Numerical Analysis

Lecture 13: Solution of Differential Equations

Instructor: Prof. Qiwei Zhan

Zhejiang University

Outline

Systems of Differential Equations

An mth-order system of first-order initial-value problems has the form

$$\begin{aligned}
\frac{du_1}{dt} &= f_1(t, u_1, u_2, \cdots, u_m) \\
\frac{du_2}{dt} &= f_2(t, u_1, u_2, \cdots, u_m) \\
&\vdots \\
\frac{du_m}{dt} &= f_m(t, u_1, u_2, \cdots, u_m)
\end{aligned}$$

for $a \leq t \leq b$, with the initial conditions

$$u_1(a) = \alpha_1, u_2(a) = \alpha_2, \cdots, u_m(a) = \alpha_m.$$

Classical Runge-Kutta Order 4 Method

$$
\begin{aligned}
w_0 &= \alpha \\
k_1 &= hf(t_i, w_i) \\
k_2 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_1\right) \\
k_3 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{1}{2}k_2\right) \\
k_4 &= hf(t_{i+1}, w_i + k_3) \\
w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)
\end{aligned}
$$

Systems of Differential Equations

Runge-Kutta Order 4 Method to Differential System

$$
\begin{aligned}
w_{i,0} &= \alpha_i, i = 1, 2, \cdots, m \\
k_{1,i} &= hf_i(t_j, w_{1,j}, w_{2,j}, \cdots, w_{m,j}) \\
k_{2,i} &= hf_i(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{1,1}, w_{2,j} + \frac{1}{2}k_{1,2}, \cdots, w_{m,j} + \frac{1}{2}k_{1,m}) \\
k_{3,i} &= hf_i(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{2,1}, w_{2,j} + \frac{1}{2}k_{2,2}, \cdots, w_{m,j} + \frac{1}{2}k_{2,m}) \\
k_{4,i} &= hf_i(t_j + h, w_{1,j} + k_{3,1}, w_{2,j} + k_{3,2}, \cdots, w_{m,j} + k_{3,m}) \\
w_{i,j+1} &= w_{i,j} + \frac{1}{6}(k_{1,i} + 2k_{2,i} + 2k_{3,i} + k_{4,i})
\end{aligned}
$$

Outline

Higher-Order Differential Equations

Higher-Order Differential Equations

An general *m*th-order initial problem

$$y^{(m)}(t) = f(t, y, y', \cdots, y^{(m-1)}), a \leq t \leq b,$$

with initial conditions

$$y(a) = \alpha_1, y'(a) = \alpha_2, \cdots, y^{(m-1)}(a) = \alpha_m.$$

Solutions

Let $u_1(t) = y(t), u_2(t) = y'(t), \cdots$, and $u_m(t) = y^{(m-1)}(t)$. This produces the first-order system

$$\frac{du_1}{dt} = u_2, \frac{du_2}{dt} = u_3, \cdots, \frac{du_{m-1}}{dt} = u_m,$$

and

$$\frac{du_m}{dt} = y^{(m)}(t) = f(t, y, y', \cdots, y^{(m-1)}) = f(t, u_1, u_2, \cdots, u_m)$$

with initial conditions

$$u_1(a) = \alpha_1, u_2(a) = \alpha_2, \cdots, u_m(a) = \alpha_m.$$

Outline

Boundary-Value Problems for ODEs

The two-point boundary-value problems involve a second-order differential equation of the form

$$y'' = f(x, y, y'), \quad \text{for } \ a \leq x \leq b$$

together with the boundary conditions

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

Boundary-Value Problems for ODEs

Theorem: Existence & Uniqueness

Suppose the function f in the boundary-value problem

$$y'' = f(x, y, y'), \text{ for } a \leq x \leq b, \text{ with } y(a) = \alpha \text{ and } y(b) = \beta$$

is continuous on the set

$$D = \{(x, y, y') \mid \text{for } a \leq x \leq b, \text{ with } -\infty < y < \infty \text{ and } -\infty < y' < \infty\},$$

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

(i) $f_y(x, y, y') > 0$, for all $(x, y, y') \in D$, and

(ii) a constant M exists, with

$$\left| f_{y'}(x, y, y') \right| \leq M, \text{ for all } (x, y, y') \in D,$$

then the boundary-value problem has a unique solution.

