MA 512 Numerical Differential Equations

HW 7

Yuchen Dong

Problem 1:

(a) To find the absolute stable region for midpoint method, equivalently, find the eigenvalues of $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$. The characteristic equation is $\lambda^2 + 1 = 0$, which gives us the two eigenvalues $\pm i$. Thus, the midpoint method is absolutely stable.

(b)

Step Sizes	Maximum errors
0.1	1.658782×10^{-1}
0.01	1.650706×10^{-3}
0.001	1.650683×10^{-5}

Table 1: The Damped mechanical oscillator when $D = 0.1$

Problem 2:

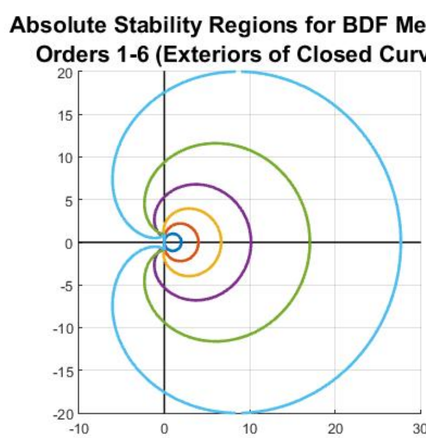

Figure 1: Absolutely-Stability Regions for BDF methods of orders 1-6

Problem 3:

Comments:

Methods	Number of successful steps	Number of unsuccessful attempts	Number of function evaluations
ode45	231	0	1387
ode113	139	0	279
ode15s(NDF)	387	0	776
ode15s(BDF)	389	2	784

Table 2: The Damped mechanical oscillator when $D = 0.1$

Methods	Number of successful steps	Number of unsuccessful attempts	Number of function evaluations
ode45	121	0	727
ode113	108	0	217
ode15s(NDF)	240	0	482
ode15s(BDF)	243	2	492

Table 3: The Damped mechanical oscillator when $D = 1$

Methods	Number of successful steps	Number of unsuccessful attempts	Number of function evaluations
ode45	121	5	757
ode113	217	8	443
ode15s(NDF)	192	1	388
ode15s(BDF)	200	2	406

Table 4: The Damped mechanical oscillator when $D = 10$

Methods	Number of successful steps	Number of unsuccessful attempts	Number of function evaluations
ode45	663	42	4231
ode113	1230	257	2718
ode15s(NDF)	177	2	360
ode15s(BDF)	187	2	380

Table 5: The Damped mechanical oscillator when $D = 100$

Problem 4:
Statistics for ode15s:
303 successful steps, 91 failed attempts, 963 function evaluations, 35 partial derivatives, 134 LU decompositions, 822 solutions of linear systems.
Statistics for ode113:

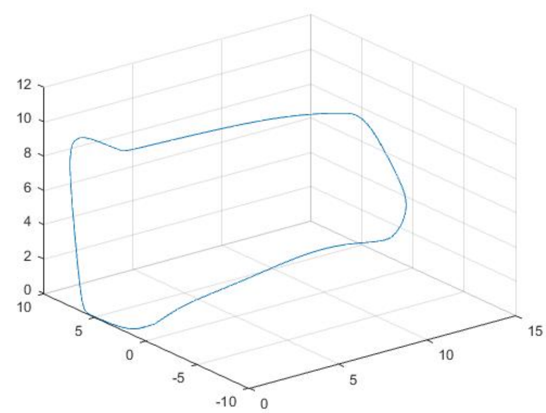

Figure 2: logs of the 3 solutions of Oregonator ODE using ode15s

4.64683×10^6 successful steps, 633604 failed attempts, 9.92726×10^6 function evaluations. Elapse time is 545.031629 seconds.

