

## lab2: 实验报告

学号: PB19111675 姓名: 德斯别尔

2021 年 12 月 10 日

### 1 三阶 Runge-Kutta 方法

Runge-Kutta 方法是用于非线性常微分方程的解的重要的一类隐式或显式迭代法。对课本 Page 149 三阶 Runge-Kutta 公式 (7.18)

$$\begin{cases} y_{n+1} = y_n + \frac{h}{9}(2k_1 + 3k_2 + 4k_3) \\ k_1 = f(x_n, y_n) \\ k_2 = f(x_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1) \\ k_3 = f(x_n + \frac{3}{4}h, y_n + \frac{3}{4}hk_2) \end{cases}$$

#### 1.1 理论证明: 证明该格式为三阶精度

将  $y(x_{n+1})$  在  $x_n$  点展成幂级数

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \frac{h^3}{3!}y'''(x_n) + O(h^4) \quad (1)$$

因为

$$\begin{aligned} y'(x_n) &= f(x_n, y(x_n)) \\ y''(x_n) &= f_x(x_n, y(x_n)) + f_y(x_n, y(x_n))f_y(x_n, y(x_n)) \\ y'''(x_n) &= f_{xx}(x_n, y(x_n)) + 2f_{xy}(x_n, y(x_n)) + f_{yy}^2(x_n, y(x_n)) \\ &\quad + f_x(x_n, y(x_n))f_y(x_n, y(x_n)) + f_y(x_n, y(x_n))f_y^2(x_n, y(x_n)) \end{aligned}$$

令

$$F = f_x(x_n, y(x_n)) + f(x_n, y_n) f_y(x_n, y(x_n))$$

$$G = f_{xx}(x_n, y(x_n)) + 2f(x_n, y_n) f_{xy}(x_n, y(x_n)) + f^2(x_n, y_n) f_{yy}(x_n, y(x_n))$$

则

$$y(x_{n+1}) = y(x_n) + hf(x_n, y(x_n)) + \frac{h^2}{2}F + \frac{h^3}{6}(Ff_y(x_n, y(x_n)) + G) + O(h^4) \quad (2)$$

对公式 (7.18) 中  $k_2, k_3$  在点  $(x_n, y_n)$  幂级数展开

则

$$k_2 = f(x_n, y(x_n)) + \frac{h}{2}F + \frac{h^2}{8}G + O(h^3)$$

$$k_3 = f(x_n, y(x_n)) + \frac{3h}{4}F + h^2\left(\frac{1}{6}(Ff_y(x_n, y(x_n))) + \frac{9}{32}G\right) + O(h^3)$$

将  $k_1, k_2, k_3$  带入式 (7.18) 中

$$y_{n+1} = y_n + hf(x_n, y_n) + \frac{h^2}{2}F + \frac{h^3}{6}(Ff_y(x_n, y_n) + G) \quad (3)$$

对比式 (3) 与式 (2), 两者相差  $O(h^4)$ , 又因为式 (2) 与式 (1) 等价, 所以式 (3) 与式 (1) 相差截断误差  $O(h^4)$  所以, 此格式具有三阶精度

## 1.2 程序实现: 对于常微分方程初值问题

$$\begin{cases} y'(x) = e^x \\ y(0) = 1 \end{cases}$$

计算  $x = 1$  处的误差绝对值 (精确解易得). 采用均匀剖分, 分别取  $N = 10, 20, 40, 80, 160$ , 完成如下误差-精度表.

N	10	20	40	80	160
error	0.000005942336	0.000000744335	0.000000093134	0.000000011647	0.000000001456
order	\	2.998575149722	2.999305486057	2.999656632008	2.999858342579

### 1.3 实验代码

实验代码如下，是使用 C 语言完成的。

```
#include<stdio.h>
#include<math.h>
#define _USE_MATH_DEFINES
//三阶龙格库塔
double runge(double a,double b,double y0,int N)
{
    double x=a,y=y0,K1,K2,K3;
    double h=(b-a)/N;
    for(int i=1;i<=N;i++)
    {
        K1=exp(x);
        K2=exp(x+h/2);
        K3=exp(x+3*h/4);
        y=y+h*(2*K1+3*K2+4*K3)/9;
        x+=h;
    }
    return y;
}
int main(void)
{
    double a=0,b=1,y0=1,Y[7],T[7],O[7];
    int N=10,i;
    for(i=0 ;N<=320;i++)
    {
        Y[i]=runge(a,b,y0,N);
        N*=2;
        T[i]=fabs(M_E-Y[i]);
        if(i!=0)
            O[i-1]=log(T[i-1]/T[i])/log(2);
    }
    N=10;
    for(i=0;N<320;i++)
    {
        printf("N=%d 时: y=%.12f error=%.12f order=%.12f\n",N,Y[i],T[i],O[i]);
        N*=2;
    }
    return 0;
}
```

## 1.4 实验总结

使用  $\text{\LaTeX}$  所呈现的结果虽然还不尽人意，但也已经让我十分惊喜并且喜欢上这种排版格式了，在以后的学习过程中我要多加使用  $\text{\LaTeX}$  来对这个工具更加了解。同时在计算机上使用计算方法的知识，这让我意识到在计算方法这门课程中学习到的知识是多么重要多么实用，我应该更认真地学习这门课程。