CHAPTER 5.

Multi-Dimensional Parabolic Problems.

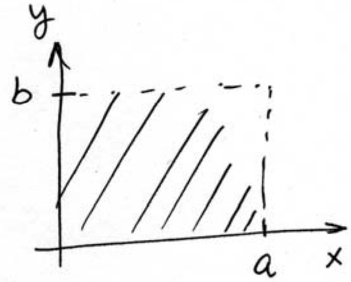
Heat Conduction (3-D)

$$
\begin{cases}
\rho c\, u_t = \nabla\cdot(K\nabla u), & (x,y,z)\in\Omega. \\
u(x,y,z,0) = \phi(x,y,z), & (x,y,z)\in\Omega\cup\partial\Omega. \\
\alpha u + \beta \frac{\partial u}{\partial n} = \mathcal{H}, & (x,y,z)\in\partial\Omega,\ t>0.
\end{cases}
$$

We will restrict our attention to 2-D problems on a rectangular domain. $K \equiv \text{const}$.

$$
\begin{cases}
u_t = \sigma\,(u_{xx}+u_{yy}), & (x,y)\in\Omega \\
u(x,y) = \phi(x,y), & (x,y)\in\Omega\cup\partial\Omega \\
u(x,y,t) = \mathcal{H}(x,y,t), & (x,y)\in\partial\Omega,\ t>0
\end{cases}
$$

$$\Omega = \{(x,y):\ 0\le x< a,\quad 0<y<b\}.$$



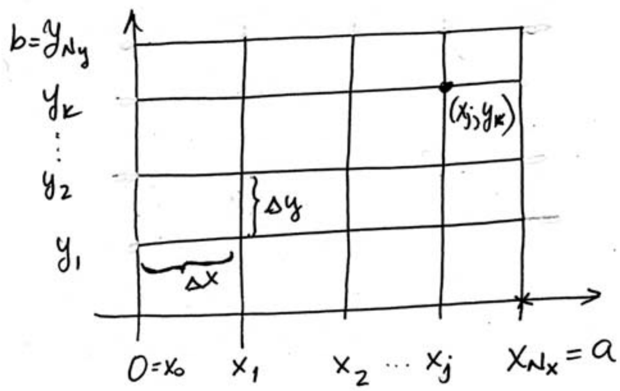
Introduce a uniform rectangular grid with space step sizes given by

$$\Delta x = \frac{a}{N_x}, \qquad \Delta y = \frac{b}{N_y}.$$

and Δt as a time step size.

Let $U_{j,k}^{n} \simeq u(x_j, y_k, t_n)$ where

$$x_j = j * \Delta x$$
$$y_k = k * \Delta y$$
$$t_n = n * \Delta t.$$



Explicit scheme FT-CS

$$\frac{U_{j,k}^{n+1} - U_{j,k}^{n}}{\Delta t} = \sigma \left[\frac{U_{j-1,k}^{n} - 2U_{j,k}^{n} + U_{j+1,k}^{n}}{(\Delta x)^2} + \frac{U_{j,k-1}^{n} - 2U_{j,k}^{n} + U_{j,k+1}^{n}}{(\Delta y)^2} \right. \qquad (1)$$

Equ. (1) can also be written as

$$U_{j,k}^{n+1} = U_{j,k}^{n} + r_x \left[U_{j-1,k}^{n} - 2U_{j,k}^{n} + U_{j+1,k}^{n} \right] + r_y \left[U_{j,k-1}^{n} - 2U_{j,k}^{n} + U_{j,k+1}^{n} \right] \qquad (2)$$

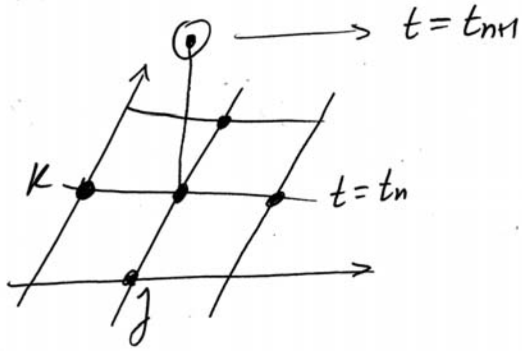
or $\boxed{U_{j,k}^{n+1} = U_{j,k}^{n} + r_x \delta_x^2 U_{j,k}^{n} + r_y \delta_y^2 U_{j,k}^{n}}$

where $r_x = \frac{\sigma \Delta t}{\Delta x^2}$, and $r_y = \frac{\sigma \Delta t}{\Delta y^2}$

