AE 370: HW6

Linyi Hou

April 10, 2020

Problem 1

a. Using the second order Lagrange basis, we approximate the solution over the interval $[x_{j-1}, x_{j+1}]$:

$$u(x) \approx \sum_{i=j-1}^{j+1} u_i L_i^{(j)}(x), \quad j = 2, ..., n \tag{1}$$

where $L_i(x)$ are the second order Lagrange polynomials centered at x_j, e.g.:

$$L_{j-1}^{j}(x) = \frac{1}{2\Delta x^2}(x - x_j)(x - x_{j-1}) \tag{2}$$

where $\Delta x = x_j - x_{j-1}$.

We plug the approximate solution to the original ODE to get

$$\sum_{i=j-1}^{j+1} u_i \frac{d^2 L_i^{(j)}}{dx^2}\bigg|_{x=x_j} + 3 \sum_{i=j-1}^{j+1} u_i \frac{dL_i^{(j)}}{dx}\bigg|_{x=x_j} = f(x_j), \quad j = 2, ..., n \tag{3}$$

Note that the boundary conditions are separately enforced via

$$\begin{aligned} u(0) &= 0 \\ u(1) &= 0. \end{aligned} \tag{4}$$

Now, we take the derivatives of $L_i^{(j)}$ and evaluate at x_j to rewrite Equation (3):

$$\frac{1}{\Delta x^2}(u_{j-1} - 2u_j + u_{j+1}) + 3\left[\frac{1}{\Delta x}\left(\frac{1}{2}u_{j+1} - \frac{1}{2}u_{j-1}\right)\right] = f(x_j) \tag{5}$$

We can further rewrite Equation (5) in matrix form:

$$Au = f \tag{6}$$

where $u = [u_1, u_2, ...u_{n+1}]^T$, $f = [0, f(x_2), ...f(x_n), 0]^T$, and $A = L + 3D$. Note that the boundary conditions have been incorporated into the first and last terms of f. We

have

$$L = \frac{1}{\Delta x^2}\begin{bmatrix} \Delta x^2 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & \Delta x^2 \end{bmatrix}$$

$$D = \frac{1}{\Delta x}\begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \\ -1/2 & 0 & 1/2 & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & -1/2 & 0 & 1/2 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & -1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7}$$

b. We define the truncation error as $\tau = A\bar{u} - f$. Therefore

$$\tau_j = \frac{1}{\Delta x^2}\left(u(x_{j-1}) - 2u(x_j) + u(x_{j+1})\right) + 3\left[\frac{1}{\Delta x}\left(\frac{1}{2}u(x_{j+1}) - \frac{1}{2}u(x_{j-1})\right)\right] - f(x_j) \tag{8}$$

Taylor expanding $u(x_{j-1})$ and $u(x_{j+1})$ as

$$u(x_{j+1}) = u(x_j) + \Delta x\frac{du}{dx} + \frac{\Delta x^2}{2}\frac{d^2u}{dx^2} + \frac{\Delta x^3}{6}\frac{d^3u}{dx^3} + \frac{\Delta x^4}{24}\frac{d^3u}{dx^4} + O(\Delta x^5) \tag{9}$$

we get

$$\tau_j = \left(\frac{d^2u}{dx^2} + \frac{\Delta x^2}{12}\frac{d^4u}{dx^4} + O(\Delta x^4)\right) + \left(3\frac{du}{dx} + \frac{\Delta x^2}{2}\frac{d^3u}{dx^3} + O(\Delta x^4)\right) - \frac{d^2u}{dx^2} - 3\frac{du}{dx} \tag{10}$$

which simplifies to

$$\tau_j = \frac{\Delta x^2}{12}\frac{d^4u}{dx^4} + \frac{\Delta x^2}{2}\frac{d^3u}{dx^3} + O(\Delta x^4) \tag{11}$$

$$\boxed{\tau_j = O(\Delta x^2)} \tag{12}$$

c. See figures and code attached in Appendix A and Appendix B, respectively.

Problem 2

a. The figures and code to Problem 2e have been attached.

Appendix A: Figures

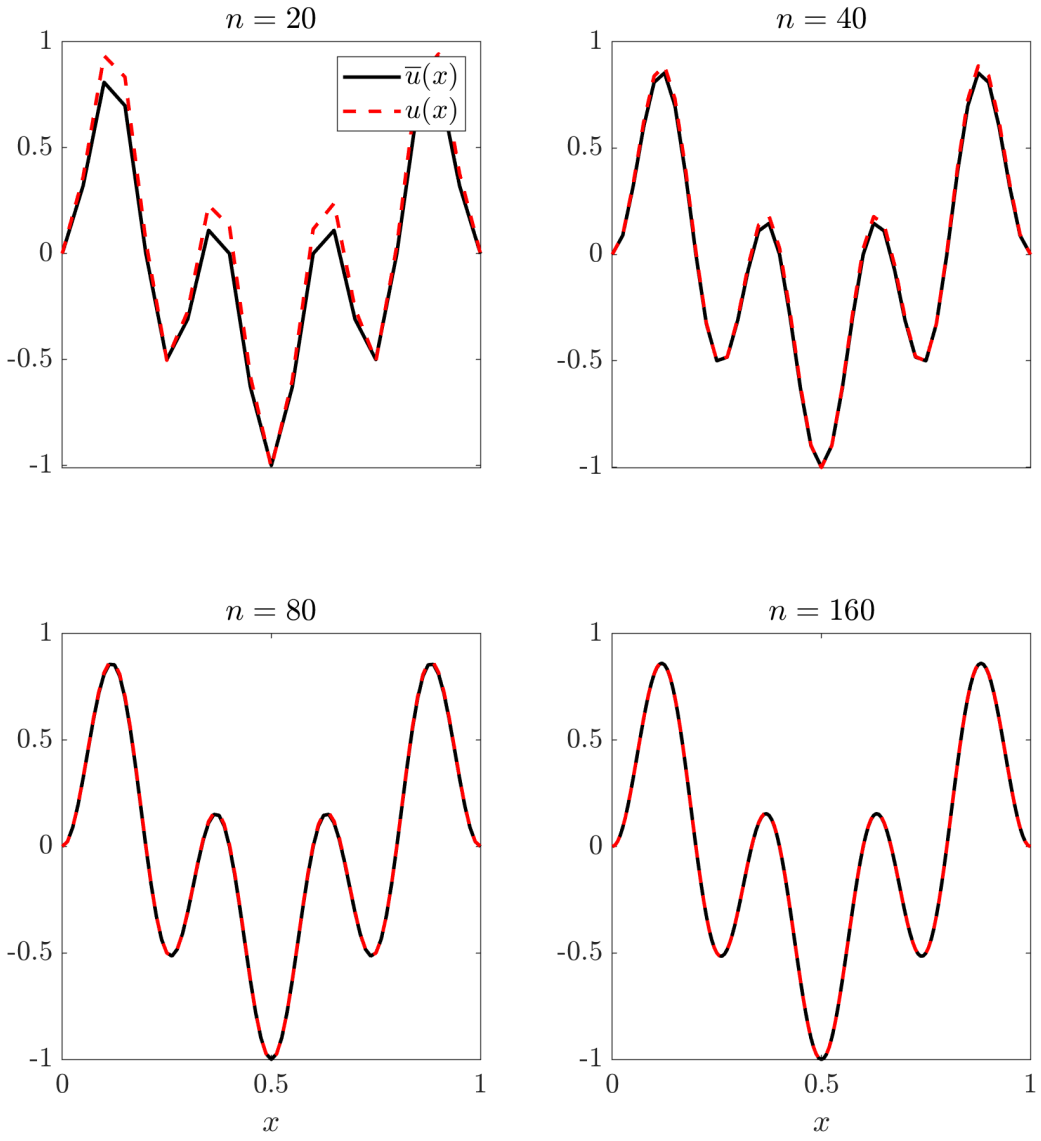

Figure 1: Problem 1 Function Plot

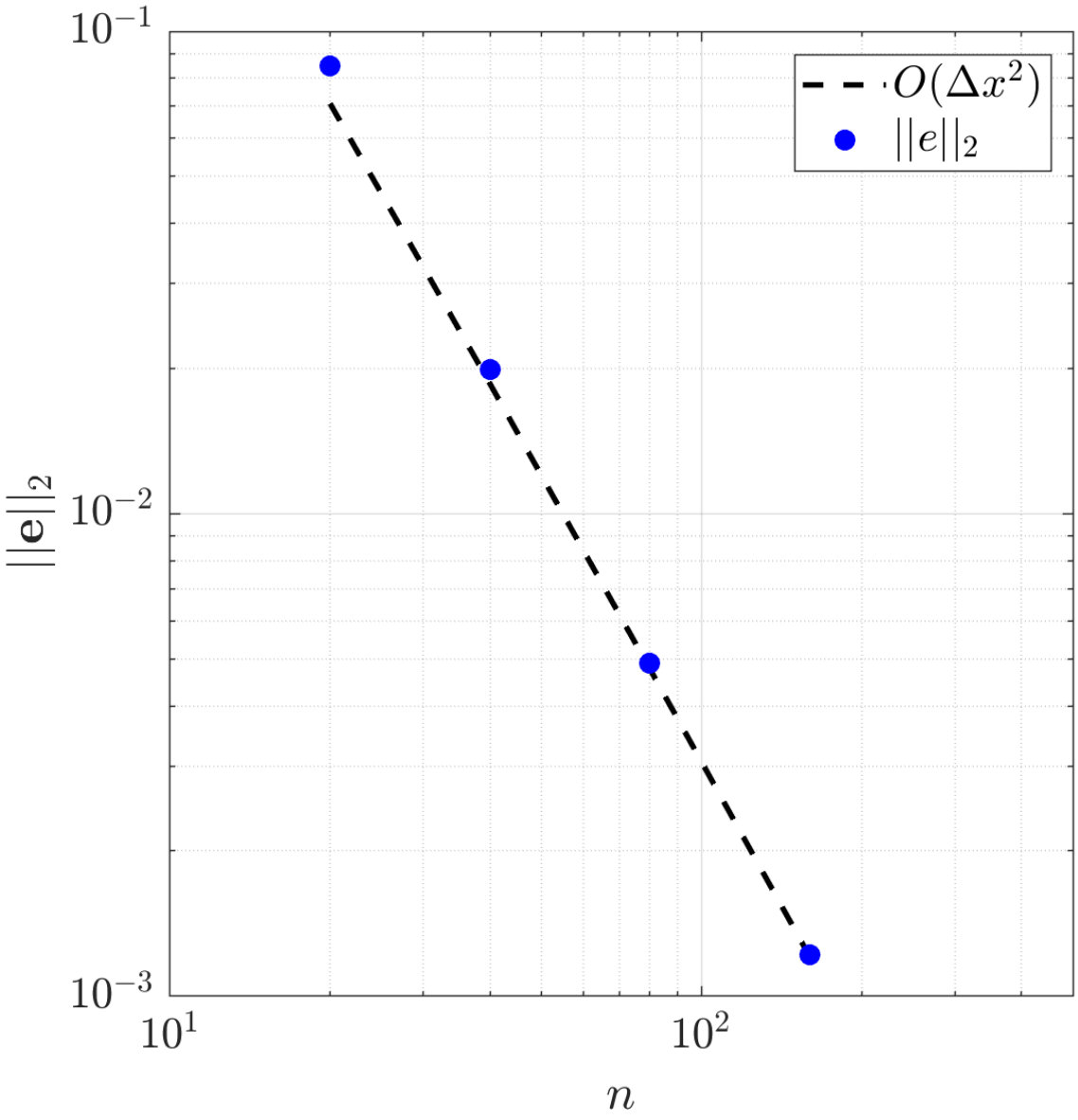

Figure 2: Problem 1 Error Plot

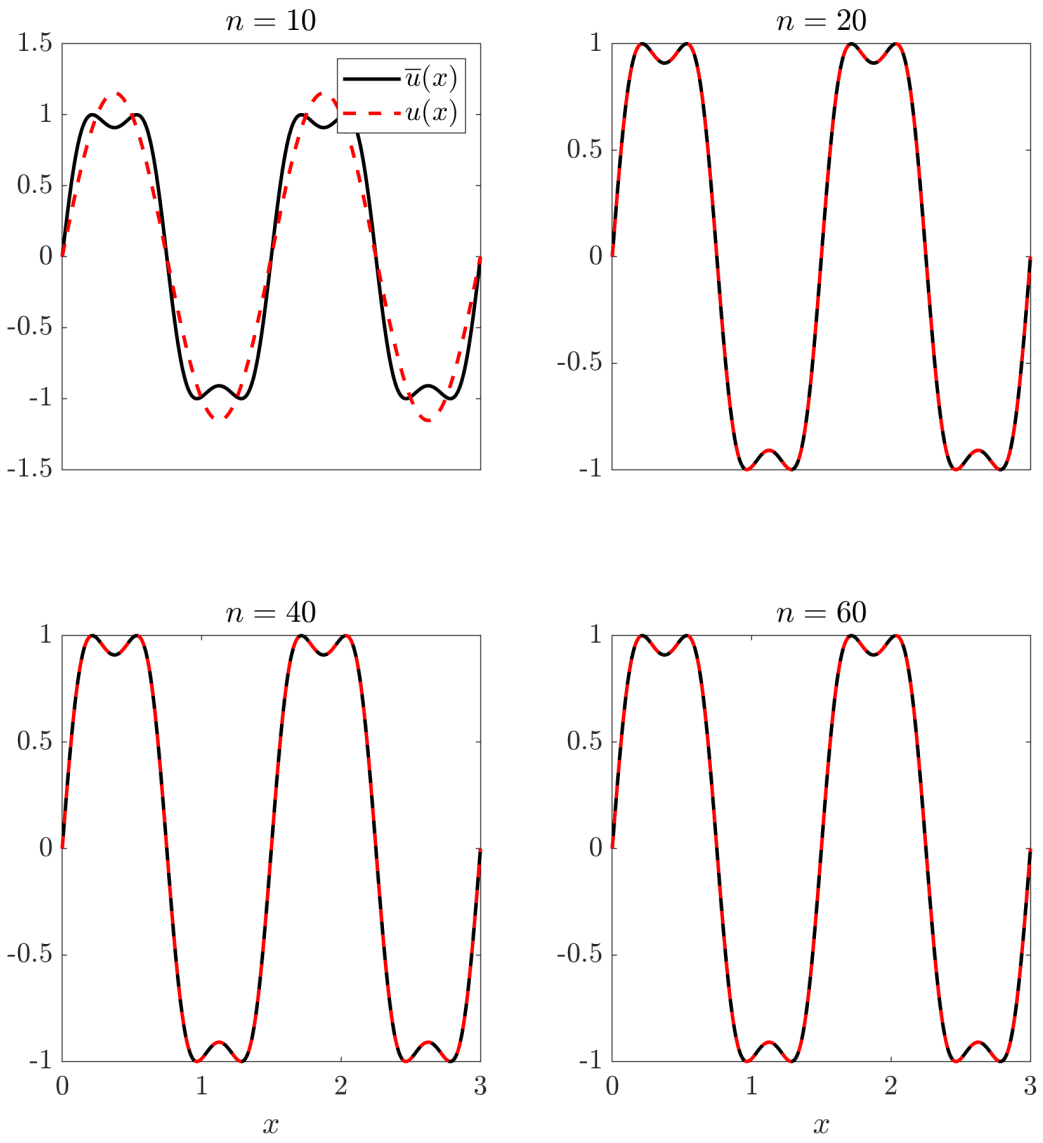

Figure 3: Problem 2 Function Plot

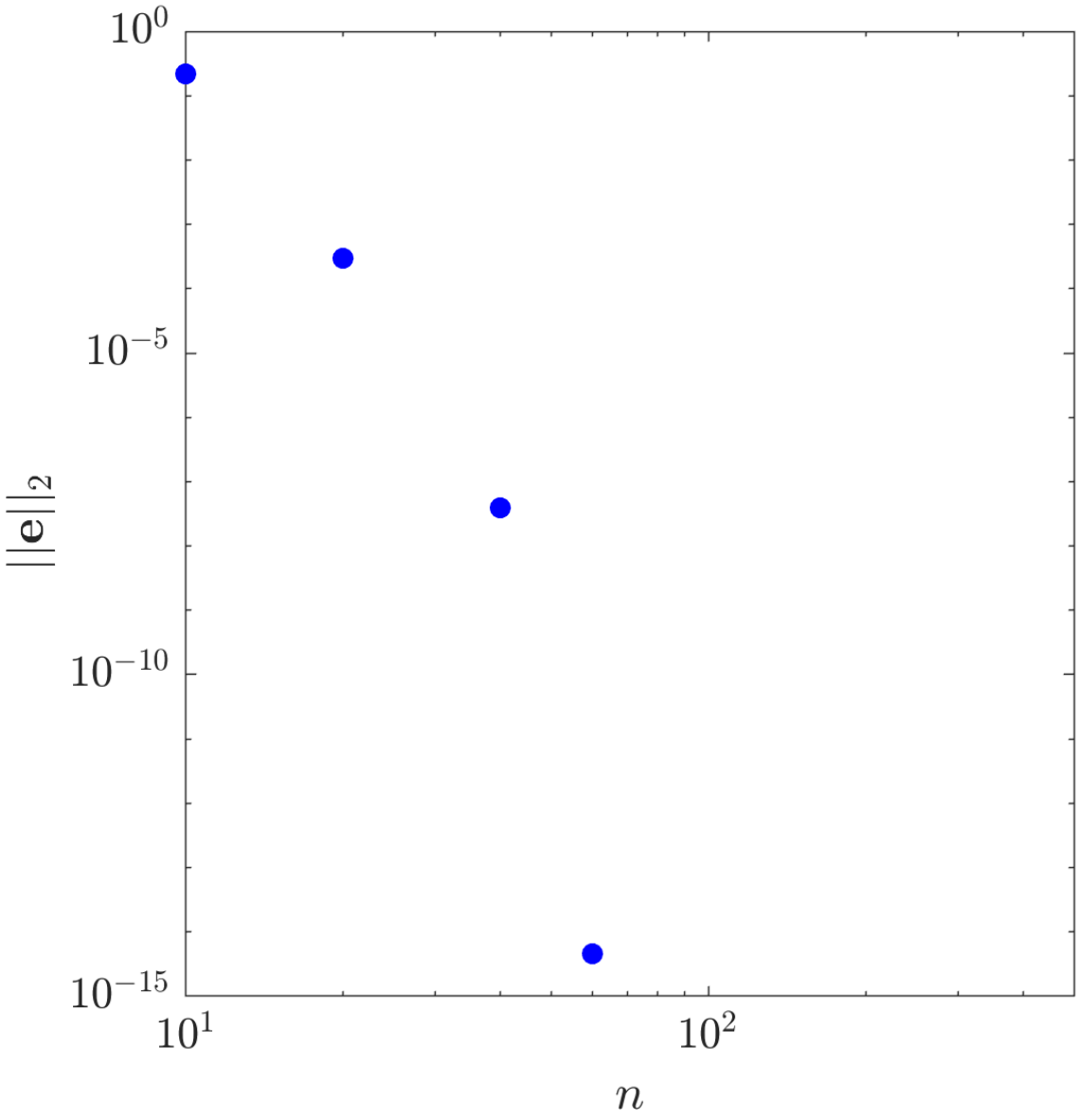

Figure 4: Problem 2 Error Plot

Appendix B: Code

```
%% Problem 1

close all
clear;clc

%solve u'' + 3*u' = f over 0 < x < 1
%with BCs u(0) = u(1) = 0;

%use central differences

%exact sol
uex = @(x) sin( 5.*pi*x ).*sin( 3*pi.*x );

%f(x) =
fcn = @(x) 30 * pi^2.*cos(3.*pi.*x).*cos(5.*pi.*x) ...
        - 34 * pi^2.*sin(3.*pi.*x).*sin(5.*pi.*x) ...
        + 3  * ( 3.*pi.*cos(3.*pi.*x).*sin(5.*pi.*x) ...
                +5.*pi.*cos(5.*pi.*x).*sin(3.*pi.*x) );

%left and right bounds
xbds = [0 1];

%# of n points to use
nvect = [20, 40, 80, 160];

%initialize error vect
err = zeros( size( nvect ) );

for j = 1 : length( nvect )

    n = nvect( j ) ;
    dx = ( xbds(2) - xbds(1) ) / n;

    %x points
    xx = linspace(xbds(1),xbds(2),n+1)';

    %--LHS matrices (2nd deriv &  1st deriv matrix)

