Boundary value problem

BVP (Boundary value problem)

$$\begin{cases} \mathbf{y}'' = \mathbf{f}(x, \mathbf{y}, \mathbf{y}') & \text{[Differential equation]} \\ \alpha_1 \mathbf{y}(a) + \beta_1 \mathbf{y}'(a) = \gamma_1 & \text{[Boundary condition on left]} \\ \alpha_2 \mathbf{y}(b) + \beta_2 \mathbf{y}'(b) = \gamma_2 & \text{[Boundary condition on right]} \end{cases}$$

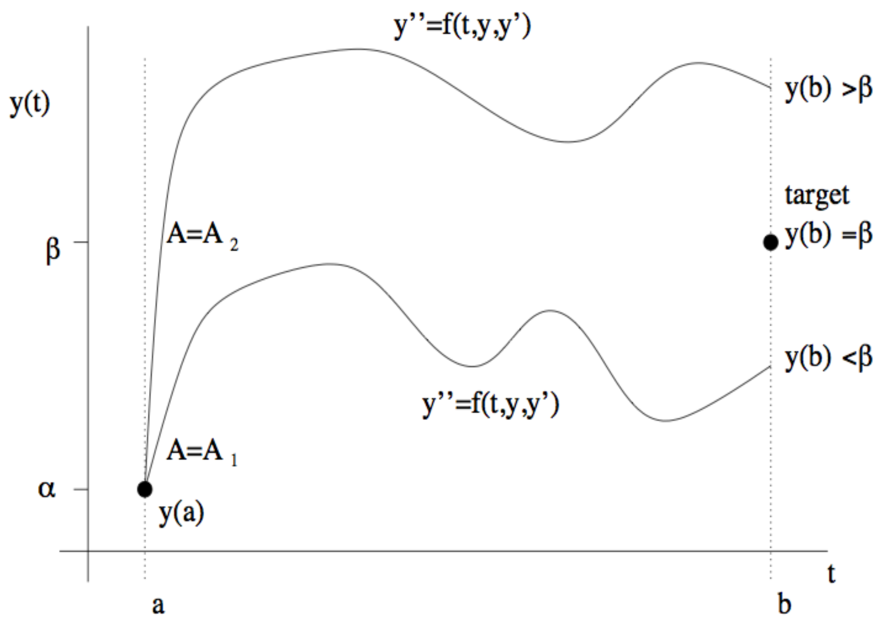
Shooting method

Repeat untill convergence:

[Step 1] Solve the following initial value problem.

$$\begin{cases} \mathbf{y}'' = \mathbf{f}(x, \mathbf{y}, \mathbf{y}') \\ \mathbf{y}(a) = \alpha \\ \mathbf{y}'(a) = A \end{cases}$$

[Step 2] Update the initial velocity $\mathbf{y}'(a) = A$ to hit the target $\mathbf{y}(b) = \beta$.

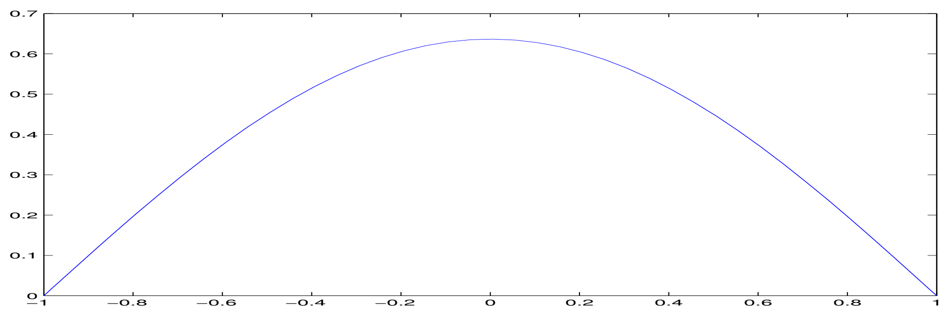


J. N. Kutz

Example - Shooting method - Part 1

Solve the following ODE with BC $y(-1) = 0$, $y(1) = 0$ with an extra parameter β:

