满时, 启动剪枝操作, 距离所有参考点都较远的点将被剔除掉, 直到满足档案容量.

采用上述策略后, 本算法可同时获得多目标问题多个参考点的多个解. 当各目标的权重发生变化时, 可以很清楚地了解到各个解集沿 Pareto 前沿运动变化情况, 这对于决策者进行更有效、更可靠的决策是大有裨益的.

## 5 仿真结果分析

为验证本文算法的有效性和可靠性, 针对一组典型的多目标 Benchmark 问题进行仿真研究. 仿真过程中参数设置如下: 对于指定目标间重要关系的偏好信息, 变量维数  $m = 14$ , 变量  $x_i \in [0, 1]$ , 粒子群规模 100, 外部种群档案容量 100, 迭代次数 200; 给定参考点时, 各项参数按具体问题设置.

### 5.1 双目标的 ZDT1 问题<sup>[11]</sup>

$$\begin{aligned} \min \{ & f_1(x), f_2(x) \}, \\ f_1(x) &= x_1, \\ f_2(x) &= g(x)h(f_1, g), \\ g(x) &= 1 + 9 \sum_{i=2}^m x_i / (m - 1), \\ h(f_1, g) &= 1 - \sqrt{f_1/g}. \end{aligned}$$

如图 1 所示, 当指定目标间重要关系时, 优化问题的 Pareto 最优解向特定区域运动, 决策者希望“两目标同等重要”时, Pareto 最优解聚集在两目标的中间区域, 见图 1 中圆圈所示部分; 希望“目标 1 比目标 2 重要”时, Pareto 最优解中目标 1 都具有较小的值, 见图 1 中靠近左边倒三角所示部分; 当“目标 2 更重要”时, Pareto 最优解都聚集在目标 2 函数

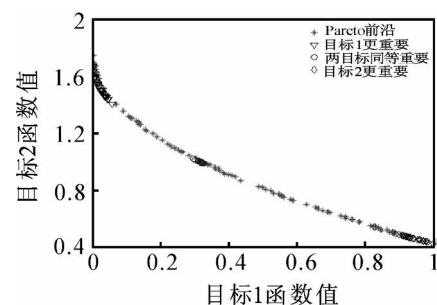


图1 指定目标间重要关系仿真结果

值较小的区域,见图 1 中靠近右边钻石形状部分.相对于整个 Pareto 前沿,只有较少的决策者感兴趣的解呈现给决策者,使决策者可以更好地评价各目标对系统综合性能的影响,有利于高效的决策.

当给定参考点时,优化问题变量的维数  $m = 30$ ,  $x_i \in [0, 1]$ ,粒子群规模为 100,外部种群档案容量为 100,迭代次数为 100,其余各参数按算法要求进行选择.该测试问题具有凸的、连续的 Pareto 前沿,如图 2 所示,其中星型线为该多目标优化问题的 Pareto 前沿.

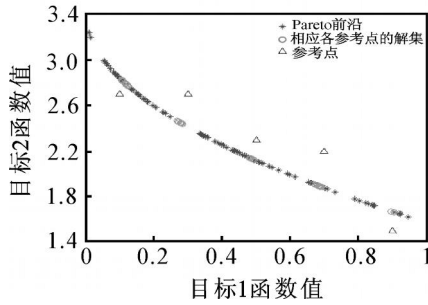


图 2 两目标给定多参考点仿真结果

本例选择 5 个参考点,其中 2 个位于可行域,3 个处于不可行域,如图 2 所示.本文介绍的算法都可找到相应参考点的 Pareto 最优解集.此处各目标权重都相等,取(0.5,0.5)时的仿真结果.为研究目标权重发生变化对结果集的影响,设定参考点为(0.5, 2.4),权重分别为(0.7,0.3), (0.5,0.5), (0.3, 0.7),算法中各参数设置不变,结果如图 3 所示.

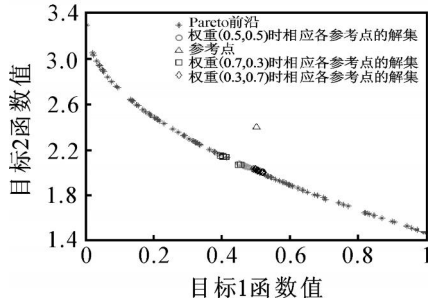


图 3 各目标权重发生变化时仿真结果

对于同一参考点,当反映偏好关系的各目标权重发生变化时,Pareto 最优解集会向着相应的区域运动.权重为(0.7,0.3)时,表示希望获得的解更有利于目标 1.由图 3 可以看出,Pareto 结果集沿 Pareto 前沿移动到使得目标 1 取较小函数值的区域;同理,当权重为(0.3,0.7)时,Pareto 结果集移动到使得目标 2 取较小函数值的区域.这一结果说明本算法能够正确反映决策人的偏好关系,而且一次运行可同时得到参考点附近的多个 Pareto 最优解.

