

文章编号: 1001-0920(2005)04-0439-05

TSK 模糊模型的协同进化学习方法

刘建成, 蒋新华, 吴今培

(中南大学 信息科学与工程学院, 湖南 长沙 410075)

摘要: 针对 TSK 模糊模型的学习是多约束和多目标优化问题, 提出将 TSK 模糊模型分解为两类不同的种群, 协作共同进化的模型学习方法。论述了所涉及的相关问题, 包括各种群的编码及其不同的进化计算, 各种群个体的合作及其适应值评估策略, 模型的后件参数估计方法。该方法要求先验知识少, 收敛速度快, 能形成简洁的模糊模型, 最后以函数近似为例说明了该方法的有效性。

关键词: TSK 模糊模型; 模糊规则; 协同进化; 进化计算

中图分类号: TP273.4 **文献标识码:** A

Method of learning TSK fuzzy model by cooperative coevolution

L I U J ian-cheng, J I A N G X in-hua, W U J in-pe i

(School of Information Science and Engineering, Central South University, Changsha 410075, China)

Correspondent: L I U J ian-cheng, E-mail: lj c@m i a l c s u . e d u . c n

Abstract: To the problem of multi-constraint and multi-target optimization in learning fuzzy model, TSK fuzzy model is decomposed of twain different species. The method of learning the model by cooperative coevolution is proposed. Some problems related to each species coding and different evolution computing means, individual cooperation and fitness evaluation strategy, consequent parameters estimation, are discussed. The characteristic of the method requests a little of previous information about objects, and is able to obtain compact fuzzy model. An example of function approximation shows the validity of the method.

Key words: TSK fuzzy model; fuzzy rule; cooperative coevolution; evolution computation

1 引言

TSK 模糊模型的一般形式^[1,2]为

$$R_i: \text{If } x_1 \text{ is } A_{i1} \text{ and } \dots x_j \text{ is } A_{ij} \text{ and } \dots \\ x_n \text{ is } A_{in} \text{ Then } y_i = f_i(X). \quad (1)$$

其中: $X = [x_1, \dots, x_j, \dots, x_n]$ 为前件变量; A_{ij} 为规则 i 中 x_j 的模糊集; $f_i(X) = a_0^i + a_1^i x_1 + \dots + a_n^i x_n$ 为规则的后件; $1 \leq i \leq R$, R 为模型中的规则数。当输入为 $X = [x_1, x_2, \dots, x_n]$ 时, 前件 T 范式采用乘积方式, 去模糊化采用重心法, 模型输出结果为

$$y = \frac{\sum_{i=1}^R \sum_{j=1}^n A_{ij}(x_j) y_i}{\sum_{i=1}^R \sum_{j=1}^n A_{ij}(x_j)}. \quad (2)$$

TSK 模型具有较好的可解释性和精确性, 在众多领域得到广泛的应用。模糊模型的学习过程是一

个或多个约束和多目标优化问题, 一些学者提出了多种解决方法^[1~3]。进化算法在 TSK 模型学习中也得到广泛的应用^[4~6], 其特点是基于单个种群的进化或不同种群分阶段的串/并行独立进化。将协同进化方法应用于 TSK 模糊模型学习还不多见。Husbands 等提出了协同进化方法^[7]。Pena 等应用协同进化方法解决 Mamdani 模糊模型的学习问题^[8]。Potter 等发展并提出其一般协同进化模型^[9], 即不同种群之间相互适应、共同进化, 协同合作产生问题完整解。该方法适合于多目标函数的优化和复杂系统的分解描述及动态搜索, 能有效避免单种群进化计算表现出的未成熟收敛, 对对象的规模比较敏感, 收敛速度慢等缺陷。应用协同进化求解, 除了一般进

收稿日期: 2004-06-10; 修回日期: 2004-09-24

基金项目: 湖南省自然科学基金项目(04JJY6036); 湖南省科技攻关项目(02JZY2006)。

作者简介: 刘建成(1964—), 男, 湖南益阳人, 副教授, 博士, 从事计算智能、模糊系统的研究; 蒋新华(1956—), 男, 湖南长沙人, 教授, 博士生导师, 从事智能控制、故障诊断等研究。

