

# Numerical Algorithms for Ordinary Differential Equations and Application

Sheng Huang, Zihui Jiang, Jiahui Li, Xiang Chi

College of Science, Shenyang Aerospace University, Shenyang Liaoning  
Email: synttt@163.com

Received: Nov. 23<sup>rd</sup>, 2019; accepted: Dec. 12<sup>th</sup>, 2019; published: Dec. 19<sup>th</sup>, 2019

---

## Abstract

In this paper, we mainly use MATLAB software to program the numerical solution of ordinary differential equation, such as Euler method and classical fourth-order Runge Kutta algorithm. Through the results of corresponding numerical example, we analyze the advantages and disadvantages of these two algorithms. Finally, we apply those algorithms to a concrete example; we solve it and compare the results.

## Keywords

Ordinary Differential Equations, Numerical Algorithm

---

# 常微分方程常用的数值算法及其应用

黄 盛, 姜紫惠, 李佳辉, 迟 祥

沈阳航空航天大学理学院, 辽宁 沈阳  
Email: synttt@163.com

收稿日期: 2019年11月23日; 录用日期: 2019年12月12日; 发布日期: 2019年12月19日

---

## 摘 要

本文主要利用了MATLAB软件对常用的常微分方程数值解法编程实现, 如欧拉法和经典的四阶龙格库塔算法, 通过相应的数值算例的结果, 分析了两种算法的优缺点。最后我们把算法应用到了一个具体的实例, 分别用欧拉法和经典的四阶龙格库塔法进行求解并对计算结果做了比较。

## 关键词

常微分方程, 数值算法

---



## 1. 常用的数值算法

常微分方程的理论和方法不仅在自然科学领域中应用非常广泛，而且在社会科学的各个领域有着越来越多的研究。但是很多时候，我们都没办法求出常微分方程的解析解，因而需要用数值算法去求出相应的数值解。常微分方程一般所用的求解初值问题的数值解法简单来说分为两类：单步法和多步法[1]-[6]。本文主要讨论了单步法中的两类方法。

考虑如下的初值问题：

$$\begin{cases} \frac{dy}{dx} = f(x, y), & x \in [a, b] \\ y(a) = y_0 \end{cases}$$

数值算法一般是：取步长  $h = \frac{b-a}{n}$ ，将  $[a, b]$  分成  $n$  等份，得到离散点  $a = x_0 < x_1 < \dots < x_n = b$ ，进而求解  $y(x)$  在这些离散点处的近似值  $y_1, y_2, \dots, y_n$ 。

### 1.1. 算法介绍

欧拉法的迭代格式为： $y_{i+1} = y_i + hf(x_i, y_i)$ ,  $i = 1, 2, \dots, n-1$ 。在已知初值  $y_0$  的情况下，我们便可以依次推算出：

$$\begin{aligned} y_1 &= y_0 + hf(x_0, y_0), \\ y_2 &= y_1 + hf(x_1, y_1), \\ &\vdots \\ y_n &= y_{n-1} + hf(x_{n-1}, y_{n-1}) \end{aligned}$$

经典的四阶龙格库塔算法的迭代格式为：

$$\begin{cases} y_{i+1} = y_i + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = f(x_i, y_i) \\ K_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_1\right) \\ K_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_2\right) \\ K_4 = f(x_i + h, y_i + hK_3) \end{cases}$$

### 1.2. 数值算例

我们分别用欧拉法和四阶的龙格库塔算法，利用 MATLAB 软件编程来计算如下的初值问题：

$$\begin{cases} y' = y - \frac{2x}{y}, & 0 < x < 1 \\ y(0) = 1 \end{cases}$$

得到结果如表 1 所示：

**Table 1.** Comparison of calculation results**表 1.** 计算结果对比表

离散点	欧拉算法近似解	四阶龙格库塔算法近似解	准确解
0.1000	1.1000	1.0954	1.0954
0.2000	1.1918	1.1832	1.1832
0.3000	1.2774	1.2649	1.2649
0.4000	1.3582	1.3416	1.3416
0.5000	1.4351	1.4142	1.4142
0.6000	1.5090	1.4832	1.4832
0.7000	1.5803	1.5492	1.5492
0.8000	1.6498	1.6125	1.6125
0.9000	1.7178	1.6733	1.6733
1.0000	1.7848	1.7321	1.7321

