11. Boundary Value Problems

We study two numerical methods for solving (BVP) of the kind:

$$(*) \quad \begin{cases} y'' = f(t, y, y') \\ y(a) = \alpha \\ y(b) = \beta \end{cases}$$

<u>Note</u> the boundary conditions could be different.

(*) has so-called <u>Dirichlet</u> B.C. on both ends

But one can also consider <u>Neumann</u> B.C.:

$y'(a) = \alpha$ (or at other end point).

or <u>Robin</u> B.C.

$\alpha y'(a) + \beta y(a) = \gamma$

Usually to get uniqueness of (BVP) we need Dirichlet B.C. on at least one end point.

<u>Note</u> Numerical methods we see here can be adapted to other B.C.

<u>Physical interpretation:</u> If $f(t, y, y'') = f(t)$, then y models temperature distribution in a bar $[a, b]$, subject to an internal heat source $f(t)$.

<u>Dirichlet B.C.</u>: means one fixes temperature at end point.

<u>Neumann B.C.</u>: " " " heat flux " " "

<u>Robin B.C.</u>: " " has another bar connected at end point.

(impedance type B.C.)

Here is an existence theorem due to Keller for (*):

Theorem: Suppose the function f in (*) is continuous on

$$D = \{ (t, y, y') \mid t \in [a,b], y \in \mathbb{R}, y' \in \mathbb{R} \}$$

and also f_y and $f_{y'}$. If

i) $f_y(t, y, y') > 0 \quad \forall (t, y, y') \in D$

ii) $\exists\, M > 0$ s.t.

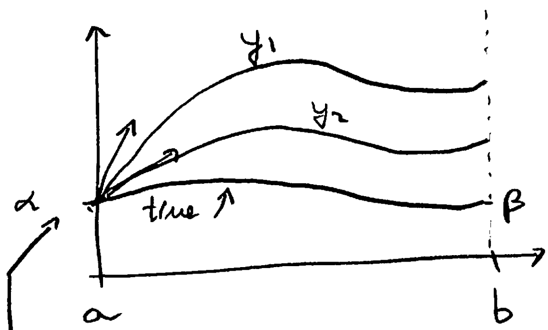
$$|f_{y'}(t, y, y')| \le M \quad \forall\, (t, y, y') \in D$$

then BVP (*) has a unique solution.

Note: this only a sufficient condition for uniqueness. There are problems (*) not satisfying hypothesis of theorem for which there is a unique solution.

§ 11.1 - 11.2 Shooting methods

Idea: Solve a related initial value problem with $y(a) = \alpha$ and a guess for $y'(a)$, hoping that $y(b) = \beta$. If it is not then we improve guess and solve again another IVP...



fixing $y'(a)$ is fixing slope at $t = a$.

The kind of (IVP) we solve is:

$$(IVP)\begin{cases} y'' = f(t,y,y') \\ y(a) = \alpha \\ y'(a) = z \end{cases}$$

Call $y_z(t)$ the sol to (IVP) w/ $y'(a) = z$.

Then we need to find z s.t. $y_z(b) = \beta$ in order to solve original BVP.

$\Leftrightarrow$ Find a zero of function.

$$\boxed{\phi(z) = y_z(b) - \beta}$$

Shooting methods are simply trying to find a root of $\phi(z)$ with for example:

- Bisection method
- Secant method
- Newton method ...

We have to keep in mind that every time we evaluate $\phi(z)$ we must solve an IVP. Also Newton method requires $\phi'(z)$. We will see how to compute this later.

Before let us see an easy case where $\phi(z)$ = line.

Linear BVP

Def: A (BVP) of form (*) with

$$f(t,y,y') = p(t)\,y' + q(t)\,y + r(t), \quad t \in [a,b].$$

For these linear (BVP) we get an easy corollary of existence theorem:

Corollary BVP (*) with $f(t,y,y') = p(t)y' + q(t)y + r(t)$ has a unique solution if p, q are continuous on $[a,b]$ and $q(t) = f_y(t,y,y') > 0$ for $t \in [a,b]$.

proof: $q(t) = f_y(t,y,y')$ and condition ii) is automatically satisfied by this particular f since

$$f_{y'}(t,y,y') = p(t) = \text{continuous on } [a,b]$$
$$= \text{bounded on } [a,b].$$

For linear BVP $\phi(\zeta)$ is a line so it is enough to solve two IVP for say ζ_1 and ζ_2 to find root.

