

基于 Memetic 框架混合群智能算法的 NARMAX 模型参数辨识

田谦益¹, 王小北²

(1. 漳州师范学院 计算机科学与工程系, 福建 漳州 363000;

2. 福建省热带作物科学研究所 科技信息管理科, 福建 漳州 363000)

摘要: 针对 NARMAX 模型的参数辨识问题, 提出一种新颖的混合群智能算法。该算法在 Memetic 算法框架的基础上, 采用粒子群算法与人工蜂群算法融合作为全局搜索策略, 采用单纯形优化法作为局部搜索策略。针对三个复杂程度不同的 NARMAX 模型进行的参数辨识仿真实验, 结果显示, 与标准人工蜂群算法和随机惯性权重粒子群算法相比, 新算法无论在鲁棒性还是求解精度上都具有一定优势。

关键词: 优化; 粒子群算法; 人工蜂群算法; 非线性系统; 参数辨识

中图分类号: TP391.4

文献标志码: A

文章编号: 1672-3767(2013)01-0088-07

Parameter Identification of NARMAX Model Using a Hybrid Swarm Intelligence Algorithm Based on Memetic Framework

Tian Qianyi¹, Wang Xiaobei²

(1. Department of Computer Science and Engineering, Zhangzhou Normal University, Zhangzhou, Fujian 363000, China;

2. Department of Science and Technology Information Management,

Fujian Institute of Tropical Crops, Zhangzhou, Fujian 363000, China)

Abstract: Aiming at the parameter identification of NARMAX model, a novel chaos-artificial bee colony algorithm was proposed. Based on the Memetic algorithm framework, the new algorithm uses the fusion of the particle swarm optimization and the artificial bee colony algorithm as the global search strategy and uses the simplex optimization as the local search strategy. Through the simulation experiment on the parameter identification of three different NARMAX models in complexity, the results show that the new algorithm, compared with the artificial bee colony algorithm and the random inertia weight particle swarm optimization, has some advantages in the robustness or solutions' quality.

Key words: optimization; particle swarm optimization; artificial bee colony; nonlinear system; parameter identification

在非线性系统辨识中, 非线性自回归滑动平均模型 (nonlinear auto regressive moving average with exogenous inputs, NARMAX)^[1] 是一类常用模型。该模型于 1982 年由 Billings 等提出, 并将其作为一类非线性系统的统一数学表示。对于此模型的系统辨识问题, 前人^[2-4] 已经做了大量的工作, 比如采用最小二乘法、非线性参数分析、神经网络等, 这些方法都能取得较好效果, 但都不同程度地存在参数较多或需要分步骤处理、运算复杂等缺点。

群体智能算法是近三十年来兴起的一类优化算法, 这类算法是模拟自然界中生物群体解决问题的行为方式而开发的元启发式搜索方式。目前, 群体智能算法已逐渐成为现代计算智能领域的研究热点, 为许多复

收稿日期: 2012-10-29

基金项目: 福建省教育厅科技项目 (JA09162)

作者简介: 田谦益 (1974—), 女, 云南昆明人, 讲师, 主要从事智能计算、信号处理与智能仪器方面的研究。

E-mail: qytian@126.com

杂非线性问题提供了一种解决方法。随着人们对群体智能算法研究的深入,也有学者将粒子群算法、蜂群算法引入到非线性系统研究中,并且取得了较好的效果^[5-6]。

本研究针对 NARMAX 模型的特点,提出一种混合群智能算法——ABCPSO(artificial bee colony & particle swarm optimization)算法。该算法基于 Memetic 框架,将 PSO(particle swarm optimization)和 ABC(artificial bees colony)算法相结合作为全局搜索策略,采用单纯形优化法作为局部搜索策略。该算法吸收了粒子群算法快速收敛与蜂群算法寻优能力强、不易陷入局部最优的特点,在对含噪声的 NARMAX 模型进行辨识过程中体现了良好的性能。