To study stability, it's convenient to write equ. (2) as

$$U^{n+1}_{j,k} = r_x U^n_{j-1,k} + r_y U^n_{j,k-1} + (1-2r_x-2r_y) U^n_{j,k} +$$
$$+ r_x U^n_{j+1,k} + r_y U^n_{j,k+1} \quad (3)$$

<u>Computational stencil</u>



<u>Stability</u>

If $r_x + r_y \le 1/2 \Rightarrow 1 - 2(r_x + r_y) \geq 0$

and $1 - 2(r_x + r_y) + 2r_x + 2r_y = 1$

Max. norm principle applies

In particular, if $\Delta x = \Delta y \Rightarrow r_x + r_y \le 1/2 \Leftrightarrow 2r_x \le 1/2$

$\Rightarrow \boxed{r_x \le 1/4}$ More restrictive than in 1-D.

Crank-Nicholson.

(Approxs are performed at $(x_j, y_k, t_{n+1/2})$ and average betw t_n and t_{n+1} is taken)

$$\frac{U_{j,k}^{n+1} - U_{j,k}^{n}}{\Delta t} = \sigma \left[\frac{\delta_x^2}{\Delta x^2} U_{j,k}^{n+1/2} + \frac{\delta_y^2}{\Delta y^2} U_{j,k}^{n+1/2} \right]$$

$$= \sigma \left[\frac{\delta_x^2}{\Delta x^2} \left[\frac{U_{j,k}^{n+1} + U_{j,k}^{n}}{2} \right] + \frac{\delta_y^2}{\Delta y^2} \left[\frac{U_{j,k}^{n+1} + U_{j,k}^{n}}{2} \right] \right] \quad (4)$$

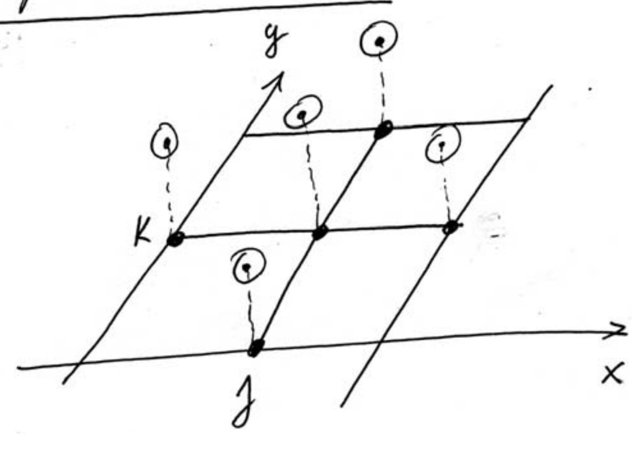
This can also be written as

$$\left(1 - \frac{r_x \delta_x^2 + r_y \delta_y^2}{2}\right) U_{j,k}^{n+1} = \left(1 + \frac{r_x \delta_x^2 + r_y \delta_y^2}{2}\right) U_{j,k}^{n}$$

$$\delta_x^2 U_{j,k}^{n} = U_{j-1,k}^{n} - 2U_{j,k}^{n} + U_{j+1,k}^{n}$$

$$\delta_y^2 U_{j,k}^{n} = U_{j,k-1}^{n} - 2U_{j,k}^{n} + U_{j+1,k}^{n}$$

Computational Stencil



Matrix Representation

Equ. (4) Can also be written as

$$U^{n+1}_{j,k} - \frac{r_x}{2}\,\delta_x^2\, U^{n+1}_{j,k} - \frac{r_y}{2}\quad U^{n+1}_{j,k} = \underbrace{\left[1 + \frac{r_x\delta_x^2 + r_y\delta_y^2}{2}\,U^{n}_{j,k}\right]}_{\equiv\, \text{rhs}}$$

or expanding $\delta_x^2 U^{n+1}_{j,k}$ in the lhs.