Let

- y_1 be sol to IVP assoc with BVP with $y_1(a)=\alpha,\ y_1'(a)=\zeta_1$
- y_2 " " " " " " " " $y_2(a)=\alpha,\ y_2'(a)=\zeta_2$

Then:

$y(t) = \lambda y_1(t) + (1-\lambda) y_2(t)$ satisfies DE in BVP:

$$\begin{aligned} y'' &= \lambda y_1'' + (1-\lambda) y_2'' \\ &= \lambda(p y_1' + q y_1 + r) + (1-\lambda)(p y_2' + q y_2 + r) \\ &= p\underbrace{(\lambda y_1' + (1-\lambda) y_2')}_{y'} + q\underbrace{(\lambda y_1 + (1-\lambda) y_1)}_{y} + \underbrace{\lambda r + (1-\lambda) r}_{r} \end{aligned}$$

Also:

$$\boxed{y(a)} = \lambda y_1(a) + (1-\lambda) y_2(a)$$
$$= \lambda \alpha + (1-\lambda)\alpha$$
$$\boxed{= \alpha}$$

$$y(b) = \lambda y_1(b) + (1-\lambda) y_2(b)$$

Now we can choose λ s.t. $y(b) = \beta$ and y is sol. we seek.

$$\Rightarrow \quad \boxed{\lambda = \frac{\beta - y_2(b)}{y_1(b) - y_2(b)}}$$

<u>Linear shooting algorithm</u>

- Solve DE of (BVP) + I.C. $y_1(a) = \alpha$, $y_1'(a) = 0$
- solve DE of (BVP) + I.C. $y_2(a) = \alpha$, $y_2'(a) = 1$

  (assumes we have an IVP solver such as RK4 and that we are keeping iterates in a vector $\underline{y_1}$ and $\underline{y_2}$).
- $\lambda = \dfrac{\beta - y_2(b)}{y_1(b) - y_2(b)}$
- output $\underline{y} = \lambda \underline{y_1} + (1-\lambda)\underline{y_2}$.

<u>Of course</u>: we assume that methods we use for solving IVP use <u>same time steps</u>. (So no adaptivity)

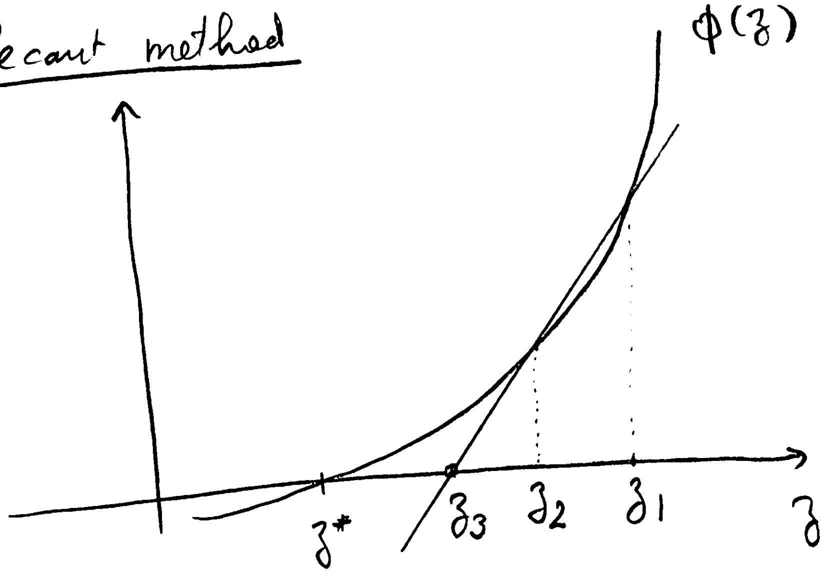
If we have an IVP solver that can deal with first order systems of DE, we can rewrite the first two steps as the solution to the system:

$$\begin{cases} y_1' = y_3 \\ y_2' = y_4 \\ y_3' = f(t, y_1, y_3) \\ y_4' = f(t, y_2, y_4) \end{cases} \iff \underline{y}' = \underline{f}(t, \underline{y})$$

$$\text{I.C.:} \begin{cases} y_1(a) = \alpha \\ y_2(a) = \alpha \\ y_3(a) = 0 \\ y_4(a) = 1 \end{cases}$$