从表 1 可以看出, 四阶龙格库塔法在的精确度是要高于欧拉算法的, 但是欧拉算法较四阶的龙格库塔算法来说, 要更简单。

## 2. 算法的实际应用

在文献[7]中, 研究了三维的微生物连续发酵生产 1,3 丙二醇的微分方程动力系统如下:

$$\begin{cases} \dot{x}_1(t) = (\mu - D)x_1 \\ \dot{x}_2(t) = D(c_{s0} - x_2(t)) - q_2x_1(t) \\ \dot{x}_3(t) = q_3x_1(t) - Dx_3(t) \\ x_i(0) = x_{0i}, \quad i \in I_3 = \{1, 2, 3\}, \quad t \in (0, T) \end{cases}$$

其中:  $\mu = \mu_m \frac{x_2(t)}{x_2(t) + k_s} \prod_{i=2}^3 \left(1 - \frac{x_1(t)}{x_i^*}\right)$ ,  $q_2 = m_2 + \frac{\mu}{Y_2} + \Delta_2 \frac{x_2(t)}{x_2(t) + k_2}$ ,  $q_3 = m_3 + \mu Y_3 + \Delta_3 \frac{x_2(t)}{x_2(t) + k_3}$ 。

参数:  $\mu_m = 0.48$ ,  $m_2 = 2.20$ ,  $m_3 = -0.97$ ,  $Y_2 = 0.0082$ ,  $Y_3 = 67.69$ ,  $\Delta_2 = 28.58$ ,  $k_s = 0.28$ ,  $k_2 = 11.43$ ,  $k_3 = 15.50$ ,  $x_1^* = 10$ ,  $x_2^* = 2039$ ,  $x_3 = 939.5$ ,  $D = 0.23$ ,  $c_{s0} = 1076$  (具体的参数意义可以参见相关文献)。

我们用欧拉法来求解该系统, 建立迭代格式如下:

$$\begin{cases} x_{n+1,1} = x_{n,1} + h(\mu - D)x_{n,1} \\ x_{n+1,2} = x_{n,2} + h(D(c_{s0} - x_{n,2}) - q_2x_{n,1}) \\ x_{n+1,3} = x_{n,3} + h(q_3x_{n,1} - Dx_{n,3}) \end{cases}$$

参数为:  $D = 0.23$ ,  $c_{s0} = 1076$ ,  $x_{0,1} = 0.94$ ,  $x_{0,2} = 864.33$ ,  $x_{0,3} = 138.6$ , 步长  $h$  取 0.01, 计算结果如表 2 所示:

**Table 2.** Calculation results of Euler method**表 2.** 欧拉法计算结果

$T$	162.42	163.28	164.15	165.08	166.22	166.88	167.88	169.05	170.38
$x_1$	0.9400	0.9370	0.9336	0.9297	0.9245	0.9213	0.9164	0.9104	0.9035
$x_2$	864.330	859.839	856.306	853.451	851.002	850.015	849.009	848.439	848.403
$x_3$	138.600	143.346	147.187	150.363	153.222	154.453	155.837	156.860	157.422

我们用四阶的龙格库塔算法来求解该系统，建立迭代格式如下：

$$\begin{cases} \dot{x}_{n+1,1} = x_{n,1} + \frac{h}{6}(k_{11} + 2k_{12} + 2k_{13} + k_{14}) \\ \dot{x}_{n+1,2} = x_{n,2} + \frac{h}{6}(k_{21} + 2k_{22} + 2k_{23} + k_{24}) \\ \dot{x}_{n+1,3} = x_{n,3} + \frac{h}{6}(k_{31} + 2k_{32} + 2k_{33} + k_{34}) \end{cases}$$

