Relaxation: An effective way to solve BVP

Lecturer: Guanfu Liu(刘观福)

School of Physics and Astronmy
Sun Yat-Sen University

December 8, 2021

Overview

How to solve BVP

Two ways to solve boundary value problem (BVP). BVP is everywhere in the research fields of astronomy.

- Shooting method
- **Relaxation** method

A visual interpretation about relaxation

In numerical mathematics, relaxation methods are iterative methods for solving systems of equations, including nonlinear systems [1].

Just imagine a half-circle rubber band whose two ends were fixed at two certain points. The half-circle rubber band is tight initially. Then, it will **relax** to the relaxation state.

It should be noted that the initial condition is not that important for the rubber band will relax to the relaxation state sooner or later, which is the same as the condition in numerical practice.

Derivation of relaxation

Replace ordinary differential equations(ODEs) with finite-difference equations(PDEs).

$$\frac{dy}{dx} = g(x, y) \tag{1}$$

$$y_k - y_{k-1} - (x_k - x_{k-1})g[\frac{1}{2}(x_k + x_{k-1}), \frac{1}{2}(y_k + y_{k-1})] = 0 \tag{2}$$

Define:

$$E_k = y_k - y_{k-1} - (x_k - x_{k-1})g[\frac{1}{2}(x_k + x_{k-1}), \frac{1}{2}(y_k + y_{k-1})] \tag{3}$$

If we have **N** varibles to solve in **M** points:

$$\vec{E}_k = \vec{y}_k - \vec{y}_{k-1} - (x_k - x_{k-1})\vec{g}(x_k, x_{k-1}, \vec{y}_k, \vec{y}_{k-1}) \qquad \text{k=2,3,4...M} \tag{4}$$

There are **(M-1)N** equations.

Derivation of relaxation

First boundary:

$$0 = \vec{E}_1 \equiv \vec{B}(x_1, \vec{y}_1) \tag{5}$$

$\vec{E}_1$ has n_1 components. There are n_1 equations at the first boundary.

Second boundary:

$$0 = \vec{E}_{M+1} \equiv \vec{C}(x_M, \vec{y}_M) \tag{6}$$

$\vec{E}_{M+1}$ has $N - n_1$ components. There are $N - n_1$ equations at the second boundary.

Derivation of relaxation

The initial guess for $y_{j,k}$ is not that important. The increments $\Delta y_{j,k}$ matters. At interior points:

$$\vec{E}_k(\vec{y}_k + \Delta\vec{y}_k, \vec{y}_{k-1} + \Delta\vec{y}_{k-1}) \approx \vec{E}_k(\vec{y}_k, \vec{y}_{k-1}) + \sum_{n=1}^{N} \frac{\partial \vec{E}_k}{\partial y_{n,k-1}} \Delta y_{n,k-1} + \sum_{n=1}^{N} \frac{\partial \vec{E}_k}{\partial y_{n,k}} \Delta y_{n,k} \tag{7}$$

We expected the updated values $\vec{E}_k(\vec{y}_k + \Delta\vec{y}_k, \vec{y}_{k-1} + \Delta\vec{y}_{k-1})$ to be **zero**. Then, we can get:

$$\sum_{n=1}^{N} S_{j,n} \Delta y_{n,k-1} + \sum_{n=N+1}^{2N} S_{j,n} \Delta y_{n-N,k} = -E_{j,k} \qquad j = 1, 2, \cdots, N \tag{8}$$

Derivation of relaxation

where

$$S_{j,n} = \frac{\partial E_{j,k}}{\partial y_{n,k-1}}, S_{j,n+N} = \frac{\partial E_{j,k}}{\partial y_{n,k}}, \qquad n = 1, 2, \cdots, N \tag{9}$$

At the first boundary:

$$\sum_{n}^{N} S_{j,n} \Delta y_{n,1} = -E_{j,1}, \qquad j = N - n_1 + 1, N - n_1 + 2, \cdots, N \tag{10}$$

where

$$S_{j,n} = \frac{\partial E_{j,1}}{\partial y_{n,1}} \qquad n = 1, 2, \cdots, N \tag{11}$$