Non-linear problem

Secant method



Instead of solving $\phi(z) = 0$ we solve
$s(z) = 0$ where s = secant

and repeat until convergence

In this example:

$$s(z) = \phi(z_2) + \frac{\phi(z_1) - \phi(z_2)}{z_1 - z_2}(z - z_2)$$

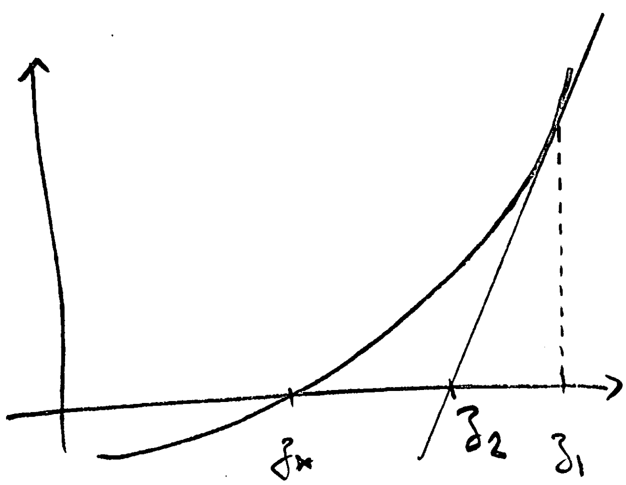
Solve for z_3 s.t. $s(z_3) = 0$

$$z_3 = z_2 - \frac{z_2 - z_1}{\phi(z_2) - \phi(z_1)}\phi(z_2)$$

or more generally:

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

We know Newton's method offers faster convergence by finding root of tangent instead of secant:



think Taylor.

$$t(z) = \phi(z_1) + \phi'(z_1)(z - z_1)$$

Solving for z_2 s.t. $t(z_2) = 0$ we get.

$$z_2 = z_1 - \frac{\phi(z_1)}{\phi'(z_1)}$$

Or more generally:

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

Now recall for us $\phi(z) = y_z(b) - \beta$ where y_z solves:

$$(PZ)\begin{cases} y_z'' = f(t, y_z, y_z') \\ y_z(a) = \alpha \\ y_z'(a) = z \end{cases}$$

Here is a trick to obtain ϕ': differentiate (PZ) w.r.t. z:

$$\begin{cases} \dfrac{\partial y_z''}{\partial z} = \dfrac{\partial t}{\partial z}\dfrac{\partial f}{\partial t} + \dfrac{\partial y_z}{\partial z}\dfrac{\partial f}{\partial y_z} + \dfrac{\partial y_z'}{\partial z}\dfrac{\partial f}{\partial y_z'} \quad \text{(Chain rule)} \\ \dfrac{\partial y_z}{\partial z}(a) = 0 \\ \dfrac{\partial y_z'(a)}{\partial z} = 1 \end{cases}$$

Letting $v = \dfrac{\partial y_z}{\partial z}$ we get the <u>first variational equation</u>:

$$\begin{cases} v'' = v f_{y_z}(t, y_z, y_z') + v' f_{y'}(t, y_z, y_z') \\ v(a) = 0 \\ v(a) = 1 \end{cases}$$

which can be solved at the same time as (P2) by formulating a system:

Let $y = y_1$
$v = y_2$

$$\begin{cases} y_1' = y_3 \\ y_2' = y_4 \\ y_3' = f(t, y_1, y_3) \\ y_4' = y_2 f_y(t, y_1, y_3) + y_4 f_{y'}(t, y_1, y_3) \end{cases}$$

with I.C.:

$$\begin{cases} y_1(a) = \alpha \\ y_2(a) = 0 \\ y_3(a) = z \\ y_4(a) = 1 \end{cases}$$