Boundary-Value Problems for ODEs

Example: Existence & Uniqueness

Show that the boundary-value problem

$$y'' + e^{-xy} + \sin y' = 0, \quad \text{for } 1 \le x \le 2, \quad \text{with } y(1) = y(2) = 0,$$

has a unique solution.

Boundary-Value Problems for ODEs

Example: Existence & Uniqueness

Show that the boundary-value problem

$$y'' + e^{-xy} + sin y' = 0, \quad \text{for } 1 \le x \le 2, \quad \text{with } y(1) = y(2) = 0,$$

has a unique solution.

Solution

We have

$$f(x, y, y') = -e^{-xy} - \sin y'.$$

and for all x in $[1, 2]$,

$$f_y(x, y, y') = xe^{-xy} > 0 \quad \text{and} \quad |f_{y'}(x, y, y')| = |-\cos y'| \le 1.$$

So the problem has a unique solution.

Outline

The Linear Shooting Method

Definition: Linear Boundary-Value Problems

The differential equation

$$y'' = f(x, y, y')$$

is linear when functions $p(x)$, $q(x)$, and $r(x)$ exist with

$$f(x, y, y') = p(x)y' + q(x)y + r(x).$$

Corollary: Existence & Uniqueness for Linear BVPs

Suppose the linear boundary-value problem

$$y'' = p(x)y' + q(x)y + r(x). \text{ for } a \leq x \leq b, \text{ with } y(a) = \alpha \text{ and } y(b) = \beta,$$

satisfies

(i) $p(x), q(x)$, and $r(x)$ are continuous on $[a, b]$,

(ii) $q(x) > 0$ on $[a, b]$. Then the boundary-value problem has a unique solution.

The Linear Shooting Method: (1/2)

Instead of considering the linear boundary-value problem

$$y'' = p(x)y' + q(x)y + r(x), \text{ for } a \leq x \leq b, \text{ with } y(a) = \alpha \text{ and } y(b) = \beta,$$

we first consider the initial value problems

$$y'' = p(x)y' + q(x)y + r(x), \text{ with } a \leq x \leq b, y(a) = \alpha, \text{ and } y'(a) = 0, \quad (1)$$

and

$$y'' = p(x)y' + q(x)y, \quad \text{with } a \leq x \leq b, y(a) = 0, \quad \text{and } y'(a) = 1. \quad (2)$$

The Linear Shooting Method

The Linear Shooting Method: (2/2)

Let $y_1(x)$ denote the solution to (1), and $y_2(x)$ denote the solution to (2). Assume that $y_2(b) \neq 0$. Define

$$y(x) = y_1(x) + \frac{\beta - y_1(b)}{y_2(b)} y_2(x).$$

Then $y(x)$ is the solution to the linear boundary problem.

The Linear Shooting Method: Validation (1/2)

$$y'(x) = {y_1}'(x) + \frac{\beta - y_1(b)}{y_2(b)} y_2'(x)$$

and

$$y''(x) = {y_1}''(x) + \frac{\beta - y_1(b)}{y_2(b)} {y''}_2(x)$$

Substituting for $y_1''(x)$ and $y_2''(x)$ in this equation gives

$$\begin{aligned}
y'' &= p(x)y_1' + q(x)y_1 + r(x) + \frac{\beta - y_1(b)}{y_2(b)} (p(x)y_2' + q(x)y_2) \\
&= p(x)\left(y_1' + \frac{\beta - y_1(b)}{y_2(b)} y_2'\right) + q(x)\left(y_1 + \frac{\beta - y_1(b)}{y_2(b)} y_2\right) + r(x) \\
&= p(x)y'(x) + q(x)y(x) + r(x).
\end{aligned}$$

