

Boundary Value Problems, Linear

A general second order one-dimensional two-point boundary value problem (BVP) has the form

$$\begin{aligned}y'' &= f(x, y, y'), \quad a \leq x \leq b, \\ \alpha_1 y(a) + \alpha_2 y'(a) &= \alpha_3 \\ \beta_1 y(b) + \beta_2 y'(b) &= \beta_3,\end{aligned}$$

The boundary conditions at $x = a$ and $x = b$ are linear combinations of the function $y(x)$ and its derivative at $x = a$ and $x = b$ respectively. These are called Dirichlet Boundary Conditions.

Here we focus on a linear BVP of the form

$$y'' = p(x)y' + q(x)y + r(x), \quad x \in [a, b]$$

with (simple) boundary conditions

$$y(a) = \alpha, \quad y(b) = \beta.$$

Here we will use a finite difference method to approximate the value of the exact solution $\mathbf{y}(\mathbf{x})$ at a set of points x_0, x_1, \dots, x_N in interval $[a, b]$. We use the symbol \mathbf{y}_i to denote the value of the exact solution at $x = x_i$, and will denote our finite difference approximation to \mathbf{y}_i by \mathbf{w}_i . Each derivative that appears in the BVP will be replaced by a finite difference. This converts the continuous differential equation for $y(x)$ into a system of discrete algebraic equations for the values $\mathbf{w}_0, \mathbf{w}_1, \dots, \mathbf{w}_N$.

For N a positive integer we partition the interval $[a, b]$ into N subintervals using the grid points

$$a = x_0 < x_1 < x_2 < \dots < x_{N-1} < x_N = b.$$

The points x_0 and x_N are called the boundary points and x_1, x_2, \dots, x_{N-1} are called the interior points. Here we will assume that the points are equispaced by $h = (b - a)/N$ and then we have $x_i = a + ih = x_0 + ih$. Parameter h is called the stepsize or mesh size and is the key parameter in governing the accuracy of the finite difference approximation.

We will use CENTERED DIFFERENCE APPROXIMATIONS for both $y'(x)$ and $y''(x)$. At an interior point $x_i, i = 1, 2, \dots, N-1$ we have the following:

$$y'(x_i) = \frac{y_{i+1} - y_{i-1}}{2h} + O(h^2) \quad \text{and} \quad y''(x_i) = \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + O(h^2)$$

Replacing the derivatives in the DE at $x = x_i$ $y'' = p(x)y' + q(x)y + r(x)$

we have
$$\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + O(h^2) = p_i \frac{y_{i+1} - y_{i-1}}{2h} + q_i y_i + r_i + O(h^2).$$

Here we have used p_i , q_i , and r_i to denote the value $p(x_i)$, $q(x_i)$, and $r(x_i)$, respectively. Next we drop the truncation error terms $O(h^2)$ and we must replace y_i by w_i to denote the approximate values. This gives us the difference equations

$$\frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} = p_i \frac{w_{i+1} - w_{i-1}}{2h} + q_i w_i + r_i$$

for $i = 1, 2, \dots, N-1$. We need two more equations which are supplied by the boundary conditions; here we have $w_0 = \alpha$ and $w_N = \beta$. Resulting numerical method will be **second order** since $O(h^2)$ terms were dropped from both finite difference formulas. We have

Second Order Finite Difference Method

$$w_0 = \alpha$$

$$\frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} = p_i \frac{w_{i+1} - w_{i-1}}{2h} + q_i w_i + r_i \quad 1 \leq i \leq N - 1$$