Thus here: $\phi(z) = y_z(b) - \beta = y_1(b) - \beta$

$$\phi'(z) = \frac{\partial y_z(b)}{\partial z} = v(b) = y_2(b)$$

and Newton iteration becomes:

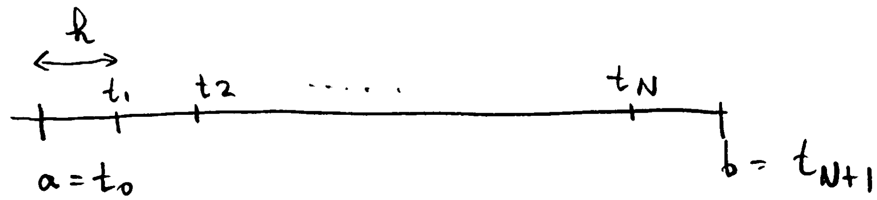
$$z_{n+1} = z_n - \frac{y_1(b) - \beta}{y_2(b)}$$

where y_1 and y_2 depend on z through I.C.

§ 11.3 – 11.4 Finite Differences

$$
\text{(BVP)} \begin{cases} y'' = f(t, y, y') \\ y(a) = \alpha \\ y(b) = \beta \end{cases}
$$

Idea. • subdivide $[a,b]$ into $N+1$ subintervals of length $h = \dfrac{b-a}{N+1}$.



in this way:

$$t_i = a + ih, \quad \text{where} \quad i = 0, \dots, N+1$$

(check $t_0 = a$ and $t_{N+1} = b$)

- Replace y'', y' by a discrete approx (finite differences)
- If BVP is <u>linear</u> i.e. $f(t, y, y') = py' + qy + r$
  $\rightsquigarrow$ get linear system to solve
- If BVP is <u>non-linear</u>
  $\rightsquigarrow$ get non-linear system of eq to solve using Newton-type method.

Finite Differences:

Taylor's theorem:

(A) $\quad y(t_{i+1}) = y(t_i) + h\,y'(t_i) + \frac{h^2}{2} y''(t_i) + \frac{h^3}{3!} y^{(3)}(t_i) + \frac{h^4}{4!} y^{(4)}(\xi_i^+)$

(B) $\quad y(t_{i-1}) = y(t_i) - h\,y'(t_i) + \frac{h^2}{2} y''(t_i) - \frac{h^3}{3!} y^{(3)}(t_i) + \frac{h^4}{4!} y^{(4)}(\xi_i^-)$

for some $\xi_i^+ \in [t_i, t_{i+1}]$ and $\xi_i^- \in [t_{i-1}, t_i]$.

Doing (A) + (B) and isolating $y''(t_i)$ we get:

$$\frac{y(t_{i+1}) - 2y(t_i) + y(t_{i-1})}{h^2} = y''(t_i) + \frac{h^2}{24}\Big(\underbrace{y^{(4)}(\xi_i^+) + y^{(4)}(\xi_i^-)}_{=2y^{(4)}(\xi_i)}\Big)$$

IVT, $\xi_i \in [t_{i-1}, t_{i+1}]$.

$$\Rightarrow \boxed{y''(t_i) = \frac{y(t_{i+1}) - 2y(t_i) + y(t_{i-1})}{h^2} + \mathcal{O}(h^2)}$$

Similarly with (A) − (B) we can show:

$$\boxed{y'(t_i) = \frac{y(t_{i+1}) - y(t_{i-1})}{2h} + \mathcal{O}(h^2)}$$

So we get discrete approx of BVP with $y_i \approx y(t_i)$:

$$\begin{cases} y_0 = \alpha \\ \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) = f\left(t_i, y_i, \frac{y_{i+1}-y_{i-1}}{2h}\right), \quad i = 1, \dots, N. \\ y_{N+1} = \beta \end{cases}$$

<u>Linear case</u>. $f(t, y, y') = p y' + q y + r$, $p, q, r \equiv$ functions.

$$\begin{cases} y_0 = \alpha \\ \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1}) = p_i\left(\frac{y_{i+1} - y_{i-1}}{2h}\right) + q_i y_i + r_i, \quad i = 1, \dots, N \\ y_{N+1} = \beta \end{cases}$$

we can rewrite this system of linear eq. in matrix form:

$$A \underline{y} = \underline{b}$$

where $A = L - D - Q \in \mathbb{R}^{n \times n}$ (to be def.)

and $\underline{y}, \underline{b} \in \mathbb{R}^n$.

