

# Lecture 34

## Finite Difference Method – Nonlinear ODE

### Heat conduction with radiation

If we again consider the heat in a metal bar of length  $L$ , but this time consider the effect of radiation as well as conduction, then the steady state equation has the form

$$u_{xx} - d(u^4 - u_b^4) = -g(x), \quad (34.1)$$

where  $u_b$  is the temperature of the background,  $d$  incorporates a coefficient of radiation and  $g(x)$  is the heat source.

If we again replace the continuous problem by its discrete approximation then we get

$$\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} - d(u_i^4 - u_b^4) = -g_i = -g(x_i). \quad (34.2)$$

This equation is nonlinear in the unknowns, thus we no longer have a system of linear equations to solve, but a system of nonlinear equations. One way to solve these equations would be by the multivariable Newton method. Instead, we introduce another iterative method.

### Relaxation Method for Nonlinear Finite Differences

We can rewrite equation (34.2) as

$$u_{i+1} - 2u_i + u_{i-1} = h^2 d(u_i^4 - u_b^4) - h^2 g_i.$$

From this we can solve for  $u_i$  in terms of the other quantities:

$$2u_i = u_{i+1} + u_{i-1} - h^2 d(u_i^4 - u_b^4) + h^2 g_i.$$

Next we add  $u_i$  to both sides of the equation to obtain

$$3u_i = u_{i+1} + u_i + u_{i-1} - h^2 d(u_i^4 - u_b^4) + h^2 g_i,$$

and then divide by 3 to get

$$u_i = \frac{1}{3} (u_{i+1} + u_i + u_{i-1}) - \frac{h^2}{3} (d(u_i^4 - u_b^4) - g_i).$$

Now for the main idea. We will begin with an initial guess for the value of  $u_i$  for each  $i$ , which we can represent as a vector  $\mathbf{u}^0$ . Then we will use the above equation to get better estimates,  $\mathbf{u}^1, \mathbf{u}^2, \dots$ , and hope that they converge to the correct answer.

If we let

$$\mathbf{u}^j = (u_0^j, u_1^j, u_2^j, \dots, u_{n-1}^j, u_n^j)$$

denote the  $j$ th approximation, then we can obtain that  $j + 1$ st estimate from the formula

$$u_i^{j+1} = \frac{1}{3} (u_{i+1}^j + u_i^j + u_{i-1}^j) - \frac{h^2}{3} (d((u_i^j)^4 - u_b^4) - g_i).$$

Notice that  $g_i$  and  $u_b$  do not change. In the resulting equation, we have  $u_i$  at each successive step depending on its previous value and the equation itself.

## Implementing the Relaxation Method

In the following program we solve the finite difference equations (34.2) with the boundary conditions  $u(0) = 0$  and  $u(L) = 0$ . We let  $L = 4$ ,  $n = 4$ ,  $d = 1$ , and  $g(x) = \sin(\pi x/4)$ . Notice that the vector  $\mathbf{u}$  always contains the current estimate of the values of  $\mathbf{u}$ .

```
% mynonlinheat (lacks comments)
% Purpose:
L = 4; %
n = 4; %
h = L/n; %
hh = h^2/3; %
u0 = 0; %
uL = 0; %
ub = .5; %
ub4 = ub^4; %
x = 0:h:L; %
g = sin(pi*x/4); %
u = zeros(1,n+1); %
steps = 4; %
u(1)=u0; %
u(n+1)=uL; %
for j = 1:steps
    %
    u(2:n) = (u(3:n+1)+u(2:n)+u(1:n-1))/3 + hh*(-u(2:n).^4+ub4+g(2:n));
end
plot(x,u)
```

If you run this program with the given  $n$  and  $steps$  the result will not seem reasonable.

We can plot the initial guess by adding the command `plot(x,u)` right before the `for` loop. We can also plot successive iterations by moving the last `plot(x,u)` before the `end`. Now we can experiment and see if the iteration is converging. Try various values of  $steps$  and  $n$  to produce a good plot. You will notice that this method converges quite slowly. In particular, as we increase  $n$ , we need to increase  $steps$  like  $n^2$ , i.e. if  $n$  is large then  $steps$  needs to be *really* large.

**Exercises**

- 34.1 Modify the script program `mynonlinheat` to plot the initial guess and all intermediate approximations. Add complete comments to the program. Print the program and a plot using  $n = 12$  and `steps` large enough to see convergence.
- 34.2 Modify your improved `mynonlinheat` to `mynonlinheattwo` that has the boundary conditions

$$u(0) = 5 \quad \text{and} \quad u(L) = 10.$$

Fix the comments to reflect the new boundary conditions. Print the program and a plot using  $n = 50$  and large enough `steps` to see convergence.