The Linear Shooting Method: Validation (2/2)

Moreover,

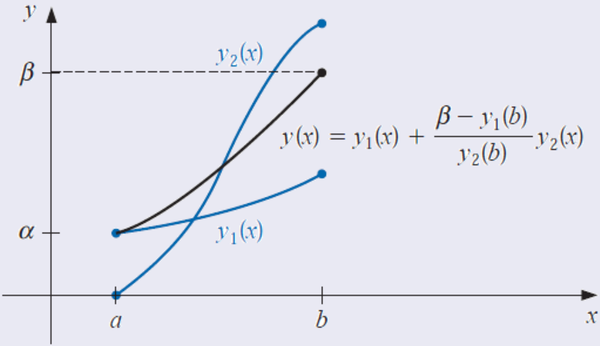
$$y(a) = y_1(a) + \frac{\beta - y_1(b)}{y_2(b)} y_2(a) = \alpha + \frac{\beta - y_1(b)}{y_2(b)} \cdot 0 = \alpha$$

and

$$y(b) = y_1(b) + \frac{\beta - y_1(b)}{y_2(b)} y_2(b) = y_1(b) + \beta - y_1(b) = \beta$$

The Linear Shooting Method

Illustration



The Linear Shooting Method

Linear Shooting Algorithm

INPUT endpoints a, b; boundary conditions α, β; number of subintervals N.

OUTPUT approximations $w_{1,i}$ to $y(x_i)$; $w_{2,i}$ to $y'(x_i)$ for each $i = 0, 1, \ldots, N$.

Step 1 Set $h = (b - a)/N$;
$u_{1,0} = \alpha$; $u_{2,0} = 0$; $v_{1,0} = 0$; $v_{2,0} = 1$.

Step 2 For $i = 0, \ldots, N - 1$ do Steps 3 and 4.
(*The Runge-Kutta method for systems is used in Steps* 3 *and* 4.)

Step 5 Set $w_{1,0} = \alpha$;

$$w_{2,0} = \frac{\beta - u_{1,N}}{v_{1,N}};$$

OUTPUT $(a, w_{1,0}, w_{2,0})$.

Step 6 For $i = 1, \ldots, N$
set $W1 = u_{1,i} + w_{2,0}v_{1,i}$;
$W2 = u_{2,i} + w_{2,0}v_{2,i}$;
$x = a + ih$;
OUTPUT $(x, W1, W2)$. (*Output is* $x_i, w_{1,i}, w_{2,i}$.)

The Linear Shooting Method

Example

Using the linear shooting method with $N = 10$ to approximate the solution to the linear boundary-value problem

$$y'' = -\frac{2}{x}y' + \frac{2}{x^2}y + \frac{\sin(\ln x)}{x^2}$$

for $1 \le x \le 2$, with $y(1) = 1$ and $y(2) = 2$, and compare the results to those of the exact solution

$$y = c_1 x + \frac{c_2}{x^2} - \frac{3}{10}\sin(\ln x) - \frac{1}{10}\cos(\ln x)$$

with $c_1 \approx 1.139$ and $c_2 \approx -0.0392$.

The Linear Shooting Method

Example

x_i	$u_{1,i} \approx y_1(x_i)$	$v_{1,i} \approx y_2(x_i)$	$w_i \approx y(x_i)$	$y(x_i)$	$\lvert y(x_i) - w_i \rvert$
1.0	1.00000000	0.00000000	1.00000000	1.00000000	
1.1	1.00896058	0.09117986	1.09262917	1.09262930	1.43×10^{-7}
1.2	1.03245472	0.16851175	1.18708471	1.18708484	1.34×10^{-7}
1.3	1.06674375	0.23608704	1.28338227	1.28338236	9.78×10^{-8}
1.4	1.10928795	0.29659067	1.38144589	1.38144595	6.02×10^{-8}
1.5	1.15830000	0.35184379	1.48115939	1.48115942	3.06×10^{-8}
1.6	1.21248372	0.40311695	1.58239245	1.58239246	1.08×10^{-8}
1.7	1.27087454	0.45131840	1.68501396	1.68501396	5.43×10^{-10}
1.8	1.33273851	0.49711137	1.78889854	1.78889853	5.05×10^{-9}
1.9	1.39750618	0.54098928	1.89392951	1.89392951	4.41×10^{-9}
2.0	1.46472815	0.58332538	2.00000000	2.00000000	

The fourth-order Runge-Kutta method gives $O(h^4)$ approximations.

The Shooting Method for Nonlinear Problems

Shooting Method

- We have nonlinear boundary-value problems with the following form,

$$y'' = f(x, y, y'), \quad \text{for } \; a \leq x \leq b, \text{with } y(a) = \alpha \text{ and } y(b) = \beta.$$

The Shooting Method for Nonlinear Problems

Shooting Method

- We approximate the solution to the nonlinear boundary-value problems by using the solutions to a sequence of initial-value problems involving a parameter t.
- These problems have the form

$$y'' = f(x, y, y'), \quad \text{for } \; a \leq x \leq b, \text{with } y(a) = \alpha \text{ and } y'(a) = t.$$

- The parameters $t = t_k$ is chosen in a manner to ensure that

$$\lim_{k \to \infty} y(b, t_k) = y(b) = \beta$$

The Shooting Method for Nonlinear Problems

Shooting Method

- We need to determine t with

$$y(b,t) - \beta = 0$$

- This is a nonlinear equation in the variable t, and can be solved by the Secant method if we choose two initial approximations t_0 and t_1.

$$t_k = t_{k-1} - \frac{(y(b,t_{k-1}) - \beta)(t_{k-1} - t_{k-2})}{y(b,t_{k-1}) - y(b,t_{k-2})}, \quad k = 2, 3, \cdots$$

- It can also be solved by the Newton's method or others.

The Shooting Method for Nonlinear Problems

Shooting Method

- The Newton's iteration has the form

$$t_k = t_{k-1} - \frac{y(b, t_{k-1}) - \beta}{\frac{dy}{dt}(b, t_{k-1})},$$

  and it requires the knowledge of $(dy/dt)(b, t_{k-1})$. This presents a difficulty because an explicit representation for $y(b,t)$ is not known; we know only the values $y(b,t_0), \cdots, y(b, t_{k-1})$.

- But we can rewrite the initial-value problem, emphasizing that the solution depends on both x and the parameter t:

$$y''(x,t) = f(x, y(x,t), y'(x,t)),$$

$$\text{for } \; a \le x \le b, \;\; \text{with } \; y(a,t) = \alpha \;\; \text{and } \; y'(a,t) = t.$$

The Shooting Method for Nonlinear Problems

- This implies that

$$
\begin{aligned}
\frac{\partial y''}{\partial t}(x,t) &= \frac{\partial f}{\partial t}(x, y(x,t), y'(x,t)) \\
&= \frac{\partial f}{\partial x}(x, y(x,t), y'(x,t))\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}(x, y(x,t), y'(x,t))\frac{\partial y}{\partial t}(x,t) \\
&\qquad + \frac{\partial f}{\partial y'}(x, y(x,t), y'(x,t))\frac{\partial y'}{\partial t}(x,t) \\
&= \frac{\partial f}{\partial y}(x, y(x,t), y'(x,t))\frac{\partial y}{\partial t}(x,t) + \frac{\partial f}{\partial y'}(x, y(x,t), y'(x,t))\frac{\partial y'}{\partial t}(x,t)
\end{aligned}
$$

- The initial conditions give

$$
\frac{\partial y}{\partial t}(a,t) = 0 \quad \text{and} \quad \frac{\partial y'}{\partial t}(a,t) = 1
$$

The Shooting Method for Nonlinear Problems

- If we simplify the notation by using $z(x,t)$ to denote $(\partial y/\partial t)(x,t)$, then the initial-value problem becomes

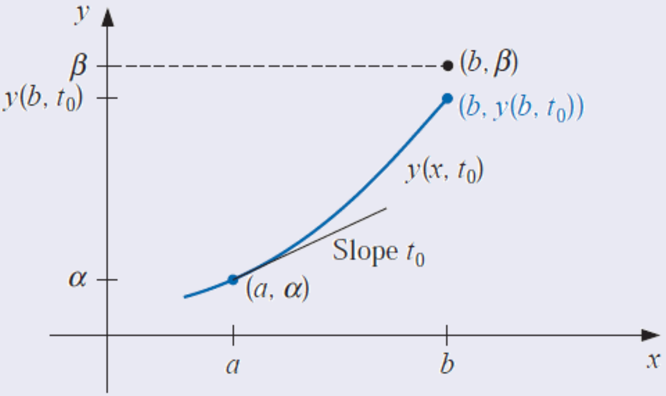
$$z''(x,t) = \frac{\partial f}{\partial y}(x,y,y')z(x,t) + \frac{\partial f}{\partial y'}(x,y,y')z'(x,t),$$

  for $a \le x \le b$ with $z(a,t) = 0$ and $z'(a,t) = 1$.
- Newton's method therefore requires that two initial-value problems be solved for each iteration. Then, we have

$$t_k = t_{k-1} - \frac{y(b,t_{k-1}) - \beta}{z(b,t_{k-1})}$$

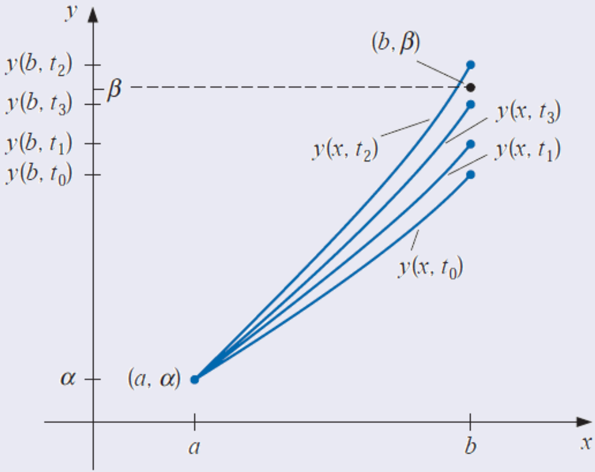
The Shooting Method for Nonlinear Problems

Shooting Method: Illustration



The Shooting Method for Nonlinear Problems

Shooting Method: Illustration



The Shooting Method

Remarks

- Both the linear and nonlinear Shooting methods for boundary-value problems can present problems of instability.

Outline

Finite-Difference Methods for Linear Problems

Finite-Difference Methods

- The finite difference method for the linear second-order boundary-value problem

$$y'' = p(x)y' + q(x)y + r(x), \quad \text{for } \; a \leq x \leq b,$$
$$\text{with } y(a) = \alpha \text{ and } y(b) = \beta,$$

  requires that difference-quotient approximations be used to approximate both y' and y''.
- We select an integer $N > 0$ and divide the interval $[a, b]$ into $(N + 1)$ equal subintervals whose endpoints $x_i = a + ih$ for $i = 0, 1, \cdots, N + 1$, where $h = (b - a)/(N + 1)$.

Finite-Difference Methods for Linear Problems

Centered-Difference Formula for $y''(x_i)$

$$y''(x_i) = \frac{1}{h^2}[y(x_{i+1}) - 2y(x_i) + y(x_{i-1})] - \frac{h^2}{12}y^{(4)}(\xi_i)$$

for some ξ_i in (x_{i-1}, x_{i+1}).

Centered-Difference Formula for $y'(x_i)$

$$y'(x_i) = \frac{1}{2h}[y(x_{i+1}) - y(x_{i-1})] - \frac{h^2}{6}y'''(\eta_i)$$

for some η_i in (x_{i-1}, x_{i+1}).

Finite-Difference Methods for Linear Problems

Finite-Difference Methods

The use of these centered-difference formulas results in the equation

$$\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} = p(x_i)\left[\frac{y(x_{i+1}) - y(x_{i-1})}{2h}\right] + q(x_i)y(x_i)$$
$$+r(x_i) - \frac{h^2}{12}[2p(x_i)y'''(\eta_i) - y^{(4)}(\xi_i)].$$

A Finite-Difference method with truncation error of order $O(h^2)$ results by using this equation together with the boundary conditions $y(a) = \alpha$ and $y(b) = \beta$ to define the system of linear equations.

$$w_0 = \alpha, w_{N+1} = \beta$$

and

$$(\frac{-w_{i+1} + 2w_i - w_{i-1}}{h^2}) + p(x_i)(\frac{w_{i+1} - w_{i-1}}{2h}) + q(x_i)w_i = -r(x_i),$$

for each $i = 1, 2, ..., N$.

Finite-Difference Methods

$$A\mathbf{w} = \mathbf{b}, \quad \text{where}$$

$$A = \begin{bmatrix} 2 + h^2 q(x_1) & -1 + \frac{h}{2}p(x_1) & 0 & \cdots & 0 \\ -1 - \frac{h}{2}p(x_2) & 2 + h^2 q(x_2) & -1 + \frac{h}{2}p(x_2) & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 + \frac{h}{2}p(x_{N-1}) \\ 0 & \cdots & 0 & -1 - \frac{h}{2}p(x_N) & 2 + h^2 q(x_N) \end{bmatrix},$$

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{N-1} \\ w_N \end{bmatrix}, \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} -h^2 r(x_1) + \left(1 + \frac{h}{2}p(x_1)\right) w_0 \\ -h^2 r(x_2) \\ \vdots \\ -h^2 r(x_{N-1}) \\ -h^2 r(x_N) + \left(1 - \frac{h}{2}p(x_N)\right) w_{N+1} \end{bmatrix}.$$

Example

Using the finite-difference method with $N = 9$ to approximate the solution to the linear boundary-value problem

$$y'' = -\frac{2}{x}y' + \frac{2}{x^2}y + \frac{\sin(\ln x)}{x^2}$$

for $1 \leq x \leq 2$, with $y(1) = 1$ and $y(2) = 2$.

Example

x_i	w_i	$y(x_i)$	$\lvert w_i - y(x_i)\rvert$
1.0	1.00000000	1.00000000	
1.1	1.09260052	1.09262930	2.88×10^{-5}
1.2	1.18704313	1.18708484	4.17×10^{-5}
1.3	1.28333687	1.28338236	4.55×10^{-5}
1.4	1.38140205	1.38144595	4.39×10^{-5}
1.5	1.48112026	1.48115942	3.92×10^{-5}
1.6	1.58235990	1.58239246	3.26×10^{-5}
1.7	1.68498902	1.68501396	2.49×10^{-5}
1.8	1.78888175	1.78889853	1.68×10^{-5}
1.9	1.89392110	1.89392951	8.41×10^{-6}
2.0	2.00000000	2.00000000	

Finite-Difference Methods for Linear Problems

Finite-Difference Methods: Remarks

- The difference method used here has local truncation error of order $O(h^2)$.
- Instead of attempting to obtain a difference method with a higher-order truncation error in this manner, it is generally more satisfactory to consider a reduction in step size.
- In addition, Richardson's extrapolation technique can be used effectively for this method.

Finite-Difference Methods

According to the centered-difference formulas, we get

$$\frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1})}{h^2} = f\left(x_i, y(x_i), \frac{y(x_{i+1}) - y(x_{i-1})}{2h} - \frac{h^2}{6}y'''(\eta_i)\right) + \frac{h^2}{12}y^{(4)}(\xi_i)$$

Finite-Difference Methods

$$w_0 = \alpha, w_{N+1} = \beta,$$

and

$$-\frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} + f(x_i, w_i, \frac{w_{i+1} - w_{i-1}}{2h}) = 0,$$

for each $i = 1, 2, ..., N$.