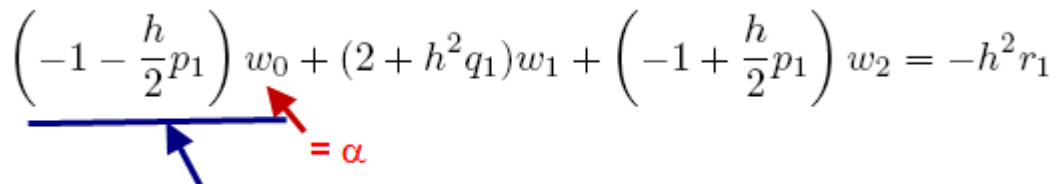
$$w_N = \beta$$

Since the difference equations are linear in the unknowns w_0, w_1, \dots, w_N we can express the system of equations in matrix form. We first multiply both sides of the difference equation at the interior points by h^2 and then gather like terms in the approximation w_i to obtain the equations

$$\left(-1 - \frac{h}{2}p_i\right) w_{i-1} + (2 + h^2q_i)w_i + \left(-1 + \frac{h}{2}p_i\right) w_{i+1} = -h^2r_i$$

for $i = 1, 2, \dots, N-1$. Since there are only 3 unknowns in each equation we can express the set of equations in matrix form using a tridiagonal matrix. In doing so we must be careful in the following cases.

Case $i = 1$
The equation is

$$\left(-1 - \frac{h}{2}p_1\right) w_0 + (2 + h^2q_1)w_1 + \left(-1 + \frac{h}{2}p_1\right) w_2 = -h^2r_1$$


So this term will be taken to the right side.

It now appears as

$$(2 + h^2q_1)w_1 + \left(-1 + \frac{h}{2}p_1\right) w_2 = -h^2r_1 + \left(1 + \frac{h}{2}p_1\right) \alpha$$

Case $i = N-1$ Here $w_N = \beta$ and in a similar manner the equation becomes

$$\left(-1 - \frac{h}{2}p_{N-1}\right) w_{N-2} + (2 + h^2q_{N-1})w_{N-1} = -h^2r_{N-1} + \left(1 - \frac{h}{2}p_{N-1}\right) \beta$$

Example

Consider the BVP

$$\begin{aligned} -u'' + \pi^2 u &= 2\pi^2 \sin(\pi x) \\ u(0) &= u(1) = 0. \end{aligned}$$

Recall that we need to convert the equation to the form

$$y'' = p(x)y' + q(x)y + r(x), \quad x \in [a, b]$$

(Multiply both sides by -1 and solve for u'' .)

We see that $p(x) = 0$, $q(x) = \pi^2$ and $r(x) = -2\pi^2 \sin(\pi x)$.

Since $p(x) = 0$, we are guaranteed of a unique solution to the finite difference equations for any value of h .

$$d_i = 2 + h^2 q_i \Rightarrow d_i = 2 + h^2 \pi^2$$

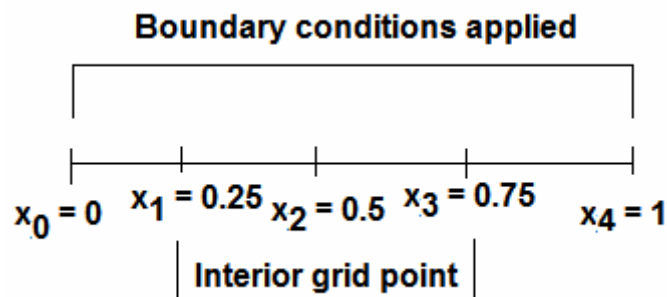
$$u_i = -1 + \frac{h}{2} p_i \Rightarrow u_i = -1$$

$$l_i = -1 - \frac{h}{2} p_i \Rightarrow l_i = -1$$

$$r_i = -2\pi^2 \sin(\pi x_i)$$

$$b_i = -h^2 r_i$$

If we choose $h = 1/4$, then the grid points are as shown below.



The corresponding linear system is

$$\begin{bmatrix} 1 & 0 & & & & & & \\ -1 & 2 + (\pi/4)^2 & & -1 & & & & \\ & -1 & 2 + (\pi/4)^2 & & -1 & & & \\ & & -1 & 2 + (\pi/4)^2 & & -1 & & \\ & & & -1 & 2 + (\pi/4)^2 & & -1 & \\ & & & & 0 & & 1 & \\ & & & & & & & 1 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} = \begin{bmatrix} 0 \\ \sqrt{2}(\pi/4)^2 \\ 2(\pi/4)^2 \\ \sqrt{2}(\pi/4)^2 \\ 0 \end{bmatrix}$$