核心代码如下：

```
k11 = f1(x1(i),x2(i),x3(i));
k21 = f2(x1(i),x2(i),x3(i));
k31 = f3(x1(i),x2(i),x3(i));
k12 = f1(x1(i)+h*k11/2,x2(i)+k21*h/2,x3(i)+h*k31/2);
k22 = f2(x1(i)+h*k11/2,x2(i)+k21*h/2,x3(i)+h*k31/2);
k32 = f3(x1(i)+h*k11/2,x2(i)+k21*h/2,x3(i)+h*k31/2);
k13 = f1(x1(i)+h*k12/2,x2(i)+k22*h/2,x3(i)+h*k32/2);
k23 = f2(x1(i)+h*k12/2,x2(i)+k22*h/2,x3(i)+h*k32/2);
k33 = f3(x1(i)+h*k12/2,x2(i)+k22*h/2,x3(i)+h*k32/2);
k14 = f1(x1(i)+h*k13,x2(i)+h*k23,x3(i)+h*k33);
k24 = f2(x1(i)+h*k13,x2(i)+h*k23,x3(i)+h*k33);
k34 = f3(x1(i)+h*k13,x2(i)+h*k23,x3(i)+h*k33);
x1(i+1) = x1(i)+h*(k11+2*k12+2*k13+k14)/6;
x2(i+1) = x2(i)+h*(k21+2*k22+2*k23+k24)/6;
x3(i+1) = x3(i)+h*(k31+2*k32+2*k33+k34)/6;
计算结果如表 3 所示：
```

**Table 3.** Calculation results of fourth-order classical Runge Kutta algorithm  
**表 3.** 四阶经典的龙格库塔算法的计算结果

$T$	162.42	163.28	164.15	165.08	166.22	166.88	167.88	169.05	170.38
$x_1$	0.9400	0.9371	0.9337	0.9297	0.9245	0.9214	0.9165	0.9105	0.9036
$x_2$	864.330	859.937	856.386	853.516	851.050	850.054	849.037	848.457	848.411
$x_3$	138.600	143.260	147.100	150.292	153.168	154.407	155.802	156.835	157.406

从两组计算结果可以看出，欧拉法和经典的四阶龙格库塔算法都能够对该系统进行求解，两组数据相差不大。因为系统本身和实验数据都是带有误差的，而且该系统没有解析解，所以没有对两组数据的精确度作对比。

### 3. 结论

本文主要研究了利用 MATLAB 软件对欧拉法和经典的四阶龙格库塔算法进行了编程并上机实现，通过数值算例的结果分析了两种数值算法的特点。最后，分别用欧拉算法和经典的龙格库塔算法求解了一个三阶的微分方程动力系统，从计算结果来看，两种方法都适合对该系统的求解。常微分方程的数值算法还有很多，以后会再讨论。

---

## 基金项目

沈阳航空航天大学大学生创新创业训练项目，项目编号：201810143271。

## 参考文献

- [1] 钟巍. 一种新的常微分方程初值问题数值解法[J]. 重庆文理学院学报(自然科学版), 2011(4): 15-18.
- [2] 林爽, 张杰. 常微分方程初值问题的基本数值解法分析[J]. 海南师范大学学报(自然科学版), 2012, 25(2): 119-121.
- [3] 霍晓程. 常微分方程数值解法的研究[J]. 佳木斯教育学院报, 2011(5): 167-167.
- [4] 孙美玲. 常微分方程数值解法的 Matlab 计算与可视比较[J]. 高教学刊, 2016(19): 60-61.
- [5] 余丽. 常微分方程数值解法的若干求解格式[J]. 鄂州大学学报, 2013(S1): 123-125.
- [6] 孙燕新, 李娟. 关于几种常用一阶常微分方程数值解法的导出[J]. 常州工程学院报, 2010(6): 71-74.
- [7] 李艳杰. 非线性时滞脉冲动力系统的辨识模型, 优化算法及应用[D]: [硕士学位论文]. 大连: 大连理工大学, 2005, 12.