

# CAAM 336 · DIFFERENTIAL EQUATIONS

## Homework 29 · Solutions

Posted Friday 21 March 2014. Due 1pm Friday 28 March 2014.

29. [25 points]

Let  $N$  be a positive integer, let  $h = \frac{1}{N+1}$  and let  $x_k = kh$  for  $k = 0, 1, \dots, N+1$ . Let the continuous piecewise quadratic functions  $\phi_j \in C[0, 1]$  be such that

$$\phi_j(x) = \begin{cases} \frac{(x - x_{j-1})(2x - x_{j-1} - x_j)}{h^2} & \text{if } x \in [x_{j-1}, x_j], \\ \frac{(x_j + x_{j+1} - 2x)(x_{j+1} - x)}{h^2} & \text{if } x \in [x_j, x_{j+1}], \\ 0 & \text{otherwise,} \end{cases}$$

for  $j = 1, \dots, N$  and let the continuous piecewise quadratic bubble functions  $\psi_j \in C[0, 1]$  be such that

$$\psi_j(x) = \begin{cases} \frac{4(x - x_{j-1})(x_j - x)}{h^2} & \text{if } x \in [x_{j-1}, x_j], \\ 0 & \text{otherwise,} \end{cases}$$

for  $j = 1, \dots, N+1$ .

- Write a MATLAB function for  $\phi_j(x)$ . It should take in as input  $x$ ,  $j$ , and  $N$ . It should return the value  $\phi_j(x)$ . It should also be able to take in a vector for  $\mathbf{x} = (\hat{x}_1, \dots, \hat{x}_m)$  and return the vector  $\phi_j(\mathbf{x}) = (\phi_j(\hat{x}_1), \dots, \phi_j(\hat{x}_m))$ .
- Write a MATLAB function for  $\psi_j(x)$ . It should take in as input  $x$ ,  $j$ , and  $N$ . It should return the value  $\psi_j(x)$ . It should also be able to take in a vector for  $\mathbf{x} = (\hat{x}_1, \dots, \hat{x}_m)$  and return the vector  $\psi_j(\mathbf{x}) = (\psi_j(\hat{x}_1), \dots, \psi_j(\hat{x}_m))$ .
- For  $N = 3$ , plot  $\phi_1(x), \phi_2(x), \phi_3(x), \psi_1(x), \psi_2(x), \psi_3(x), \psi_4(x)$  on the same figure. If you experience problems with displaying  $\psi$  using the latex interpreter then you may use the command `legend('phi_1(x)', 'phi_2(x)', 'phi_3(x)', 'psi_1(x)', 'psi_2(x)', 'psi_3(x)', 'psi_4(x)')` to produce the legend.

**Solution.**

- [5 points] One way of coding the function is:

```
function phi_j=cpg1(x,j,N)

h=1/(N+1);
xj=j*h;
xjml=xj-h;
xjpl=xj+h;
hs=h^2;

phi_j=((x>=xjml)&(x<=xj)).*(x-xjml).*(2*x-xjml-xj)/hs+...
      ((x>=xj)&(x<=xjpl)).*(xj+xjpl-2*x).*(xjpl-x)/hs;
```

- [5 points] One way of coding the function is:

```

function psi_j=cpq2(x,j,N)

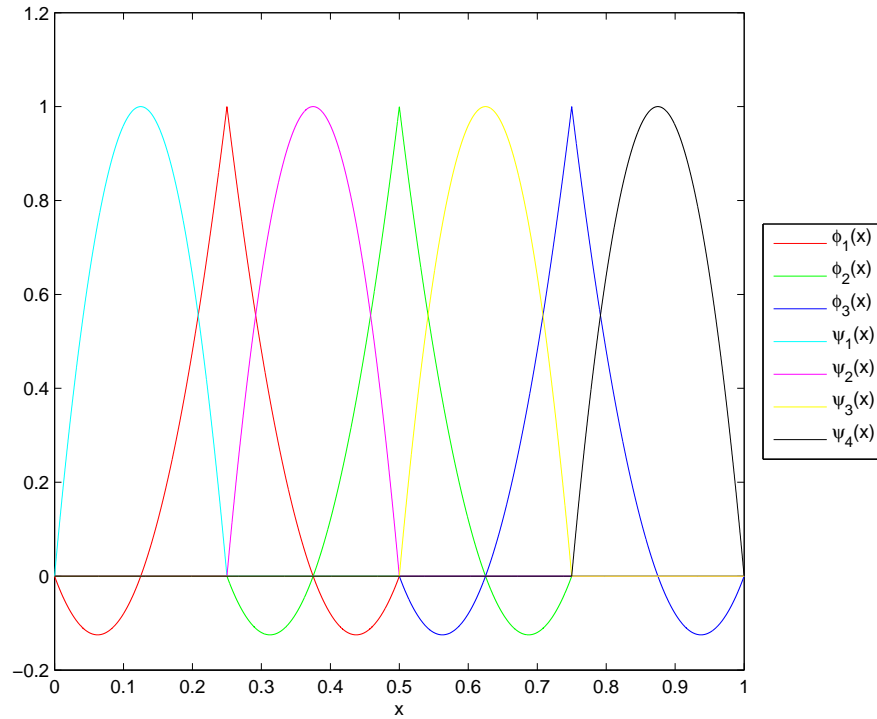
h=1/(N+1);
xj=j*h;
xjml=xj-h;
fohs=4/h^2;

psi_j=((x>=xjml)&(x<xj)).*(x-xjml).*(xj-x)*fohs;

```

(c) [15 points]

The plot and code used to create it are below.



```

clear
clc
N=3;
x=linspace(0,1,10000);

figure(1)
clf
colors='rgbcmyk';
for k=1:3
    plot(x,cpq1(x,k,N),colors(k))
    hold on
end
for k=1:4
    plot(x,cpq2(x,k,N),colors(k+3))
end

legend('\phi_1(x)', '\phi_2(x)', '\phi_3(x)', '\psi_1(x)', '\psi_2(x)', '\psi_3(x)', ...
        '\psi_4(x)', 'Location', 'EastOutside')
xlabel('x');
saveas(figure(1), 'hw29c.eps', 'psc2')

```