Some Comments on Homework/Project Assignments of MA402 (title)

Zhilin Li, ID: 000-00-0000

Abstract

The format of homework/project assignments of MA402 will be emphasized below. Please read and pay attention to it.

1 Some important facts about MA402

- This is a three credits course.
- There are no in-class tests.
- This is a reading/writing requirement in this course that will be reflected in the homework/project assignments.
- The homework should be done individually. The projects/lab. reports can be done as a team work.
- Tardiness (absence, late arrival, early departure, sleeping in class etc.) will affect your grade.
- Late homework will be accepted up to a time, usually the end of the same week before I have graded the homework.

2 Specific comments on homework # 2.

- An algorithm is a description that can be understood by the readers with reasonable background. A computer code is *not* an algorithm but it is a implementation of a particular algorithm. An example of an algorithm is

$$u_i^{k+1} = \Delta t F_i^k + \alpha(u_{i-1}^k + u_{i+1}^k) + (1-2\alpha)u_i^k, \quad i = 2, 3, \cdots, n, \quad k = 0, 1, \cdots \tag{1}$$

for modeling the heat diffusion in a thin rod with little insulation. We also need to state the time step constraint

$$0 < \Delta t < \frac{h^2 \rho c}{2\kappa} \tag{2}$$

to complete the algorithm.

- The stability condition for the heat diffusion in a wire with little insulation is

$$1 - 2\alpha - \frac{2c_{sur}\Delta t}{\rho c} > 0, \quad \text{and} \quad \alpha > 0 \tag{3}$$

which can be found on the Notes just before Section 1.3.4. Solve for Δt we get

$$0 < \Delta t < \frac{h^2 \rho c}{2\kappa + 2c_{sur}/r}. \tag{4}$$

Therefore, as we increase c_{sur}, we should also decrease the time step size Δt. The maximum iterations is determined from $k_{max} = \frac{t_{final}}{\Delta t}$. Many students got unstable solutions when c_{sur} is large because they did not change the time step size Δt.

- How to present your simulations? You should label your plots using *xlabel, ylabel, zlabel, title*. For example:

```
xlabel('x'); ylabel('time'); zlabel('Temperature')
title('Fig.1, n=20, rho=1, cond=0.01, dt=0.001, csur=0.1')
```

It is advised to itemize your tests, specify all the parameters in the tests, and explain the results, for example

  - Test 1: The parameters are $n = 20, \rho = 1, c = 0.01, \Delta t = 0.001, c_{sur} = 0.1$. The computed solution is plotted in Fig.1. We can see the temperature increases with time because the source term and the zero boundary condition. The solution gradually approaches to the steady state solution, ...
  - Test 2: Now we set $c_{sur} = 0.01$ and the rest parameters are the same as in Test 1. The computed solution is plotted in Fig.2, ...
  - Test 3: ...

- If you are not clear with the assignments, you should ask the instructor (zhilin@math.ncsu.edu) or Jordan E. Massad (jemassad@unity.ncsu.edu).

3 Summary

This course requires some reading, writing, and programming. It should not be difficult if one spends reasonable time. However, one may fail this class if he/she does want to invest reasonable time for a three credits class.

References

[1] R. White, *Computational Mathematics: Models, Methods and Analysis*, NCSU, 2001

A A sample computer code for HW#3.

```
%  Heat Diffusion in a Thin Insulated Wire

clear all; close all;
L = 1.0;                             %  Length of the Wire
tfinal = 1;                          %  Final Time
n = 40.; dx = L/n;                   %Number of Space Steps
rho = 1.; cond = 1/(pi*pi); spheat = 1.0;    % Material parameters
dt = dx*dx*rho*spheat/(cond*2.1);
maxk = fix(tfinal/dt),                %  Number of Time Steps
b = dt/(dx*dx); a = cond/(spheat*rho); d = a*b;

%  Initial Temperature
for i = 1:n+1
        x(i) = (i-1)*dx;
        u0(i) = sin(pi*x(i));
end

plot(x,u0); hold

t = 0;
for k=1:maxk                  % Time Loop
   u1(1) = 0;
   u1(n+1) = 0;
   for i=2:n;                 % Space Loop
      u1(i) = 0.01*dt/(spheat*rho)+(1-2*d)*u0(i) + d*(u0(i-1)+u0(i+1));
   end
   u0 = u1;   plot(x,u1)
   t = t + dt;
end

for i=1:n+1
  err(i) = u1(i)- uexact(x(i),t);
end

  figure(2); plot(x,err)
  e2 = [norm(err,1),norm(err,2),norm(err,inf)],
```

B uexact.m

```
function y = uexact(x,t)
    y = exp(-t)*sin(pi*x);
```