Finite-Difference Methods

The $N \times N$ nonlinear system obtained from this method,

$$\begin{aligned}
2w_1 - w_2 + h^2 f(x_1, w_1, \frac{w_2 - \alpha}{2h}) - \alpha &= 0, \\
-w_1 + 2w_2 - w_3 + h^2 f(x_2, w_2, \frac{w_3 - w_1}{2h}) &= 0, \\
&\vdots \\
-w_{N-2} + 2w_{N-1} - w_N + h^2 f(x_{N-1}, w_{N-1}, \frac{w_N - w_{N-2}}{2h}) &= 0, \\
-w_{N-1} + 2w_N + h^2 f(x_N, w_N, \frac{\beta - w_{N-1}}{2h}) - \beta &= 0
\end{aligned}$$

has a unique solution provided that $h < 2/L$.

Outline

Elliptic Partial Differential Equations

Elliptic Equations

- The partial differential equation that involves $u_{xx}(x,y) + u_{yy}(x,y)$ is an elliptic equation.
- The particular elliptic equation we will consider is known as the Poisson equation:

$$\frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y)$$

- When $f(x,y) = 0$, resulting in a simplification to Laplace's equation.

Elliptic Partial Differential Equations

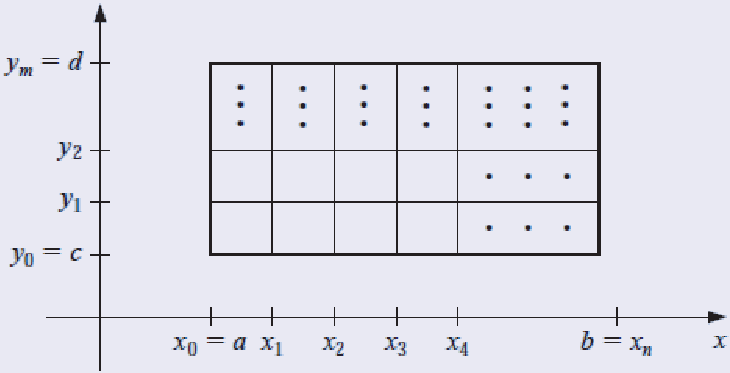
Poisson Equations

The Poisson Equation is an elliptic partial differential equation

$$\nabla^2 u(x,y) \equiv \frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y)$$

on $R = \{(x,y) | a < x < b, c < y < d\}$, with $u(x,y) = g(x,y)$ for $(x,y) \in S$, where S denotes the boundary of R. If f and g are continuous on their domains, then there is a unique solution to this equation.

Finite Difference Method



Finite Difference Method

We can use the Taylor series in the variable x about x_i to generate the centered-difference formula

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) = \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h^2} - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, y_j),$$

where $\xi_i \in (x_{i-1}, x_{i+1})$. We can also use Taylor series in the variable y about y_j to generate the centered-difference formula

$$\frac{\partial^2 u}{\partial y^2}(x_i, y_j) = \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{k^2} - \frac{k^2}{12}\frac{\partial^4 u}{\partial y^4}(x_i, \eta_j),$$

where $\eta_j \in (y_{i-1}, y_{i+1})$.

Finite Difference Method

Using these formulas allows us to express the Poisson equation at the points (x_i, y_j) as

$$\frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h^2} + \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{k^2}$$
$$= f(x_i, y_j) + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, y_j) + \frac{h^2}{12}\frac{\partial^4 u}{\partial y^4}(x_i, \eta_j)$$

for each $i = 1, 2, ..., n-1$ and $j = 1, 2, ..., m-1$. The boundary conditions are

$$u(x_0, y_j) = g(x_0, y_j) \quad \text{and} \quad u(x_n, y_j) = g(x_n, y_j) \text{ for each } j = 1, 2, ..., m,$$
$$u(x_i, y_0) = g(x_i, y_0) \quad \text{and} \quad u(x_i, y_m) = g(x_i, y_m) \text{ for each } i = 1, 2, ..., n-1.$$

Finite Difference Method

In difference-equation form, this results in the **Finite-Difference method**:

$$2\left[\left(\frac{h}{k}\right)^2+1\right]w_{ij}-(w_{i+1,j}+w_{i-1,j})-\left(\frac{h}{k}\right)^2(w_{i,j+1}+w_{i,j-1})=-h^2f(x_i,y_j),$$

for each $i=1,2,...,n-1$ and $j=1,2,...,m-1$, and

$$\begin{aligned} w_{0j}&=g(x_0,y_j) \quad \text{and} \quad w_{nj}=g(x_n,y_j) \text{ for each } j=1,2,...,m,\\ w_{i0}&=g(x_i,y_0) \quad \text{and} \quad w_{im}=g(x_i,y_m) \text{ for each } i=1,2,...,n-1. \end{aligned}$$

where w_{ij} approximates $u(x_i,y_j)$. This method has local truncation error of order $O(h^2+k^2)$.

Parabolic Partial Differential Equations

Parabolic Equations

The parabolic partial differential equation we consider is the heat, or diffusion, equation

$$\frac{\partial u}{\partial t}(x,t) = \alpha^2 \frac{\partial^2 u}{\partial^2 x^2}(x,t), \quad 0 < x < l, \ \ t > 0,$$

subject to the conditions

$$u(0,t) = u(l,t) = 0, \ \ t > 0, \ \text{ and } \ u(x,0) = f(x), \ \ 0 \leq x \leq l.$$

Parabolic Partial Differential Equations

Forward Difference Method

We obtain the difference method using the Taylor series in t to form the difference quotient

$$\frac{\partial u}{\partial t}(x_i, t_j) = \frac{u(x_i, t_{j+k}) - u(x_i, t_j)}{k} - \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j),$$

for some $\mu_j \in (t_j, t_{j+1})$, and the Taylor series in x to form the difference quotient

$$\frac{\partial^2 u}{\partial x^2}(x_i, t_j) = \frac{u(x_i + h, t_j) - 2u(x_i, t_j) + u(x_i - h, t_j)}{h^2} - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, t_j),$$

where $\xi_i \in (x_{i-1}, x_{i+1})$.

Parabolic Partial Differential Equations

Forward Difference Method

The parabolic partial difference equation implies that at interior gridpoints (x_i, t_j), for each $i = 1, 2, ..., m-1$ and $j = 1, 2, ...,$ we have

$$\frac{\partial u}{\partial t}(x_i, t_j) - \alpha^2 \frac{\partial^2 u}{\partial x^2}(x_i, t_j) = 0,$$

so the difference method using the difference quotients is

$$\frac{w_{i,j+1} - w_{ij}}{k} - \alpha^2 \frac{w_{i+1,j} - 2w_{ij} + w_{i-1,j}}{h^2} = 0,$$

where w_{ij} approximates $u(x_i, t_j)$.
The local truncation error for this difference equation is

$$\tau_{ij} = \frac{k}{2} \frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) - \alpha^2 \frac{h^2}{12} \frac{\partial^4 u}{\partial x^4}(\xi_i, t_j).$$

Hyperbolic Partial Differential Equations

Wave Equation

Wave equation is a hyperbolic partial differential equation in the form of

$$\frac{\partial^2 u}{\partial^2 t^2}(x,t) - \alpha^2 \frac{\partial^2 u}{\partial^2 x^2}(x,t) = 0, \quad 0 < x < l, \;\; t > 0,$$

subject to the conditions

$$u(0,t) = u(l,t) = 0, \;\; \text{for} \;\; t > 0,$$

$$u(x,0) = f(x), \;\; \text{and} \; \frac{\partial u}{\partial t}(x,0) = g(x), \;\; \text{for} \;\; 0 \le x \le l,$$

where α is a constant dependent on the physical conditions of the problem.

Assignment

- Reading assignment: Chap. 11 & 12.