《数值分析实验报告》

列主元消去法
和四阶阿达梅斯预测-矫正方法

学　院　计算机科学与技术

专　业　计算机科学与技术

年　级　**2012 级 3 班**

学　号　**3012216083**

姓　名　王雨朦

时　间　**2015 年 6 月 20 日**

一、算法概述

1. 列主元消去法求线性方程组的解

高斯消去法是一种解线性方程组的基本解法，有消元和迭代两个过程，可以求出矩阵的秩和 N 个未知数的解。列主元消去法是高斯消去法的变形，出现的原因是为了避免小主元把计算误差扩大（因为计算中需要将其作为除数），与高斯的差异是用换行的方法选取绝对值最大的元素作为主元素，最终能有有效的避免误差的扩散，得到矩阵的秩和未知数的解。

2. 四阶阿达梅斯预测-矫正方法求微分方程的解

初始阶段用四阶龙格－库塔方法启动：

数值分析中，龙格－库塔法（Runge-Kutta）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。经典四阶龙格库塔法是龙格库塔法的家族中的一个成员，它非常常用以至于经常被称为“RK4”或者就是“龙格库塔法”。该方法主要是在已知方程导数和初值信息，利用计算机仿真时应用，省去求解微分方程的复杂过程。其初值问题表述如下：

$$y' = f(t, y), \quad y(t_0) = y_0$$

则，对于该问题 RK4 由如下方程给出：

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

其中，

$$\begin{aligned}
k_1 &= f\left(t_n, y_n\right) \\
k_2 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right) \\
k_3 &= f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right) \\
k_4 &= f\left(t_n + h, y_n + hk_3\right)
\end{aligned}$$

这样，下一个值(y_{n+1})由现在的值(y_n)加上时间间隔(h)和一个估算的斜率的乘积决定。该斜率是以下斜率的加权平均：

k1 是时间段开始时的斜率；

k2 是时间段中点的斜率，通过欧拉法采用斜率 k1 来决定 y 在点 tn＋h/2 的值；

k3 也是中点的斜率，但是这次采用斜率 k2 决定 y 值；

k4 是时间段终点的斜率，其 y 值用 k3 决定。

阿当姆斯方法是用多个点的斜率的加权平均来预测斜率，显示方法要求与阶数同样数目的前面的点，而隐式方法也要求阶数个点，但是包括自己在内。阿当姆斯预测——矫正方法是先用显示方法估值，再用隐式方法矫正。

$$y_{n+4} = y_{n+3} + h\left(\frac{55}{24}f(t_{n+3}, y_{n+3}) - \frac{59}{24}f(t_{n+2}, y_{n+2}) + \frac{37}{24}f(t_{n+1}, y_{n+1}) - \frac{3}{8}f(t_n, y_n)\right)$$

$$y_{n+3}=y_{n+2}+h\left(\frac{3}{8}f(t_{n+3},y_{n+3})+\frac{19}{24}f(t_{n+2},y_{n+2})-\frac{5}{24}f(t_{n+1},y_{n+1})+\frac{1}{24}f(t_n,y_n)\right)$$

步骤：先通过 RK 四阶方法，利用初始值，得出前三个点的值（第一个点已给），然后用 adams 显示方法求出第四个点，用隐式方法矫正，然后第五个第六个点同样预测矫正，直到达到要求的点的数目。

二、程序接口

double GaussElimination(double A[n+1][n+1],double b[n+1]);
设 Ax = b. A 为系数矩阵，b 为结果向量，消元结果冲掉 A，乘数冲掉主元下面那些数，计算值 x 冲掉 b，返回行列式的值 det。

double R_K(double x,double y,double h);
龙格库塔单步法

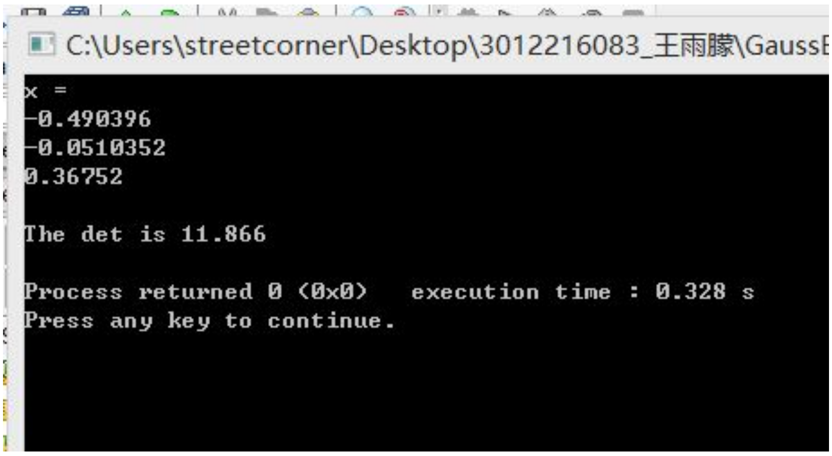
double Adams1(double *px, double* py, double h)
阿达姆斯预测法（显示方法）

double Adams2(double *px, double* py, double h)
阿达姆斯矫正法（隐式方法）

三、示例和运行结果

1. 列主元消去法求线性方程组的解

```
double A[n+1][n+1] = {
                    {0.000,0.000,0.000,0.000},
                    {0.000,0.001,2.000,3.000},
                    {0.000,-1.000,3.712,4.623},
                    {0.000,-2.000,1.072,5.643},
          };
double b[n+1] = { 0.000, 1.000, 2.000, 3.000 }
```



输出第一行为解向量，第二行为行列式的值。

2. 四阶阿达梅斯预测-矫正方法求微分方程的解

```
double f(double x, double y){
    return -y+x+1;
}
```

输入：步长和初值

输出：原函数的数值解

```
C:\Users\streetcorner\Desktop\3012216083_王雨朦\Ada
Please input h:0.2
Please input x[0],y[0]:0 1
x[0] = 0  y[0] = 1
x[1] = 0.2  y[1] = 1.0154
x[2] = 0.4  y[2] = 1.06426
x[3] = 0.6  y[3] = 1.14052
x[4] = 0.8  y[4] = 1.2549
x[5] = 1  y[5] = 1.39461
x[6] = 1.2  y[6] = 1.55317
x[7] = 1.4  y[7] = 1.72482
x[8] = 1.6  y[8] = 1.90553
x[9] = 1.8  y[9] = 2.09242
x[10] = 2  y[10] = 2.28354
x[11] = 2.2  y[11] = 2.47752
x[12] = 2.4  y[12] = 2.67344
x[13] = 2.6  y[13] = 2.87068
x[14] = 2.8  y[14] = 3.06881
x[15] = 3  y[15] = 3.26755
x[16] = 3.2  y[16] = 3.46669
x[17] = 3.4  y[17] = 3.66611
x[18] = 3.6  y[18] = 3.86572
x[19] = 3.8  y[19] = 4.06546

Process returned 0 (0x0)   execution time : 6.378 s
微软拼音 半 : to continue.
```

四、感想

程序本身其实并不难，通过这次手动编程，对两个算法的原理的了解更深的一步，也查阅了相关的知识和内容，知识得到了拓展，收获颇丰。