At the second boundary:

$$\sum_{n}^{N} S_{j,n} \Delta y_{n,M} = -E_{j,M+1}, \qquad j = 1, 2, \cdots, n_1 \tag{12}$$

where

$$S_{j,n} = \frac{\partial E_{j,M+1}}{\partial y_{n,M}} \qquad n = 1, 2, \cdots, N \tag{13}$$

Derivation of relaxation

The dimension of S is **MN**$\times$**MN**, with at most $MN \times N \times 2 - 2N = \mathbf{2MN^2 - 2N}$ non-zero elements (**Sparse matrix!**).
About convergence:

$$\text{err} = \frac{1}{MN}\sum_{k=1}^{M}\sum_{j=1}^{N}\frac{|\Delta y(j,k)|}{\text{scalv(j)}} \tag{14}$$

We need to supply to an array **scalv** which measures **typical size** of each varible.

conv is a preset parameters, which corresponds the accuracy of the ODEs.

When the $\text{err} \leq \text{conv}$, the method has converged.

Derivation of relaxation

If **err** is large, it perhaps means the corrections generated from a first-order Taylor series are inaccurate. Under this circumstances, we apply only **a fraction of** the corrections.(Why?)

$$\vec{E}_k(\vec{y}_k + \Delta\vec{y}_k, \vec{y}_{k-1} + \Delta\vec{y}_{k-1}) \approx \vec{E}_k(\vec{y}_k, \vec{y}_{k-1}) + \sum_{n=1}^{N} \frac{\partial \vec{E}_k}{\partial y_{n,k-1}} \Delta y_{n,k-1} + \sum_{n=1}^{N} \frac{\partial \vec{E}_k}{\partial y_{n,k}} \Delta y_{n,k} \tag{15}$$

$$y_{j,k} \rightarrow y_{j,k} + \frac{\mathrm{slowc}}{\max(\mathrm{slowc}, \mathrm{err})} \Delta y_{j,k} \tag{16}$$

A simple example

$$\psi'' + k^2\psi = 0 \tag{17}$$
$$\psi(-1) = \psi(1) = 0 \tag{18}$$

$$y_1 = \psi \tag{19}$$
$$y_2 = \psi' \tag{20}$$
$$y_3 = k^2 \tag{21}$$

$$E_{1,k} = y_{1,k} - y_{1,k-1} - \frac{h}{2}(y_{2,k} + y_{2,k-1}) \tag{22}$$
$$E_{2,k} = y_{2,k} - y_{2,k-1} + \frac{1}{4}h(y_{3,k} + y_{3,k-1})(y_{1,k} + y_{1,k-1}) \tag{23}$$
$$E_{3,k} = y_{3,k} - y_{3,k-1} \tag{24}$$

Here, k=2,3,4...M

$$h = x_k - x_{k-1} = \text{constant} \tag{25}$$

At interior points:

$$\begin{cases} S_{1,1} = -1, & S_{1,2} = -\frac{1}{2}h, \quad S_{1,3} = 0 \\ S_{1,4} = 1, & S_{1,5} = -\frac{1}{2}h, \quad S_{1,6} = 0 \end{cases} \tag{26}$$

$$\begin{cases} S_{2,1} = \frac{h}{4}(y_{3,k} + y_{3,k-1}), \quad S_{2,2} = -1, & S_{2,3} = \frac{h}{4}(y_{1,k} + y_{1,k-1}) \\ S_{2,4} = \frac{h}{4}(y_{3,k} + y_{3,k-1}), \quad S_{2,5} = 1, & S_{2,6} = \frac{h}{4}(y_{1,k} + y_{1,k-1}) \end{cases} \tag{27}$$

$$\begin{cases} S_{3,1} = 0, \quad S_{3,2} = 0, & S_{3,3} = -1 \\ S_{3,4} = 0, \quad S_{3,5} = 0, & S_{3,6} = 1 \end{cases} \tag{28}$$

