Boundary Value Problems
Ch. 27

Lecture Objectives

• To understand the difference between an initial value and boundary value ODE
• To be able to understand when and how to apply the shooting method and FD method.
• To understand what an Eigenvalue Problem is.

Initial Value Problems

• These are the types of problems we have been solving with RK methods

\[
\frac{dy_1}{dt} = f_1(t, y_1, y_2) \\
\frac{dy_2}{dt} = f_2(t, y_1, y_2)
\]

Subject to:

\[
\begin{align*}
t &= 0 \\
y_1(t = 0) &= Y_1 \\
y_2(t = 0) &= Y_2
\end{align*}
\]
Initial Value Problems

- These are the types of problems we have been solving with RK methods

\[
\frac{dy_1}{dt} = f_1(t, y_1, y_2)
\]
\[
\frac{dy_2}{dt} = f_2(t, y_1, y_2)
\]

All conditions are specified at the same value of the independent variable!

Subject to:
\[ t = 0 \]
\[ y_1(t = 0) = Y_1 \]
\[ y_2(t = 0) = Y_2 \]

Boundary Value Problems

- Auxiliary conditions are specified at the boundaries (not just a one point like in initial value problems)

Two Methods:
- Shooting Method
- Finite Difference Method

conditions are specified at different values of the independent variable!
Shooting Method

• Applicable to both linear & non-linear Boundary Value (BV) problems.
• Easy to implement
• No guarantee of convergence
• Approach:
  – Convert a BV problem into an initial value problem
  – Solve the resulting problem iteratively (trial & error)
  – Linear ODEs allow a quick linear interpolation
  – Non-linear ODEs will require an iterative approach similar to our root finding techniques.

Shooting Method – Cooling fin Example

\[ \frac{d^2 T}{dx^2} - \frac{hP}{kA}(T - T_c) = 0 \]

\[ T(x = 0) = T_0 \]
\[ T(x = L) = T_1 \]

Analytical Solution

\[ m^2 = \frac{hP}{kA} \]
\[ \theta(x) = T(x) - T_c \]
\[ \frac{d^2 \theta}{dx^2} - m^2 \theta = 0 \]

\[ \theta(x) = e^{mx} + e^{-mx} \]

Boundary Conditions

\[ T(x = 0) = T_0 \quad \theta(x = 0) = \theta_0 \]
\[ T(x = L) = T_1 \quad \theta(x = L) = \theta_L \]
\[ \theta(x) = c_1e^{mx} + c_2e^{-mx} = \theta(x) \]

\[ \theta(x) = \frac{\theta_0 - \theta_L}{mL} \sinh mx + \sinh m(L - x) \]

\[ \frac{\theta(x)}{\sinh mL} \]
Shooting Method – Basic Method - Cooling fin Example

\[ \frac{dT}{dx} \cdot \frac{hP}{kA} (T - T_\infty) = 0 \]

\[ T(x = 0) = T_0 \]

\[ T(x = l) = T_1 \]

1. Rewrite as two first order ODEs

\[ \frac{dT}{dx} = z \]

\[ \frac{dz}{dx} = \frac{hP}{kA} (T - T_\infty) \]

2. We need an initial value for \( z \); Guess:

\[ T(x = 0) = T_0 \]

\[ z(x = 0) = z_i \]

3. Integrate the two equations using RK4 and \( z_1 \); this will yield a solution at \( x = l \)

4. Integrate the two equations again using a 2nd guess guess for \( z(x=0)=z_2 \).

5. Linearly interpolate the \( z \) results to obtain the correct initial condition (Note: this only works for Linear ODEs.

Example:

\[ z_{as} = z_2 + \frac{z_2 - z_1}{T_1 - T_0} (T_{as} - T_0) \]

Matlab implementation (rung4_fin_multieqn.m)

Recast the problem:

\[ T = y_1 \]

\[ \frac{dT}{dx} = y_2 \]

\[ \frac{dy_1}{dx} = y_2 \]

\[ \frac{dy_2}{dx} = m^2 (y_1 - T_\infty) \]

\[ T(x) = 23.62e^{0.1x} + 196.12e^{-0.1x} \]

\[ y_1(x = 0) = T_0 = 200 \]

\[ y_1(x = 0) = G_i \]
Non-Linear BV Problems - Shooting Method

- Linear interpolation between 2 solutions will not necessarily result in a good estimate of the required boundary conditions.
- Recast the problem as a Root finding problem.
- The solution of a set of ODEs can be considered a function \( g(z_0) \) where \( z_0 \) is the initial condition that is unknown.
  \[ g(z_0) = f(z_0) - y_e \]
- Drive \( g(z_0) \rightarrow 0 \) to get our solution.
- Iteratively adjust your guess.

Non-linear Shooting method – Secant Method

Consider the following ODEs system:
\[
\begin{align*}
\frac{dy}{dx} &= z \\
\frac{dz}{dx} &= f(x,y,z) \\
y(a) &= 0 \\
z(b) &= y_e
\end{align*}
\]

1. Guess an initial value of \( z \) (i.e., \( z(a) \)) just as was done with the linear method. Using RK4 or some other ODE method, we will obtain solution at \( y(b) \).