1 问题描述

NARMAX 模型的类型很多,其中最常用的是多项式表达的 NARMAX 模型,也是在实际系统中运行良好的一种模型^[7],其表达式可简记为

$$y(k) = \theta_0 + \sum_{i=1}^n \theta_i f_i(y(k), u(k)) + e(k) \quad (1)$$

其中: $f_i(\cdot)$ —由输入变量和输出变量相应延迟项的单项或乘积形成的单项式,如 $u(k-1), u(k-1)y(k-1), y^2(k-2)$ 等; $e(k)$ —噪声项,为不可测量项; θ_i —标量系数,即需要辨识的参数向量,其估计值为 $\hat{\theta}_i$ 。估计的偏差可用准则函数式(2)来衡量:

$$J(k) = \frac{1}{2N} \sum_{i=0}^n [y_p(k-i) - \hat{y}_p(k-i)]^2 \quad (2)$$

其中: N 为辨识窗口宽度。

2 群体智能算法

2.1 粒子群算法

粒子群算法(PSO)是 Eberhart 和 Kennedy^[8]于 1995 年提出的一种群体智能优化算法,该算法起源于对鸟群或者鱼群觅食行为的模拟。

假设在一个 D 维的目标搜索空间中,有 $PNum$ 个粒子组成一个群体,其中第 i 个粒子表示为一个 D 维向量 $\bar{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$,即第 i 个粒子在 D 维搜索空间的位置是 \bar{x}_i 。将 \bar{x}_i 代入一个目标函数就可以计算出其适应值,根据适应值的大小衡量 \bar{x}_i 的优劣。第 i 个粒子的“飞翔”速度也是一个 D 维向量,记为 $\bar{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ 。记第 i 个粒子迄今为止搜索到的最优位置为 $\bar{P}_i = (p_{i1}, p_{i2}, \dots, p_{iD})$,整个粒子群迄今为止搜索到的最优位置为 $\bar{P}_g = (p_{g1}, p_{g2}, \dots, p_{gD})$ 。基本 PSO 算法采用下式对粒子进行操作:

$$v_{id} = \omega \times v_{id} + c_1 \times \text{rand}() \times (p_{id} - x_{id}) + c_2 \times \text{rand}() \times (p_{gd} - x_{id}); \quad (3)$$

$$x_{id} = x_{id} + v_{id} \quad (4)$$

其中: $i=1, 2, \dots, PNum; d=1, 2, \dots, D; c_1, c_2$ —学习因子,为非负整数; $\text{rand}()$ —随机函数,产生 $[0, 1]$ 之间的随机数; ω —惯性权重; $v_{id} = [-v_{\max}, v_{\max}]$, v_{\max} 是常数,由用户设定。

2.2 蜂群算法

人工蜂群算法(ABC)是 Karaboga 等^[9]于 2005 年提出的一种模拟蜂群采蜜行为的群体智能优化算法。蜜蜂根据各自的分工进行不同活动,并实现蜂群信息的共享和交流,从而找到问题的最优解。在 ABC 算法中,人工蜂群可分为采蜜蜂、观察蜂和侦察蜂三种类型,采蜜蜂与蜜源一一对应,通过摇摆舞与其他蜜蜂分享蜜源信息;观察蜂根据采蜜蜂所在位置的蜜源量选择搜索区域,当采蜜蜂在其周围经过一定迭代次数仍未找到更好的蜜源,则成为侦察蜂,重新初始化。每个蜜源的位置代表优化问题的一个可能解,蜜源的花蜜量对应于相应解的质量或适应度,寻找并采集蜜源的速度对应问题求解的速度。

算法具体描述为:首先,ABC 随机产生初始群体即 SN 个初始解(SN 为采蜜蜂数也为蜜源数目)。将群体分为两部分,50%为采蜜蜂,50%为观察蜂。蜜源个数始终是 SN 个,不会随着迭代的进行而改变。随机产生的可行解为

$$x_{ij} = x_{\min j} + \text{rand}(0, 1) \times (x_{\max j} - x_{\min j})。 \quad (5)$$

其中, $j \in \{1, 2, \dots, Q\}$ 为 Q 维解向量的某个分量。

采蜜蜂记住自己当前的最优解, 在采蜜源附近邻域搜索, 搜索公式为

$$v_{ij} = x_{ij} + \varphi_{ij} (x_{ij} - x_{kj})。 \quad (6)$$

其中: $j \in \{1, 2, \dots, Q\}, k \in \{1, \dots, i-1, i+1, \dots, EN\}, k$ 随机生成, 且 $k \neq i, \varphi_{ij}$ 为 $[-1, 1]$ 之间的随机数, 随着迭代次数累加。

$(x_{ij} - x_{kj})$ 缩小, 则搜索的空间缩小, 即搜索的步长缩小。动态调整步长, 有助于提高算法精度, 并最终获得最优解或非常接近最优解的次优解。采蜜蜂采用贪婪准则, 比较记忆中的最优解和邻域搜索解, 当搜索解优于记忆最优解时, 替换记忆解; 反之, 保持不变。

在所有的采蜜蜂完成邻域搜索后, 采蜜蜂跳摆尾舞与观察蜂共享蜜源信息。观察蜂根据蜜源信息, 以一定概率选择采蜜源, 蜜量大的采蜜蜂吸引观察蜂的概率大于蜜量小的采蜜蜂。同样, 观察蜂在采蜜源附近邻域搜索, 采用贪婪准则, 比较观察蜂搜索解与原采蜜蜂的解, 当搜索解优于原采蜜蜂的解时, 替换原采蜜蜂的解, 完成角色互换; 反之, 保持不变。确定选择采蜜蜂招募度(招募其他蜜蜂的概率)的计算公式为

$$p_i = \frac{0.9 \text{fit}_i}{\sum_{i=1}^{SN} \text{fit}_i + 0.1}。 \quad (7)$$

其中: fit_i 为第 i 个解的适应度(函数值), 该值对应于该点的蜜量。当某只采蜜蜂在其周围搜索次数 Bas 达到一定阈值 $Limit$ 而仍未找到更优解时, 放弃该蜜源, 此处的采蜜蜂变为侦察蜂, 该蜜源被侦察蜂在解空间中随机搜索的新蜜源(即新可行解)代替。假设被放弃的可行解为 x_i , 则侦察蜂发现并替换 x_i 的公式为

$$x_{ij} = \begin{cases} x_{\min j} + \text{rand}(0, 1)(x_{\max j} - x_{\min j}), & Bas_i \geq Limit \\ x_{ij}, & Bas_i < Limit \end{cases}。 \quad (8)$$

作为群智能算法的代表之一, PSO 算法由于其简单的数学模型和明确的应用背景, 得到了广泛应用, 但由于目前该算法的理论还不完善, 而且较易陷入局部最优, 因此对该算法的研究主要在理论研究、算法改进及应用拓展方面。同为群智能算法代表的 ABC 算法出现的时间虽不长, 但该算法突出的优点是种群多样性好, 每次迭代中都进行全局和局部搜索, 提高了找到最优解的概率, 并在较大程度上避免了局部最优。由此可见, 两种算法具有一定的互补性, 可以考虑将二者相融合, 构成一种新型的混合优化算法。

3 基于 Memetic 框架的 ABCPSO 算法

算法融合的方式有多种, 可以是串联融合、并联融合、部分融合或者完全融合等。这些方法可以单独使用, 也可以根据实际情况结合使用。考虑到 ABC 算法和 PSO 算法都是基于种群的算法, 而且二者具有一定的互补性, 本研究提出了一种基于 Memetic 框架的混合群智能算法。该算法在 ABC 算法与 BSO 算法的基础上进行创新, 引入了 Memetic 框架, 采用串联融合与部分融合相结合的方法将 PSO 算法与 ABC 算法融合在一起, 既能够提高算法的收敛精度, 又可以增加单纯形法作为局部搜索策略, 构成新型的全局搜索算法。

3.1 Memetic 框架

Memetic algorithm(MA) 是 Pablo Moscato 于 1989 年首次提出的概念, 并称之为一种基于群体的混合全局启发式搜索算法^[10]。事实上, MA 可以看作是一种全局搜索算法与局部搜索算法相结合的混合算法框架, 各个环节都有多种实施策略, 采用不同的策略就可以构成不同的基于 Memetic 框架的混合算法。本研究提出的基于 Memetic 框架的混合算法, 就是采用 PSO 算法与 ABC 算法相结合, 构成新型的全局搜索策略, 既能保证算法种群的多样性, 又能增强算法的鲁棒性。局部搜索策略采用单纯形优化方法, 以增强算法的整体寻优能力。

3.2 PSO 与 ABC 相融合的全局搜索策略

PSO 算法在辨识 NARMAX 模型时, 具有较强的鲁棒性, 收敛速度快, 但很容易陷入局部最优; ABC 算法则搜索能力较强。针对这两种算法的优点, 本研究提出了一种新的全局搜索策略。该策略基本框架与 ABC 算法类似, 将所有种群进行三种分工, 觅食粒子、观察粒子和侦察粒子分别对应 ABC 算法中的采蜜蜂、

观察蜂和侦察蜂。

首先,随机产生初始群体即 SN 个初始解。位置初始化公式如式(5)所示,速度初始化公式为

$$v_{ij} = \text{rand}(0,1)v_{\max j} - \frac{v_{\max j}}{2} \quad (9)$$

将群体分为两部分:50%为觅食粒子,50%为观察粒子。觅食粒子采用式(3)、式(4)进行迭代,搜索群中的最优粒子,逐步向食物源靠近。其中惯性权重 ω 采用公式(10)进行随机调整。

$$\omega = 0.5 + 0.5\text{rand}(0,1) \quad (10)$$

观察粒子采用式(5)、式(6)进行迭代,同样,观察粒子在食物源附近领域搜索,采用贪婪准则,比较观察粒子搜索解与觅食粒子的解,当搜索解优于原觅食粒子的解时,替换原觅食粒子的解,完成角色互换;反之,保持不变。侦察粒子与侦察蜂的作用一样,主要是为了增强种群的多样性,以降低种群陷入局部最优的几率。

基于 Memetic 框架的 ABCPSO 算法的局部搜索策略采用单纯形优化方法,这有助于进一步提高局部搜索的精度。下面给出该算法的伪代码:

```

/* 初始化阶段 */
按照式(5)初始化食物源种群  $\bar{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD}), i = 1, 2, \dots, SN/2$ 
按照式(9)初始化种群速度
while ( cycle  $\leq$  Maxcycle ) do
/* 觅食粒子阶段 */
for  $i = 1$  to  $SN/2$  do
    利用公式(3),(4)优化粒子,惯性权重  $\omega$  采用公式(10)进行计算
end for
/* 利用公式(7)计算每个食物源被观察粒子选中的概率 */
/* 观察粒子阶段 */
for  $i = 1$  to  $SN/2$  do
    利用公式(5),(6)优化粒子
end for
利用单纯性优化方法进行局部搜索
/* 侦察粒子阶段 */
利用公式(8)生成侦察蜂
end while
    
```

4 仿真实验

为了验证 ABCPSO 算法的有效性,将 ABCPSO 算法与随机惯性权重粒子群算法(RIWPSO, random inertia weight PSO)^[11]和 ABC 算法进行对比,采用三个 SISO(single input single output)NARMAX 多项式模型进行实验。

第一个 NARMAX 模型为

$$y(t) = 0.5y(t-1) + u(t-2) + 0.1u^2(t-1) + 0.5e(t-1) + 0.2u(t-1)e(t-2) + e(t) \quad (11)$$

其中:输入 $u(t)$ 为区间 $[-\sqrt{3}, \sqrt{3}]$ 上的零均值均匀分布随机序列; $e(t)$ 为区间 $[-0.04, 0.04]$ 上的零均值均匀分布加性噪声;辨识参数为 $\theta = [-0.3, 0.4, 1.25, -2.5]$ 。

第二个 NARMAX 模型为

$$y(t) = (0.8 - 0.5e^{-y(t-1)^2})y(t-1) + (-0.3 - 0.9e^{-y(t-1)^2})y(t-2) + u(t-1) + 0.2u(t-2) + 0.1u(t-1)u(t-2) + e(t) \quad (12)$$

其中:输入 $u(t)$ 为区间 $[-\sqrt{3}, \sqrt{3}]$ 上的零均值均匀分布随机序列; $e(t)$ 为区间 $[-0.04, 0.04]$ 上的高斯白噪

声; 辨识参数为 $\theta = [0.8, -0.5, -0.3, -0.9, 1, 0.2, 0.1]$ 。

第三个 NARMAX 模型为

$$y(t) = -0.4y(t-1) + 0.2y(t-2)y(t-3) + 0.4y(t-4) + 0.2y(t-2)^2 + 0.8u(t-1)^3 + 0.3u(t-3) + e(t). \quad (13)$$

其中: 输入 $u(t)$ 为区间 $[-1, 1]$ 上的零均值均匀分布随机序列; $e(t)$ 为区间 $[-0.05, 0.05]$ 上的零均值均匀分布加性噪声; 辨识参数为 $\theta = [-0.4, 0.2, 0.4, 0.2, 0.8, 0.3]$ 。

仿真在 PIV/1.7G 的计算机上进行, 算法用 Matlab7.1 实现, 在 Windows 2000 环境下运行。仿真采用离线开环辨识的方法, 对模型参数进行批处理辨识, 辨识窗口宽度统一为 500。为了研究噪声对算法性能的影响, 本文采用的模型都是带噪声的模型, 其中模型 3 只含有加性噪声, 而模型 1 和模型 2 既含有加性噪声也含有乘性噪声。对一个模型每做一次辨识, 三种算法各随机运行 30 次, 参数取 30 次实验结果的平均值 (表 1)。种群数设置为 20。

表 1 辨识结果对比表

Tab.1 Comparisons of identification results

| 模型 | | ABCPSO | RWPSO | ABC |
|--------|----------------------------|------------------------|----------------------------|------------------------|
| Model1 | 参数 | 0.496 706 319 426 034 | 0.467 628 805 979 354 | 0.513 157 719 793 574 |
| | | 1.006 230 253 098 877 | 0.895 382 655 702 131 | 0.904 998 835 970 176 |
| | | 0.101 674 317 585 375 | 0.107 873 827 965 362 | 0.110 984 845 387 896 |
| | | 0.475 618 328 310 499 | 0.230 731 134 631 021 | 0.357 191 856 212 092 |
| | | 0.179 464 848 208 851 | 0.194 750 551 449 437 | 0.407 880 525 297 194 |
| | 平均误差 | 0.000 986 777 862 128 | 0.038 024 782 472 111 | 0.013 980 110 291 842 |
| Model2 | 参数 | 0.773 022 132 334 824 | 0.826 847 505 086 102 | 0.851 975 985 963 987 |
| | | -0.464 991 106 798 965 | -0.204 977 593 687 474 | -0.341 173 133 611 801 |
| | | -0.286 568 036 158 284 | -0.311 512 427 623 463 | -0.437 201 243 301 803 |
| | | -0.910 825 739 324 685 | -0.989 859 872 963 998 | -0.627 467 104 557 023 |
| | | 1 | 1.030 776 901 813 363 | 0.786 180 386 730 345 |
| | 平均误差 | 0.008 109 840 052 197 | 0.039 993 765 366 811 | 0.077 081 801 137 700 |
| Model3 | 参数 | -0.399 999 991 837 134 | -0.397 899 641 846 501 | -0.399 506 799 069 568 |
| | | 0.200 000 006 453 294 | 0.192 753 343 318 156 | 0.204 086 816 791 390 |
| | | 0.400 000 007 129 107 | 0.403 116 724 629 077 | 0.399 917 657 807 854 |
| | | 0.200 000 003 468 182 | 0.194 296 374 091 577 | 0.201 778 350 815 503 |
| | | 0.800 000 002 140 229 | 0.804 178 392 059 816 | 0.799 482 274 162 526 |
| | 平均误差 | 0.300 000 005 706 089 | 0.301 389 916 323 736 | 0.299 999 047 291 959 |
| | 5.054 389 643 519 085e-017 | 0.000 229 079 631 518 | 1.365 869 536 570 960e-005 | |

表 1 为利用三种算法进行辨识得出的参数及平均误差。可以看出, ABCPSO 算法比其他两种算法具有明显的优势。在对模型 3 进行辨识时, 三种算法都能得到较高精度的参数, 误差较小, 其中以 ABCPSO 算法更胜一筹。究其原因, 主要是模型 3 相对其他两个模型较简单, 而且只含有加性噪声。模型 1 和模型 2 都较复杂, 既含有加性噪声又含有乘性噪声, 在对模型 1 和模型 2 辨识时, 三种算法的精度都有不同程度的降低, 但和 RWPSO 算法和 ABC 算法相比, ABCPSO 算法鲁棒性较强。

从图 1 可以看出, 无论辨识哪个模型, ABCPSO 算法的误差都较小。图 2 给出的是三种算法在迭代过

程中的适应值曲线。需要特别说明的是:为了更清晰地显示趋势,纵坐标使用了实验中适应值的对数形式(最优解)。从图 2 可以看出,ABCPSO 算法具有较强的寻优能力,且精度很高。ABC 算法在处理模型 1 和模型 3 时,能力居中。在处理模型 2 时,RWPSO 算法则胜过 ABC 算法。

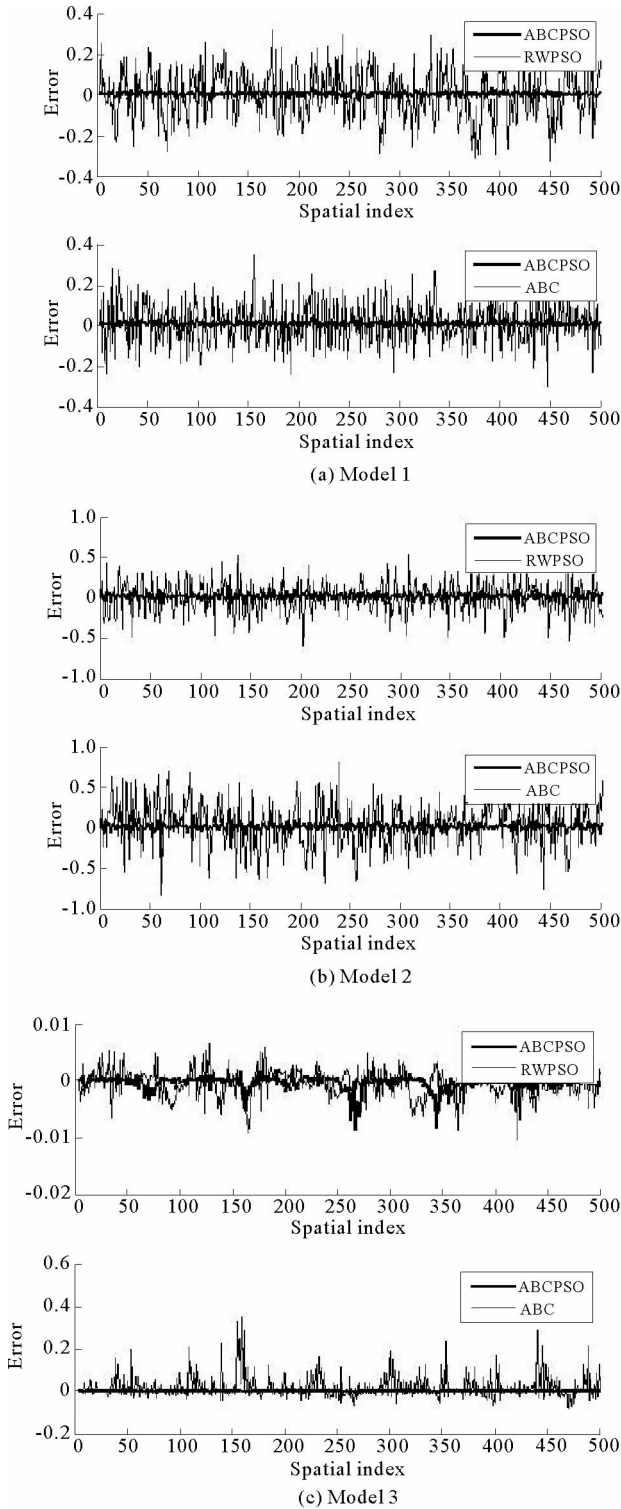


图 1 三种算法的辨识误差曲线对比图

Fig. 1 Comparisons on the identification errors for the three algorithms

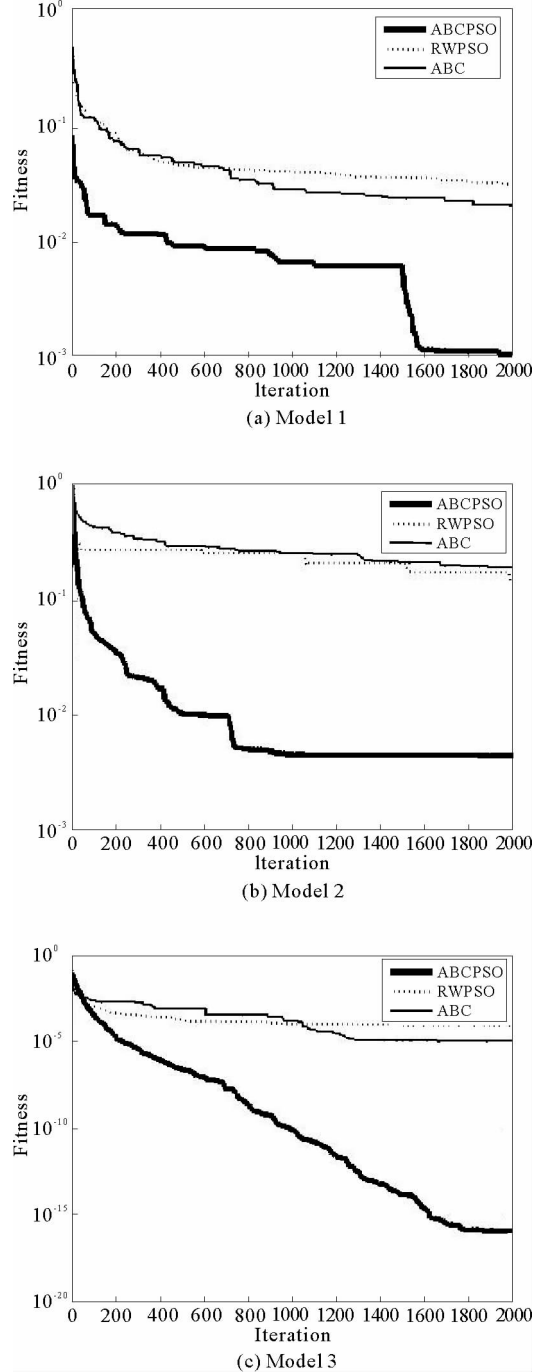


图 2 三种算法的辨识进化曲线图

Fig. 2 Convergences of the three algorithms for the models

5 结束语

提出了一种能够有效进行非线性系统辨识的优化算法——基于 Memetic 框架的 ABCPSO 算法。利用 PSO 算法简单且收敛速度快,ABC 算法不易陷入局部最优的特点,将二者有机融合,构成基于 Memetic 框架的全局搜索策略,同时局部搜索策略采用单纯形优化方法。这样既能保证算法种群的多样性,不易陷入局部最优,又能增强算法的鲁棒性,提高算法的整体寻优能力。

实验结果表明:首先,在对非线性系统进行辨识时,ABCPSO 算法不但在精度上优于 RWPSO 算法和 ABC 算法,而且寻优能力也比 RWPSO 算法和 ABC 算法有明显提高;其次,在处理一般含加性噪声的模型时,三种算法都能取得较好的结果,而对于更为复杂的模型,RWPSO 和 ABC 两种算法性能受到较大影响,说明 ABCPSO 算法的鲁棒性较强,这一点在处理非线性系统时非常重要。

研究表明,基于 Memetic 框架的 ABCPSO 算法是一种能够对复杂非线性系统进行有效辨识的新型混合群智能算法,进一步的研究是对算法的收敛性进行理论分析。

参考文献:

- [1]Leontaritis I J,Billings S A. Input-output parametric models for non-linear systems Part I;Deterministic non-linear systems [J]. International Journal of Control,1985,41(2):311-344.
- [2]Mendes E M A M,Billings S A. An alternative solution to the model structure selection problem[J]. IEEE Transactions on Systems,Man and Cybernetics-Part A;Systems and Humans,2001,31(6):597-608.
- [3]Chiras N,Evans C,Rees D. Nonlinear gas turbine modeling using NARMAX structures [J]. IEEE Transactions on Instrumentation and Measurement,2001,50(4):893-898.
- [4]Rahim N A,Taib M N,Yusof M I. Nonlinear system identification for a DC motor using NARMAX Approach[C]//Asian Conference on Sensors (AsiaSense). Kuala Lumpur,Malaysia,July 18,2003:305-311.
- [5]Supeni E,Yassin I M,Ahmad A, et al. NARMAX identification of DC motor model using repulsive particle swarm optimization[C]//5th International Colloquium on Signal Processing & Its Applications (CSPA),Kuala Lumpur,Malaysia, Mar. 6-8,2009:1-7.
- [6]Rao R V,Pawar P J. Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms [J]. Applied Soft Computing,2010,10(2):445-456.
- [7]Chen S,Billings S A. Representation of non-linear systems;the NARMAX model[J]. International Journal of Control,1989,49(3):1013-1032.
- [8]Kennedy J,Eberhar T R C. Particle swarm optimization[C]//IEEE International Conference on Neural Networks, Perth West Australia,Nov. 27-Dec. 1,1995(4):1942-1948.
- [9]Karaboga D. An idea based on bee swarm for numerical optimization[R/OL]. Kayseri, Turkey:Erciyes University,2005. [2012-09-05]. http://www.dmi.unict.it/mpavone/nc-cs/materiale/tr06_2005.pdf.
- [10]Moscato P. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms[R/OL]. Technical Report No. 790,Caltech Concurrent Computation Program,California Institue of Technology,1989:1-68. [2012-09-05]. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.27.9474&rep=rep1&type=pdf>.
- [11]Eberhart R C,Shi Y. Tracking and optimizing dynamic systems with particle swarms[C]//The IEEE International Congress on Evolutionary Computation, Seoul,May 27-30,2001:94-100.

(责任编辑:吕文红)