Obviously: $\underline{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix} =$ vector w/ values of y that we do not know (i.e. excluding B.C.)

$$L = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad L\underline{y} \approx y''(t)$$

$$D = \frac{1}{2h}\begin{bmatrix} 0 & p_1 & & & & \\ -p_2 & 0 & p_2 & & & \\ & -p_3 & 0 & p_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -p_{N-1} & 0 & p_{N-1} \\ & & & & -p_N & 0 \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad D\underline{y} \approx p(t)\,y'(t)$$

$$Q = \begin{bmatrix} q_1 & & & \\ & q_2 & & \\ & & \ddots & \\ & & & q_N \end{bmatrix} \in \mathbb{R}^{N\times N}, \quad Q\underline{y} \approx q(t)\,y(t)$$

$$\underline{b} = \begin{bmatrix} r_1 - \alpha/h^2 - p_1\alpha/2h \\ r_2 \\ \vdots \\ r_{N-1} \\ r_N - \beta/h^2 + p_N\beta/2h \end{bmatrix} \in \mathbb{R}^N.$$

<u>Note</u> how B.C. appear as "corrections" in R.H.S. $\underline{b}$.

<u>Note</u>: A is tridiagonal. Solving tridiagonal systems is a cheap $\mathcal{O}(n)$ operation.

<u>Note</u>: Accuracy of approx is $\mathcal{O}(h^2)$.

Theorem: Suppose p, q, r are continuous on $[a,b]$. If $q(t) \geq 0$ on $[a,b]$, then system $A\underline{y} = \underline{b}$ has a unique sol provided.

$$h < \frac{2}{M}, \quad M = \max_{t \in [a,b]} |p(t)|.$$

Non-Linear Case

$$\begin{cases} y' = f(t, y, y') \\ y(a) = \alpha \\ y(b) = \beta \end{cases}$$

We assume:

- $f, f_y, f_{y'}$ continuous on $D = \{(t, y, y') \mid t \in [a,b],\ y, y' \in \mathbb{R}\}$
- $f_y(t, y, y') \geq \delta > 0$ on D.
- $\exists\, k, L > 0$ s.t.

$$k = \max_{(t,y,y') \in D} |f_y(t, y, y')|$$

$$L = \max_{(t,y,y') \in D} |f_{y'}(t, y, y')|$$

this guarantees that systems involved are non-singular, and existence of a unique solution to BVP.

Using finite differences we get the following system of N nonlinear eq. with N variables:

$$\begin{cases} \dfrac{\alpha - 2y_1 + y_2}{h^2} - f\left(t_1, y_1, \dfrac{y_2 - \alpha}{2h}\right) \equiv F_1(\underline{y}) = 0 \\ \dfrac{y_1 - 2y_2 + y_3}{h^2} - f\left(t_2, y_2, \dfrac{y_3 - y_1}{2h}\right) \equiv F_2(\underline{y}) = 0 \\ \vdots \\ \dfrac{y_N - 2y_{N-1} + y_{N-2}}{h^2} - f\left(t_{N-1}, y_{N-1}, \dfrac{y_N - y_{N-2}}{2h}\right) \equiv F_{N-1}(\underline{y}) = 0 \\ \dfrac{\beta - 2y_N + y_{N-1}}{h^2} - f\left(t_N, y_N, \dfrac{\beta - y_{N-1}}{2h}\right) \equiv F_N(\underline{y}) = 0 \end{cases}$$

which we can rewrite compactly as:

$$\underline{F}(\underline{y}) = \underline{0} \qquad (\text{here } \underline{F}: \mathbb{R}^N \to \mathbb{R}^N)$$

where

$$\underline{F}(\underline{y}) = \begin{bmatrix} F_1(\underline{y}) \\ F_2(\underline{y}) \\ \vdots \\ F_N(\underline{y}) \end{bmatrix}$$

This is a multidimensional zero finding problem that can be solved using Newton's method. If $\underline{y}^{(k)}$ is our current iterate, Newton's method finds zero of "best linear model" of $\underline{F}(\underline{y})$ around $\underline{y} = \underline{y}^{(k)}$.