2. Denote the difference between the boundary condition and our result from the integration as some function \( m \).
   \[ m(z) = y(b) - y_{true}(b) \]
   Find the zero of this function
   \[ m(z) = y_{true}(b) - y_{true}(b) \]

3. Check to see if \( m \) is within an acceptable tolerance. Have we satisfied the boundary condition \( y(b) \)?
   \[ \varepsilon \leq \varepsilon_i \]
   \[ \varepsilon = \left| \frac{m_i - m_{i-1}}{m_i} \right| \]

4. If not, then use the Secant Method to determine our next guess.
   \[
   z_i = z_{i-1} - \frac{m(z_{i-1})}{m(z_{i-1})} \\
z_{i} = z_{i-1} - \frac{z_{i-1} - z_{i-2}}{m_{i-1} - m_{i-2}} m_{i-1}
   \]
Non-linear Shooting method – Secant Method

y(a) = 0
y(b) = y₁

z(α) = Guess
y(a) = 0

Solve with RK4

y(x)
z(x)
y(b)

m = y₁(b) - y_guess(b)

Is m small enough?

write out solution

Finite Difference Method

• Alternative to the shooting method
• Substitute finite difference equations for derivatives in the original ODE.
• This will give us a set of simultaneous algebraic equations that are solved at nodes.
• Recall using central differencing:

\[
\frac{dT}{dx} = \frac{T_{i+1} - T_{i-1}}{2\Delta x}
\]

\[
\frac{d^2 T}{dx^2} = \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2}
\]

Finite Difference Method

\[
\frac{d^2 T}{dx^2} - m^2 (T - T_e) = 0
\]

Rewrite in finite (central) difference form:

\[
\frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} - m^2 (T_i - T_e) = 0
\]

Multiply by \(\Delta x^2\) and solve for \(T_i\):

\[
T_{i+1} - 2T_i + T_{i-1} - m^2 (T_i - T_e)\Delta x^2 = 0
\]

\[
-T_{i+1} + 2T_i - T_{i-1} + m^2 (T_i - T_e)\Delta x^2 = 0
\]

\[
T_{i+1} + (2 + m^2 \Delta x^2)T_i - T_{i-1} = m^2 (T_e)\Delta x^2
\]
Finite Difference Method

General Equation:
\[-T_{i,j} + (\frac{2}{m^2} + m \Delta x^2)T_{i,j} - T_{i-1,j} = m^2 (T_{i,j} - \Delta x^2)\]

Write out for all nodes:
\[-T_{1,j} + (\frac{2}{m^2} + m \Delta x^2)T_{1,j} - T_{0,j} = m^2 (T_{1,j} - \Delta x^2)\]
\[-T_{2,j} + (\frac{2}{m^2} + m \Delta x^2)T_{2,j} - T_{1,j} = m^2 (T_{2,j} - \Delta x^2)\]
\[-T_{3,j} + (\frac{2}{m^2} + m \Delta x^2)T_{3,j} - T_{2,j} = m^2 (T_{3,j} - \Delta x^2)\]
\[-T_{4,j} + (\frac{2}{m^2} + m \Delta x^2)T_{4,j} - T_{3,j} = m^2 (T_{4,j} - \Delta x^2)\]

Apply boundary conditions:
\[
m^2 = 50\]
\[
T_1 = 200\]
\[
T_5 = 30\]
\[
\Delta x = 0.1\]
\[
-200 + (2 + 0.5)T_2 - T_3 = 12.5\]
\[
-T_2 + (2 + 0.5)T_2 - T_3 = 12.5\]
\[
-T_1 + (2 + 0.5)T_2 - T_3 = 12.5\]
\[
-T_5 + (2 + 0.5)T_1 - 30 = 12.5\]

4 Equations, 4 unknowns

Finite Difference Method

Put in Matrix form:
\[
\begin{bmatrix}
2.5 & -1 & 0 & 0 & T_1 \\
-1 & 2.5 & -1 & 0 & T_2 \\
0 & -1 & 2.5 & -1 & T_3 \\
0 & 0 & -1 & 2.5 & T_4
\end{bmatrix}
= \begin{bmatrix}
12.5 \\
12.5 \\
42.5
\end{bmatrix}
\]

Fast, easy to implement technique for solving ODEs

Finite Difference Method – Extended to PDEs

Consider a simple Elliptical Equation: LaPlace’s Equation
\[
\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = R(x,y)
\]

This could describe the steady state temperature distribution in 2D metal plate.

Discretize (write in finite difference form) our PDE using Central Difference technique:
\[
\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = R_{i,j}
\]
### Finite Difference Method – Extended to PDEs

Consider a simple Elliptical Equation: Laplace’s Equation

\[
\frac{T_{i,j} - 2T_{i,j} + T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1} - R_{i,j}}{\Delta x^2} = \Delta y^2
\]

\[
\Delta = x = y
\]

If \( R = 0 \)

\[
\frac{T_{i,j} + T_{i+1,j} + T_{i,j+1} + T_{i,j-1}}{4}
\]

Suppose we have a heated plate with Dirchlet boundary conditions

\[
T_{i,j} = T_{i-1,j} + T_{i+1,j} + T_{i,j-1} + T_{i,j+1}
\]

We can easily use Gauss-Seidel to solve our system of equations until:

\[
\varepsilon \leq \varepsilon_0
\]

\[
\varepsilon = \frac{|T_{i,j}^{n+1} - T_{i,j}^{old}|}{T_{i,j}^{new}}
\]
Finite Difference Method – Extended to PDEs

Heated Plate Matlab Example