

6-2-2020

## Identification of Takagi-Sugeno Model with Self-gravitational Search Algorithm

Ao Yuan

*College of Computing & Electrical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China;*

Xueming Ding

*College of Computing & Electrical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China;*

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

---

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

---

## Identification of Takagi-Sugeno Model with Self-gravitational Search Algorithm

### Abstract

**Abstract:** An approach of Self-Gravitational Search Algorithm (SGSA) was proposed for the identification of T-S fuzzy model. In the identification of T-S model, *structure parameters and consequent parameters were encoded into a particle of SGSA*. Based on new strategies, *all the individuals in SGSA dynamically adjusted the distance and force between particles according to the intensity of the swarm, and the gravitational constant was adaptively alerted*. As a result, the global search ability of the proposed SGSA was enhanced in the earlier stage of the search process to prevent the algorithm from trapping into local regions, in the mean while; the local search capability was improved in the latter iterations of optimization to reduce the probability of the algorithm fruitlessly searching around the local optima. The simulation results and comparative analysis demonstrate the good performance of the proposed method with high accuracy and great robustness.

### Keywords

T-S model, system identification, self-adaption, GSA

### Recommended Citation

*Ao Yuan, Ding Xueming*. Identification of Takagi-Sugeno Model with Self-gravitational Search Algorithm[J]. Journal of System Simulation, 2017, 29(3): 487-493.

# 基于自适应引力搜索算法的 T-S 模型辨识

敖媛, 丁学明

(上海理工大学光电信息与计算机工程学院, 上海 200093)

**摘要:** 提出一种基于自适应引力搜索算法(Self-Gravitational Search Algorithm, SGSA)的 T-S 模型辨识方法, 把 T-S 模型的前件参数和后件参数编码进一个粒子中用 SGSA 辨识。SGSA 是针对标准引力搜索算法(GSA)收敛过快的缺点, 在 GSA 的基础上, 根据群体密集程度动态调整粒子间的距离和受力大小, 并自适应修改引力常数  $G$  的改进引力搜索算法。不仅增加了算法在前期的全局搜索能力, 防止其过早收敛; 而且降低了算法在后期最优解附近震荡的影响, 提高了算法的开采能力。仿真结果表明该方法能获得较高的辨识精度, 验证了算法的有效性。

**关键词:** T-S 模型; 系统辨识; 自适应; 引力搜索算法

中图分类号: TP 237 文献标识码: A 文章编号: 1004-731X (2017) 03-0487-07

DOI: 10.16182/j.issn1004731x.joss.201703003

## Identification of Takagi-Sugeno Model with Self-gravitational Search Algorithm

Ao Yuan, Ding Xueming

(College of Computing & Electrical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

**Abstract:** An approach of Self-Gravitational Search Algorithm (SGSA) was proposed for the identification of T-S fuzzy model. In the identification of T-S model, *structure parameters and consequent parameters were encoded into a particle of SGSA*. Based on new strategies, *all the individuals in SGSA dynamically adjusted the distance and force between particles according to the intensity of the swarm, and the gravitational constant was adaptively alerted*. As a result, the global search ability of the proposed SGSA was enhanced in the earlier stage of the search process to prevent the algorithm from trapping into local regions, in the mean while; the local search capability was improved in the latter iterations of optimization to reduce the probability of the algorithm fruitlessly searching around the local optima. The simulation results and comparative analysis demonstrate the good performance of the proposed method with high accuracy and great robustness.

**Keywords:** T-S model; system identification; self-adaption; GSA

## 引言

在实际工程问题中, 许多系统是具有强耦合性、迟滞性的非线性系统, 使用传统的方法建立精

确的数学模型非常困难。对于这类问题, 模糊模型由于其对非线性系统具有良好的逼近能力, 被证明是一种有效的辨识工具。T-S 模糊模型<sup>[1]</sup>就是其中著名的模糊模型之一, 由 Takagi 和 Sugeno 于 1985 年提出, 是一种基于输入模糊划分思想的辨识模型, 已经被许多学者广泛的研究和应用<sup>[2-10]</sup>。目前, 辨识 T-S 模型通常的步骤分为结构辨识和参数辨识两步<sup>[5-10]</sup>。结构辨识常用的方法是聚类算法, 参数辨识主要的方法是小二乘法或者智能算法<sup>[11-12]</sup>。



