

## Mackey-Glass 方程参数反演的微分进化算法及其灵敏度分析

闵涛, 孙瑶, 邱李祯  
(西安理工大学理学院, 陕西 西安 710054)

**摘要:** 本文研究了 Mackey-Glass 混沌时滞微分方程的参数反演问题. 利用微分进化算法对其进行求解, 并对参数的灵敏度进行了详细的分析, 最后给出了数值模拟, 结果表明了该算法的可行性与有效性.

**关键词:** Mackey-Glass 方程; 微分进化算法; 参数反演; 时滞微分方程; 灵敏度

MR(2010) 主题分类号: 34K29; 49J15      中图分类号: O175.14

文献标识码: A      文章编号: 0255-7797(2016)04-0775-07

### 1 引言

时滞微分方程是具有时间滞后的微分方程, 它用于描述既依赖于当前状态也依赖于过去状态的发展系统. 对于一些事物的发展趋势, 若不考虑时滞, 则这样的模型是毫无意义的. 因此对时滞系统模型的研究是有十分重要意义的课题. 当对参数变化情况掌握之后, 其对现实生活有着指导性的作用, 它能更准确的刻画实际问题. 在流体力学、电子学、种群生态学、核反应堆动力学、经济学及现代控制理论等领域中存在着大量的时滞微分方程, 例如, 经济学中价值法则的作用是由于生产与消费之间的时滞造成的; 在生态系统中, 为了更真实地反映自然, 时滞常是一种不应忽略的因素等等<sup>[1]</sup>.

自从 Mackey 和 Glass<sup>[2]</sup> 发现时滞系统中的混沌现象以来, 时滞混沌系统引起了人们浓厚的兴趣. Mackey-Glass 混沌时滞微分方程具有如下的形式:

$$\frac{dy(t)}{dt} = -p_1 y(t) + \frac{p_2 y(t-\tau)}{1 + y^{p_3}(t-\tau)}, \quad (1)$$

初始条件为

$$y(t_0) = y_0.$$

此方程最初是用于描述白血球繁殖的模型, 后来成为了混沌理论中超混沌系统的典型代表, 其中  $y$  代表循环白血球的浓度,  $p_1, p_2, p_3$  是参数,  $\tau$  为时滞参数,  $y_0$  为初始值, 通常取  $p_1=0.1$ ,  $p_2=0.2$ ,  $p_3=10$ . Famer<sup>[3]</sup> 对 Mackey-Glass 方程的行为特性做了深入的研究, 当  $\tau > 17$  时表现出混沌性,  $\tau$  的值越大, 其混沌程度就越高. 估计时滞混沌系统的未知参数是混沌控制同步中必须解决的关键问题. 近几十年来, 关于参数反演的研究十分广泛, 众多学者相继提出了许多有效的方法, 这些方法<sup>[4,6]</sup> 解反演问题时大都是把反演转化为优化问题, 而微分进化

\*收稿日期: 2014-04-03      接收日期: 2014-07-31

基金项目: 国家自然科学基金重大项目 (51190093); 国家自然科学基金 (51179151).

作者简介: 闵涛 (1963-), 男, 陕西西安, 教授, 主要研究方向: 数学物理方程反问题, 科学工程计算与计算机模拟等.

算法是求解最优化问题的一种非常有效的方法,它属于全局最优化算法,具有鲁棒性强,收敛速度快,计算精度高的优点.据文献报告,微分进化算法在收敛速度和稳定性方面都超过了其他几种知名的随机算法,如退火单纯形算法 (annealed nelder and mend strategy, 简称 ANM)、自适应模拟退火算法 (adaptive simulate annealing, 简称 ASA)、演化策略 (evolution strategies, 简称 ES) 和随机微分方程 (stochastic differential equations). 本文以微分进化算法为基础,讨论了 Mackey-Glass 混沌时滞微分方程的参数反演问题,并对参数的灵敏度进行了详细的分析,最后给出了数值模拟,结果表明了该算法的可行性与有效性.