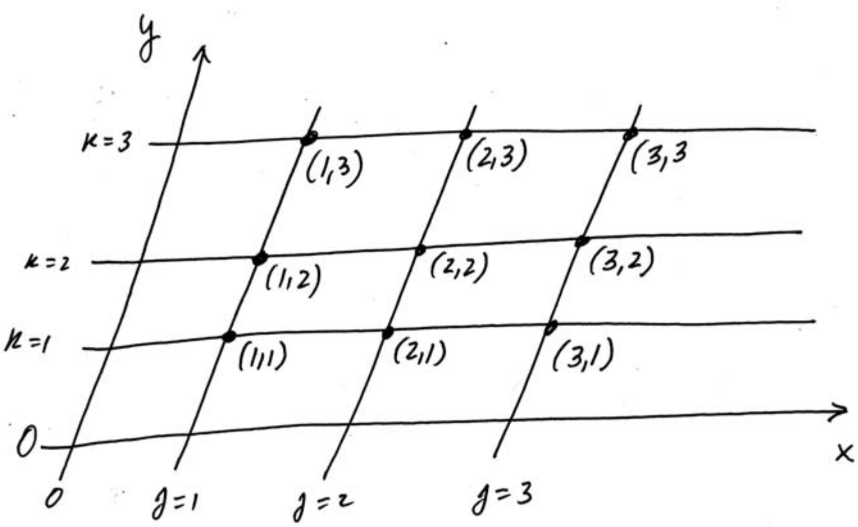
$$U^{n+1}_{j,k} + \frac{1}{2}\left[-r_x U^{n+1}_{j-1,k} + 2r_x U^{n+1}_{j,k} - r_x U^{n+1}_{j+1,k}\right] +$$

$$+\,\frac{1}{2}\left[-r_y U^{n+1}_{j,k-1} + 2r_y U^{n+1}_{j,k} - r_y U^{n+1}_{j,k+1}\right] = \text{rhs}$$

or $$U^{n+1}_{j,k} + \frac{1}{2}\left[-r_y U^{n+1}_{j,k-1} - r_x U^{n+1}_{j-1,k} + 2(r_x+r_y)U^{n+1}_{j,k} - r_x U^{n+1}_{j+1,k} - r_y U^{n+1}_{j,k+1}\right] =$$

$$= \text{rhs}.$$

To construct the matrix consider



If BC's are homogeneous (DIRICHLET).

$\boxed{K=1}$

$$U_{11}^{n+1}+\frac{1}{2}\left[2(r_x+r_y)U_{11}^{n+1}-r_x U_{2,1}^{n+1}-r_y U_{1,2}^{n+1}\right]=\text{rhs}.$$

$$U_{2,1}^{n+1}+\frac{1}{2}\left[-r_x U_{11}^{n+1}+2(r_x+r_y)U_{21}^{n+1}-r_x U_{3,1}^{n+1}-r_y U_{22}^{n+1}\right]=\text{rhs}.$$

$$\vdots$$

$$U_{Nx-1,1}^{n+1}+\frac{1}{2}\left[-r_x U_{Nx-2,1}^{n+1}+2(r_x+r_y)U_{Nx-1,1}^{n+1}-r_y U_{Nx-1,2}^{n+1}\right]=\text{rhs}$$

$\boxed{K=2}$

$$U_{1,2}^{n+1}+\frac{1}{2}\left[-r_y U_{11}^{n+1}+2(r_x+r_y)U_{1,2}^{n+1}-r_x U_{2,2}^{n+1}-r_y U_{1,3}^{n+1}\right]=\text{rhs}$$

$$U_{2,2}^{n+1}+\frac{1}{2}\left[-r_y U_{2,1}^{n+1}-r_x U_{1,2}^{n+1}+2(r_x+r_y)U_{2,2}^{n+1}-r_x U_{3,2}^{n+1}-r_y U_{2,3}^{n+1}\right]=\text{rhs}$$

$$\vdots$$

$$U_{Nx-1,2}^{n+1}+\frac{1}{2}\left[-r_y U_{Nx-1,1}^{n+1}-r_x U_{Nx-2,2}^{n+1}+2(r_x+r_y)U_{Nx-1,2}^{n+1}-r_y U_{Nx-1,3}^{n+1}\right]=\text{rhs}$$

$$\vdots$$

$K=N_y-1$.