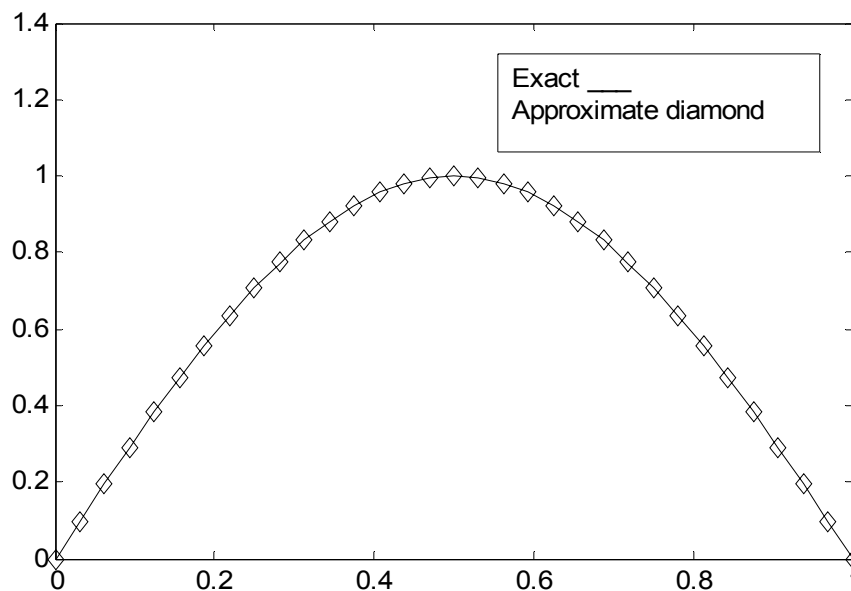
and its solution is

$$\mathbf{w} = [0.725371 \quad 1.025830 \quad 0.725371]^T$$

The following table compares this approximate solution with the exact solution: $u(x) = \sin(\pi x)$. The accuracy of the approximate solution is quite reasonable given the crudeness of the computational grid.

x_i	approximate solution, w_i	exact solution, u_i	absolute error
0.00	0.000000	0.000000	
0.25	0.725371	0.707107	0.018264
0.50	1.025830	1.000000	0.025830
0.75	0.725371	0.707107	0.018264
1.00	0.000000	0.000000	

The figure below compares the exact solution to the BVP and the finite difference approximation using a uniform partition of $[0, 1]$ containing 32 subintervals.



In the table below we present the results of doubling the number points in a partition of $[0, 1]$. We expect the absolute error approximation is reduced by roughly a factor of 4 since both derivative approximation are $O(h^2)$. We verify this in the error ratio column.

N	maximum absolute error	error ratio
4	0.0258297765	
8	0.0064337127	4.014754
16	0.0016068959	4.003814
32	0.0004016275	4.000961
64	0.0001004008	4.000241
128	0.0000250998	4.000060
256	0.0000062749	4.000015
512	0.0000015687	4.000000

Next we provide a second example in which $p(x) \neq 0$.

Example

Consider the BVP

$$u'' = -(x+1)u' + 2u + (1-x^2)e^{-x}$$

$$u(0) = -1, \quad u(1) = 0.$$

For this problem, we see that $p(x) = -(x+1)$, whose maximum absolute value on the interval $[0, 1]$ is 2; hence, we are guaranteed of a unique solution to the finite difference equations for any value of h less than 1.