## 2 问题提出

考虑如下形式的 Mackey-Glass 混沌时滞微分方程

$$\frac{dy(t)}{dt} = -p_1 y(t) + \frac{p_2 y(t-17)}{1 + y^{10}(t-17)}, \quad (2)$$

初始条件为

$$y(t_0) = p_3, \quad (3)$$

其中  $p_1, p_2, p_3$  为参数. 如果  $p_1, p_2, p_3$  已知, 通过求解 (2) 便可得到该方程的变化规律. 而在实际问题中,  $p_1, p_2, p_3$  常常是未知的, 如何根据上述方程及其它已知信息确定未知参数, 便构成了一类参数识别反演问题.

## 3 算法分析和算法实现步骤

为了反演参数  $p_1, p_2, p_3$ , 一般要附加先验条件, 不妨取解在有限个时间点上的值是已知的, 例如,  $y(t_l)$  为已知,  $l = 1, 2, \dots, r$ . 这时, 参数反演问题就可以表示为求下面的最小值问题

$$\min_{(p_1, p_2, p_3) \in R^3} f(p_1, p_2, p_3) = \sum_{l=1}^r [y(t_l, p_1, p_2, p_3) - y(t_l)]^2, \quad (4)$$

其中  $y(t_l, p_1, p_2, p_3)$  是以  $p_1, p_2, p_3$  为参数求解原问题 (2) 获得的相应的计算值. 微分进化算法是通过种群内个体间的合作与竞争来实现对优化问题的求解, 其本质上是一种基于实数编码的具有保优思想的进化算法. 该算法得到了广泛关注, 因为它的实现技术简单, 在对各种测试问题的实验中表现优异. 对于连续变量的函数优化, 微分进化算法能更快、更稳定地收敛到问题的全局最优解<sup>[7,8]</sup>.

微分进化算法求解非线性最小二乘问题  $\min_{(p_1, p_2, p_3) \in R^3} f(p_1, p_2, p_3)$  的算法流程如下:

### 步骤 1 种群初始化

首先, 输入演化参数: 种群规模  $N$ , 最大演化代数  $\text{sub\_gen}$ , 交叉概率  $P_c$ , 交叉因子  $\alpha \in (0, 1)$ , 自变量的下界  $lb$  和上界  $ub$ ; 并令演化代数  $t = 0$ , 随机生成初始种群  $Z(0) = \{P_1, P_2, \dots, P_N\}$ , 其中  $P_i = \{p_{i1}, p_{i2}, p_{i3}\}$ .

随机生成初始种群的具体方法如下:

$$p_{ij}(t) = p_{\min,j} + \text{rand}_j(0, 1)(p_{\max,j} - p_{\min,j}), \quad i = 1, 2, \dots, N; \quad j = 1, 2, 3, \quad (5)$$

其中 3 为变量空间的维数,也就是所需要反演参数的个数;  $\text{rand}_j(0, 1)$  是  $[0, 1]$  区间满足正态分布的随机数;  $[p_{\min,j}, p_{\max,j}]$  是搜索空间第  $j$  个变量的下界与上界.

### 步骤 2 个体评价

计算每个个体  $P_i(t)$  的适应值, 其中评价函数  $f$  就是优化的目标函数  $f(p_1, p_2, p_3)$ .

### 步骤 3 变异

对于每一个目标个体向量  $P_i(t), i = 1, 2, \dots, N$ , 按照下面的方式产生变异个体  $V_i(t)$ , 其中  $V_i(t) = (v_{i1}, v_{i2}, v_{i3})$ , 具体方式如下:

$$v_{ij}(t) = p_{r_1j}(t) + xf \cdot (p_{r_2j}(t) - p_{r_3j}(t)), \quad (6)$$

其中  $r_1, r_2, r_3 \in \{1, 2, \dots, N\}$  并且  $i \neq r_1 \neq r_2 \neq r_3$ , 所以种群的大小  $N$  是大于或者等于 4 的整数. 参数  $xf$  是指控制差分变量  $(p_{r_2j}(t) - p_{r_3j}(t))$  的缩放因子, 它通常在  $[0, 2]$  上取值.