收稿日期: 2015-06-02 修回日期: 2015-08-30;  
基金项目: 宝山区科技创新专项资金(bkw201408),  
沪江基金(C14002);  
作者简介: 敖媛(1992-), 女, 贵州盘县, 硕士生, 研究方向为系统辨识、智能算法; 丁学明(1971-), 男, 安徽无为, 博士, 副教授, 研究方向为智能算法。

<http://www.china-simulation.com>

然而两步的方法通常会影响到 T-S 模型的紧密性, 并且不利于获得一个更精确的 T-S 模糊模型。所以解决两步辨识过程中模型参数的一致性和寻找有效的搜索算法是 T-S 模型辨识的重点以及难点。

引力搜索算法(GSA)是由 Rashedi 等人于 2009 年提出的启发式智能优化算法<sup>[7]</sup>, 该算法源于牛顿万有引力定律。文献[13]表明, 引力搜索算法处理各种非线性函数的性能明显优于其他一些优化算法, 如 PSO 和 CFO。尽管标准的引力搜索算法有着相对优秀的搜索能力, 但是它仍然会出现过早收敛的问题。对此, 许多学者对 GSA 算法做出改进<sup>[3,14]</sup>。文献[3]中, 把混沌引入 GSA 算法中。文献[14]中, 增加了随机扰动。但都很难找到防止其过早收敛的方法。

本文提出一种基于 SGSA 的 T-S 模型辨识方法, 把 T-S 模型的前件参数和后件参数编码进一个粒子, 消除了辨识结构和后件参数过程中产生的一致。并采用基于种群密度的自适应策略和引力常数  $G$  的自适应策略, 改进了传统 GSA 过早收敛的缺点。最后用静态非线性系统和 Box-Jenkins 系统进行实验仿真, 验证算法的有效性。

## 1 T-S 模型

T-S 模型是把非线性系统看成是多个线性系统的加权组合, 通过模糊推理的方法实现全局的非线性, 能够逼近任意非线性系统。用 T-S 模型描述非线性系统时, 第  $i$  条模糊规则的数学表达式如下:

$$R^i: \text{if } x_1 \text{ is } A_1^i, x_2 \text{ is } A_2^i, \dots, x_n \text{ is } A_n^i \\ \text{then } y_i = \omega_i^T x + b_i (i=1, 2, \dots, R)$$

式中:  $x=[x_1, x_2, \dots, x_n]$  是系统的输入;  $A_j^i$  是模糊集;  $n$  是输入变量的维数;  $R$  是模糊规则总数;  $\omega=[\omega_1, \omega_2, \dots, \omega_n]$  是后件参数;  $y_i$  是第  $i$  条模糊规则的输出。当给定输入  $x=[x_1, x_2, \dots, x_n]$  时, 则模型的输出可由公式(1)计算:

$$y = \frac{\sum_{i=1}^m \mu_i y_i}{\sum_{i=1}^m \mu_i} \quad (1)$$

其中, 第  $i$  条规则占总体输出权重  $\mu$  的表达式为:

$$\mu_i(x) = \prod_{j=1}^n h_j(x) \quad h_j(x) \text{ 为隶属函数 } A_j^i \text{ 的隶属度。}$$

本文隶属函数选择高斯函数, 即  $h_j(x) = \exp\left(-\frac{(x_j - c_{ij})^2}{\sigma_{ij}^2}\right)$ ,  $c_{ij}$  和  $\sigma_{ij}$  分别代表类中心和基宽参数。

## 2 自适应引力搜索算法

### 2.1 引力搜索算法

引力搜索算法是一种基于牛顿万有引力定律和运动定律的启发式搜索算法。假设在一个  $D$  维的搜索空间中, 由  $N$  个粒子组成, 其中第  $i$  个粒子在  $D$  维搜索空间中的位置为:  $x_i = [x_i^1, x_i^2, \dots, x_i^d, \dots, x_i^D]$  ( $i=1, 2, \dots, N$ )。以在第  $t$  次迭代中 GSA 的执行过程为例, 算法的实现如下。

在 GSA 算法中, 首先在可行解空间内随机初始化粒子的位置, 并计算出当前粒子的适应度值。适应度的值决定了 GSA 中粒子惯性质量的大小, 使用以下公式更新粒子的惯性质量:

$$m_i(t) = \frac{\text{fit}_i(t) - \text{worst}(t)}{\text{best}(t) - \text{worst}(t)} \quad (2)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^N m_j(t)} \quad (3)$$

式(3)中,  $\text{fit}_i(t)$  代表第  $i$  个粒子的适应度值。best( $t$ ) 为最好的适应度值, worst( $t$ ) 为最差的适应度值。

此时, 粒子  $i$  受到其他粒子的作用力大小为:

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) + M_{pj}(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \\ j = 1, 2, \dots, N (j \neq i) \quad (4)$$

其中:  $\varepsilon$  是一个很小的常量;  $G(t)$  是引力常量,  $R_{ij}(t)$  是第  $i$  个粒子和第  $j$  个粒子之间的欧式距离:  $R_{ij}(t) = \|x_i(t), x_j(t)\|_2$ 。

粒子  $i$  上总的的作用力为:

$$F_i^d(t) = \sum_{j \in kbest} \text{rand}_j F_{ij}^d(t) \quad (5)$$

其中:  $\text{rand}_j$  是范围在  $[0, 1]$  之间的随机数。

又由牛顿第二定理知,  $a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)}$ , 因此,

GSA 中粒子速度和位置更新公式为:

$$v_i^d(t+1) = \text{rand}_i \times v_i^d(t) + a_i^d(t) \quad (6)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (7)$$

通过 GSA 的实现过程, 不难发现, GSA 算法中粒子间因存在着引力的作用, 迫使较轻的粒子(适应度较低的粒子)不断大步长运动寻找全局最优解, 而适应度较高的粒子由于质量大而移动小, 保留目前最优值的能力, 这两点使得 GSA 算法有很强的全局搜索能力。但是, 文献[3,14]中也指出, GSA 算法的收敛速度比较快, 当遇到比较复杂的问题时, 极易早熟而陷入局部最优。

## 2.2 自适应改进策略

为避免 GSA 陷入局部最优, 加强粒子之间信息交流, 平衡算法的勘探和开采能力, 提出以下两方面改进策略。

### 2.2.1 基于群体密集度的自适应改进策略

牛顿万有引力定律中, 物体受到的力是和物体间的距离的平方成正比。而在 GSA 中并没有完全遵循牛顿万有引力定律, 文献[13]通过实验发现采用  $R$  的效果比较好, 计算引力时用  $R$  代替了  $R^2$ 。然而, 固定距离对粒子间相互作用的影响有以下两个明显的不足:

(1) 当种群比较分散的时候, 粒子间的距离较大, 则它们之间的相互作用变得非常小, 从而使 GSA 算法退化成随机运动。

(2) 当种群比较密集时候, 粒子的距离非常小, 使得相互作用力变得非常大, 算法中的粒子会在最优解附近高速震荡而减慢收敛速度。

因此, GSA 丧失了其优秀的全局搜索能力和局部搜索能力。合理的算法应该随着种群密集程度来改变相应的作用力大小。由此, 对公式(4)进行如下改进。

首先, 定义描述种群密集程度的变量  $\delta(\delta \geq 0)$ , 其公式为:

$$\delta = \frac{1}{n} \sum_{i=1}^n \delta_i \quad (8)$$

其中:  $n$  为粒子个数;  $\delta_i$  为粒子  $i$  和其他所有粒子

的平均距离

$$\delta_i = \frac{1}{n-1} \sum_{j=1, j \neq i}^n \sqrt{\sum_{k=1}^D (x_i^k - x_j^k)^2} \quad (9)$$

然后, 将公式(4)改进为:

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) + M_{pj}(t)}{R_{ij}^{Rp(\delta)}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \quad (10)$$

其中:  $Rp(\delta)$  设计为:

$$Rp(\delta) = \begin{cases} Rp_{\min} + (Rp_{\max} - Rp_{\min})e^{-1/\delta} & 0 \leq \delta < 1 \\ Rp_{\min} + (Rp_{\max} - Rp_{\min})e^{-1} & \delta \geq 1 \end{cases} \quad (11)$$

式中:  $Rp_{\min}$  和  $Rp_{\max}$  分别是  $R$  的指数的最小和最大值。

公式(11)的设计思路为: 当种群密集度  $\delta$  太大或太小时将导致算法中粒子之间作用力过大或过小, 影响算法的收敛性。因此, 如果当群体密度不在一个合适的范围内时, 应当减少因粒子间距离对相互作用力的影响, 提高算法的收敛能力。

### 2.2.2 引力常数 $G$ 的自适应策略

在公式(4)中, 影响算法性能的另一个参数为引力常数  $G$ , 它直接影响粒子受到的合力和加速度的大小。也就是说,  $G$  描述了算法中粒子“步长”的大小, 是决定粒子能否摆脱局部最优和最优解精细度最直接的因素。基本 GSA 中的引力常量公式为:

$$G(t) = G_0 e^{-\alpha \frac{t}{T}} \quad (12)$$

式中:  $G_0$  表示引力常数的初始值,  $\alpha$  是描述衰减速率的量,  $T$  为总的迭代次数,  $t$  为当前迭代的次数。

图 1 中的曲线 1 为公式(12)中引力常数  $G(t)$  随迭代次数  $t$  的变化情况。由图可知  $G(t)$  在刚开始就快速下降。这样做虽然能使算法快速收敛, 但是算法也极易陷入局部最优。显然这与我们所希望的在算法刚开始有很强的勘探能力是不相符的。

为了解决上面所提出的问题, 设计引力常数  $G'$  公式如下:

$$G'(t) = G'_0 e^{-\alpha' (\frac{t}{T})^n}, \quad n > 0 \quad (13)$$

式中:  $G'_0$  表示引力常数的初始值;  $\alpha'$  是描述衰减速率的量;  $T$  为总的迭代次数;  $t$  为当前迭代的次数;  $n$  为描述勘探和开发比例的一个参数。

图 1 中曲线 2 为所设计的  $G'$ 。通过图 1 可以发现新设计的  $G'$  可以由  $n$  来调节算法在搜索过程中的开发和勘探的比例,  $n$  越大算法初期勘探的比例就越大。需要说明的是, 当  $n=1$  时公式(13)和原算法中公式(12)等效。

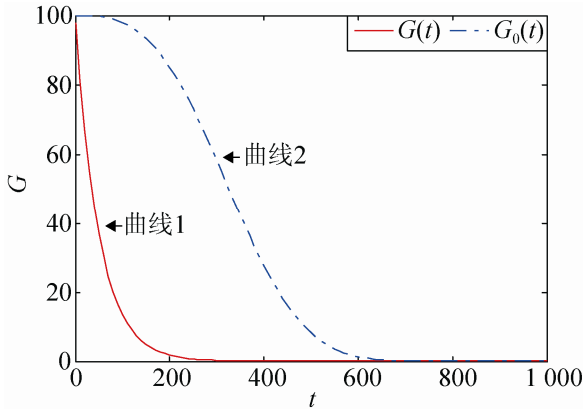


图 1 引力常量  $G(t)$  和  $G'(t)$  与迭代次数  $t$  的关系  
Fig. 1 Gravitational constant  $G(t)$  and  $G'(t)$  relationship with the number of iterations  $t$

通过对原始的 GSA 采用基于种群密度的自适应策略和引力常数  $G$  的自适应策略, 改进了传统 GSA 过早收敛的缺点。改进后的算法 SGSA 其实现流程如下所示:

#### 1、初始化

(a) 设置算法的迭代次数, 种群大小, 引力常数  $G'_0$ 、引力常数的衰减速率  $\alpha'$ 、勘探与开发的比例  $n$ 、公式(11)中距离的指数最大值和最小值  $Rp_{\min}$  和  $Rp_{\max}$

(b) 随机初始化粒子的位置和速度

(c) 计算  $\begin{cases} \text{best}(t) = \arg \{ \min f(x_i) \} \\ \text{worst}(t) = \arg \{ \max f(x_i) \} \end{cases}$

( $f(\cdot)$  是适应度函数)

#### 2、算法结束检查

If 算法满足终止条件, 结束

输出最优解  $\text{best}(t)$

Else 执行步骤 3

#### 3、SGSA 算法优化参数

(a) 根据公式(2)、(3)计算当前粒子的质量  $M$

(b) 由公式(8)、(9)、(10)、(11)、(13)、(5)计算出当前粒子受到其他粒子的合力

(c) 根据公式(6)、(7)更新粒子的位置和速度

(d) 更新第  $t$  次迭代中粒子适应度值, 计算出  $\text{best}(t)$ ,  $\text{worst}(t)$

4、执行步骤 2

5、结束

## 3 基于 SGSA 的 T-S 模型辨识

本文提出的基于 SGSA 的 T-S 模型辨识方法, 使用上文改进的 GSA 算法优化 T-S 模型的前件参数和后件参数。算法实现过程的详细描述如下。

### 3.1 编码策略

在算法的编码上, 为提高辨识出 T-S 模型的精度, 克服传统前件参数和后件参数分步辨识所引起模型结构内部不紧密的问题。本文所设计的编码策略, 每个 SGSA 粒子包含 T-S 模型的所有规则, 每条规则又由前件结构参数和后件参数两个部分构成, 这样使每个粒子可以代替一个完整的 T-S 模糊模型, 保持了前件参数和后件参数内部的关系。算法中每个粒子的编码方式如图 2 所示。

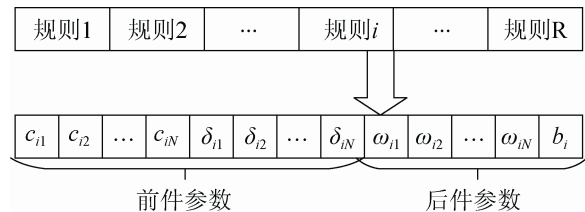


图 2 SGSA 算法粒子编码方式

Fig. 2 SGSA particle encoding

### 3.2 适应度函数

合理的适应度函数选择可以获得一个满意的结果。本文使用均方差(MSE)作为 SGSA-TS 算法的适应度函数。均方差的值越小, 表明该粒子的适应度值越高。MSE 的数学表达式如下:

$$MSE = \frac{1}{k} \sum_{i=1}^k (y_i - \hat{y}_i)^2 \quad (14)$$

其中:  $k$  是训练样本数据的个数;  $y_i$  是样本的目标输出;  $\hat{y}_i$  是 T-S 模型的输出。

### 3.3 算法实现

图 3 描述了在第  $t$  次迭代、种群中一个粒子基于 SGSA 的 T-S 模型辨识算法的实现过程。

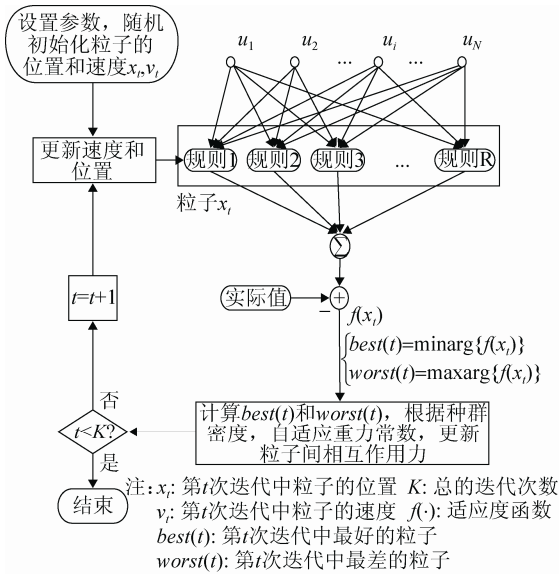


图 3 基于 SGSA 的 T-S 模型辨识算法的实现过程  
Fig. 3 The flowchart in SGSA identification of T-S model

### 4 仿真

为验证本文所提出算法的辨识精度和泛化能力, 选取静态非线性系统和 Box-Jenkins<sup>[8]</sup>系统进行实验仿真。静态非线性系统实质上是一个函数逼近问题, 它能有效的测试出算法的辨识精度。Box-Jenkins 系统是一个复杂的非线性系统, 常用来验证新模型方法的性能。

#### 4.1 静态非线性系统

选择如下双输入单输出的静态非线性系统作为辨识的对象:

$$y = (1 + u_2^{-2} + u_1^{-1.5})^2, 1 \leq u_1, u_2 \leq 5 \quad (15)$$

其中:  $u_1, u_2$  为输入;  $y$  为输出。

为方便和其他文献中的算法进行对比, 本仿真采用与文献[5]中相同的 50 组数据([1.400 0, 1.800 0, 3.702 9], [4.280 0, 4.960 0, 1.311 3], ... [2.220 0, 1.350 0, 3.387 2])。用 SGSA 辨识该系统的 T-S 模型, T-S 模型的规则数设为 6 条, SGSA 算法中种群的大小设置为 40 个, 迭代 2 000 次, 引力常数  $G_0'$  设为 6, 引力常数的衰减速率  $\alpha'$  设为 10, 勘探与开发的比例  $n$  设为 2, 公式(11)中距离的指数最大值和最小值  $Rp_{max}$ ,  $Rp_{min}$  分别设为 1.5 和 0.5。按图 2 的编码方式进行编码, 用 SGSA 迭代寻优, 得

到静态非线性系统的 T-S 模型参数见表 1。非线性静态系统实际输出与 T-S 模型输出对比见图 4。

表 1 静态非线性系统的 T-S 模型辨识参数

Tab. 1 The parameters in SGSA identification of T-S model in the static nonlinear system

TS 参数	规则 1	规则 2	规则 3
$c_{i1}$	3.570 4	1.727 9	1.346 7
$c_{i2}$	2.989 9	1.826 7	1.702 9
$\sigma_{i1}$	2.945 0	0.903 5	3.664 0
$\sigma_{i2}$	7.598 0	1.733 3	1.377 0
$\omega_{i1}$	2.5778	-0.266 9	-7.287 6
$\omega_{i2}$	-2.691 4	-4.059 2	2.902 2
$b_i$	4.371 3	5.323 6	2.159 5
TS 参数	规则 4	规则 5	规则 6
$c_{i1}$	4.158 1	3.261 8	3.910 6
$c_{i2}$	2.030 4	3.110 4	3.442 0
$\sigma_{i1}$	8.323 0	5.677 7	8.637 0
$\sigma_{i2}$	6.485 1	3.981 4	7.438 3
$\omega_{i1}$	-2.651 5	-3.476 2	1.094 6
$\omega_{i2}$	0.312 6	-10.502 2	5.068 9
$b_i$	16.101 0	10.153 8	14.221 8

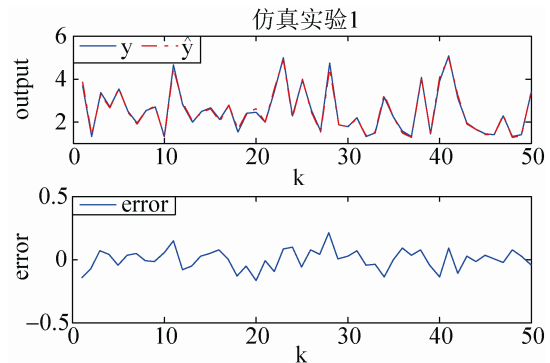


图 4 静态非线性系统实际输出与 T-S 模型输出对比  
Fig. 4 Comparison of T-S model and the real system for the static nonlinear system

图 4 中,  $y$  为非线性系统的实际输出,  $\hat{y}$  为辨识出的 T-S 模型输出, error 是辨识出的 T-S 模型输出与实际系统输出的误差, 反映 T-S 模型逼近对象的能力。从图中可看出, 用本文所提出的算法辨识出的模型误差在  $\pm 0.2$  之间, 精度较高。

表 1 是不同方法辨识静态非线性系统的结果对比。从表 2 中可知本文方法的均方差(MSE)为 0.006 1, 通过和其他文献中的方法进行对比发现, 提出的方法具有较高的辨识精度。



表 2 不同方法辨识静态非线性系统的结果对比  
Tab. 2 Comparison of different models for the static nonlinear system

模型	规则数	参数个数	MSE
Sugeno, et al. <sup>[5]</sup>	6	65	0.079 0
Kim, et al. <sup>[6]</sup>	3	21	0.009 0
Kung and Su <sup>[7]</sup>	3	21	0.019 6
Cheung, et al. <sup>[11]</sup>	6	42	0.006 4
本文模型	6	42	0.006 1

## 4.2 Box-Jenkins 系统

为进一步验证本文提出算法的泛化能力,选择 Box-Jenkins<sup>[8]</sup>系统作为辨识对象。该系统的数据来自于甲烷和空气混合物的燃烧过程,共有 296 组 ( $[-0.109\ 0, 53.800\ 0]$ ,  $[0, 53.600\ 0]$ , ...,  $[-0.262\ 0, 57]$ )。本实验中,把前 148 组数据作为训练数据来训练 T-S 模型,后 148 组数据作为测试数据测试模型的有效性,以此来验证本文算法的鲁棒性。

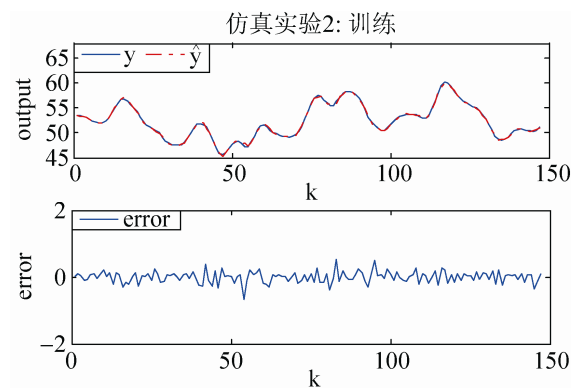
Box-Jenkins 系统在每个采样时刻  $k$ , 输入  $u(k)$  是甲烷气体的流量,输出  $y(k)$  是  $CO_2$  的浓度。在本实验,选择  $u(k)$ ,  $u(k-1)$  及  $y(k-1)$ ,  $y(k-2)$  作为 T-S 模糊模型的输入,  $y(k)$  为 T-S 模糊模型的输出。T-S 模型规则数设为 3 条,SGSA 算法的种群大小设置为 40 个,迭代 2 000 次,引力常数  $G_0$  设为 10,引力常数的衰减速率  $\alpha'$  设为 25,勘探与开发的比例  $n$  设为 3,式(11)中距离的指数最大值和最小值  $Rp_{max}$ ,  $Rp_{min}$  分别设为 1.5 和 0.5。用图 2 所示编码方式把 T-S 模型的前件参数和后件参数编码进一个粒子中,由前 148 组数据用 SGSA 算法训练 T-S 模型,最后得到的 T-S 模型的参数如表 3 所示。Box-Jenkins 系统前 148 组数据的实际输出与 T-S 模型输出对比如图 5(a)所示。再用后 148 组数据测试所得 T-S 模型,Box-Jenkins 系统后 148 组数据的实际输出与 T-S 模型输出对比如图 5(b)所示。

图 5(a)和(b)中,  $y$  为 Box-Jenkins 系统的实际输出,  $\hat{y}$  为辨识出的 T-S 模型输出, error 是辨识出的 T-S 模型输出与实际系统输出的误差。图 5(a)反映了前 148 组数据训练出的 T-S 模型与实际系统十分接近,并且通过图 5(b)可知,用后 148 组数据

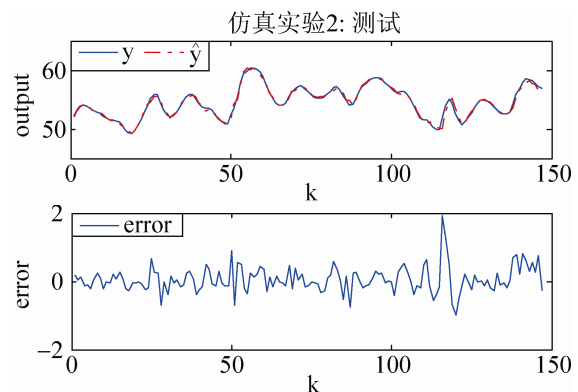
测试所得 T-S 模型,误差也较小,说明算法的鲁棒性较高。

表 3 Box-Jenkins 系统的 T-S 模型辨识参数  
Tab. 3 The parameters in SGSA identification of T-S model in the Box-Jenkins system

规则	TS 参数	参数值			
规则 1	$c_1$	56.806 8	49.728 8	2.827 0	1.613 8
	$\sigma_1$	19.278 2	21.460 2	37.418 3	23.557 3
	$\omega_1$	1.000 4	-0.247 9	-1.964 3	-0.684 3
	$b_1$	1.358 6			
规则 2	$c_2$	49.587 4	57.133 9	-0.976 4	-0.822 3
	$\sigma_2$	24.785 8	29.900 0	32.955 2	28.004 3
	$\omega_2$	2.899 4	-1.371 7	1.561 1	-1.264 1
	$b_2$	0.578 6			
规则 3	$c_3$	57.948 5	50.458 3	-0.681 2	0.452 6
	$\sigma_3$	32.316 3	43.700 7	29.755 2	22.768 3
	$\omega_3$	1.215 3	-0.580 9	1.381 5	0.010 6
	$b_3$	2.474 5			



(a) 训练数据上 Box-Jenkins 系统与辨识 T-S 模型输出对比



(b) 测试数据上 Box-Jenkins 系统与辨识 T-S 模型输出对比

图 5 辨识出的 T-S 模型在训练集和测试集上的输出与实际对比图

Fig. 5 Comparison of T-S model and the real system for the Box and Jenkins system using training and testing data.



表 4 是本文的模型和其他文献中模型的对比。本文模型的训练均方差(MSE)为 0.029 5, 测试均方差(MSE)为 0.141。与其他模型对比发现本文所提出的方法辨识出的 Box-Jenkins 系统的 T-S 模型比传统的方法拥有更高的精度。

表 4 不同方法辨识 Box-Jenkins 系统的结果对比  
Tab. 4 Comparison of different models for the Box-Jenkins system

模型	规则数	参数个数	训练 MSE	测试 MSE
Kim et al. <sup>[6]</sup>	6	38	0.034 0	0.244
Lin et al. <sup>[9]</sup>	5	44	0.071 0	0.261
Tsekouras <sup>[10]</sup>	6	91	0.022 0	0.236
Cheung et al. <sup>[11]</sup>	3	39	0.059 0	0.152
本文模型	3	39	0.029 5	0.141

## 5 结论

本文提出基于 SGSA 的 T-S 模型辨识算法。该方法把模型的结构前件参数和后件参数编码进一个粒子, 由于解决了辨识结构和后件参数过程中产生的不一致性问题, 该方法比传统的两步处理的方法拥有更高的精度。并且针对 GSA 算法容易陷入局部最优的缺陷, 提出基于群体密度的自适应策略和基于引力常数  $G$  的自适应策略, 有效抑制算法易陷入局部最优的缺点, 平衡了算法的勘探和开发能力。最后, 通过对静态非线性系统和 Box-Jenkins 系统的仿真, 证明了文中所提出的方法是有效的, 可以减少规则之间的冗余度, 优化模型结构, 提高辨识精度。

## 参考文献:

- [1] Takagi T, Sugeno M. Fuzzy identification of systems and its application to modeling and control [J]. IEEE Syst, Man, Cybern (S0018-9472), 1985, 15(1): 116-132.
- [2] Haiping D, Nong Z. Application of evolving Takagi-Sugeno fuzzy model to nonlinear system identification [J]. Applied Soft Computing (S1568-4946), 2008, 8(1): 676-686.
- [3] Li C, Zhou J, Xiao J. Parameters identification of chaotic system by chaotic gravitational search algorithm [J]. Chaos, Solitons & Fractals (S0960-0779), 2012, 45(4): 539-547.
- [4] Chao S L, Jiang Z Z, Jian X, et al. Hydraulic turbine governing system identification using T-S fuzzy model optimized by chaotic gravitational search algorithm [J]. Engineering Applications of Artificial Intelligence (S0952-1976), 2013, 26(9): 2073-2082.
- [5] Sugeno M, Yasukawa T. A fuzzy-logic-based approach to qualitative modeling [J]. IEEE Transactions on Fuzzy Systems (S1063-6706), 1993, 1(1): 7-31.
- [6] Kim E, Park M, Kim S, et al. A transformed input-domain approach to fuzzy modeling [J]. IEEE Transactions on Fuzzy Systems (S1063-6706), 1998, 6(4): 596-604.
- [7] Kung C, Su J. Affine takagi-sugeno fuzzy modeling algorithm by fuzzy c-regression models clustering with a novel cluster validity criterion [J]. IET Control Theory (S1751-8644), 2007, 1(5): 1255-1265.
- [8] Box G E P, Jenkins G M. Time Series Analysis Forecasting and Control [D]. Holden Day, San Francisco, CA, USA: John Wiley & Sons (Hoboken, New Jersey), 1970.
- [9] Lin Y, Cunningham I. A new approach to fuzzy-neural system modeling [J]. IEEE Transactions on Fuzzy Systems (S1063-6706), 1995, 3 (2): 190-198.
- [10] Tsekouras G. On the use of the weighted fuzzy c-means in fuzzy modeling [J]. Advances in Engineering Software (S0965-9978), 2005, 36 (5): 287-300.
- [11] Cheung N J, Ding X M, Shen H B. OptiFel: A convergent heterogeneous particle swarm optimization algorithm for takagi sugeno fuzzy modeling [J]. IEEE Transactions on Fuzzy Systems (S1063-6706), 2014, 22(4): 919-933.
- [12] Ding X M, Xu Z K, Cheung N J, et al. Parameter estimation of Takagi-Sugeno fuzzy system using heterogeneous cuckoo search algorithm [J]. Neurocomputing (S0925-2312), 2015, 151(3): 1332-42.
- [13] Rashedi E, Nezamabadi-Pour H, Saryazdi S. GSA: A Gravitational Search Algorithm [J]. Information Sciences (S0020-0255), 2009, 170(13): 2232-2248.
- [14] Khajehzadeh M, Taha M R, El-Shafie A. A modified gravitational search algorithm for slope stability analysis [J]. Engineering Applications of Artificial Intelligence (S0952-1976), 2012, 25(8): 1589-1597.