Numerical Methods - Boundary Value Problems for ODEs

Y. K. Goh

Universiti Tunku Abdul Rahman

2013

Outline

Boundary Value Problems for ODEs

- In Initial Value Problem, $y'' = f(t, y, y')$, the value of y and y' are provided at a certain point, ie $y(a) = \alpha$ and $y'(a) = \beta$.
- In Boundary Value Problem, instead of the values of y and its derivative are given, the values of y at two different points (the boundaries) are given.

Definition (Two-Point Boundary Value Problems)

The two-point boundary value problems is a second-order differential equations of the form

$$y'' = f(t, y, y'), \text{ for } a \le t \le b,$$

subjected to the boundary conditions

$$y(a) = \alpha \quad \text{and} \quad y(b) = \beta.$$

Uniqueness of the solution for BVP

Theorem

Suppose the function f in the boundary value problem

$$y'' = f(t, y, y'), \quad \text{for } a \le t \le b, \text{ with } y(a) = \alpha \text{ and } y(b) = \beta,$$

is continuous on the set

$$D = \{(t, y, y') | \text{for } a \le t \le b, \text{ with } -\infty < y, y' < \infty\},$$

and that its partial derivatives f_y and $f_{y'}$ are also continuous on D. If

- *$f_y(t, y, y') > 0$, for all $(t, y, y') \in D$, and*
- *a constant M exists, with $|f_y(t, y, y')| \le M$, for all $(t, y, y') \in D$,*

then the boundary value problem has a unique solution.

Uniqueness of the solution for Linear BVP

Definition (Linear ODE)

A differential equation $y'' = f(t, y, y')$ is linear if f could be written as $f(t, y, y') = p(t)y' + q(t)y + r(t)$.

Theorem

Suppose the linear boundary value problem

$$y'' = p(t)y' + q(t)y + r(t), \quad \text{for } a \leq t \leq b, \text{ with } y(a) = \alpha \text{ and } y(b) = \beta,$$

satisfies

- $p(t), q(t)$ *and* $r(t)$ *are continuous on* $[a, b]$,
- $q(t) > 0$ *on* $[a, b]$.

Then the linear boundary value problem has a unique solution.

Outline

Shooting Method

- Suppose that we have a two-point BVP:

$$y'' = f(t, y, y'), \quad y(a) = \alpha, y(b) = \beta.$$

- Techniques for IVP such as Euler or Runge-Kutta will not work as two initial conditions are required to find the unique solution.
- This make BVP more difficult to solve than IVP.
- The shooting strategy to solve BVP is as follow:
  1. make a guess for $y'(a)$, say $y'(a) = z$.
  2. solve the IVP $y'' = f(t, y, y'), \quad y(a) = \alpha, y'(a) = z$ for $a \leq t \leq b$
  3. if $y(b) = \beta$ then stop, else if $y(b) \neq \beta$ then repeat the procedure with different guess of $y'(a)$.
- From the strategy we can say that $y(b)$ depends on our guess z for $y'(a)$, or we say $y(b) = \phi(z)$.
- We do not have much information on $\phi(z)$, but we can compute its value for different z.
- The shooting method uses root finding scheme to find the appropriate value of z such that $\phi(z) - \beta = 0$.

Shooting Method (Example)

Example

Apply the shooting method with secant method to the BVP:

$$y'' = \frac{1}{8}(32 + 2t^3 - yy'), \quad \text{for } 1 \le t \le 3, \text{ with } y(1) = 17 \text{ and } y(3) = \frac{43}{3}.$$

ANSWER: MATLAB code: `nm07_shoot.m`

- Setup the IVP. Let $x_1 = y$ and $x_2 = y'$, thus the ODE is now

$$\mathbf{X}'(t) = \begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{1}{8}(32 + 2t^3 - x_1 x_2) \end{bmatrix}$$

  and $\mathbf{X}(1) = [17, z]'$

- Choose a value z_0, and solve the IVP using an ODE solver, such as `ode45`.
- Choose another value z_1 and solve the IVP.
- Calculate z_3 using the secant method

$$z_{n+1} = z_n - (\phi(z_n) - \beta)\frac{z_n - z_{n-1}}{\phi(z_n) - \phi(z_{n-1})}$$

Outline

- Consider the BVP:

$$y'' = f((t, y, y'), \quad y(a) = \alpha, y(b) = \beta.$$

- Partition $[a, b]$ into N equally spaced subintervals with nodes $a = t_0 < t_1 < \cdots < t_{N-1} < t_N = b$, and spacing $h = (b - a)/N$.
- We could approximate the BVP with the corresponding discretised version by using the finite different fomula for the derivatives:

$$\begin{aligned} y'(t) &= \frac{y(t+h) - y(t-h)}{2h} \\ y''(t) &= \frac{y(t+h) - 2y(t) + y(t-h)}{h^2} \end{aligned}$$

- The approximated BVP is

$$\begin{cases} y_0 = \alpha \\ \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} = f(t_i, y_i, \frac{y_{i+1} - y_{i-1}}{2h}), 1 \leq i \leq N-1 \\ y_N = \beta \end{cases}$$

Discretisation Method for Linear BVP

- Let the ODE be $y'' = p(x)y' + q(x)y + r(x)$ and denote p_i, q_i and r_i are values of $p(t), q(t)$ and $r(t)$ at t_i repectively.
- Then the discretised ODE is

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

$$(-\frac{hp_i}{2} - 1)y_{i-1} + (2 + h^2 q_i)y_i + (\frac{hp_i}{2} - 1)y_{i+1} = -h^2 r_i$$

- Simplify the notation by defining:

$$\begin{aligned} a_i &= -(\frac{hp_i}{2} + 1), \\ b_i &= 2 + h^2 q_i, \\ c_i &= (\frac{hp_i}{2} - 1), \\ d_i &= -h^2 r_i, \end{aligned}$$

gives

$$a_i y_{i-1} + b_i y_i + c_i y_{i+1} = d_i.$$

- Since we know $y_0 = \alpha$ and $y_N = \beta$, put this information all into the equation $a_i y_{i-1} + b_i y_i + c_i y_{i+1} = d_i$ for $1 \leq i \leq N-1$, we have

$$\begin{cases} b_1 y_1 + c_1 y_2 &= d_1 - a_1 \alpha \\ a_i y_{i-1} + b_1 y_i + c_i y_{i+1} &= d_i \quad 2 \leq i \leq N-2 \\ a_{N-1} y_{N-2} + b_{N-1} y_{N-1} &= d_{N-1} - c_{N-1} \beta \end{cases}$$

- Which reduced to solving a linear system of $\mathbf{A}\mathbf{y} = \mathbf{d}$.

$$\begin{bmatrix} b_1 & c_1 & & & & \\ a_2 & b_2 & c_2 & & & \\ & a_3 & b_3 & c_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & a_{N-2} & b_{N-2} & c_{N-2} \\ & & & & a_{N-1} & b_{N-1} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{N-2} \\ y_{N-1} \end{bmatrix} = \begin{bmatrix} d_1 - a_1 \alpha \\ d_2 \\ d_3 \\ \vdots \\ d_{N-2} \\ d_{N-1} - c_{N-1} \beta \end{bmatrix}$$

Discretisation Method for Linear BVP (Example)

Example

Discretise the following BVP and solve the corresponding linear system:

$$y''(t) = \frac{2t}{1+t^2} y'(t) - \frac{2}{1+t^2} y(t) + 1, \quad y(0) = 1.25, y(4) = -0.95.$$

ANSWER: MATLAB code : `nm07_discretise.m`

Example

Discretise the following BVP and solve the corresponding linear system:

$$y''(t) = e^t - 3\sin(t) + y' - y, \quad y(1) = 1.09737491, y(2) = 8.63749661.$$

ANSWER: MATLAB code : `nm07_discretise.m`
[Exact solution is $y(t) = e^t - 3\cos(t)$.]

THE END