代码 – 微分方程数值解

笔记本：	我的第一个笔记本		
创建时间：	2017/5/10 11:44	更新时间：	2017/5/10 11:50
URL：	https://iclass.bupt.edu.cn/webapps/blackboard/content/listContent.jsp?course...		

打开快速链接

快速链接

页面标志

内容大纲

键盘快捷键

注销

全局菜单

理学院 罗畅

活动更新

顶框表格

顶框选项卡

我的机构 My_institution 4的选项卡1	**课程 Courses 4的选项卡2 (活动选项卡)**	社区 Community 4的选项卡3	帮助中心 4的选项卡4

“选项卡列表”表格

当前位置

.

1. **H 微分方程数值解**

1. **课程资料**

1. **代码**
2. .

.
←

菜单管理选项

-
-
- .

.
微分方程数值解

H
»
→

- 课程主页.
- 课程公告.

-

  .
- A 课程导论.
- B 初值问题.
- C 椭圆方程.
- D 抛物方程.
- E 双曲方程.
- F 有限元法.

-

  .
- 作业习题.
- 实验日记.
- 学习风采.

-

  .
- 课程介绍.
- 教师介绍.
- 教学大纲.

- 课程资料.
-

  .
- 课堂讲义.
- 延伸学习.
- 课外阅读.
-

  .
- 小组论坛.
- 课程小组.
- 发表博客.
-

  .
- 我的成绩.
- 教师警报.
-

  .
- 帮助.
- 工具.
-

  .
-

  .

我的小组

- **2014级本科组→→**

代码

.

内容

- **单步法**

  .

考虑如下常微分方程初值问题

$$\begin{cases} u' = -30u, 0 \le t \le 1 \\ u(0) = 1 \end{cases}$$

(1)用 Matlab 求出解析解；

(2)取 h=0.1, 0.05, 0.01, 分别用 Euler 法、改进 Euler 法和四阶的 Runge- Kutta 格式求数值解，进行收敛性数值分析，并由此讨论上述格式的稳定性.

```
clear all;
dufun = @(t,u)(-30*u);
h = 0.1;
% h=0.05;
% h=0.01;
t0=0:h:1;
u0=exp(-30*t0)';
[t1,u1] = ODEeuler(dufun,[0,1],1,h);
[t2,u2] = ODEetr(dufun,[0,1],1,h);
[t3,u3] = ODEKutta3(dufun,[0,1],1,h);
[t4,u4] = ODERK4(dufun,[0,1],1,h);

plot(t1,u1,'-*',t2,u2,'-d',t3,u3,'-o',t4,u4,'-+',t0,u0)
legend('u1','u2','u3','u4','exact solu');

err=[norm(u1-u0) norm(u2-u0) norm(u3-u0) norm(u4-u0)];
vN=[1/h+1];
for n=1:8
    h=h/2;
    t0=0:h:1;
    u0=exp(-30*t0)';
    [t1,u1] = ODEeuler(dufun,[0,1],1,h);
    [t2,u2] = ODEetr(dufun,[0,1],1,h);
    [t3,u3] = ODEKutta3(dufun,[0,1],1,h);
    [t4,u4] = ODERK4(dufun,[0,1],1,h);
    err=[err;norm(u1-u0) norm(u2-u0) norm(u3-u0) norm(u4-u0)];
    vN=[vN; 1/h+1];
    n=n+1;
end



subplot(2,2,1)
loglog(vN,err(:,1),'-*',vN,vN.^(-1))
subplot(2,2,2)
loglog(vN,err(:,2),'-d',vN,vN.^(-2))
subplot(2,2,3)
loglog(vN,err(:,3),'-o',vN,vN.^(-3))
subplot(2,2,4)
loglog(vN,err(:,4),'-+',vN,vN.^(-4));
```

- **多步法**

例3.4.1. 分别用四阶$Adams$显式方法和四阶$Adams$ 预测-校正方法求解求解常微分方程初值问题

$$\begin{cases} u' = (1 - tcosu)cot(u), 0 < t \le 2; \\ u(0) = arccos(1/2). \end{cases} \tag{3.4.20}$$

该问题的精确解为 $u = arccos\left(\frac{1}{t+1+e^t}\right), t \in [0, 2]$.

主程序 main.m：

```
clear all;
dufun =   @(t,u)(1-t.*cos(u)).*cot(u);
h = 0.01;
t0=0:h:2;
u0=acos(1./(t0+1+exp(t0)));
[t1,u1] = ODEAdams4x(dufun,[0,2],acos(1/2),h);
plot(t1,u1,'-*',t0,u0)
legend('u1','exact solu');
```

函数 ODEAdams4x.m：

```
function [t, u]=ODEAdams4x( dufun, tspan, u0, h)
% 用途：函数ODEAdams4x 用4 阶Adams 显式格式求解常微分方程
% 初值问题u'=f(t,u),u(t0)=u0.
% 格式：[t,u]=ODEAdams4x(dufun,tspan,u0,h)
% 输入中dufun 为函数f(t,u),
% tspan 为求解区间[tsan(1) tspan(2)],
% u0 为初始值u(t0),h 为步长.
% 输出中t 为自变量离散得到的节点向量,
% u 为对应的节点函数向量.
t = tspan(1):h:tspan(2);
u(1) = u0 ;
num = length(t)-1;
for n=1:3
    u(n+1) = rk4(dufun, t(n), u(n), h);
end
fn = feval(dufun, t(1:4), u(1:4));
beta = [-9; 37 ; -59 ; 55 ;] / 24;
for n=4:num
    u(n+1) = u(n) + h*fn*beta;
    fn=[fn(2:4) feval(dufun, t(n+1) ,u(n+1))];
end
t = t .';
u = u .';
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function u = rk4(dufun, t, u, h)
K1 = feval(dufun, t, u) ;
KK = u+h*K1/2;
K2 = feval(dufun, t+h/2 ,KK) ;
KK = u+h*K2/ 2 ;
K3 = feval(dufun, t+h/2 ,KK) ;
KK = u+h*K3 ;
K4 = feval(dufun, t+h ,KK) ;
u = u + h*(K1+2*K2+2*K3+K4)/6;
```

- **求解两点边值问题的有限差分法**

.

用有限差分法编程求如下两点边值问题的数值解，并对数值解结果进行分析说明。（已知解析解为 $u(x)=\cos(\pi x)$。）

$$
\begin{cases}
-u''-2u'+3u=(\pi^2+3)\cos(\pi x)+2\pi\sin(\pi x), 0<x<2 \\
u(0)=1 \\
u(2)=1
\end{cases}
$$

```
function [x, U]=two_point(a,b,ua,ub,coef, f, n)
% 用途：函数two_point用有限差分法求解两点边值问题
%          c2 u'' + c1 u' + c0 u = f(x)
% 格式：[x, U]=two_point(a,b,ua,ub,coef, f, n)
% 输入：a, b: 左右两个端点
%          ua, ub : a，b两点的Dirichlet边值条件
%          coef ： 上述微分方程中对应项系数
%          f: 右端函数 f(x)
%          n: 网格剖分数目
% 输出：x: x(1), x(2), ..., x(n-1) 为网格节点
%          U: U(1), U(2), ..., U(n-1) 为对应的节点函数值

h=(b-a)/n;
h2=h*h;
c2=coef(1);
c1=coef(2);
c0=coef(3);

A = sparse(n-1,n-1);
F = zeros(n-1,1);

for i = 1:n-2,
      A(i,i)=-2*c2/h2+c0;
      A(i+1,i)=c2/h2-c1/(2*h);
      A(i,i+1)=c2/h2+c1/(2*h);
end
A(n-1,n-1)=-2*c2/h2+c0;
```

```matlab
for i = 1:n-1
    x(i)=a+i*h;
    F(i)=feval(f,x(i));
end

F(1)=F(1)-ua*(c2/h2-c1/(2*h));
F(n-1)=F(n-1)-ub*(c2/h2+c1/(2*h));

U=A\F;
return


%%%%%%%%%%%%%%%%% 右端函数 f(x) %%%%%%%%%%%%%%%%%%%
function f=f(x)
f=(pi*pi+3)*cos(pi*x)+2*pi*sin(pi*x);


%%%%%%%%%%%%%%%%% 主程序 main.c %%%%%%%%%%%%%%%%
clear; close all
% Input
a = 0; b=2; n=40;
ua=1;    ub=1;
coef=[-1,-2,3];
[x,U]=two_point(a,b,ua,ub,coef,'f',n);

u=cos(pi*x)';

plot(x,U,'-o', x,u);

% plot error
figure(2); plot(x,U-u)
norm(U-u,inf)
```