We have that $p(x) = -(x+1)$, $q(x) = 2$, and $r(x) = (1-x^2)e^{-x}$. It follows that

$$p_i = -(x_i + 1) = -(1 + i/4);$$

$$q_i = 2; \text{ and}$$

$$r_i = (1 - x_i^2)e^{-x_i} = [1 - (i/4)^2] e^{-i/4}$$

Using the computational template

$$\left(-1 - \frac{h}{2}p_i\right) w_{i-1} + (2 + h^2q_i)w_i + \left(-1 + \frac{h}{2}p_i\right) w_{i+1} = -h^2r_i$$

and identifying the terms in the tridiagonal matrix **A** we have for the interior points

$$d_i = 2 + h^2q_i \Rightarrow d_i = 2 + 2h^2$$

$$u_i = -1 + \frac{h}{2}p_i \Rightarrow u_i = -1 - \frac{h}{2}(x_i + 1)$$

$$l_i = -1 - \frac{h}{2}p_i \Rightarrow l_i = -1 + \frac{h}{2}(x_i + 1)$$

$$r_i = (1 - x_i^2) e^{-x_i}$$

$$b_i = -h^2r_i$$

Choosing $h = 1/6$ we can generate the terms of the coefficient matrix **A** and right side **b** as follows.

```
>> x=0:1/6:1'
x =
    0    1/6    1/3    1/2    2/3    5/6    1
```

Recall that the first and last x values we have the boundary conditions so for we need only use values of x at the interior values in the following computations. Those values are in positions 2 through 6 in vector x .

```
>> format rat
>> p=-(x(2:6)+1)
p =
```

```

    -7/6    -4/3    -3/2    -5/3    -11/6
>> q=2*ones(1,5)
q =
    2        2        2        2        2

```

Next we compute the values diagonal, upper, and lower triangular entries of matrix for the interior points in the grid.

```

>> di=2+2*h^2*ones(1,5)
di =
    37/18    37/18    37/18    37/18    37/18
>> ui=-1-(h/2)*(x(2:6)+1)
ui =
   -79/72   -10/9    -9/8    -41/36   -83/72
>> li=-1+(h/2)*(x(2:6)+1)
li =
   -65/72   -8/9     -7/8    -31/36   -61/72

```

Next compute the values of r_i at the interior grid points. Here it is best to show the values in decimal form.

```

>> format short
>> r=(1-x(2:6).^2).*exp(-x(2:6))
r =

    0.8230    0.6369    0.4549    0.2852    0.1328

```

Use the values of r_i to get the values of the right side b at the interior points.

```

bi= -(h^2)*(1-x(2:6).^2).*exp(-x(2:6))
bi =

   -0.0229   -0.0177   -0.0126   -0.0079   -0.0037

```

Now define vectors for the tridiagonal matrix \mathbf{A} and the right side \mathbf{b} . Here we incorporate the boundary conditions and get the correct size for matrix \mathbf{A} and right side \mathbf{b} .

```

main=[1 di 1];
firstsuper=[0 ui];
firstsub=[li 0];

```

We display the coefficient matrix \mathbf{A} and right side \mathbf{b} next.
 $\mathbf{A} = \text{diag}(\text{main}) + \text{diag}(\text{firstsuper}) + \text{diag}(\text{firstsub})$

```

A =
    1.0000     0         0         0         0         0         0
   -0.9028    2.0556   -1.0972     0         0         0         0
         0   -0.8889    2.0556   -1.1111     0         0         0
         0         0   -0.8750    2.0556   -1.1250     0         0
         0         0         0   -0.8611    2.0556   -1.1389     0
         0         0         0         0   -0.8472    2.0556   -1.1528
         0         0         0         0         0         0    1.0000

```

```

b=[-1 bi 0]

```

```
b =  
-1.0000 -0.0229 -0.0177 -0.0126 -0.0079 -0.0037 0
```

```
>> approx=tridiag(firstsub,main,firstsuper,b)
```

```
approx =
```

```
-1.0000 -0.7049 -0.4769 -0.3025 -0.1705 -0.0721 0
```

The table of x-coordinates for the grid points with the approximations is

```
>> [x;approx]
```

```
ans =
```

```
0 0.1667 0.3333 0.5000 0.6667 0.8333 1.0000  
-1.0000 -0.7049 -0.4769 -0.3025 -0.1705 -0.0721 0
```

To the approximations and superimpose a graph of the true solution we use the commands

```
>> exact=(x-1).*exp(-x);  
>> plot(x,approx,'dk','markersize',6)  
>> hold on  
>> plot(x,exact,'-k')
```

(The legend shown on the graph was added after the plot.)