$$y'' + (100 - \beta)y = 0$$

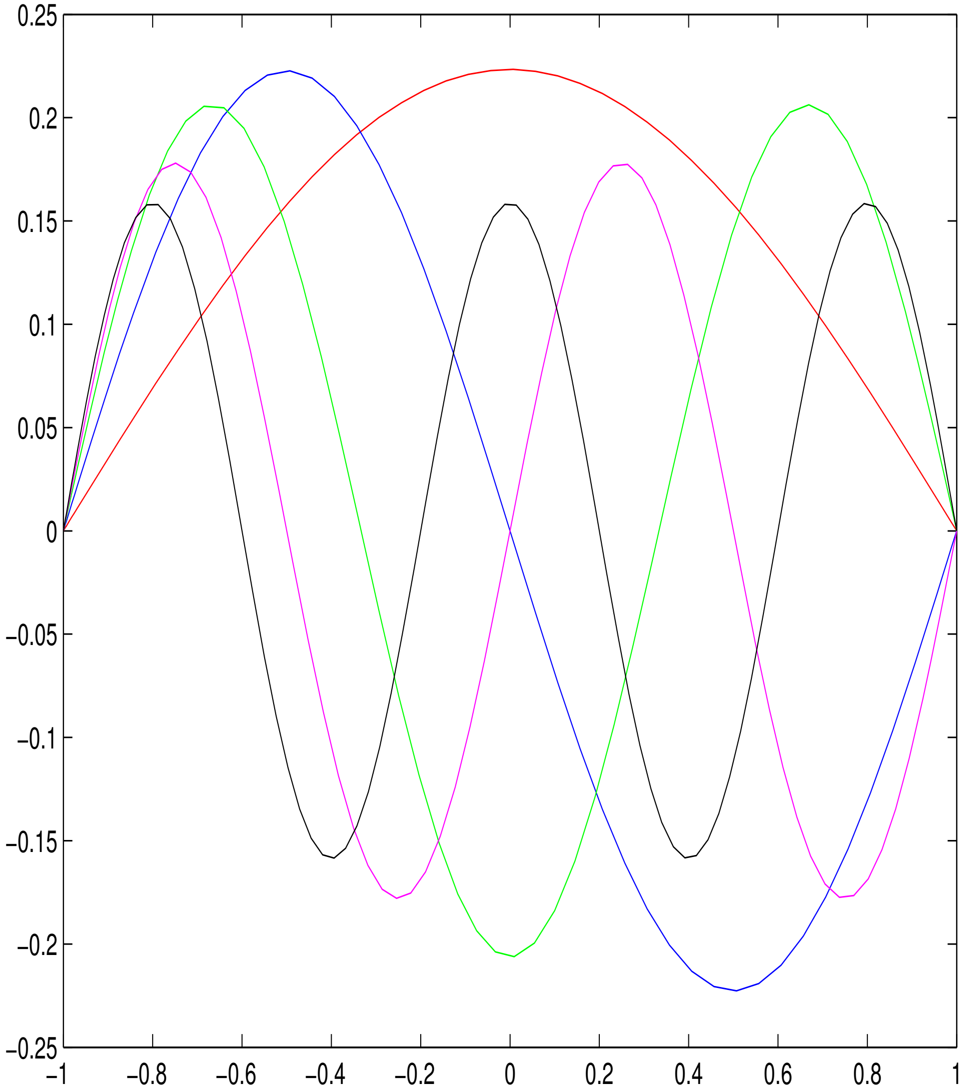



```
clear all; close all; clc;

tol = 1e-4;
NIter = 1000;

rhs_0615_2014 = @(t,y,bt) [y(2); (bt-100)*y(1)];
tspan = [-1 1];
A = 1; ic = [0; A];

bt = 99;
dbt = 1;

for j = 1:NIter

    [t,y] = ode45(rhs_0615_2014,tspan,ic,[],bt);

    if abs(y(end,1))<tol, j, bt, y(end,1), break, end

    if y(end,1)>0, bt = bt-dbt;
    else bt = bt+dbt/2; dbt = dbt/2; end

end

plot(t,y(:,1))
```

Example - Shooting method - Part 2

Find five different solutions of the following ODE with BC $y(-1) = 0$, $y(1) = 0$ with an extra parameter β:

$$y'' + (100 - \beta)y = 0$$



```
clear all; close all; clc;

tol = 1e-4;
NIter = 1000;

rhs_0615_2014 = @(t,y,bt) [y(2); (bt-100)*y(1)];
tspan = [-1 1];
A = 1; ic = [0; A];

bt = 99;
dbt = 1;

color = 'rbgmk';

for jj = 1:5

    for j = 1:NIter

        [t,y] = ode45('rhs_0615_2014',tspan,ic,[],bt);

        if (abs(y(end,1))<tol), j, bt, break, end

        if (y(end,1)*((-1)^(jj+1))>0), bt = bt-dbt;
        else bt = bt+dbt/2; dbt = dbt/2; end

    end

    no = norm(y(:,1));
    plot(t,y(:,1)/no,color(jj)); hold on

    bt = bt-1;
    dbt = 1;

end
```

$O(\Delta t^2)$ center-difference scheme

$$
\begin{aligned}
y'_n &= \frac{y_{n+1} - y_{n-1}}{2h} \\
y''_n &= \frac{y_{n+1} - 2y_n + y_{n-1}}{h^2} \\
y'''_n &= \frac{y_{n+2} - 2y_{n+1} + 2y_{n-1} - y_{n-2}}{2h^3} \\
y''''_n &= \frac{y_{n+2} - 4y_{n+1} + 6y_n - 4y_{n-1} + y_{n-2}}{h^4}
\end{aligned}
$$

$$
\begin{aligned}
f(t+\Delta t) &= f(t) + f'(t)\Delta t + \frac{1}{2}f''(t)\Delta t^2 + O(\Delta t^3) \\
f(t-\Delta t) &= f(t) - f'(t)\Delta t + \frac{1}{2}f''(t)\Delta t^2 + O(\Delta t^3) \\
& \Rightarrow \quad f(t+\Delta t) - f(t-\Delta t) = 2f'(t)\Delta t + O(\Delta t^3) \\
& \Rightarrow \quad f'(t) = \frac{f(t+\Delta t) - f(t-\Delta t)}{2\Delta t} + O(\Delta t^2)
\end{aligned}
$$

$$
\begin{aligned}
f(t+\Delta t) &= f(t) + f'(t)\Delta t + \frac{1}{2}f''(t)\Delta t^2 + O(\Delta t^3) \\
f(t-\Delta t) &= f(t) - f'(t)\Delta t + \frac{1}{2}f''(t)\Delta t^2 + O(\Delta t^3) \\
& \Rightarrow \quad f(t+\Delta t) + f(t-\Delta t) = 2f(t) + f''(t)\Delta t^2 + O(\Delta t^3) \\
& \Rightarrow \quad f''(t) = \frac{f(t+\Delta t) - 2f(t) + f(t-\Delta t)}{\Delta t^2} + O(\Delta t) \\
& \Rightarrow \quad f''(t) = \frac{f(t+\Delta t) - 2f(t) + f(t-\Delta t)}{\Delta t^2} + O(\Delta t^2) \quad (\Leftarrow \quad \text{Why ?})
\end{aligned}
$$

FDM

Divide x interval

Divide x interval of interest into small subintervals with equal length $\Delta x = h$.

$$x_0 \quad < \quad x_1 \quad < \quad x_2 \quad < \quad \cdots \quad < \quad x_N \quad < \quad x_{N+1}$$

$N+2$ unknowns and $N+2$ equations

$N+2$ unknowns	$\mathbf{y}_0,\ \mathbf{y}_1,\ \mathbf{y}_2, \cdots,\ \mathbf{y}_N,\ \mathbf{y}_{N+1}$
$N+2$ equations	2 equations from BC + N equations from ODE

$$\begin{array}{ccccccccccc}
\mathbf{y}_0 & & \mathbf{y}_1 & & \mathbf{y}_2 & & \cdots & & \mathbf{y}_N & & \mathbf{y}_{N+1} \\
x_0 & < & x_1 & < & x_2 & < & \cdots & < & x_N & < & x_{N+1} \\
\uparrow & & & & & & & & & & \uparrow \\
\text{BC} & & & & & & & & & & \text{BC}
\end{array}$$

$$\begin{array}{ccccccccccc}
\mathbf{y}_0 & & \mathbf{y}_1 & & \mathbf{y}_2 & & \cdots & & \mathbf{y}_N & & \mathbf{y}_{N+1} \\
x_0 & < & x_1 & < & x_2 & < & \cdots & < & x_N & < & x_{N+1} \\
 & & \uparrow & & \uparrow & & \uparrow & & \uparrow & & \\
 & & \text{ODE} & & \text{ODE} & & \text{ODE} & & \text{ODE} & &
\end{array}$$

FDM

Linear BVP	$\xrightarrow{\text{Discretization}}$	Linear equations	`Ax=b`
Non-linear BVP	$\xrightarrow{\text{Discretization}}$	Non-linear equations	`f(x)=b`

Example - FDM - Linear

Solve the following BVP:

$$y'' - y = 0, \qquad y(0) = 1, \quad y(1) = \frac{e + e^{-1}}{2}$$

$\boxed{\text{Divide } x \text{ interval}}$

Divide x interval of interest into small subintervals with equal length $\Delta x = h$.

$$x_0 \; < \; x_1 \; < \; x_2 \; < \; \cdots \; < \; x_N \; < \; x_{N+1}$$

$\boxed{\text{2 equations from BC}}$

$$\begin{array}{ccccccccccc} y_0 = 1 & & y_1 & & y_2 & & \cdots & & y_N & & y_{N+1} = \frac{e+e^{-1}}{2} \\ x_0 & < & x_1 & < & x_2 & < & \cdots & < & x_N & < & x_{N+1} \end{array}$$

$\boxed{\text{N equations from ODE}}$

$$\begin{array}{lll} \text{At } x_1 & \frac{y_2 - 2y_1 + y_0}{h^2} - y_1 = 0 & \Rightarrow \; y_2 - (2 + h^2)y_1 + y_0 = 0 \\ \text{At } x_2 & \frac{y_3 - 2y_2 + y_1}{h^2} - y_2 = 0 & \Rightarrow \; y_3 - (2 + h^2)y_2 + y_1 = 0 \\ \vdots & \vdots & \vdots \quad \vdots \\ \text{At } x_{N-1} & \frac{y_N - 2y_{N-1} + y_{N-2}}{h^2} - y_{N-1} = 0 & \Rightarrow \; y_N - (2 + h^2)y_{N-1} + y_{N-2} = 0 \\ \text{At } x_N & \frac{y_{N+1} - 2y_N + y_{N-1}}{h^2} - y_N = 0 & \Rightarrow \; y_{N+1} - (2 + h^2)y_N + y_{N-1} = 0 \end{array}$$

$\boxed{\text{FDM}}$

$$\underbrace{\begin{bmatrix} -(2+h^2) & 1 & & & \\ 1 & -(2+h^2) & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -(2+h^2) & 1 \\ & & & 1 & -(2+h^2) \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} -y_0 \\ 0 \\ \vdots \\ 0 \\ -y_{N+1} \end{bmatrix}}_{\mathbf{b}}$$

$$\boxed{\mathbf{A}\mathbf{x} = \mathbf{b} \quad \Rightarrow \quad \mathbf{x} = \mathbf{A} \backslash \mathbf{b}}$$

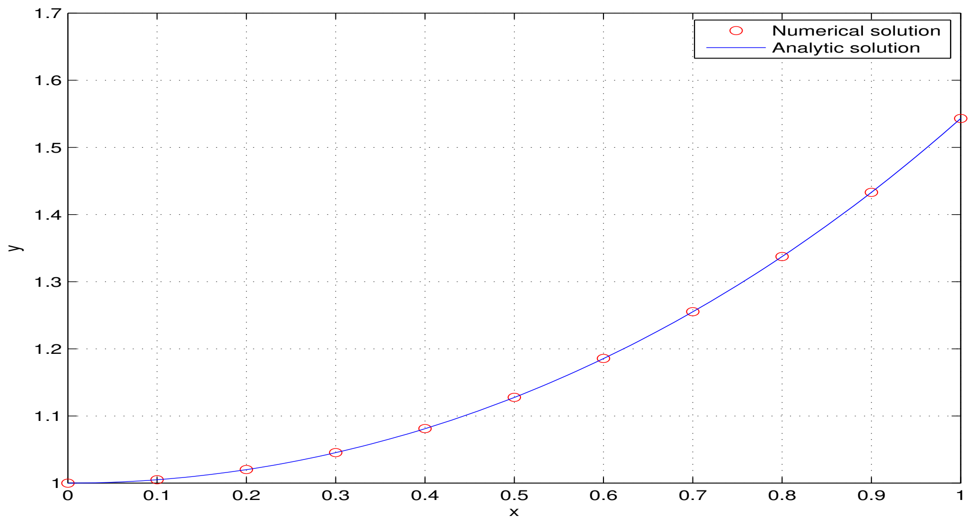


Figure 1: Comparison of analytic solution and numerical solution.

```
clear all; close all; clc;

n = 11;
h = 1/(n-1);

D = [ones(n,1) -(2+h^2)*ones(n,1) ones(n,1)];
d = [-1 0 1];
A = spdiags_Lee(D,d,n-2,n-2);
% spy(A) % Check the structure of the sparse matric A

y_0 = 1; y_end = (exp(1)+exp(-1))/2;
b = zeros(n-2,1); b(1) = -y_0; b(end) = -y_end;

y = A\b; y = [y_0; y; y_end];

x = linspace(0,1,n);
plot(x,y,'or'); grid on; hold on; xlabel('x'); ylabel('y');

% Analytic solution
z = 0:0.01:1;
w = (exp(z)+exp(-z))/2;
plot(z,w,'-'); legend('Numerical solution','Analytic solution');
```

Example - FDM - Non-linear

Solve the following BVP: With $\varepsilon = 0.1$ and $\mu = 3$,

$$y'' + \varepsilon e^{\frac{y}{1+\mu y}} = 0, \qquad y(-1) = 0, \quad y(1) = 0$$

Divide x interval

Divide x interval of interest into small subintervals with equal length $\Delta x = h$.

$$x_0 \;<\; x_1 \;<\; x_2 \;<\; \cdots \;<\; x_N \;<\; x_{N+1}$$

2 equations from BC

$$\begin{array}{ccccccccccc} y_0 = 0 & & y_1 & & y_2 & & \cdots & & y_N & & y_{N+1} = 0 \\ x_0 & < & x_1 & < & x_2 & < & \cdots & < & x_N & < & x_{N+1} \end{array}$$

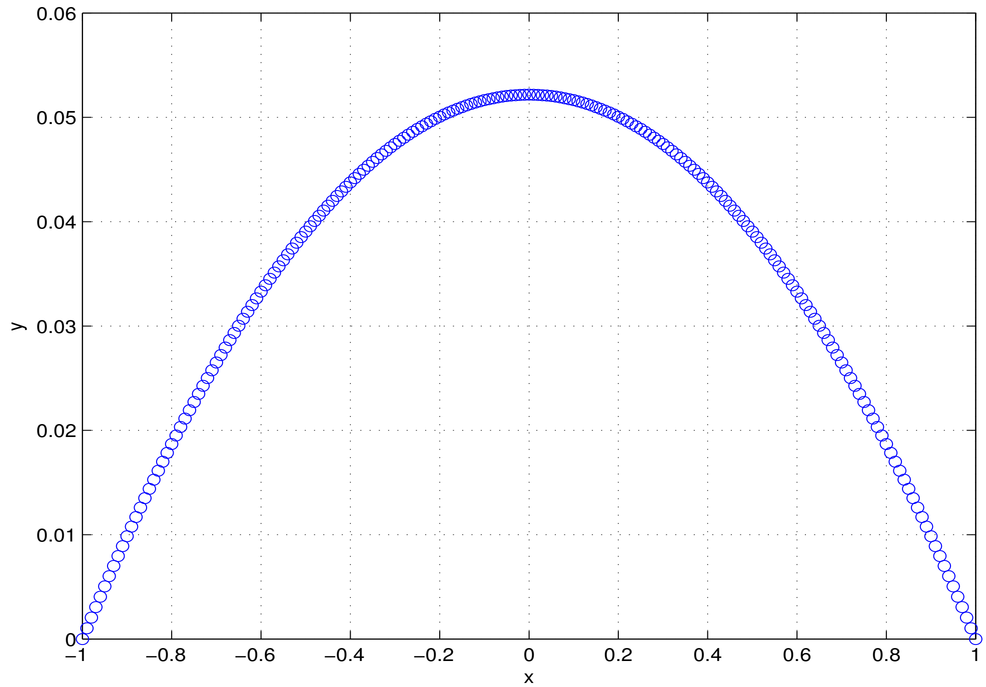
N equations from ODE

$$\begin{array}{llll}
\text{At } x_1 & \frac{y_2 - 2y_1 + y_0}{h^2} + \varepsilon e^{\frac{y_1}{1+\mu y_1}} = 0 & \Rightarrow & y_2 - 2y_1 + y_0 + h^2 \varepsilon e^{\frac{y_1}{1+\mu y_1}} = 0 \\
\text{At } x_2 & \frac{y_3 - 2y_2 + y_1}{h^2} + \varepsilon e^{\frac{y_2}{1+\mu y_2}} = 0 & \Rightarrow & y_3 - 2y_2 + y_1 + h^2 \varepsilon e^{\frac{y_2}{1+\mu y_2}} = 0 \\
\vdots & \vdots & \vdots & \vdots \\
\text{At } x_{N-1} & \vdots & \Rightarrow & y_N - 2y_{N-1} + y_{N-2} + h^2 \varepsilon e^{\frac{y_{N-1}}{1+\mu y_{N-1}}} = 0 \\
\text{At } x_N & \vdots & \Rightarrow & y_{N+1} - 2y_N + y_{N-1} + h^2 \varepsilon e^{\frac{y_N}{1+\mu y_N}} = 0
\end{array}$$

FDM

$$\underbrace{\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix} + \begin{bmatrix} h^2 \varepsilon e^{\frac{y_1}{1+\mu y_1}} \\ h^2 \varepsilon e^{\frac{y_2}{1+\mu y_2}} \\ \vdots \\ h^2 \varepsilon e^{\frac{y_{N-1}}{1+\mu y_{N-1}}} \\ h^2 \varepsilon e^{\frac{y_N}{1+\mu y_N}} \end{bmatrix} - \begin{bmatrix} -y_0 \\ 0 \\ \vdots \\ 0 \\ -y_{N+1} \end{bmatrix}}_{f(\mathbf{y})} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{0}}$$

$\boxed{f(\mathbf{y}) = \mathbf{0} \quad \Rightarrow \quad \mathbf{y} = fsolve(f, y_0)}$




```
clear all; close all; clc;

n = 201;
h = 2/(n-1);

ep = 0.1;
mu = 3;

% A = spdiags(D,d,m,n)
D = [-2*ones(n,1), ones(n,1), ones(n,1)];
d = [0, 1, -1];
A = spdiags_Lee(D,d,n-2,n-2);

% FDM
f = @(y) A*y + h^2*ep*exp(y./(1+mu*y));
yi = zeros(n-2,1);
y = fsolve(f,yi); y = [0; y; 0];

% Plot of FDM solution
x = linspace(-1,1,n);
plot(x,y,'o'); grid on; xlabel('x'); ylabel('y');
```

Example - How to implement BC at infinity

$$\text{[Differential equation]} \qquad \frac{d^2\Psi_n}{dx^2} + (n(x) - \beta_n)\Psi_n = 0$$
$$\text{[BC at infinity]} \qquad \Psi_n \to 0 \quad \text{as} \quad x \to \pm\infty$$

where

$$n(x) = \begin{cases} n_0(1 - |x|^2) & \text{for } |x| < 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\boxed{\Psi_n \to 0 \text{ as } x \to \infty \text{ means } \frac{d\Psi_n}{dx}(L) + \sqrt{\beta_n}\Psi_n(L) = 0}$$

$$n(x) = 0 \text{ for large } x \quad \Rightarrow \quad \text{[ODE]} \qquad \frac{d^2\Psi_n}{dx^2} - \beta_n\Psi_n = 0$$
$$\Rightarrow \quad \text{[Solution]} \qquad \Psi_n = C_1 e^{\sqrt{\beta_n}x} + C_2 e^{-\sqrt{\beta_n}x} = C_2 e^{-\sqrt{\beta_n}x}$$
$$\Rightarrow \quad \text{[ODE]} \qquad \frac{d\Psi_n}{dx} + \sqrt{\beta_n}\Psi_n = 0$$
$$\Rightarrow \quad \text{[BC at } x = L\text{]} \qquad \frac{d\Psi_n}{dx}(L) + \sqrt{\beta_n}\Psi_n(L) = 0$$

$$\boxed{\Psi_n \to 0 \text{ as } x \to -\infty \text{ means } \frac{d\Psi_n}{dx}(-L) - \sqrt{\beta_n}\Psi_n(-L) = 0}$$

$$n(x) = 0 \text{ for large } |x| \quad \Rightarrow \quad \text{[ODE]} \qquad \frac{d^2\Psi_n}{dx^2} - \beta_n\Psi_n = 0$$
$$\Rightarrow \quad \text{[Solution]} \qquad \Psi_n = C_1 e^{\sqrt{\beta_n}x} + C_2 e^{-\sqrt{\beta_n}x} = C_1 e^{\sqrt{\beta_n}x}$$
$$\Rightarrow \quad \text{[ODE]} \qquad \frac{d\Psi_n}{dx} - \sqrt{\beta_n}\Psi_n = 0$$
$$\Rightarrow \quad \text{[BC at } x = L\text{]} \qquad \frac{d\Psi_n}{dx}(-L) - \sqrt{\beta_n}\Psi_n(-L) = 0$$

bvp4c

bvp4c

```
solinit  =  bvpinit   (xinit,       yinit)
                                      ↑
                                 yinit=@(x)

  sol    =  bvp4c     (rhs,          bc,        solinit)
                        ↑            ↑
                   rhs=@(x,y)   bc=@(yl,yr)

   y     =  deval     (sol,          x)
                                     ↑
                              evaluation points
```

bvp4c with parameters

```
solinit  =  bvpinit   (xinit,       yinit,          bt)
                                      ↑              ↑
                               yinit=@(x,bt)     Parameter

  sol    =  bvp4c     (rhs,          bc,        solinit)
                        ↑            ↑
                  rhs=@(x,y,bt)  bc=@(yl,yr,bt)
```

bvp4c with options

```
options  =  bvpset  ('RelTol',  1e-2,  'AbsTol',  [1e-4 1e-5])

sol      =  bvp4c   (rhs,       bc,    solinit,   options);
```

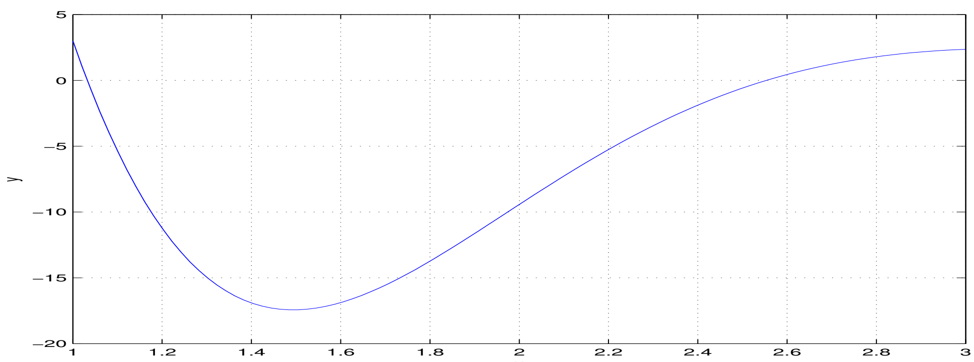
Example - bvp4c

Solve the following BVP:

$$y'' + 3y' + 6y = 5, \qquad y(1) = 3, \quad y(3) + 2y'(3) = 5$$

Change the 2nd order ODE into a system of first order ODE; with $y_1 = y$, $y_2 = y'$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ -6 & -3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 5 \end{bmatrix}, \qquad \begin{bmatrix} y_1(1) - 3 \\ y_1(3) + 2y_2(3) - 5 \end{bmatrix} = \mathbf{0}$$




```
clear all; close all; clc;

% [Step 1] bvpinit
xinit = linspace(1,3,10);
yinit = @(x)[x+2;0];
solinit = bvpinit(xinit,yinit);

% [Step 2] bvp4c
A = [0 1; -6 -3]; b = [0; 5];
RHS = @(x,y) A*y + b;
BC = @(yl,yr) [ yl(1) - 3; yr(1) + 2*yr(2) - 5 ];
sol = bvp4c(RHS,BC,solinit);

% [Step 3] deval
x = linspace(1,3);
y = deval(sol,x);

plot(x,y(1,:)); grid on; xlabel('x'); ylabel('y');
```

Example - Continuation

`solinit   --->   sol   --->   sol2`

For a fixed p, a standard test for any BVP code is to solve the following BVP:

$$y'' + \frac{3py}{(p+t^2)^2} = 0, \qquad y(-0.1) = -\frac{0.1}{\sqrt{p+0.01}}, \quad y(0.1) = \frac{0.1}{\sqrt{p+0.01}}$$

The above BVP has an analytic solution:

$$y = \frac{t}{\sqrt{p+t^2}}$$

```
clear all; close all; clc;

p = 1e-5;

% solinit
xinit = [-0.1 0.1];
yinit = @(x)[0 10]';
solinit = bvpinit(xinit,yinit);

% First solution
RHS = @(x,y)[y(2), -3*p*y(1)/(p+x^2)^2]';
BC = @(yl,yr)[yl(1)+0.1/sqrt(p+0.01), yr(1)-0.1/sqrt(p+0.01)]';
sol = bvp4c(RHS,BC,solinit);
x = linspace(-0.1,0.1,51); y = deval(sol,x); plot(x,y(1,:),'o'); hold on;

% Continuation solution
solinit2 = sol;
options = bvpset('RelTol',1e-4);
sol2=bvp4c(RHS,BC,solinit2,options);
y2 = deval(sol2,x); plot(x,y2(1,:))

% Analytic solution
xa = -0.1:0.001:0.1; ya = xa./sqrt(p+xa.^2); plot(xa,ya,'-r')
legend('First solution','Continuation solution','Analytic solution')
```

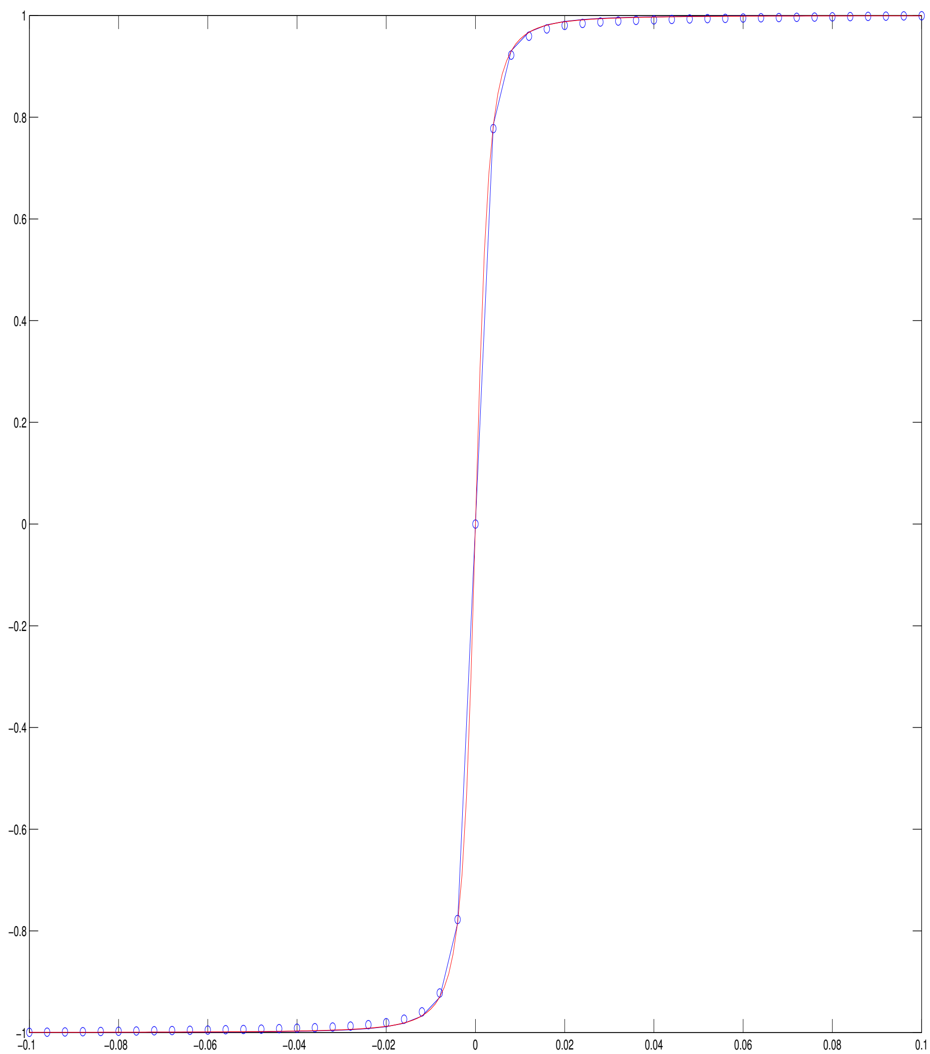


Figure 2: Red curve is the analytic solution. Circles are the first solution. Using this first solution as an initial guess Blue curve is the continuation solution.

Example - Solution of boundary value problem may be not unique - Version 1

Solve the following ODE:

$$y'' + |y| = 0, \qquad y(0) = 0, \quad y(4) = -2$$

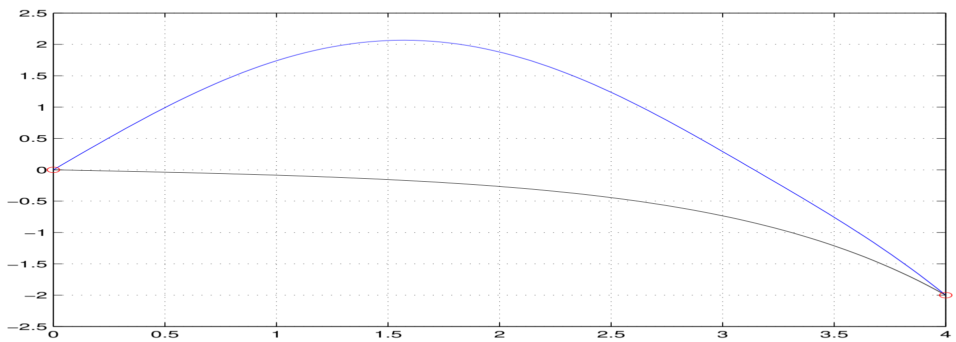


Figure 3: With an initial guess $yinit = @(x)[1;0]$, we have Blue curve as a solution. With a different initial guess $yinit = @(x)[-1;0]$, we have Black curve as a solution.

```
% [Step 1] bvpinit
xinit = [0 4]; yinit_first_component_list = [1 -1];
% yinit_first_component_list(1) leads first solution
% yinit_first_component_list(2) leads second solution

color_list = 'bk';
for i=1:2
% [Step 2] bvp4c
    yinit = @(x)[yinit_first_component_list(i), 0]';
    solinit = bvpinit(xinit,yinit);
    fode = @(x,y)[y(2), -abs(y(1))]'; fbc = @(yl,yr)[yl(1), yr(1)+2]';
    sol = bvp4c(fode,fbc,solinit);

    % [Step 3] deval
    x = linspace(0,4); y = deval(sol,x);
    plot(x,y(1,:),'-','color',color_list(i)) % plot of solution
    axis([0 4 -2.5 2.5]); hold on; grid on;
end

plot(0,0,'or',4,-2,'or')                        % plot of boundary condition
```

Example - Solution of boundary value problem may be not unique - Version 2

Solve the following BVP:

$$y'' + (100 - \beta)y + 10y^3 = 0, \qquad y(-1) = 0, \quad y(1) = 0$$

Change the 2nd order ODE into a system of first order ODE; with $y_1 = y$, $y_2 = y'$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}' = \begin{bmatrix} 0 & 1 \\ (\beta - 100) & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -10y_1^3 \end{bmatrix}, \qquad \begin{bmatrix} y_1(-1) \\ y_1(1) \\ y_2(-1) - 0.1 \end{bmatrix} = \mathbf{0}$$

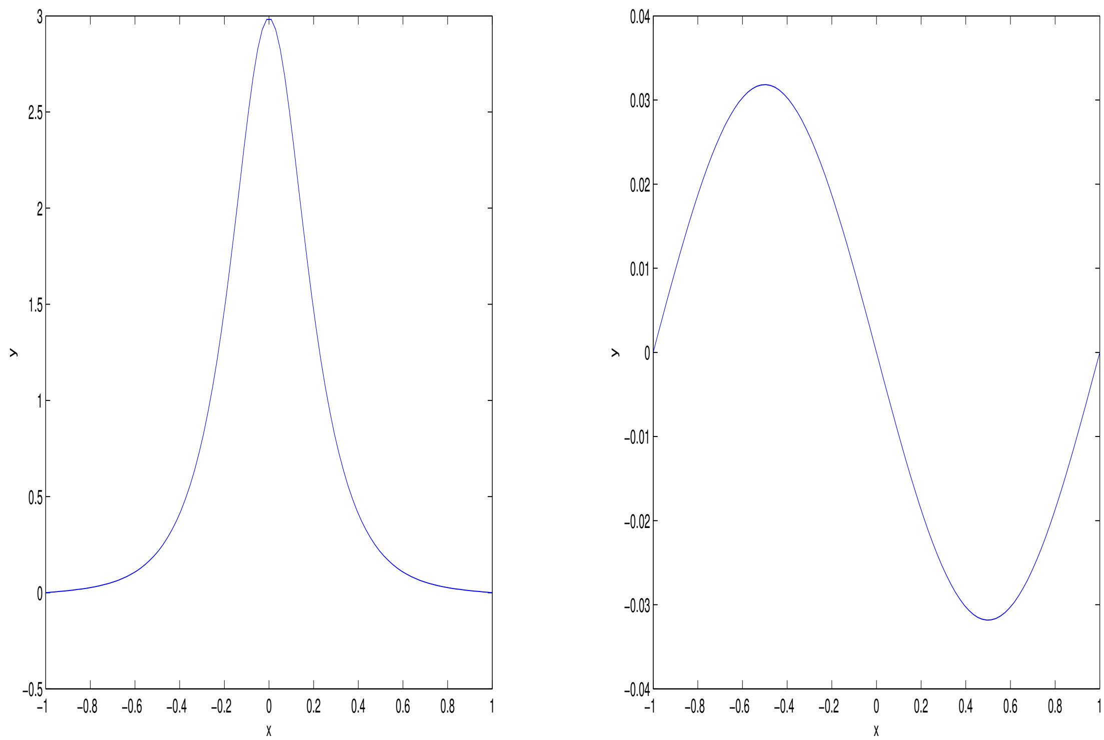


Figure 4: With an initial guess $yinit = @(x)[cos((pi/2)*x); -(pi/2)*sin((pi/2)*x)]$, we have left with $\beta = 144.6665$ as a solution. With a different initial guess $yinit = @(x)[sin(x); cos(x)]$, we have right with $\beta = 90.1380$ as a solution. With yet another initial guess $yinit = @(x)[0; 0]$, we have an error!

```
%% Different yinit produces different solution %%%%%%%%%%%%%%%%%%%%%%%%%

clear all; close all; clc;

% [Step 1] bvpinit
btinit = 99;
xinit = linspace(-1,1,50);
yinit = @(x) [cos((pi/2)*x); -(pi/2)*sin((pi/2)*x)];
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                  %
% Different yinit produces different solution      %
%                                                  %
% yinit = @(x) [sin(x); cos(x)] % bt = 90.1380     %
% yinit = @(x) [0; 0]; % Error!                    %
%                                                  %
% Different yinit produces different solution      %
%                                                  %
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
solinit = bvpinit(xinit,yinit,btinit);

% [Step 2] bvp4c
RBS = @(x,y,bt) [0 1; (bt-100) 0]*y + [0; -10*y(1)^3];
BC = @(yl,yr,bt) [ yl(1); yr(1); yl(2) - 0.1 ];
sol = bvp4c(RHS,BC,solinit);

% [Step 3] deval
x = linspace(-1,1);
y = deval(sol,x);

bt = sol.parameters
plot(x,y(1,:)); grid on; xlabel('x'); ylabel('y');

%% Output %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

bt =

  144.6665

%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```