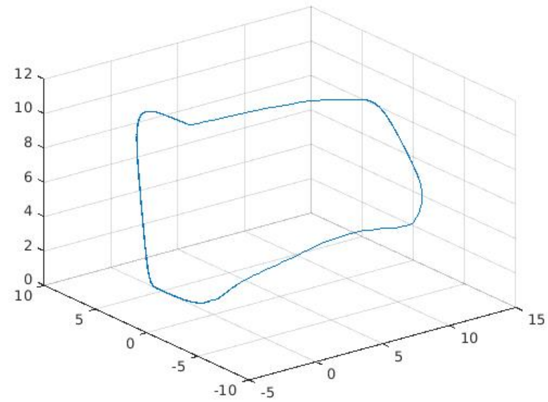

Figure 3: logs of the 3 solutions of Oregonator ODE using ode113

Code:

```
% MA 512 Numerical Differential Equations
% HW 7 Problem 1(b)
% The midpoint method of undamped ocsillator ode
clear all
clc
%% Initial part
h = [0.1, 0.01, 0.001];
a = 0;
b = 100;
A = [0,1;-1,0];
y(:,:,1) = [1,0]'; % Initial value
%% Midpoint method
for i = 1:length(h)
t = (a:h(i):b);
% To begin, use forward method to obtain the solution at x = h;
y(:,:,2) = y(:,:,1) + h(i)*A*y(:,:,1);
% Midpoint method
for j = 2:length(t)-1
y(:,:,j+1) = y(:,:,j-1) + 2*h(i)*A*y(:,:,j);
end
% Exact solution
for k = 1: length(t)
yext(:,:,k) = [cos(t(k)), -sin(t(k))]';
end
err(i) = max(max(y - yext)); % Maximum Error
fprintf('The error is %d, when h is %d.\n',err(i),h(i));
end

% MA 512 Numerical Differential Equations
% HW 7 Problem 3
% The damped mechanical oscillator function in vector form
function yp = fHW7P3(t,x)
yp = zeros(2,1);
yp(1) = x(2);
yp(2) = -x(1)-100*x(2);
end

% MA 512 Numerical Differential Equations
% HW 7 Problem 3
% The damped mechanical oscillator
% Using ode45, ode113, ode15s
```

```
%% Now illustrate setting tolerances
clear all
clc
y_10 = 1; % Initial value of x
y_20 = 1; % Initial value of x'
t_0 = 0;
t_f = 20;

% Set the error tolerances to 1e-8 in structure "myopts".
myopts = odeset('RelTol',1e-8,'AbsTol',1e-8,'Jacobian',[0 1;-1 -100],'Stats','on');
% Set  ode15s with BDF
myopts1 = odeset('RelTol',1e-8,'AbsTol',1e-8,'Jacobian',[0 1;-1 -100],'BDF','on','Stats','on');
myopts2 = odeset('RelTol',1e-8,'AbsTol',1e-8,'Stats','on'); % ode45
% Call ode45
fprintf('Method: ode45\n');
[tvals1,yvals1] = ode45(@fHW7P3,[t_0 t_f],[y_10 y_20],myopts2);
% Call ode113
fprintf('Method: ode113\n');
[tvals2,yvals2] = ode113(@fHW7P3,[t_0 t_f],[y_10 y_20],myopts);
% Call ode15s with BDF
fprintf('Method: ode15s with BDF\n');
[tvals3,yvals3] = ode15s(@fHW7P3,[t_0 t_f],[y_10 y_20],myopts1);
% Call ode15s with NDF
fprintf('Method: ode15s with default NDF\n');
[tvals4,yvals4] = ode15s(@fHW7P3,[t_0 t_f],[y_10 y_20],myopts);

% MA 512 Numerical Differential Equations
% HW 7 Problem 4
% The Oregonator IVP
clear all
clc
%% Now illustrate setting tolerances using ode15s options

y_10 = 1; % Initial value of y_1'
y_20 = 1.8; % Initial value of y_2'
y_30 = 1.8; % Initial value of y_3'
t_0 = 0;
t_f = 290;

% Set the default error tolerances in structure "myopts".
 myopts = odeset('Stats','on');
% Call ode15s
[tvals,yvals] = ode15s('fHW7P4',[t_0 t_f],[y_10 y_20 y_30],myopts);

nsteps = size(tvals,1); % Number of steps

fprintf('\n nsteps = %d\n', nsteps);

figure(1);
plot3(log(yvals(:,1)),log(yvals(:,2)),log(yvals(:,3)));
grid on

tic
% Call ode113
[tvals113,yvals113] = ode113('fHW7P4',[t_0 t_f],[y_10 y_20 y_30],myopts);
toc
nsteps113 = size(tvals113,1); % Number of steps

fprintf('\n nsteps = %d\n', nsteps113);

figure(2);
plot3(log(yvals113(:,1)),log(yvals113(:,2)),log(yvals113(:,3)));
grid on
```