This can be obtained from Taylor's theorem:

$$\underline{F}(\underline{y}) = \underline{F}(\underline{y}^{(k)}) + D\underline{F}[y^{(k)}](\underline{y} - y^{(k)}) + o(\|\underline{y} - y^{(k)}\|)$$

$$D\underline{F}[y^{(k)}] = \text{Jacobian of } \underline{F} \text{ at } y^{(k)}$$

$$= \text{matrix} \in \mathbb{R}^{N \times N} \text{ depending on } y^{(k)}$$

$$= \begin{bmatrix} \nabla F_1^T(\underline{y}) \\ \nabla F_2^T(\underline{y}) \\ \vdots \\ \nabla F_N^T(\underline{y}) \end{bmatrix} = \begin{bmatrix} \frac{\partial F_1}{\partial y_1} & \cdots & \frac{\partial F_1}{\partial y_N} \\ \vdots & & \\ \frac{\partial F_N}{\partial y_1} & \cdots & \frac{\partial F_N}{\partial y_N} \end{bmatrix}$$

To obtain $\underline{y}^{(k+1)}$ we find zero of linear model suggested by Taylor:

$$\underline{T}(\underline{y}) = \underline{F}(\underline{y}^{(k)}) + D\underline{F}[\underline{y}^{(k)}](\underline{y} - \underline{y}^{(k)})$$

Solution $\underline{y}^{(k+1)}$ of $\underline{T}(\underline{y}) = \underline{0}$ is:

$$\boxed{\underline{y}^{(k+1)} = y^{(k)} - D\underline{F}[y^{(k)}]^{-1}\underline{F}(\underline{y}^{(k)})}$$

(notice similarity w/ one dimensional N.M.:

$$z^{(k+1)} = z^{(k)} - \frac{\phi(z^{(k)})}{\phi'(z^{(k)})}$$
)

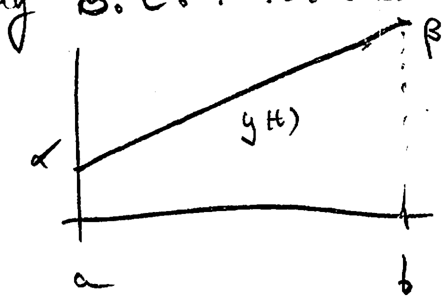
Nonlinear shooting method

- Choose $\underline{y}^{(0)}$ (see below)
- for $k = 0, \dots,$ until convergence

$$\underline{y}^{(k+1)} = \underline{y}^{(k)} - D\underline{F}[\underline{y}^{(k)}]^{-1}\,\underline{F}(\underline{y}^{(k)})$$

Choice of initial iterate $\underline{y}^{(0)}$:

Typically $\underline{y}^{(0)}$ is a vector representing a function satisfying B.C.. For example:



$$y(t) = \alpha + \frac{\beta-\alpha}{b-a}(t-a)$$

$\Rightarrow$ $\underline{y}^{(0)}$ has entries $y_i = \alpha + \dfrac{\beta-\alpha}{b-a}\, ih$, $i = 0, \dots, N+1$.

and $h = \dfrac{b-a}{N+1}$ (check $y_0 = \alpha$ and $y_{N+1} = \beta$)

The Jacobian $DF[\underline{y}]$ can be written explicitly:

$$DF[\underline{y}] = L - D - Q \in \mathbb{R}^{N \times N}$$

where $L = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix}$ (as in p 71)

$$D = \frac{1}{2h}\begin{bmatrix} 0 & f_{y'}(t_1, y_1, \frac{y_2-\alpha}{2h}) & & \\ -f_{y'}(t_2, y_2, \frac{y_3-y_1}{2h}) & 0 & f_{y'}(t_2, y_2, \frac{y_3-y_1}{2h}) & \\ & \ddots & \ddots & \ddots \\ & & -f_{y'}(t_n, y_n, \frac{\beta-y_{n-1}}{2h}) & 0 \end{bmatrix}$$