A simple example

At the first boundary:

$$x = -1 : \begin{cases} E_{3,1} = y_{1,1} \\ S_{3,4} = 1, \qquad S_{2,5} = 0, \quad S_{2,6} = 0 \end{cases} \tag{29}$$

At the second boundary:

$$x = 1 : \begin{cases} E_{1,M+1} = y_{1,M} - 0 \\ S_{1,4} = 1, \qquad S_{1,5} = 0, \quad S_{1,6} = 0 \end{cases} \tag{30}$$

$$x = 1 : \begin{cases} E_{2,M+1} = y_{2,M} - 5 \\ S_{2,4} = 0, \qquad S_{2,5} = 1, \quad S_{1,6} = 0 \end{cases} \tag{31}$$

Here, we add another boundary conditions at the second boundary: $\psi'(1) = 5$

A simple example

3 equations, 4 mesh points.

$$
\begin{bmatrix}
\boldsymbol{X} & \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\boldsymbol{X} & \boldsymbol{X} & \mathbf{0} & \boldsymbol{X} & \boldsymbol{X} & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & 0 \\
\boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & 0 & 0 & 0 & 0 & 0 & 0 \\
\mathbf{0} & \mathbf{0} & \boldsymbol{X} & \mathbf{0} & \mathbf{0} & \boldsymbol{X} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \boldsymbol{X} & \boldsymbol{X} & \mathbf{0} & \boldsymbol{X} & \boldsymbol{X} & \mathbf{0} & 0 & 0 & 0 \\
0 & 0 & 0 & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & 0 & 0 & 0 \\
0 & 0 & 0 & \mathbf{0} & \mathbf{0} & \boldsymbol{X} & \mathbf{0} & \mathbf{0} & \boldsymbol{X} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{X} & \boldsymbol{X} & \mathbf{0} & \boldsymbol{X} & \boldsymbol{X} & \mathbf{0} \\
0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{X} \\
0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} & \mathbf{0} & \boldsymbol{X} & \mathbf{0} & \mathbf{0} & \boldsymbol{X} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \boldsymbol{X} & \mathbf{0} & \mathbf{0} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{0} & \boldsymbol{X} & \mathbf{0}
\end{bmatrix}
\begin{bmatrix}
\Delta y_{1,1} \\ \Delta y_{2,1} \\ \Delta y_{3,1} \\ \Delta y_{1,2} \\ \Delta y_{2,2} \\ \Delta y_{3,2} \\ \Delta y_{1,3} \\ \Delta y_{2,3} \\ \Delta y_{3,3} \\ \Delta y_{1,4} \\ \Delta y_{2,4} \\ \Delta y_{3,4}
\end{bmatrix}
=
\begin{bmatrix}
-E_{3,1} \\ -E_{1,2} \\ -E_{2,2} \\ -E_{3,2} \\ -E_{1,3} \\ -E_{2,3} \\ -E_{3,3} \\ -E_{1,4} \\ -E_{2,4} \\ -E_{3,4} \\ -E_{1,5} \\ -E_{2,5}
\end{bmatrix}
\tag{32}
$$

Here $\boldsymbol{X}$ is the non-zero elements, $\mathbf{0}$ is the elements that are zero in this certain example but can be non-zero in another example.

A simple example

Process:

1. An educated or random guess of all $y_{j,k}$.
2. Get the square matrix in the left hand side of the equation and the column vector in the right hand side.
3. Solve the linear equations, get all $\Delta y_{j,k}$.
4. Calculate the **err**, the new $y_{j,k}$ is $y_{j,k} + \frac{\mathrm{slowc}}{\max(\mathrm{slowc},\mathrm{err})}\Delta y_{j,k}$.
5. Repeat the steps above until the **err** is less than **conv**.

Note that:

If you input some wrong elements in the square matrix or the column vector in the right hand side, you are likely to find that the **err** will oscillate, then become larger and larger, and the BVP solver will fail. Consequently, if you are in this condition, check the square matrix first.

Code Analysis

Let's come stright to the code!

[1] Wikipedia contributors. Relaxation (iterative method) — Wikipedia, the free encyclopedia, 2021. [Online; accessed 1-December-2021].