Solving the Heat Equation using Matlab

In class I derived the heat equation

$$u_t = Cu_{xx}\,, \quad u_x(t,0) = u_x(t,1) = 0\,, \quad u(0,x) = u_0(x)\,, \quad 0 < x < 1\,,$$

where $u(t,x)$ is the temperature of an insulated wire. To solve this problem numerically, we will turn it into a system of odes. We use the following Taylor expansions,

$$u(t,x+k) = u(t,x) + ku_x(t,x) + \frac{1}{2}k^2u_{xx}(t,x) + \frac{1}{6}k^3u_{xxx}(t,x) + O(k^4)\,, \tag{1}$$

$$u(t,x-k) = u(t,x) - ku_x(t,x) + \frac{1}{2}k^2u_{xx}(t,x) - \frac{1}{6}k^3u_{xxx}(t,x) + O(k^4)\,, \tag{2}$$

$$\tag{3}$$

If we add the equations in (1) and solve for $u_{xx}(t,x)$ we get

$$u_{xx}(t,x) = \frac{u(t,x-k) - 2u(t,x) + u(t,x+k)}{k^2} + O(k^2)\,.$$

Now if we divide the region $0 < x < 1$ into n pieces with $x_i = ik$ and $k = \frac{1}{n}$ and let $u_i(t) \sim u(t,x_i)$ then we will have the following system of ordinary differential equations.

$$\frac{du_i}{dt} = C\left(\frac{u_{i-1} - 2u_i + u_{i+1}}{k^2}\right)\,, \quad i = 1\ldots(n-1)\,.$$

To find the equations for u_0 and u_n, we must consider the boundary conditions $u_x(t,0) = u_x(t,1) = 0$. To approximate u_x we take the equations in (1) and subtract then and solve for u_x to get

$$u_x(t,x) = \frac{u(t,x+k) - u(t,x-k)}{2k} + O(k^2)$$

We apply this at $x = 0$ and $x = 1$ to find

$$u_x(t,0) = \frac{u_1 - u_{-1}}{2k} = 0\,,$$

$$u_x(t,1) = \frac{u_{n+1} - u_{n-1}}{2k} = 0\,,$$

We note that the points x_{-1} and x_{n+1} are not in our interval and the solution is not really valid there. However if we use these relations to solve for u_{-1} and u_{n+1}, we can eliminate these terms from the u_0 and u_n differential equation resulting in

$$\frac{du_0}{dt} = C\left(\frac{2(u_1 - u_0)}{k^2}\right)\,,$$

$$\frac{du_n}{dt} = C\left(\frac{2(u_{n-1} - u_n)}{k^2}\right)\,,$$

We code this all up with the initial condition $u(0,x) = e^{-\frac{(x-0.1)^2}{0.01}}$. The Octave code is given below. To use the `ode5r` code I had to install the octave-odepkg. To run this code with Matlab just change `ode5r` to `ode15s`.

```
function [t,u]=heat()

  n=100;
  dx=1/n;
  y0=zeros(n,1);
  for i=1:n
       x=i*dx;
```

```
y0(i)=exp(-(x-.1)^2/.01);
  end

  [t,u]=ode5r(@odes,[0,10],y0);

  for i=1:length(t)
    plot(u(i,:))
    pause(.2);
  end



end

function yp=odes(t,y)

  n=100;
  dx=1/n;
  k=1;
  yp=zeros(n,1);
  yp(1)=k*(2*(y(2)-y(1)))/dx^2;
  for i=2:n-1
  yp(i)=k*(y(i+1)-2*y(i)+y(i-1))/dx^2;
  end
  yp(n)=k*(2*(y(n-1)-y(n)))/dx^2;
end
```