化算法遇到的问题外, 还须解决其它问题, 如求解对象的分解, 子种群间的数据约束关系和相互合作关系, 子种群中个体适应值的计算方法等

本文针对 TSK 模型的学习问题(包括: 输入变量的选择、分区及其在此分区上的隶属函数参数、模型规则数、后件参数), 提出一种基于协同进化的 TSK 模型(COOPCO-TSKFM). 将模糊模型分解为两类种群: 一类种群描述模糊模型及其规则的结构; 另一类种群描述变量的模糊分区和隶属函数参数. 根据各种群的特点, 采用灵活的二维和分层树状结构编码方式以及遗传算法和遗传规划相结合的进化方法. 本文方法具有要求先验知识少、收敛速度快、适应能力强、模型简洁等特点

2 TSK 模糊模型的协同进化

2.1 模型的种群分解与编码原则

对于式(1)所示的模糊模型学习问题, 可分解为多个子问题或种群, 各种群的典型个体组合成模糊模型. 模糊模型的分解需要考虑以下几方面问题:

- 1) 各种群是模型的有效部分, 对模型的品质有足够的影响, 并且避免冲突;
- 2) 便于各种群的合作及典型个体的产生;
- 3) 各种群合作的适应值计算过程应相对简单. 从总体上说, 在能正确解决模糊模型学习问题的情况下, 分解的种群数越少, 计算的复杂性、搜索空间和时间降低. 本文将模糊模型分解为以下两类种群:

(1) 种群 Pop₁ 描述模糊模型及其规则的一般性结构

包括: 模型的各项规则、规则所涉及的变量及其有关模糊分区. 该种群采用遗传算法(GA). 为描述模型以及变量模糊分区的动态特征, 本文对该种群采用二维编码, 如下式所示:

$$\begin{bmatrix} I_1^1 & I_1^2 & \dots & I_1^j & \dots & I_1^n & L_1 \\ I_2^1 & I_2^2 & \dots & I_2^j & \dots & I_2^n & L_2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ I_{R_M}^1 & I_{R_M}^2 & \dots & I_{R_M}^j & \dots & I_{R_M}^n & L_{R_M} \end{bmatrix} \quad (3)$$

其中: 每行代表一条规则; R_M 表示该模糊模型最大可能的规则数; n 表示规则所涉及的变量数; $I_i^j \in \{0, 1, \dots, N_{B_j}\}$, 表示规则 i 在变量 x_j 论域内的分区码, N_{B_j} 为 x_j 的最大可能分区数, $I_i^j = 0$ 表示变量 x_j 在规则 i 中不起作用; 开关参数 $L_i = 1$ 表示在模型中该规则 i 存在, $L_i = 0$ 表示在模型中该规则 i 不存在

(2) 种群 Pop₂ 描述规则的变量分区及隶属函数. 本文的隶属函数采用广义梯形函数 $T_{M_F}(x; a,$

$b, c, d): U \in [0, 1], x \in U$. 其中 a, b, c, d 是需要学习的参数, 可产生不同的模糊集, 使模型具有较强的适应能力, $b = c$ 为三角形隶属函数, $b = a$ 为降半梯形函数, $c = d$ 为升半梯形函数. 各变量的分区结构以及定义的隶属函数是不定的, 因此采用分层树状结构的遗传规划(GP), 每个个体描述模糊模型各变量的分区结构及隶属函数. 种群个体树状结构如图 1 所示

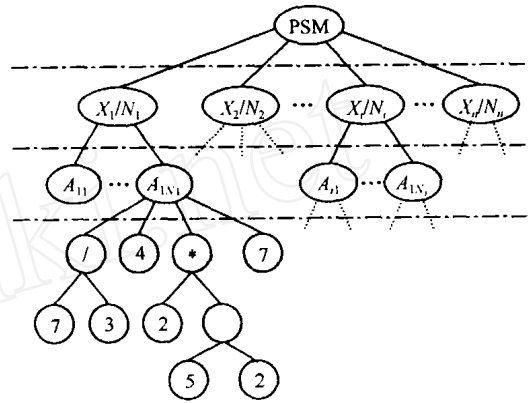


图 1 种群 Pop₂ 个体树状结构

种群个体由 4 层组成: 第 1 层是 PSM 根结点, 用于连接各树枝, 不参与进化操作; 第 2 层由各变量及其分区数结点组成, 称为 V_T 结点; 第 3 层表示变量的每个分区及其隶属函数, 称为 M_F 结点; 第 4 层产生各分区隶属函数参数 (a, b, c, d) , 由算术算子 A_0 结点和变量论域参数 V_U 结点(终结点)构成各隶属函数参数子树 $x_j/N_j (N_j \in N_{B_j})$ 表示变量 x_j 当前有 N_j 个分区; $A_{j1}(1 \sim N_j)$ 表示变量 x_j 第 l 分区的隶属函数, A_{1N_1} 为变量 x_1 第 N_1 分区的隶属函数, 即 $T_{M_F}(x_1; 2, 3, 3, 4, 6, 7)$.

上述两类种群既相互独立各自进化, 又相互依赖, 来自各种群任意个体的组合都是模糊模型的一个可能解. 下面给出几个定义, 用于对上述 Pop₁ 编码和 Pop₂ 各结点及其关系在进化过程中的约束

定义 1 I_i^j 编码的有效性是其子基因串长度 l 满足 $[(l - 1)_2]_{\max} < N_{B_j} < [(l + 1)_2]_{\max}$, 其中 $[(l)_2]_{\max}$ 表示长度为 l 的二进制最大值

定义 2 结点向量 $N_{F_{N_M}}$ 是各结点的可能值域; 其中 N_M 为结点类型

本文约定上述 4 类结点的结点向量为

$$N_{V_{A_0}} = \{+, -, *, / \},$$

$$N_{V_{V_U}} = \{x_j/pa \mid pa \in \{1, 2, 3, U_{\min}, U_{\mid}, U_{\max}\}\}.$$

其中: $U_{\min}/U_{\mid}/U_{\max}$ 分别为对应于变量 x_j 论域的较

小 / 中间 / 较大值;

$$N_{V_{M_F}} = \{x_j / A_{j_l} \mid l \in \{1, 2, \dots, N_{j_l}\}\},$$

$$N_{V_{V_T}} = \{x_j / N_{j_l} \mid N_{j_l} \in \{0, 1, \dots, N_{B_j}\}\}.$$

当 $V_T = 0$ 时, 说明该变量 x_j 不在规则中出现

定义 3 结点协议 $N_P(N_M)$ 是各结点输入 / 输出约束关系, 描述为

$$N_P(N_M) = \{I_n, O_n, N_V, I_x, O_y\}.$$

其中: I_n 为结点 N_M 的输入个数, O_n 为输出个数, N_V 为该结点向量, I_x 为输入结点类型, O_y 为输出结点类型

2.2 种群个体的适应值评估策略

在协同进化中, 个体适应值的评估依赖于该个体与其他种群个体合作产生可能解的品质, 即种群 Pop_1 个体适应值是它与种群 Pop_2 的代表合作产生的模糊模型性能 (或适应值); 对于种群 Pop_2 个体适应值的评估也是如此. 种群代表可以是种群全体或典型个体, 本文采用典型个体作为代表. 各种群代表选取策略由种群两类个体组成: 一类是选择 N_{p_f} 个适应值最好的个体; 另一类是随机选择 N_{p_r} 个个体. 每个种群都有代表 (或称代表团, 成员个数 $N_p = N_{p_f} + N_{p_r}$) 参与其他种群个体适应值的评估. 各种群代表的更新方式采用该种群进化完成对每个适应值评估后产生新的代表, 但其合作种群的当前代适应值尚未评估, 因此合作种群的代表还是上一代的代表. 任意个体 D_j^i 的适应值评估方法为

$$\text{fit} - D_j^i = \text{MAX}_{\substack{I \\ \text{Rep}}} (\text{fit} - f_m(D_j^i, D_{\text{Rep}}^m)). \quad (4)$$

其中: $\text{fit} - f_m(D_j^i, D_{\text{Rep}}^m)$ 为个体 D_j^i 与其他种群的代表 D_{Rep}^m 合作产生的模糊模型适应值; $i, m \in \{1, 2\}, i \neq m$.

在协同进化框架中, 各种群是同步独立进化而不考虑其它种群的影响. 实际上, 种群之间存在一定的数据约束, 如 Pop_1 个体的子基因串 I_i^j 指向 Pop_2 个体中 V_T 结点下的 M_F 结点, 可能使某个 I_i^j 大于 V_T , 即 Pop_1 某个体的规则 i 中, 变量 x_j 的较大分区码可能没有对应的分区及其隶属函数. 在合作中遇此情况, 本文采用的措施是将 I_i^j 改为 V_T , 即指向 V_T 结点下的最大 M_F 结点, 为保持种群的差异性, 在 Pop_1 中对该个体的 I_i^j 并不修改. 这是一种简单的处理方法, 不一定是最合理的, 但从算例结果看是有效的.

2.3 种群的遗传操作

本文提出的 COOPCO-TSKFM 采用两类种群和两种不同的编码方式, 但都依据遗传算法的思想

设计遗传操作 (选择、杂交和变异), 两种种群选择操作都采用精英保留 (比例为 E_r) 与轮盘赌选择相结合的方法. 下面分析其他遗传操作在各种种群进化过程中的一些相关问题.

2.3.1 种群 Pop_1 的遗传操作

Pop_1 中个体是否交叉由交叉概率 p_{c1} 决定. 一旦进行交叉, 则从当前代群体中随机选取两个个体作为父代, 随机产生二维块指示器

$$\text{ND}_1 = \{(i_1, j_1), (i_2, j_2) \mid (i_1, j_1) \in R_M(n+2), (i_2, j_2) \in R_M(n+2)\}, \quad (5)$$

指向父代的子二维块, 相互交换各自的子二维块产生新的两个个体.

Pop_1 中个体是否变异由变异概率 p_{m1} 决定. 变异采用多子基因串下的单点变异, 即在待变异的个体中随机选取 m_1 个二维指示器

$$\text{ND}_2 = \{(i, j_1) \mid (i, j_1) \in R_M(n+2)\}, \quad (6)$$

指向子基因串 (I_i^j 和 L_i 或 y_i), 对其执行单点变异.

2.3.2 种群 Pop_2 的遗传操作

Pop_2 中个体是否交叉由概率 p_{c2} 决定. 进行交叉时, 在 Pop_2 中随机选取个体作为支配者, 在支配者的树结构中随机选取某个结点, 并选取另一个个体作为从属者参与交叉. 不同的结点类型其交叉操作也有差别: 如果支配者随机选取的结点为某一变量的 V_T , 则从属者的结点也为同一变量的 V_T , 二者的该结点及其树枝完全交叉; 如果支配者随机选取的结点为 M_F (树的第 3 层), 则从属者的第 3 层中随机选取某个结点, 二者的该结点及其树枝完全交叉; 如果支配者随机选取的结点为 A_o 或 V_U , 则从属者在第 4 层随机选取某个结点, 二者的该结点及其树枝完全交叉; 或从属者随机选取同类型的结点, 完成二者的结点交叉.

Pop_2 中个体是否变异由变异概率 p_{m2} 决定, 在待变异的个体中随机选取某个结点进行变异. 不同的结点类型其交叉操作也有差别: 如果支配者随机选取的结点为 V_T , 则在其对应的结点向量中随机选取另一个值作为变异结果 (即某个变量的分区发生变化), 假如需要增加新的隶属函数 (M_F 结点), 则按 M_F 的结点协议随机产生; 否则, 随机删除多余的 M_F 结点. 如果支配者随机选取的结点为 M_F , 则按该结点协议随机产生新 M_F 结点取代原先结点. 如果支配者随机选取的结点为 A_o 或 A_U , 则可随机选取结点变异或树枝变异; 对于结点变异, 可从对应的结点向量中随机选取另一个值作为变异结果; 对于树枝变异, 则按 A_o 和 A_U 结点协议随机产生新的树枝取

代原来的结点

在上述操作过程中, Pop_2 各结点的选取概率服从均匀分布. 个体中各 M_F 结点的4个输入参数值无论其大小如何排列, 在建立隶属函数 $T_{M_F}(x_j; a, b, c, d)$ 时, 通过调整其隶属函数4个参数树枝的次序, 使其满足 $a < b < c < d$. 如果4个参数相等, 则随机产生新的参数树枝取代其中一个; 如果4个参数中有的超过变量论域, 则以论域边界值代替.

2.4 TSK 模型的后件参数估计

本文采用最小二乘法对后件参数进行估计. 设通过协作进化产生 R 条规则, 对于输入样本 $X_j = [x_{1j} \ x_{2j} \ \dots \ x_{nj}]$, 根据式(2), 输出为

$$y_j = \frac{\sum_{i=1}^R \mu_j^i y_j^i}{\sum_{i=1}^R \mu_j^i} = \frac{\sum_{i=1}^R \beta_{ji} y_j^i}{\sum_{i=1}^R \mu_j^i}$$

其中

$$y_j^i = a_0^i + a_{1j}^i x_{1j} + \dots + a_{nj}^i x_{nj}$$

于是 N 个训练样本产生的输出向量为

$$Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} \beta_{11} & \dots & \beta_{1R} \\ \vdots & \ddots & \vdots \\ \beta_{N1} & \dots & \beta_{NR} \end{bmatrix} \begin{bmatrix} a_0^1 & \dots & a_n^1 \\ \vdots & \ddots & \vdots \\ a_0^R & \dots & a_n^R \end{bmatrix} \times \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_{11} & x_{12} & \dots & x_{1N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nN} \end{bmatrix} \quad (7)$$

通过矩阵重组, 式(7)可改写为

$$Y = XA^T \quad (8)$$

其中

$$X = \begin{bmatrix} \beta_{11} & \dots & \beta_{1R} & \beta_{11}x_{11} & \dots & \beta_{1R}x_{n1} \\ \beta_{N1} & \dots & \beta_{NR} & \beta_{N1}x_{1N} & \dots & \beta_{NR}x_{nN} \end{bmatrix},$$

$$A = [a_0^1 \ \dots \ a_n^1 \ \dots \ a_0^R \ \dots \ a_n^R],$$

X 是 $N \times R(n+1)$ 矩阵, A 是 $R(n+1)$ 维向量, 满足 $N > R(n+1)$. 对于式(8), 应用最小二乘法可得到参数 A , 即

$$A = (X^T X)^{-1} X^T Y \quad (9)$$

2.5 COOPCO-TSKFM 算法描述

1) COOPCO-TSKFM 的控制参数初始化, 按式(3)和图1生成 $Pop_1(0)$ 和 $Pop_2(0)$.

2) 根据式(4)和(9)评估初始种群中个体的适应值, 产生各种群的初始代表.

3) 对 $Pop_1(t)$ 按2.3.1节方法分别进行选择、杂交和变异等操作, 生成下一代种群 $Pop_1(t+1)$.

4) $Pop(t+1)$ 的个体与 $Pop_2(t)$ 的代表合作产生模糊模型, 并用式(9)对后件参数进行辨识; 计算

各模糊模型的适应值, 通过式(4)评估 $Pop_1(t+1)$ 中各个体的适应值, 采用精英策略保存最好个体并产生该种群代表. 如果算法终止条件满足, 则转7); 否则, 转6).

5) 类似于3), 对 $Pop_2(t)$ 按2.3.2节所述遗传操作生成 $Pop_2(t+1)$.

6) 类似于4), $Pop_2(t+1)$ 与 $Pop_1(t+1)$ 的代表合作, 计算 $Pop_2(t+1)$ 中各个体的适应值, 并产生该种群代表. 如果算法终止条件满足, 则转7); 否则, $t = t+1$, 转3).

7) 输出最好解, 算法终止.

上述算法中模糊模型的适应值计算方法为

$$\text{fit} - f_m = \alpha \left[1 / \left(\frac{1}{N_T} \sum_{j=1}^{N_T} (y_j - \hat{y}_j)^2 \right) \right] + \alpha_c (R_M / M_L) \quad (10)$$

其中: N_T 为测试样本个数, y_j 和 \hat{y}_j 分别为实际输出

和模糊模型输出, $M_L = \sum_{i=1}^{R_M} L_i > 0$ 为模型实际规则数, α 为精确性权值, α_c 为简洁性权值. 该计算方法从精确性和简洁性两方面衡量模型.

3 算例分析

下面以函数近似为例说明 COOPCO-TSKFM 算法的有效性. 采用文献[4]给出的非线性函数

$$x \in [0, \frac{1}{3}]: f(x) = -9x^2 + 3x + \frac{1}{3},$$

$$x \in [\frac{1}{3}, \frac{1}{2}]: f(x) = 2x - \frac{1}{3},$$

$$x \in [\frac{1}{2}, 1]: f(x) = \frac{0.03}{0.03 + (4.5x - 3.85)^2}$$

该函数的特征是: 在 $x = 1/3$ 处是非连续的导数, 在 $x = 1/2$ 处是不连续的, 在 $x = 0.85$ 附近有较高曲率的单驼峰. 在文献[4]中, 编码方式和进化方法决定了模糊模型的后件只能为单值实数, 需要预先确定规则数和模糊分区数. 按照[4]的方法, 种群规模为500, 规则数为20, 变量模糊分区数为20. 在第367代出现最大适应值, 均方差 $MSE = 0.005$.

应用 COOPCO-TSKFM 算法, 模型后件采用单值实数, 协同进化中种群 Pop_1 和 Pop_2 的各控制参数为: 种群大小(30, 40), $R_M = 25, N_B = 25$, 种群代表数(1+1, 1+1), 精英比例(0.1, 0.1), 交叉概率(0.7, 0.7), 变异概率(0.02, 0.03), $m_1 = 1, \alpha_1 = 1, \alpha_c = 5$, 最小二乘训练数据及适应值测试数据都为100, 算法终止条件: 协同进化400代或个体的适应值大于200. 按2.5提供的 COOPCO-TSKFM 算法,

第153代出现最大适应值,此时将输入变量 x 自动分解为16个模糊分区(如图2所示),形成16条规则,均方差 $MSE = 0.0047$.

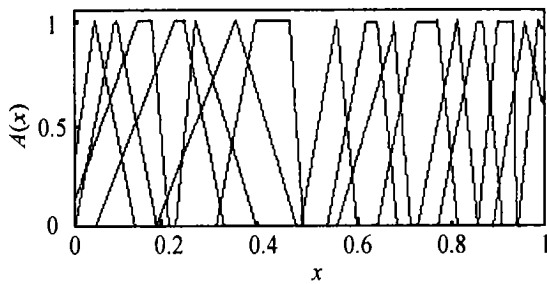


图2 输入变量的模糊划分

在COOPCO-TSKFM算法中,模型后件采用线性函数 $f(x) = a_0 + a_1x$ 的形式,按上述给出的控制参数,在第36代出现最大适应值249.此时规则数为12, $MSE = 0.0042$.

从上述结果可以看出,COOPCO-TSKFM算法能在较小种群规模和较少进化代数得到最优解,并能提供更灵活、更精确、更简洁的模糊模型.

4 结 语

本文提出一种基于协作共同进化的TSK模糊模型,并论述了与之相关的问题.该方法具有要求先验知识少、收敛速度快、自适应能力强等特点.对于COOPCO-TSKFM,还存在以下问题:1)两种群共同进化时,个体之间存在一定的数据约束关系,目前只是采用一种硬性处理的方法,如何在理论和应用上更好地解决还有待于进一步研究;2)COOPCO-TSKFM涉及的控制参数较多,各参数对共同进化过程和最优模糊模型的形成都有一定的影响,如何给出最优的参数选择方法也是需要深入研究的问题.

参考文献(References)

- [1] Takagi T, Sugeno M. Fuzzy identification of systems and its applications to modeling and control[J]. *IEEE Trans on Systems, Man and Cybernetics*, 1985, 15: 116-132.
- [2] Bastian A. Sequential fuzzy system identification[J]. *J of Control Cybernet*, 1996, 25(2): 199-223.
- [3] Nelles O, Sinsel S, Isemann R. Basis function networks for interpolation local linear models[A]. *Proc of IEEE Conf on Decision and Control*[C]. Kobe, 1996: 470-475.
- [4] Siarry P, Guely F. A genetic algorithm for optimizing Takagi-Sugeno fuzzy rule bases[J]. *Fuzzy Sets and Systems*, 1998, 99(1): 37-47.
- [5] Gonzalez A, Herrera F. Multi-stage genetic fuzzy systems based on the iterative rule learning approach[J]. *Mathware and Soft Computing*, 1997, 4(3): 233-249.
- [6] Delgado M R, Von Zuben F, Gomide F. Coevolutionary genetic fuzzy system: A hierarchical collaborative approach[J]. *Fuzzy Sets and Systems*, 2004, 141(1): 89-106.
- [7] Husband P, Mill F. Simulated co-evolution as the mechanism for emergent planning and scheduling[A]. *Proc of the 4th Int Conf on Genetic Algorithms*[C]. San Diego, 1991: 264-270.
- [8] Pena-Reyes C A, Sipper M. Applying fuzzy CoCo to breast cancer diagnosis[A]. *Proc of the 2000 Congress on Evolutionary Computation* [C]. Piscataway: IEEE Press, 2000, 2: 1168-1175.
- [9] Potter M A, De Jong K A. Cooperative coevolution: An architecture for evolving coadapted subcomponents[J]. *Evolutionary Computation*, 2000, 8(1): 1-29.