```

```
39          %**second deriv. interior points (Central Difference)
40          L = (1/dx^2) * (diag(ones(n,1),1) + diag(ones(n,1),-1) - 2*eye(n
                +1));
41
42          %** first deriv. interior points (Central Difference)
43          D = (1/dx) * (0.5 * diag(ones(n,1),1) - 0.5 * diag(ones(n,1),-1))
                ;
44
45          %**combine and incorporate BCs
46          A = L + 3*D;
47
48      %---
49
50      %remove erroneous entries in first & last rows
51      A(1,:) = 0;
52      A(n+1,:) = 0;
53
54      %correct entries in first & last rows
55      A(1,1) = 1;
56      A(end) = 1;
57
58      %rhs vector
59      f = fcn(xx);
60      f(1) = 0;
61      f(end) = 0;
62
63      %approximate solution
64      u = A \ f;
65
66      % exact solution
67      ue = uex(xx);
68
69      %---plot soln
70          figure(1),subplot(2,2,j)
71          plot( xx, ue, 'k-', 'linewidth', 2 ), hold on
72          plot( xx, u , 'r--', 'linewidth', 2 ), hold on
73
74          %make plot pretty
75          title( ['$n = ', num2str( n ),'$'] ,'interpreter', 'latex',...
76              'fontsize', 16)
```

```
        if j == 1
            h = legend( '$\overline{u}(x)$', '$u(x)$');
        end

        if j <= 2
            set( gca, 'XTick', [] )
        else
            xlabel( '$x$', 'interpreter', 'latex', 'fontsize', 16)

        end
        set(h, 'location', 'NorthEast', 'Interpreter', 'Latex', 'fontsize
            ', 16 )
        set(gca, 'TickLabelInterpreter','latex', 'fontsize', 16 )

        set(gcf, 'PaperPositionMode', 'manual')
        set(gcf, 'Color', [1 1 1])
        set(gca, 'Color', [1 1 1])
        set(gcf, 'PaperUnits', 'centimeters')
        set(gcf, 'PaperSize', [25 25])
        set(gcf, 'Units', 'centimeters' )
        set(gcf, 'Position', [0 0 25 25])
        set(gcf, 'PaperPosition', [0 0 25 25])

    %--

    %error
    err( j ) = sqrt(dx)*norm( u - ue );


end

svnm = 'q1_plot';
print( '-dpng', svnm, '-r200' )

%plot err
figure(100)

%plot dx^2 line to show err scales correctly
c = err(end)/(dx^2);
loglog( nvect, c./((nvect+1).^2), 'k--', 'linewidth', 2 ), hold on
```

```
116
117 %plot err
118 loglog( nvect, err , 'b.', 'markersize', 26 )
119 xlim([10 500])
120
121 h = legend( '$O(\Delta x^2)$', '$||e||_2$');
122 set(h, 'location', 'NorthEast', 'Interpreter', 'Latex', 'fontsize', 16 )
123
124 %make pretty
125 xlabel( '$n$', 'interpreter', 'latex', 'fontsize', 16)
126 ylabel( '$||\textbf{e}||_2$ ', 'interpreter', 'latex', 'fontsize', 16)
127
128 grid(gca,'minor')
129 grid on
130
131 set(gca, 'TickLabelInterpreter','latex', 'fontsize', 16 )
132
133 set(gcf, 'PaperPositionMode', 'manual')
134 set(gcf, 'Color', [1 1 1])
135 set(gca, 'Color', [1 1 1])
136 set(gcf, 'PaperUnits', 'centimeters')
137 set(gcf, 'PaperSize', [15 15])
138 set(gcf, 'Units', 'centimeters' )
139 set(gcf, 'Position', [0 0 15 15])
140 set(gcf, 'PaperPosition', [0 0 15 15])
141
142 svnm = 'q1_error';
143 print( '-dpng', svnm, '-r200' )
144
145
146 %% Problem 2
147
148 close all
149 clear;clc
150
151 %solve -u'' + 2*u = f over 0 < x < 3
152 %with BCs u(0) = u(3) = 0;
153
154 %left and right bounds
155 xl = 0; xr = 3;
```

```
ln = xr - xl;

%exact sol
uex = @(x) sin( 2* sin(4*pi*(x-xl)/ln) );


%f(x)
fcn = @(x) ...
    (1/ln^2) ...
        * ( (32*pi^2*cos(2*sin((4*pi*(x-xl))/ln)).*sin((4*pi*(x-xl))/ln))
            ...
          + (64*pi^2*sin(2*sin((4*pi*(x-xl))/ln)).*cos((4*pi*(x-xl))/ln)
             .^2) ) ...
  + 2*(sin( 2* sin(4*pi*(x-xl)/ln) ));

syms x
dfcn = matlabFunction(diff(fcn(x)));

%# of n points to use
nvect = [10; 20; 40; 60];

%initialize error vect
err = zeros( size( nvect ) );

%x points on which to evaluate u
xx = linspace(xl,xr,1000);

for j = 1 : length( nvect )

    n = nvect( j );

    %Store coefficients in c vector
    c = zeros(n,1);
    for jj = 1 : n

        c(jj) = trapz(xx,fcn(xx).*sin(jj.*pi.*(xx-xl)/ln)) / ...
          ( 1*trapz(xx, (cos(jj.*pi.*(xx-xl)/ln)*jj*pi/ln) .* (cos(jj.*pi
             .*(xx-xl)/ln)*jj*pi/ln) ) ...
           +2*trapz(xx, (sin(jj.*pi.*(xx-xl)/ln)) .* (sin(jj.*pi.*(xx-xl)
              /ln)) ) );
```

```
192
193     end
194
195
196     %Represent u on the finer xx domain
197     u = 0;
198     for kk = 1 : n
199
200         u = u + c(kk) .* sin(kk.*pi.*(xx-xl)/ln);
201
202     end
203
204     ue = uex(xx);
205
206     %--plot soln
207         figure(1),subplot(2,2,j)
208         plot( xx, ue, 'k-', 'linewidth', 2 ), hold on
209         plot( xx, u, 'r--', 'linewidth', 2 ), hold on
210
211         %make plot pretty
212         title( ['$n = ', num2str( n ),'$'] ,'interpreter', 'latex',...
213             'fontsize', 16)
214         if j == 1
215             h = legend( '$\overline{u}(x)$', '$u(x)$');
216         end
217
218         if j <= 2
219             set( gca, 'XTick', [] )
220         else
221             xlabel( '$x$', 'interpreter', 'latex', 'fontsize', 16)
222
223         end
224         set(h, 'location', 'NorthEast', 'Interpreter', 'Latex', 'fontsize
                ', 16 )
225         set(gca, 'TickLabelInterpreter','latex', 'fontsize', 16 )
226
227         set(gcf, 'PaperPositionMode', 'manual')
228         set(gcf, 'Color', [1 1 1])
229         set(gca, 'Color', [1 1 1])
230         set(gcf, 'PaperUnits', 'centimeters')
```

```
        set(gcf, 'PaperSize', [25 25])
        set(gcf, 'Units', 'centimeters' )
        set(gcf, 'Position', [0 0 25 25])
        set(gcf, 'PaperPosition', [0 0 25 25])
    %--

    %error
    err( j ) = norm( u - uex(xx) )/norm(uex(xx)) ;


end

svnm = 'q2_plot';
print( '-dpng', svnm, '-r200' )

%plot err
figure(100)

% %plot dx^2 line to show err scales correctly
% c = err(end)/(dx^2);
% loglog( nvect, c./((nvect+1).^2), 'k--', 'linewidth', 2 ), hold on

%plot err
loglog( nvect, err , 'b.', 'markersize', 26 )
xlim([10 500])

%make pretty
xlabel( '$n$', 'interpreter', 'latex', 'fontsize', 16)
ylabel( '$||\textbf{e}||_2$ ', 'interpreter', 'latex', 'fontsize', 16)

set(gca, 'TickLabelInterpreter','latex', 'fontsize', 16 )

set(gcf, 'PaperPositionMode', 'manual')
set(gcf, 'Color', [1 1 1])
set(gca, 'Color', [1 1 1])
set(gcf, 'PaperUnits', 'centimeters')
set(gcf, 'PaperSize', [15 15])
set(gcf, 'Units', 'centimeters' )
set(gcf, 'Position', [0 0 15 15])
set(gcf, 'PaperPosition', [0 0 15 15])
```

```
271
272 svnm = 'q2_error';
273 print( '-dpng', svnm, '-r200' )
```