### 5.2 双目标的 ZDT3 问题<sup>[11]</sup>

$$\begin{aligned} \min \{ & f_1(x), f_2(x) \}, \\ f_1(x) = & x_1, \\ f_2(x) = & g(x) h(f_1, g), \\ g(x) = & 1 + 9 \sum_{i=2}^m x_i / (m - 1), \\ h(f_1, g) = & 1 - \sqrt{f_1/g} - (f_1/g) \sin(10 \pi f_1). \end{aligned}$$

本测试问题具有不连续的 Pareto 前沿,给定目标间重要关系时,可得到所关心区域内的 Pareto 最优解集的子集.如图 4 所示,靠近左边的倒三角部分、中间圆圈部分以及靠近右边的钻石部分,分别对应偏好信息“目标 1 更重要”、“两目标同等重要”和“目标 2 更重要”的 Pareto 最优解,这些解都位于优化问题的 Pareto 前沿.

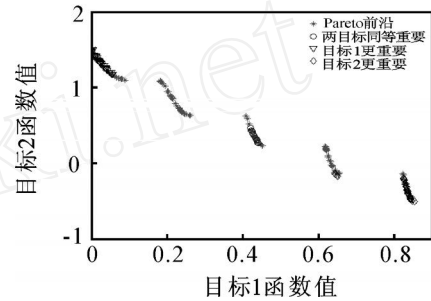


图 4 两目标指定目标间重要关系仿真结果

当指定不同区域参考点时,相应参数设置为:优化问题变量的维数  $m = 30$ ,  $x_i \in [0, 1]$ ,粒子群规模、外部种群档案和迭代次数均设为 100,各目标权重取(0.5,0.5).选取两参考点(0.1,2.7)和(0.5, 2.4),结果见图 5.对于参考点(0.1,2.7),相邻两段 Pareto 前沿上距离该点最近的 Pareto 集都可以得到.

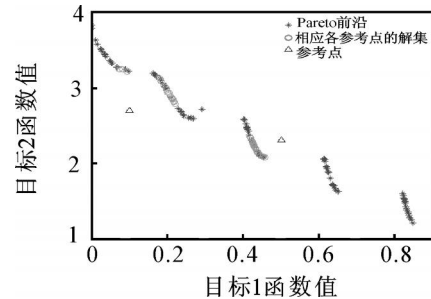


图 5 Pareto 前沿不连续两目标多参考点仿真结果

通过以上两个测试问题的仿真研究可以发现,本文算法对 Pareto 前沿不敏感,不论是凸、凹、连续还是不连续,都可在一次运行过程中获得多个特定区域的 Pareto 最优解.

### 5.3 3 目标的 DTLZ2 问题<sup>[11]</sup>

$$\min \{ f_1(x), f_2(x), f_3(x) \},$$

$$\begin{aligned}
 f_1(x) &= (1 + g(x)) \cos(x_1/2) \cos(x_2/2), \\
 f_2(x) &= (1 + g(x)) \cos(x_1/2) \sin(x_2/2), \\
 f_3(x) &= (1 + g(x)) \sin(x_1/2), \\
 g(x) &= \sum_{i=3}^m (x_i - 0.5)^2.
 \end{aligned}$$

其中:  $m = 12$  为优化问题变量的维数;  $x_i \in [0, 1]$ .

本例选择粒子群规模 300, 外部种群档案容量 300, 迭代次数 100, 其余参数按要求设置. 选择两参考点 (0.3, 0.3, 0.5) 和 (0.8, 0.6, 0.6), 计算所得部分最优解对应的目标值, 如表 1 所示.

表 1 3 目标优化问题的优化结果

	$f_1$	$f_2$	$f_3$	$f_i$
1	0.31876	0.27324	0.9495	1.077
2	0.51768	0.43501	0.79519	1.089
...	...	...	...	...
14	0.60767	0.57312	0.61492	1.076

因为该测试问题的 Pareto 最优解满足  $\sum_{i=1}^3 f_i^2 = 1$ , 为验证所得解确实是 Pareto 最优解, 对所得到的 14 个解计算其  $\sum_{i=1}^3 f_i^2$ . 结果显示, 所有解的  $\sum_{i=1}^3 f_i^2$  都在 (1.076, 1.089) 内, 非常接近 Pareto 前沿, 因此, 这些解都逼近该测试问题的 Pareto 最优解.

#### 5.4 5 目标的 DTLZ2 问题<sup>[11]</sup>

5 目标的 DTLZ2 测试问题, 可由 3 目标 DTLZ2 进行扩展得到, 变量维数  $m = 14$ ,  $x_i \in [0, 1]$ . 选取两参考点 (0.5, 0.5, 0.5, 0.5, 0.5) 和 (0.2, 0.2, 0.2, 0.2, 0.8), 仿真结果见表 2.

表 2 5 目标优化问题的优化结果

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_i$
1	0.64	0.44	0.51	0.62	0.40	1.40
2	0.47	0.48	0.40	0.43	0.62	1.18
...	...	...	...	...	...	...
15	0.23	0.35	0.33	0.08	0.86	1.04

同样, 需要验证这些解是否是本优化问题的 Pareto 最优解. 因该测试问题的 Pareto 最优解满足  $\sum_{i=1}^5 f_i^2 = 1$ , 故判断所得到的解是否满足该式即可.

对获得的 15 个解计算其  $\sum_{i=1}^5 f_i^2$  值, 见表 2. 结果表明, 所有解的  $\sum_{i=1}^5 f_i^2$  都在 (1.04, 1.50) 内, 接近优化问题的 Pareto 前沿, 因此这些解都逼近该优化问题

的 Pareto 最优解, 而且, 获得的两结果集基本上都位于两参考点附近.

## 6 结 论

通常, 决策者在决策过程中, 并不需要获得问题的全部解, 而只需要了解某些区域解的特性. 对此, 本文提出了一种基于偏好信息的多目标微粒群优化算法. 该方法可同时得到多个不同区域的多组 Pareto 最优解集, 能方便地了解到特定区域内解集的运动变化规律, 有助于决策者更好地进行决策. 本文方法对 Pareto 前沿不敏感, 对于凸、凹、不连续等问题均可得到满意的结果.

## 参考文献(References)

- [1] 夏蔚军, 吴智铭. 基于混合微粒群优化的多目标柔性 Job-shop 调度[J]. 控制与决策, 2005, 20(2): 137-141. (Xia W J, Wu Z M. Hybrid particle swarm optimization approach for multi-objective flexible job-shop scheduling problems[J]. Control and Decision, 2005, 20(2): 137-141.)
- [2] Nanbo Jin, Yahya Rahmat-Samii. Advances in particle swarm optimization for antenna designs: Real-number, binary, single-objective and multiobjective implementations [J]. IEEE Trans on Antennas and Propagation, 2007, 55(3): 556-567.
- [3] Xiaohui Hu, Russell Eberhart. Multiobjective optimization using dynamic neighborhood particles swarm optimization [C]. Proc of Congress on Evolutionary Computation. Hawaii, 2002: 1677-1681.
- [4] Jonathan E Fieldsend, Sameer Singh. A multi-objective algorithm based upon particle swarm optimization: An efficient data structure and turbulence [C]. Proc of the 2002 UK Workshop on Computational Intelligence. Manchester, 2002: 37-44.
- [5] Carlos A Coello Coello, Maximino Salazar Lechuga. MOPSO: A proposal for multiple objective particle swarm optimization [C]. Proc of Congress on Evolutionary Computation. Hawaii, 2002: 1051-1056.
- [6] Miettinen K. Nonlinear multiobjective optimization [M]. Boston: Kluwer, 1999.
- [7] Kalyanmoy Deb. Solving goal programming problems using multi-objective genetic algorithms [C]. Proc of Congress on Evolutionary Computation. Washington, 1999: 77-83.
- [8] Cvetkovic D, Parmee I. Evolutionary design and multi-objective optimization [C]. Proc of the 6th European Congress on Intelligent Techniques and Soft Computing. Lille, 1998: 397-401.

(下转第 75 页)

- Economics, 1991, 22(3): 195-202.
- [4] Kalagnanam J R, Dawande M, Trumbo M, et al. Matching orders with surplus inventory in a steel mill [J]. Operations research, 2000, 48(4): 505-516.
- [5] Kalagnanam J R, Lee H S, Trumbo M E. Fast inventory matching algorithm for the process industry [P]. US Patent 6044361, 2000.
- [6] 聂义勇, 申志勇, 王宏, 等. 考虑库存余材利用的杆材下料方案[J]. 小型微型计算机系统, 2001, 7(22): 830-832.  
(Nie Y Y, Shen Z Y, Wang H, et al. ID cutting stock scheme with utilizing surplus in storage[J]. Mini-micro Systems, 2001, 7(22): 830-832.)
- [7] 胡琨元, 朱云龙, 汪定伟. 自适应 PBIL 算法求解合同优化匹配问题[J]. 系统工程, 2004, 12(22): 87-91.  
(Hu K Y, Zhu Y L, Wang D W. Adaptive PBIL algorithms for order optimal matching problems [J]. Systems Engineering, 2004, 12(22): 87-91.)
- [8] 胡琨元, 高政威, 汪定伟. 钢铁企业合同匹配多目标优化模型与算法[J]. 东北大学学报, 2004, 6(25): 527-530.  
(Hu K Y, Gao Z W, Wang D W. Optimal multi-objective model and algorithm for order matching problems in iron & steel plants[J]. J of Northeastern University, 2004, 6(25): 527-530.)
- [9] 胡琨元, 常春光, 郑秉林, 等. 钢铁企业中库存匹配与生产计划集成优化模型与算法[J]. 信息与控制, 2004, 33(2): 177-181.  
(Hu K Y, Chang C G, Zheng B L, et al. The model and algorithm for joint optimization of inventory matching and production planning in steel plant [J]. Information and Control, 2004, 33(2): 177-181.)
- [10] Shixin Liu, Jiafu Tang, Jianhai Song. Order-planning model and algorithm for manufacturing steel sheets[J]. Int J of Production Economics, 2006, 100(1): 30-43.
- [11] 黄可为, 卢克斌, 汪定伟. 炼钢组炉问题优化模型及其动态规划算法[J]. 东北大学学报, 2006, 27(2): 138-141.  
(Huang K W, Lu K B, Wang D W. Dynamic programming algorithm and optimization model of charge design for steel-making [J]. J of Northeastern University, 2006, 27(2): 138-141.)
- [12] Hwai-En Tseng, Wen-Pai Wang, Hsur Yi Shih. Using memetic algorithms with guided local search to solve assembly sequence planning[J]. Expert Systems with Applications, 2007, 33(2): 451-467.
- [13] Wang D W. Earliness/tardiness production planning approaches for manufacturing systems[J]. Computers & Industry Engineering, 1995, 34(3): 425-436.
- [14] 玄光南, 程润伟. 遗传算法与工程设计[M]. 北京: 科学出版社, 2000.  
(Xuan G N, Cheng R W. Genetic algorithm and engineering design[M]. Beijing: Science Press, 2000.)

## (上接第 70 页)

- [9] Jin Y, Sendhoff B. Fuzzy preference incorporation into evolutionary multi-objective optimization [C]. Proc of the 4th Asia-Pacific Conf on Simulated Evolution and Learning. Melbourne, 2002: 26-30.
- [10] Branke J, Kau ler T, Schmeck H. Guidance in evolutionary multi-objective optimization[J]. Advances in Engineering Software, 2001, 32(2): 499-507.
- [11] Kalyanmoy Deb, Sundar J, Udaya Byaskara Rao N. Reference point based multi-objective optimization using evolutionary algorithms [J]. Int J of Computational Intelligence Research, 2006, 2(3): 273-286.
- [12] James Kennedy, Russell Eberhart. Particle swarm optimization[C]. Proc of the 4th IEEE Int Conf on Neural Networks. Piscataway, 1995: 1942-1948.
- [13] Joshua Knowles, David Corne. Properties of an adaptive archiving algorithm for storing nondominated vectors[J]. IEEE Trans on Evolutionary Computation, 2003, 7(2): 100-116.
- [14] Carlos A Coello Coello, Gregorio Toscano Pulido, Maximino Salazar Lechuga. Handling multiple objectives with particle swarm optimization[J]. IEEE Trans on Evolutionary Computation, 2004, 8(3): 256-279.
- [15] 徐玖平, 李军. 多目标决策的理论与方法[M]. 北京: 清华大学出版社, 2005.  
(Xiu J P, Li J. Multiple objective decision theory and methods [M]. Beijing: Tsinghua University Press, 2005.)