It's convenient to define the unknown Vector $\vec{U}^n$ by rows.

$$\vec{U}^n=\begin{bmatrix} \left.\begin{matrix} U_{11}^n \\ U_{21}^n \\ \vdots \\ U_{Nx-1,1}^n \end{matrix}\right\} \text{1st row in x-direction} \\ \left.\begin{matrix} U_{1,2}^n \\ U_{22}^n \\ \vdots \\ U_{Nx-1,2}^n \end{matrix}\right\} \text{2nd row in x-direction} \\ \vdots \\ \left.\begin{matrix} U_{1,Ny-1}^n \\ U_{2,Ny-1}^n \\ \vdots \\ U_{Nx-1,Ny-1}^n \end{matrix}\right\} \text{Last row in x-direction.} \end{bmatrix}$$

MATRIX REPRESENTATION.

$$
\begin{array}{ll|cccccccccccc}
 & & U_{1,1}^{n+1} & U_{2,1}^{n+1} & \cdots & U_{N_x-2,1}^{n+1} & U_{N_x-1,1}^{n+1} & U_{1,2}^{n+1} & U_{2,2}^{n+1} & \cdots & U_{N_x-1,2}^{n+1} & U_{1,3}^{n+1} & \cdots & \\
\hline
 & j=1 & 2(r_x+r_y) & -r_x & 0 & 0 \cdots & 0 & -r_y & 0 & 0 \cdots & 0 & 0 & \cdots 0 & 0 \\
 & j=2 & -r_x & 2(r_x+r_y) & -r_x & 0 \cdots & 0 & 0 & -r_y & 0 \cdots & 0 & 0 & \cdots 0 & 0 \\
k=1 & j=3 & 0 & \ddots & \ddots & \ddots & & & & \ddots & & & & \\
 & \vdots & \vdots & & & 2(r_x+r_y) & -r_x & 0 & 0 & \cdots -r_y & & & & \\
 & j=N_x-1 & 0 & \cdots & & -r_x & 2(r_x+r_y) & 0 & 0 & \cdots & -r_y & 0 & & \\
 & j=1 & -r_y & 0 & 0 & 0 \cdots & 0 & 2(r_x+r_y) & -r_x & 0 \cdots & 0 & -r_y & & \\
 & j=2 & 0 & -r_y & & 0 \cdots & 0 & -r_x & 2(r_x+r_y) & -r_x \; 0 \cdots & 0 & & -r_y & \\
k=2 & \vdots & \vdots & & \ddots & & & & \ddots & \ddots & -r_x & & & \\
 & j=N_x-1 & 0 & \cdots & & 0 & -r_y & 0 & \cdots & -r_x & 2(r_x+r_y) & & & \\
 & & \vdots & & & \vdots & \vdots & & & & \vdots & & &
\end{array}
$$

The matrix Representation of the previous page Can be written in compact form as

$$\boxed{\left[I + \tfrac{1}{2} C\right] \vec{U}^{\,n+1} = \left[I - \tfrac{1}{2} C\right] \vec{U}^{\,n}} \qquad (8.1)$$

where

$$C \equiv \begin{bmatrix} D_x & D_y & 0 & 0 & \cdots & 0 \\ D_y & D_x & D_y & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & 0 \\ \vdots & & & D_y & D_x & D_y \\ 0 & \cdots & \cdots & 0 & D_y & D_x \end{bmatrix}_{\left[(N_x-1)(N_y-1)\right] \times \left[(N_x-1)(N_y-1)\right]}$$

$$D_x \equiv \begin{bmatrix} 2(r_x+r_y) & -r_x & 0 & \cdots & \cdots & 0 \\ -r_x & 2(r_x+r_y) & -r_x & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & 0 \\ \vdots & & \ddots & 2(r_x+r_y) & & -r_x \\ \vdots & & & -r_x & & \\ 0 & \cdots & \cdots & 0 & -r_x & 2(r_x+r_y) \end{bmatrix}_{(N_x-1) \times (N_x-1)}$$

$$D_y \equiv \begin{pmatrix} -r_y & 0 & \cdots & 0 \\ 0 & -r_y & \cdots & 0 \\ \vdots & & \ddots & 0 \\ 0 & 0 & \cdots & -r_y \end{pmatrix}$$

The linear system (8.1) can be solved by an extension of the tridiagonal algorithm to block systems.

It requires $\frac{5}{3} N_y (N_x)^3$ multiplications per time step.

Too expensive!

where

$$a = -\frac{\alpha\,\Delta t}{2(\Delta y)^2} = -\frac{1}{2}r_y$$

$$b = -\frac{\alpha\,\Delta t}{2(\Delta x)^2} = -\frac{1}{2}r_x$$

$$c = 1 + r_x + r_y$$

$$d_{i,j}^n = u_{i,j}^n + \frac{\alpha\,\Delta t}{2}(\hat{\delta}_x^2 + \hat{\delta}_y^2)u_{i,j}^n$$

If we apply Eq. (4-101) to the two-dimensional (6 × 6) computational mesh shown in Fig. 4-15, the following system of 16 linear algebraic equations must be solved at each $(n+1)$ time level.

$$\left[\begin{array}{cccc|cccc|cccc|cccc}
c & b & 0 & 0 & a & 0 & & & 0 & & & & & & & 0 \\
b & c & b & & & a & & & & & & & & & & \\
0 & b & c & b & & & a & & & & & & & & & \\
0 & & b & c & 0 & & & a & & & & & & & & \\
\hline
a & & & 0 & c & b & 0 & \cdots & a & & & & & & & \\
0 & a & & & b & c & b & & & a & & & & & & \\
& & a & & & b & c & b & & & a & & & & & \\
& & & a & & & b & c & 0 & & & a & & & & \\
\hline
& & & & a & & & 0 & c & b & & & a & & & \\
& & & & & a & & & b & c & b & & & a & & \\
& & & & & & a & & & b & c & b & & & a & 0 \\
& & & & & & & a & & & b & c & 0 & & & a \\
\hline
& & & & & & & & a & & & 0 & c & b & & 0 \\
& & & & & & & & & a & & & b & c & b & 0 \\
& & & & & & & & & & a & & & b & c & b \\
0 & & & & & & & & & & 0 & a & 0 & 0 & b & c
\end{array}\right]
\begin{bmatrix}
u_{2,2}^{n+1} \\ u_{3,2}^{n+1} \\ u_{4,2}^{n+1} \\ u_{5,2}^{n+1} \\ u_{2,3}^{n+1} \\ u_{3,3}^{n+1} \\ u_{4,3}^{n+1} \\ u_{5,3}^{n+1} \\ u_{2,4}^{n+1} \\ u_{3,4}^{n+1} \\ u_{4,4}^{n+1} \\ u_{5,4}^{n+1} \\ u_{2,5}^{n+1} \\ u_{3,5}^{n+1} \\ u_{4,5}^{n+1} \\ u_{5,5}^{n+1}
\end{bmatrix}
=
\begin{bmatrix}
d_{2,2}''' \\ d_{3,2}' \\ d_{4,2}' \\ d_{5,2}''' \\ d_{2,3}'' \\ d_{3,3} \\ d_{4,3} \\ d_{5,3}'' \\ d_{2,4}'' \\ d_{3,4} \\ d_{4,4} \\ d_{5,4}'' \\ d_{2,5}''' \\ d_{3,5}' \\ d_{4,5}' \\ d_{5,5}'''
\end{bmatrix}
\qquad (4\text{-}102)$$

where $d' = d - au_0$

$d'' = d - bu_0$

$d''' = d - (a+b)u_0$

A system of equations, like Eq. (4-102), requires substantially more computer time to

ADI METHOD.

It's a predictor-corrector scheme

For example, Peaceman-Rachford Version of it.

PREDICTOR STEP:

$$\boxed{U_{j,k}^{n+1/2} - \frac{r_x}{2}\delta_x^2 U_{j,k}^{n+1/2} = U_{j,k}^{n} + \frac{r_y}{2}\delta_y^2 U_{j,k}^{n}} \qquad (1)$$

Formula (1) can be obtained this way

Split the continuous wave equation in 2. as

$$\frac{1}{2} u_t = \sigma u_{xx} \quad \text{and} \quad \frac{1}{2} u_t = \sigma u_{yy}$$

Use BT-CS at $(x_j, y_k, t_{n+1/2})$ time step $\Delta t/2$

$$\frac{1}{2}\left[\frac{U_{j,k}^{n+1/2} - U_{j,k}^{n}}{\Delta t/2}\right] = \sigma \delta_x^2 U_{j,k}^{n+1/2}$$

$$\Rightarrow U_{j,k}^{n+1/2} - r_x \delta_x^2 U_{j,k}^{n+1/2} = U_{j,k}^{n} \qquad (2)$$

Use FT-CS at (x_j, y_k, t_n) with time step size $\Delta t/2$.

$$\frac{1}{2}\left[\frac{U_{j,k}^{n+1/2} - U_{j,k}^{n}}{\Delta t/2}\right] = \sigma \delta_y^2 U_{j,k}^{n}$$

$$\Rightarrow U_{j,k}^{n+1/2} = r_y \delta_y^2 U_{j,k}^{n} + U_{j,k}^{n} \qquad (3)$$

Adding (2) and (3).

$$2U_{j,k}^{n+1/2} - r_x \delta_x^2 U_{j,k}^{n+1/2} = r_y \delta_y^2 U_{j,k}^{n} + 2U_{j,k}^{n}$$

$$\Rightarrow \quad \boxed{U_{j,k}^{n+1/2} - \frac{r_x}{2}\delta_x^2 U_{j,k}^{n+1/2} = \frac{r_y}{2}\delta_y^2 U_{j,k}^{n} + U_{j,k}^{n}} \qquad (1)$$

IMPLICIT in The x-Direction.

Corrector step:

$$\boxed{U_{j,k}^{n+1} - \frac{r_y}{2}\delta_y^2 U_{j,k}^{n+1} = U_{j,k}^{n+1/2} + \frac{r_x}{2}\delta_x^2 U_{j,k}^{n+1/2}} \qquad (4)$$

Formula (4) can be obtained as follows:

$\frac{1}{2} u_{tt} = \sigma u_{xx}$ and $\frac{1}{2} u_{tt} = \sigma u_{yy}$

FT-CS at $(x_j, y_k, t_{n+1/2})$ with time step size $\Delta t/2$

$$\frac{1}{2}\left[\frac{U_{j,k}^{n+1} - U_{j,k}^{n+1/2}}{\Delta t/2}\right] = \sigma \delta_x^2 U_{j,k}^{n+1/2}$$

$$\Rightarrow U_{j,k}^{n+1} = r_x \delta_x^2 U_{j,k}^{n+1/2} + U_{j,k}^{n} \qquad (5)$$

BT-CS at (x_j, y_k, t_{n+1}) with time step size $\Delta t/2$.

$$\frac{1}{2}\left[\frac{U_{j,k}^{n+1} - U_{j,k}^{n+1/2}}{\Delta t/2}\right] = \sigma \delta_y^2 U_{j,k}^{n+1}$$

$$\Rightarrow U_{j,k}^{n+1} - r\,\delta_y^2 U_{j,k}^{n+1} = U_{j,k}^{n+1/2} \qquad (6)$$

Adding (5) and (6).

$$2U_{j,k}^{n+1} - r\,\delta_y^2 U_{j,k}^{n+1} = r_x \delta_x^2 U_{j,k}^{n+1/2} + 2U_{j,k}^{n+1/2}$$

$$\Rightarrow \boxed{U_{j,k}^{n+1} - \frac{r_y}{2}\delta_y^2 U_{j,k}^{n+1} = \frac{r_x}{2}\delta_x^2 U_{j,k}^{n+1/2} + U_{j,k}^{n+1/2}} \qquad (7)$$

Implicit in the y-Direction.

The two finite difference equations to be solved are

(I) Implicit in the x-direction:

$$\boxed{U_{j,k}^{n+1/2} - \frac{r_x}{2}\delta_x^2 U_{jk}^{n+1/2} = \frac{r_y}{2}\delta_y^2 U_{jk}^{n} + U_{jk}^{n}.}$$ Predictor.

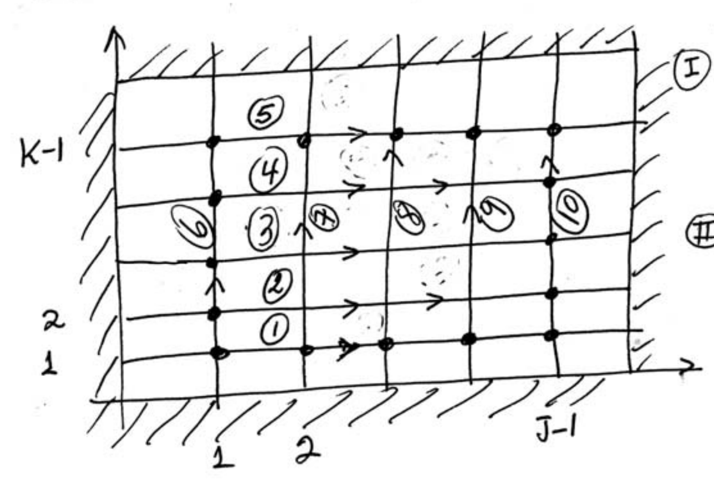
(II) Implicit in the y direction:

$$\boxed{U_{jk}^{n+1} - \frac{r_y}{2}\delta_y^2 U_{jk}^{n+1} = \frac{r_x}{2}\delta_x^2 U_{jk}^{n+1/2} + U_{jk}^{n+1/2}}$$ Corrector.

Solution:

(I) is solved using the tridiagonal algorithm row by row. That way, we can obtain all the approximate values at time level $t_{n+1/2}$. Each row solution is obtained independently from all other rows.

Order in the Computation



(I) Predictor comp. row - row to obtain $\vec{U}^{n+1/2}$.

(II) Corrector comp. Column-Column to obtain $\vec{U}^{n+1}$

(I) Matrix Representation of Predictor step.

Row k:

$$\boxed{[I + C_x]\,\vec{U}_k^{\,n+1/2} = \vec{g}_{y,k}^{\,n}} \qquad (8)$$

where

$$\vec{U}_k^{\,n+1/2} = \begin{bmatrix} U_{1,k}^{n+1/2} \\ U_{2,k}^{n+1/2} \\ \vdots \\ U_{J-1,k}^{n+1/2} \end{bmatrix}, \qquad C_x = \frac{r_x}{2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 2 \end{bmatrix}$$

$$g_{y,k}^{n} = \begin{bmatrix} U_{1,k}^{n} + \frac{r_y}{2}\delta_y^2 U_{1,k}^{n} \\ \vdots \\ U_{J-1,k}^{n} + \frac{r_y}{2}\delta_y^2 U_{J-1,k}^{n} \end{bmatrix}$$

We need to solve this system for each row $k = 1, 2, , K-1$. using the tridiagonal algorithm.

Similarly, Once all Values at time level $t_{n+1/2}$ have been obtained, the next corrector step will compute all Values at time level t_{n+1}. This is done column by column using

$$[I + C_y]\,U_j^{n+1} = g_{x,j}^{n+1/2}$$

$$\vec{U}_j^{n+1} = \begin{bmatrix} U_{j,1}^{n+1} \\ U_{j,2}^{n+1} \\ \vdots \\ U_{j,K-1} \end{bmatrix}, \quad C_y = \frac{r_y}{2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & & 0 \\ & & \ddots & & \\ & & & -1 & 2 \end{bmatrix}, \quad g_{x,j}^{n+1/2} = \begin{bmatrix} U_{j,1}^{n+1/2} + \frac{r_x}{2}\delta_x^2 U_{j,1}^{n} \\ \vdots \\ U_{j,K-1}^{n+1/2} + \frac{r_x}{2}\delta_x^2 U_{j,K-1}^{n} \end{bmatrix}$$

Of operations (Complexity of algorithm).

PREDICTOR: Each tridiag. System requires $\simeq 5J$ operations

Operation: One multipl/division + One add/subtr.

A tridiag. System is solved $K-1$ times, then

operations Predictor step $= 5J(K-1) \simeq 5JK$.

Corrector: $J-1$ tridiag. Systems require $\simeq 5K$ operations of $K-1$ dimension. This is done $J-1$ times

$\Rightarrow$ # Operations Corrector step $\simeq 5K(J-1) \simeq 5JK$

TOTAL # operations $\simeq 10JK$. much less than $\frac{5}{3}KJ^3$. employed when solving Crank-Nicholson tridiag. block System.

It can be proved (Homework).

a) Local discretization error of ADI is given by

$$\tilde{\tau}^n_{J,K} = (\tilde{\tau}^n_{J,K})_{C-N} + \mathcal{O}(\Delta t^2) = \mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta y^2) + \mathcal{O}(\Delta t^2).$$

b) Using Von Neumann method. that Peaceman-Rachford numerical scheme is unconditionally stable, Stable for all positive Values of r_x and r_y.