### 步骤 4 杂交

为了增加群体多样性, 微分进化算法引入了杂交操作, 最终来产生新的个体  $P'_i(t) = (p'_{i1}(t), p'_{i2}(t), p'_{i3}(t))$ ; 具体方法如下所示:

$$p'_{ij}(t) = \begin{cases} v_{ij}(t), & \text{如果 } \text{rand}[0, 1] < P_c \text{ 或者 } j = j_{\text{rand}}, \\ p_{ij}(t), & \text{其他,} \end{cases} \quad (7)$$

其中  $i = 1, 2, \dots, N$  且  $j = 1, 2, 3$ ;  $P_c$  是交叉概率,  $j_{\text{rand}}$  是  $[1, 3]$  之间的一个随机整数, 它保证了新个体  $P'_i(t)$  至少有一个分量来自变异个体  $V_i(t)$ , 从而避免与父个体  $P_i(t)$  相同. 微分进化算法常用的杂交方式有: 指数杂交、二进制杂交、二项式杂交.

### 步骤 5 选择

微分进化算法与其它智能优化算法一样, 也采用的是达尔文的适者生存定律, 即是适应值好的个体进入下一代, 适应值差的个体被淘汰. 微分进化算法采用的是贪婪选择策略. 选择过程如下:

$$P_i(t+1) = \begin{cases} P'_i(t), & \text{如果 } f(P'_i(t)) \leq f(P_i(t)), \\ P_i(t), & \text{其他,} \end{cases} \quad (8)$$

其中  $f(P_i(t))$  为个体  $P_i(t)$  的适应值, 由于微分进化算法采用一对一选择操作, 与排序或竞标赛选择相比更能维持群体的多样性, 接下来令  $t = t + 1$ .

### 步骤 6 终止检验

如果演化代数  $t = \text{sub\_gen}$ , 则满足终止条件并输出种群  $Z(t)$  中具有最小适应值的个体为最优解. 否则转到步骤 2.

## 4 灵敏度分析

灵敏度分析<sup>[9,11]</sup>是研究与分析一个系统(或模型)的状态或输出变化对系统参数或周围条件变化的敏感程度的方法. 在最优化方法中经常利用灵敏度分析来研究原始数据不准确或发生变化时最优解的稳定性. 通过灵敏度分析还可以决定哪些参数对系统或模型有较大的影响. 因此, 灵敏度分析几乎在对各种方案进行评价时都是很重要的<sup>[12]</sup>.

考虑如下 Mackey-Glass 混沌时滞微分方程

$$\frac{dy(t)}{dt} = -p_1y(t) + \frac{p_2y(t-17)}{1+y^{10}(t-17)}, y(t_0) = p_3, y \in R, P \in R^3, \quad (9)$$

也可写成如下的形式

$$y'(t) = f(t, y, P), y(t_0) = p_3, y \in R, P \in R^3, \quad (9)'$$

其中  $P = (p_1, p_2, p_3)$  为参数, 变量  $y$  对  $p_i$  的绝对灵敏度定义为

$$S_j(t) = \frac{\partial y(t, p)}{\partial p_j}, j = 1, 2, 3, \quad (10)$$

为了消除量纲的影响, 也可定义相对灵敏度. 其表达式为

$$S_j(t) = \frac{\partial y(t, p)}{\partial p_j} \frac{p_j}{y}, j = 1, 2, 3. \quad (11)$$

本文对这两种定义都给出了计算. 为了得到  $S_j(t)$  满足的关系式, 方程 (9) 两边对  $p_j$  求导

$$\frac{\partial y'(t, y, p)}{\partial p_j} = \frac{d}{dt} \left( \frac{\partial y(t, p)}{\partial p_j} \right) = \frac{d}{dt} S_j(t) = \frac{\partial f}{\partial y} \bullet \frac{\partial y}{\partial p_j} + \frac{\partial f}{\partial p_j},$$

其中  $j = 1, 2, 3$ . 上式中  $f$  是已知的,  $\frac{\partial f}{\partial y}$  和  $\frac{\partial f}{\partial p_j}$  可通过复合函数求偏导获得. 与方程 (9)' 联合, 可得到求灵敏度  $S_j(t)$  的微分方程组, 写成矩阵如下所示

$$Y'(t) = F(t, Y, P), Y(t_0) = Y_0, Y \in R^4, P \in R^3, \quad (12)$$

其中

$$Y(t) = \begin{pmatrix} y(t) \\ S_1(t) \\ S_2(t) \\ S_3(t) \end{pmatrix}, \quad F(t, Y, P) = \begin{pmatrix} f(t, y, p) \\ \frac{\partial f}{\partial y} S_1(t) + \frac{\partial f}{\partial p_1} \\ \frac{\partial f}{\partial y} S_2(t) + \frac{\partial f}{\partial p_2} \\ \frac{\partial f}{\partial y} S_3(t) + \frac{\partial f}{\partial p_3} \end{pmatrix},$$

$$(S_i(t) = \frac{\partial y(t, p)}{\partial p_i}, \quad S'_i(t) = \frac{\partial f}{\partial y} S_i(t) + \frac{\partial f}{\partial p_i}, i = 1, 2, 3,$$

$$Y(t_0) = (y(t_0), S_1(t_0), S_2(t_0), S_3(t_0))^T).$$

由于方程 (9)' 的初始条件为  $y(t_0) = p_3$ , 只与  $p_3$  有关, 与参数  $p_1, p_2$  无关, 便可得  $S_1(t_0) = S_2(t_0) = 0, S_3(t_0) = 1$ . (12) 式是一个微分方程组, 可通过龙格-库塔法求出其数值解, 从而可得到变量  $y$  对参数  $p_j$  在任意时刻  $t$  处的绝对灵敏度  $S_j(t)$ , 其相对灵敏度也可用类似的方法求得.

## 5 数值试验

### 5.1 参数反演

以上述的 Mackey-Glass 混沌时滞微分方程为例进行数值模拟. 为了反演参数  $p_1, p_2, p_3$  需要附加先验条件, 模拟时先给  $p_1, p_2, p_3$  赋真值, 取  $p_1 = 0.1, p_2 = 0.2, p_3 = 1.2$ , 可通过龙格-库塔法求解出  $y$  在任意时刻  $t$  处的数值解, 并把  $y$  的值作为已知条件来反求参数. 利用微

表 4-1: 参数反演结果

迭代次数	$p_1$ (真值为 0.1)	$p_2$ (真值为 0.2)	$p_3$ (真值为 1.2)	误差	运行时间
迭代 300 次	0.0962388588	0.1982489672	1.2118209891	0.00300180	459.99584s
迭代 350 次	0.1007455032	0.2004240655	1.1963364074	8.275834e-005	486.36747s
迭代 400 次	0.1005064401	0.2009273493	1.2000522778	3.030493e-005	504.12096s
迭代 500 次	0.1000314811	0.2000436008	1.1999607243	9.426289e-008	570.16506s
迭代 800 次	0.0999999769	0.1999999632	1.2000000496	4.507721e-014	725.25417s
迭代 1000 次	0.0999999999	0.2000000000	1.2000000001	3.115144e-019	863.10017s
迭代 2000 次	0.1	0.2	1.2	4.972597e-029	1315.7718 s

分进化算法反演参数时, 控制参数为交叉因子  $xf=0.9$ , 交叉概率  $P_c=0.3$ , 种群规模  $N=150$ . 其反演结果见表 4-1.

从表 4-1 中可以看出, 经过 1000 次迭代的反演值是  $p_1 = 0.0999999999$ ,  $p_2 = 0.2000000000$ ,  $p_3 = 1.2000000001$ , 与真值  $p_1 = 0.1$ ,  $p_2 = 0.2$ ,  $p_3 = 1.2$  相比精度很高, 误差只有  $3.115144e-019$ , 而经过 2000 次迭代的反演值几乎与真值相差无几, 误差只有  $4.972597e-029$ ; 另外一方面, 本文算法与其他智能算法相比, 其运行时间也相对较短. 这充分说明了本文算法在解决此类 Mackey-Glass 混沌时滞微分方程的参数反演问题时是可行的且十分有效的.

## 5.2 灵敏度分析

参数对每个变量的影响程度可以由计算灵敏度得到, 以上述的 Mackey-Glass 方程为例, 求其绝对灵敏度的方程组如下所示:

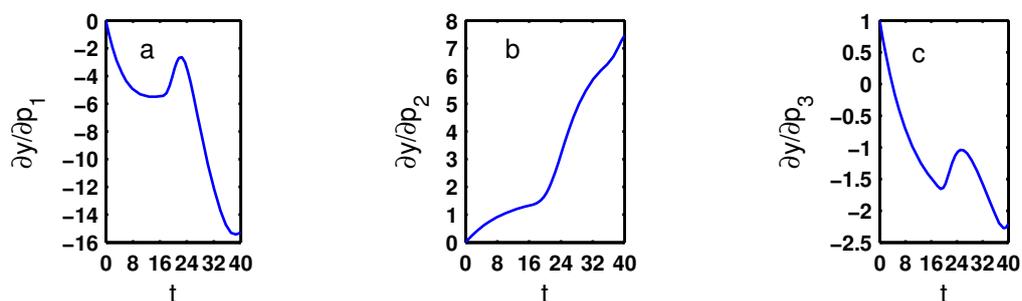
$$\begin{cases} y' = \frac{p_2 y(t-17)}{1+y^{10}(t-17)} - p_1 y(t), \\ S'_1 = \frac{p_2 S_1(t-17)[1-9y^{10}(t-17)]}{[1+y^{10}(t-17)]^2} - y(t) - p_1 S_1, \\ S'_2 = \frac{y(t-17)+y^{11}(t-17)+p_2 S_2(t-17)-9p_2 S_2(t-17)y^{10}(t-17)}{[1+y^{10}(t-17)]^2} - p_1 S_2, \\ S'_3 = \frac{p_2 S_3(t-17)[1-9y^{10}(t-17)]}{[1+y^{10}(t-17)]^2} - p_1 S_3, \end{cases} \begin{cases} y(0) = p_3, \\ S_1(0) = 0, \\ S_2(0) = 0, \\ S_3(0) = 1, \end{cases} \quad (13)$$

其中  $S_i = \frac{\partial y(t,p)}{\partial p_i}$ ,  $S'_i = \frac{\partial S_i}{\partial t}$ ,  $i = 1, 2, 3$ . (13) 式是一个时滞微分方程组, 通过龙格 - 库塔法求出其数值解, 从而可得到变量  $y$  对参数  $p_j$  在任意时刻  $t$  处的绝对灵敏度  $S_j(t)$ , 其相对灵敏度也可用类似的方法求得, 如图 1 所示.

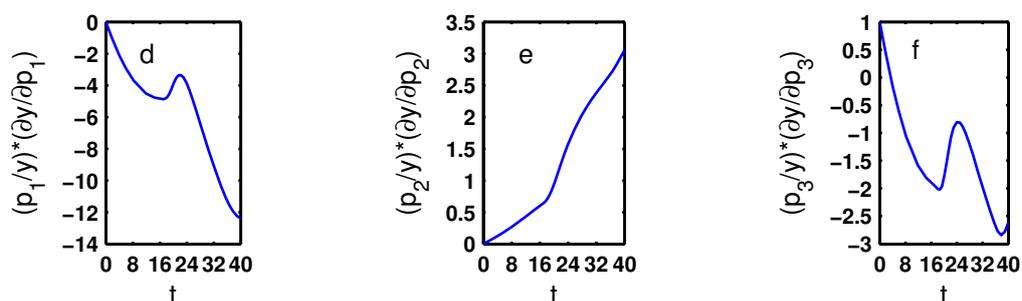
由图 a 和图 d 可以看出,  $y$  对  $p_1$  的绝对灵敏度及相对灵敏度均小于零, 这说明  $p_1$  的增加将会引起  $y$  的减小. 例如, 当  $t=0.5$  时, 通过计算可知  $\frac{\partial y}{\partial p_1}|_{t=0.5} = -0.5747$ , 也就是说  $t=0.5$  时  $p_1$  增加 10%, 将会导致  $y$  减少 5.747%. 由图 b 和图 e 可以看出  $y$  对  $p_2$  的绝对灵敏度及相对灵敏度均大于零, 这说明  $p_2$  的增加将会引起  $y$  的增加. 通过计算可知当  $t=19$  时,  $\frac{\partial y}{\partial p_2}|_{t=19} = 1.5314$ , 也就是说此时  $p_2$  增加 10%, 将会导致  $y$  增加 15.341%. 关于  $y$  对  $p_3$  的灵敏度也可以类似的分析.

## 6 结论

本文应用微分进化算法对 Mackey-Glass 混沌时滞微分方程的未知参数进行了反演, 同时对此类问题的灵敏度给出了具体的分析, 数值模拟的结果表明了该算法对解决此类时滞微



$a$  是  $y$  对  $p_1$  的绝对灵敏度     $b$  是  $y$  对  $p_2$  的绝对灵敏度     $c$  是  $y$  对  $p_3$  的绝对灵敏度



$d$  是  $y$  对  $p_1$  的相对灵敏度     $e$  是  $y$  对  $p_2$  的相对灵敏度     $f$  是  $y$  对  $p_3$  的相对灵敏度

图 1:  $y$  对  $p_1, p_2, p_3$  的绝对灵敏度 ( $a, b, c$ ) 及相对灵敏度 ( $d, e, f$ )

分方程参数反演的问题是可行而且十分有效的. 文中选择的数值算例是一个非线性方程和三个参数, 其实对多个非线性方程组成的方程组及多个参数本文方法也是可行的. 在微分进化算法中, 交叉因子  $xf$ , 交叉概率  $P_c$ , 种群规模  $N$  的选择对算法的收敛性有一定的影响, 经多次计算可知, 选取  $xf=0.9$ ,  $P_c=0.3$ ,  $N=150$  时反演效果最佳.

## 参 考 文 献

- [1] 刘桂荣. 时滞微分方程的周期正解及其在种群模型中的应用 [D]. 山西: 山西大学博士学位论文, 2007.
- [2] Mackey M C. Glass L. Oscillation and chaos in physiological control system[J]. Sci., 1977, 197: 287-289.
- [3] Farmer J D. Chaotic attractors of infinite-dimensional dynamical systems[J]. Phys. D.: Nonl. Phen., 1982.
- [4] Bjorck A. Numerical methods for least squares problems[M]. SIAM Phil., 1996.
- [5] Tarantola A. Inverse problem theory and methods for model parameter estimation[M]. Amsterdam: Elsevier, 2009.
- [6] Andreas Kirsch. An introduction to the mathematical theory of inverse problem [M]. New York: Springer-Verlag, 1996.
- [7] 陈子仪, 康立山. 多父体杂交演化算法求解约束优化问题 [J]. 武汉大学学报 (信息科学版), 2006, 31(5): 440-443.

- [8] Mezura-Montes E, Coello Coello C A. Adding a diversity mechanism to a simple evolution strategy to solve constrained optimization problems[C]. Canberra, Australia: the 2003 Congress on Evolutionary Computation (CEC'03), 2003.
- [9] Morris W. Hirsch, Stephen Smale, Robert L. Devaney, differential equations, dynamical systems, and linear algebra[M]. San Diego, CA: Academic Press, 2004.
- [10] Mark M, Meerscheart. 数学建模方法与分析 [M]. 北京: 北京机械工业出版社, 2005.
- [11] Maly T, Petzold L. Numerical methods and software for sensitivity analysis of differential-algebraic systems[J]. Appl. Numer. Math., 1996, 20: 57-79.
- [12] 闵涛, 成瑶, 胡刚, 毕妍妍. Lorenz 方程参数反演的 L-M 算法及灵敏度分析 [J]. 科技通报, 2012, 28(11): 19-21.
- [13] 胡健伟, 汤怀民. 微分方程数值方法 [M]. 北京: 科学出版社, 2007.

## DIFFERENTIAL EVOLUTION ALGORITHM OF MACKEY-GLASS EQUATION PARAMETER ESTIMATION AND SENSITIVITY

MIN Tao, SUN Yao, QIU Li-zhen

(School of Science, Xi'an University of Science and Technology, Xi'an 710054, China)

**Abstract:** In this paper, we study the Inverse problems about Mackey-Glass equation. By using the differential evolution algorithm, we solve the problem and the sensitivity of parameters is analyzed in detail. We give the numerical simulation, which shows the feasibility and effectiveness of the proposed algorithm.

**Keywords:** Mackey-Glass equation; differential evolution algorithm; parameter estimation; differential equation; sensitivity

**2010 MR Subject Classification:** 34K29; 49J15