$= \frac{1}{2h} \times$ matrix with all zero entries except:

superdiagonal:

$$f_{y'}(t_1, y_1, \frac{y_2-\alpha}{2h}), f_{y'}(t_2, y_2, \frac{y_3-y_1}{2h}), \dots, f_{y'}(t_{N-1}, y_{N-1}, \frac{y_N - y_{N-2}}{2h})$$

subdiagonal

$$-f_{y'}(t_2, y_2, \frac{y_3-y_1}{2h}), -f_{y'}(t_3, y_3, \frac{y_4-y_2}{2h}), \dots, -f_{y'}(t_N, y_N, \frac{\beta - y_{N-1}}{2h})$$

$$Q = \mathrm{diag}\left(f_y\left(t_1, y_1, \frac{y_2-\alpha}{2h}\right), f_y\left(t_2, y_2, \frac{y_3-y_1}{2h}\right), \ldots, f_y\left(t_N, y_N, \frac{\beta - y_{N-1}}{2h}\right)\right)$$
$\in \mathbb{R}^{N\times N}$.

Notice $DF[\underline{y}] = A$ when problem is linear!

(Check by comparing to p71).

$DF[\underline{y}] =$ tridiagonal matrix

so each iteration in N.M. involves relatively cheap tridiagonal system solves ($\mathcal{O}(N)$).

Numerical Partial Differential Equations

We shall consider three types of PDE's:

- Elliptic : $\Delta u = 0$ (Laplace eq)

  steady state heat distribution
  " " flow in porous medium
  voltage distribution on a plate (DC)

- Parabolic :

  $$u_t - \Delta u = 0 \quad \text{(heat eq)}$$

  heat conduction
  flow in porous media
  diffusion

- Hyperbolic :

  $$u_{tt} - \Delta u = 0 \quad \text{(wave eq)}$$

  $$u_t + \underline{v} \cdot \nabla u = 0 \quad \text{(advection)}$$

wave propagation (EM, acoustic, ...)
transport.

Recall: $$\nabla u = \begin{bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{bmatrix}$$

$$\nabla \cdot \underline{v} = \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y}$$

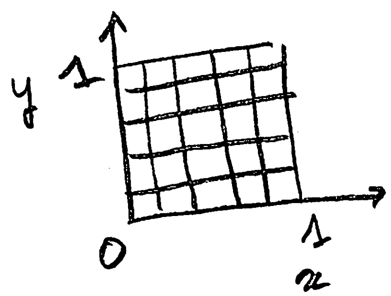
$$\nabla \cdot [\nabla u] = \Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \text{Laplacian}$$

Finite differences method for elliptic PDE.

We will study Poisson's problem

$$\boxed{\Delta u = f} \quad (*)$$

The 5-point stencil for the Laplacian



$x_i = i\Delta x, \quad i = 0, \dots, m+1$

$y_j = j\Delta y, \quad j = 0, \dots, n+1$

here $\Delta x = \frac{1}{m+1}, \quad \Delta y = \frac{1}{n+1}$

Let $u_{ij} \approx u(x_i, y_j)$. We discretize Poisson's eq $(*)$ by replacing the x and y derivatives by finite differences:

why? Use Taylor's theorem!

$$\frac{1}{(\Delta x)^2}\left(u_{i-1,j} - 2u_{ij} + u_{i+1,j}\right)$$

$$+ \frac{1}{(\Delta y)^2}\left(u_{ij-1} - 2u_{ij} + u_{ij+1}\right) = f_{ij}$$

when $\Delta x = \Delta y = h$ we obtain the linear system:

$$\boxed{\frac{1}{h^2}\left(u_{i-1j} + u_{i+1j} + u_{ij-1} + u_{ij+1} - 4u_{ij}\right) = f_{ij}}$$

This is known as the 5-point stencil: