ADI (Summary)

Predictor: at $(x_j, y_k, t_{n+1/2}$ in x, t_n in $y)$

$BT-CS_x^{n+1/2}-CS_y^{n}$

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

Implicit in the x-direction at $t_{n+1/2}$

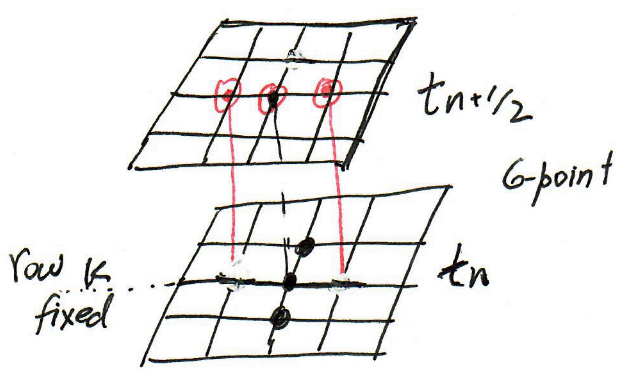
Corrector:

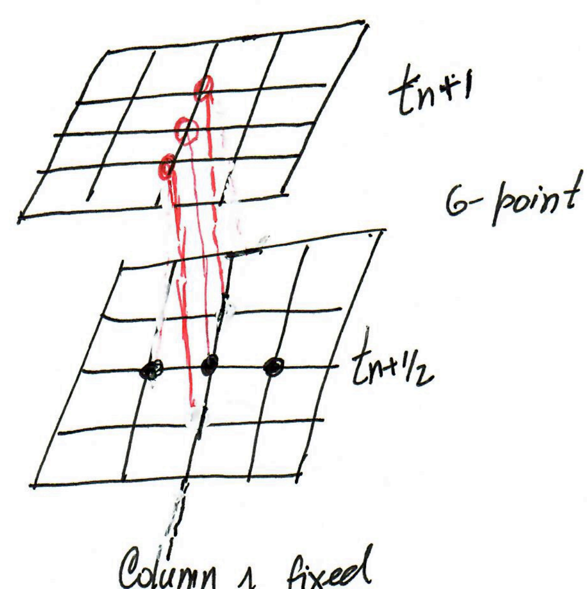
$BT-CS_y^{n+1}-CS_x^{n+1/2}$

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

Implicit in the y-direction at t_{n+1}.

Predictor





Splitting Heat Equation

PREDICTOR:

$$\frac{1}{2} u_t = \sigma u_{xx} \qquad (2.1)$$

$BT^{n+1/2} - CS_x^{n+1/2}$ $(\Delta t/2)$

$$U_{j,k}^{n+1/2} - r_x \delta_x^2 U_{j,k}^{n+1/2} = U_{j,k}^{n} \qquad (2.3)$$

$$\tfrac{1}{2} u_t = \sigma u_{yy} \qquad (2.2)$$

$FT^{n} - CS_y^{n}$ $(\Delta t/2)$

$$U_{j,k}^{n+1/2} = r_y \delta_y^2 U_{j,k}^{n} + U_{j,k}^{n} \qquad (2.4)$$

Notice $(2.1) + (2.2) \rightarrow u_t = \sigma (u_{xx} + u_{yy})$

The new Numerical scheme from (2.3) and (2.4)

$$(2.3) + (2.4) \longrightarrow \boxed{U_{j,k}^{n+1/2} - \frac{r_x}{2} \delta_x^2 U_{j,k}^{n+1/2} = U_{j,k}^{n} + \frac{r_y}{2} \delta_y^2 U_{j,k}^{n}} \qquad (2.5)$$

CORRECTOR:

$$\frac{1}{2} u_t = \sigma u_{xx}$$

$FT^{n+1/2} - CS_x^{n+1/2}$ $(\Delta t/2)$

$$U_{j,k}^{n+1} = r_x \delta_x^2 U_{j,k}^{n+1/2} + U_{j,k}^{n+1/2} \qquad (2.6)$$

$$\tfrac{1}{2} u_t = \sigma u_{yy}$$

$BT^{n+1} - CS_y^{n+1}$ $(\Delta t/2$

$$U_{j,k}^{n+1} - r_y \delta_y^2 U_{j,k}^{n+1} = U_{j,k}^{n+1/2} \qquad (2.7)$$

$$(2.5) + (2.6) \longrightarrow \boxed{U_{j,k}^{n+1} - \frac{r_y}{2} \delta_y^2 U_{j,k}^{n+1} = U_{j,k}^{n+1/2} + \frac{r_x}{2} \delta_x^2 U_{j,k}^{n+1/2}} \qquad (2.8)$$

0°

ADI Motivation.

The main motivation of ADI method is to construct a FDM that overcome the high computational cost of Crank-Nicholson but maintaining its good properties.

More precisely, we want to construct a FDM with the following properties:

i) Order of accuracy: $\mathcal{O}(\Delta x^2) + \mathcal{O}(\Delta y^2) + \mathcal{O}(\Delta t^2)$.

ii) Unconditionally stable. It means no restrictions on $\Delta x, \Delta y, \Delta t$.

iii) The number of operations per time step of the same order of # of unknowns, i.e,

$$\# \text{ Operations} = \mathcal{O}((N_x)(N_y))$$

Conditions (i) and (ii) are satisfied by C-N.

However, (iii) is not since

$$\# \text{ operations C-N} = \mathcal{O}(N_x (N_y)^3).$$

ADI METHOD. Construction Strategy:

The idea is to compute all unknowns at time $t_{n+1/2}$ from known values at $t=t_n$. along a row "k" independently of the other rows.

This will reduce two things compared with C-N:

i) The size of the bandwith $\longrightarrow$ tridiagonal.

ii) The size of the linear system for each row is $[N_x-1]_\times[N_x-1]$ (tridiagonal)$^{(*)}$. Of course, we need to multiply the work done by the total number of rows. So, total # operations $= \mathcal{O}((N_x)(N_y))$.

Then, perform a second step to go from $t=t_{n+1/2} \longrightarrow t=t_{n+1}$

This time computing unknowns along a column "j" independently of the other columns.

The first step will be called Predictor-step.
The second step will be called Corrector-step.

(*) Remark: The total # operations of a tridiagonal system $M\times M$ is $\mathcal{O}(M)$.

ADI METHOD. DERIVATION

Consider heat equation:

$$u_t = \sigma u_{xx} + \sigma u_{yy}$$

An alternative to C-N. is to use BT-CS to approximate only u_t and u_{xx}, at $(x_j, y_k, t_{n+1/2})$ with $\Delta t/2$ time step and at the same time use a centered approximation of u_{yy} at (x_j, y_k, t_n).

In fact,

$$(u_t)_{jk}^{n+1/2} = \sigma (u_{xx})_{jk}^{n+1/2} + \sigma (u_{yy})_{jk}^{n}$$

Thus;

$$U_{jk}^{n+1/2} - U_{jk}^{n} = \frac{\sigma \Delta t}{2\Delta x^2} \delta_x^2 U_{jk}^{n+1/2} + \frac{\sigma \Delta t}{2\Delta y^2} \delta_y^2 U_{jk}^{n}$$

or

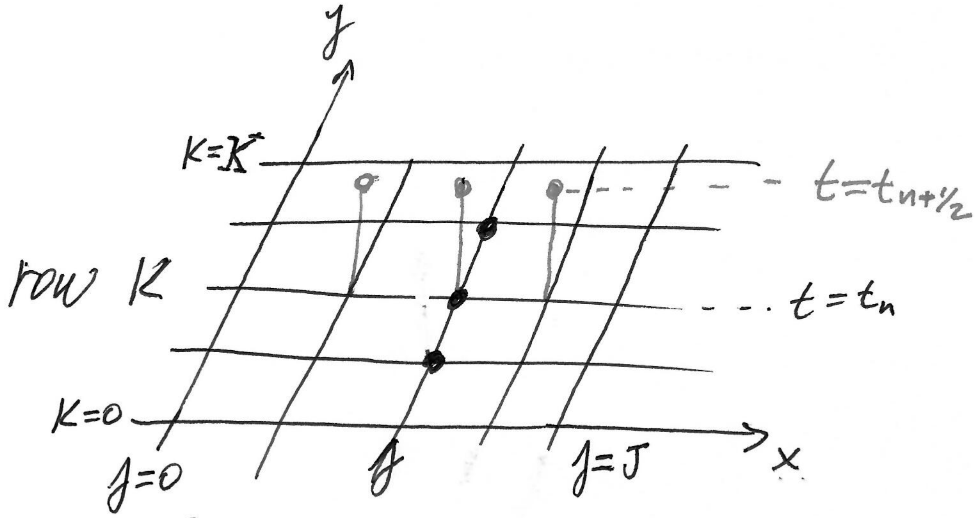
$$\boxed{U_{jk}^{n+1/2} - \frac{r_x}{2} \delta_x^2 U_{jk}^{n+1/2} = U_{jk}^{n} + \frac{r_y}{2} \delta_y^2 U_{jk}^{n}} \qquad (1.1)$$

Predictor-step

$k = 1, 2, \ldots, \underline{K}-1$, y-axis

$j = 1, 2, \ldots, J-1$, x-axis

Implicit in x-direction.



Computational stencil. Predictor-step

This is equivalent to split the continuous H.E. operator in two:

$\frac{1}{2} u_t = \sigma u_{xx}$	$\frac{1}{2} u_t = \sigma u_{yy}$
$BT-CS_x$ at	$FT-CS_y$ at
$(x_j, y_k, t_{n+1/2})$ with $\Delta t/2$	(x_j, y_k, t_n) with $\Delta t/2$
$U_{jk}^{n+1/2} - r_x \delta_x^2 U_{jk}^{n+1/2} = U_{jk}^n$ (2.1)	$U_{jk}^{n+1/2} = r_y \delta_y^2 U_{jk}^n + U_{jk}^n$ (2.2)

Adding: $(2.1) + (2.2) = (1.1)$.

So, the predictor step can be seen as a combined contribution of

Implicit in x from $t_{n+1/2}$ and explicit in y from t_n.
both with $\Delta t/2$ time-step.

ADI

$k=1$ $\quad t_n \to t_{n+1/2}$ $\quad$ (1.1) page 1.

$$
\begin{cases}
j=1 & U_{11}^{n+1/2}\left(1+2\frac{r_x}{2}\right) - \frac{r_x}{2} U_{21}^{n+1/2} = rhs^n + \frac{r_x}{2} \underbrace{U_{0,1}^{n+1/2}}_{=BC} \\
j=2 & -\frac{r_x}{2} U_{11}^{n+1/2} + (1+r_x) U_{2,1}^{n+1/2} - \frac{r_x}{2} U_{3,1}^{n+1/2} = rhs^n \\
\vdots & \\
j=J-1: & -\frac{r_x}{2} U_{J-2,1} + (1+r_x) U_{J-1,1} = rhs^n + \frac{r_x}{2} \underbrace{U_{J1}^{n+1/2}}_{=BC}
\end{cases}
$$

$$
\begin{array}{c}
j=1 \\ j=2 \\ \vdots \\ \vdots \\ j=J-1
\end{array}
\underbrace{\begin{bmatrix}
(1+r_x) & -\frac{r_x}{2} & \cdots & & 0 \\
-\frac{r_x}{2} & (1+r_x) & -\frac{r_x}{2} & 0 \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \\
 & & & 1+r_x & -r_x/2 \\
0 & & & -\frac{r_x}{2} & 1+r_x
\end{bmatrix}}_{(J-1)\times(J-1)}
\begin{bmatrix}
U_{11} \\ U_{21} \\ \vdots \\ \vdots \\ U_{J-1,1}
\end{bmatrix}
= rhs
$$

Similar, for
$k=2,3,\ldots, K-1$.

or

$$\left[I + \frac{r_x}{2} C \right] \vec{v}_1^{\,n+1/2} = rhs.$$

where

$$C = \begin{bmatrix} 2 & -1 & 0 & 0 & \cdots & & 0 \\ -1 & 2 & -1 & 0 & \cdots & & 0 \\ 0 & -1 & 2 & 1 & & & \\ \vdots & & \ddots & \ddots & \ddots & & \\ \vdots & & & \ddots & \ddots & \ddots & -1 \\ & & & & -1 & 2 & \\ 0 & & & & & -1 & 2 \end{bmatrix}_{(J-1)\times(J-1)}$$

<u>Complexity</u>:

$$\# \text{ operations} = \mathcal{O}\left((J-1) \right)$$

Why (1.1) is not enough?

Stability problems.

<u>Corrector step</u> → BT-CS$_y$ at (x_j, y_k, t_{n+1}) with $\frac{\Delta t}{2}$ time step

Implicit in y-direction from time $t = t_{n+1}$ with step $\Delta t/2$ while u_{xx} is approximated at $t = t_{n+1/2}$ using centered difference.

$$(u_t)_{jk}^{n+1} = \sigma (u_{xx})_{jk}^{n+1/2} + \sigma (u_{yy})_{jk}^{n+1}$$

Then,

$$(3.2)+(3.3) \Rightarrow \boxed{v_{jk}^{n+1} - \frac{r_y}{2}\delta_y^2 v_{jk}^{n+1} = v_{jk}^{n+1/2} + \frac{r_x}{2}\delta_x^2 v_{jk}^{n+1/2}} \quad (3.1)$$

Corrector-step. $j = 1, 2, \ldots, J-1$ x-axis

Implicit in y-direction. $k = 1, 2, \ldots, K-1$, y-axis

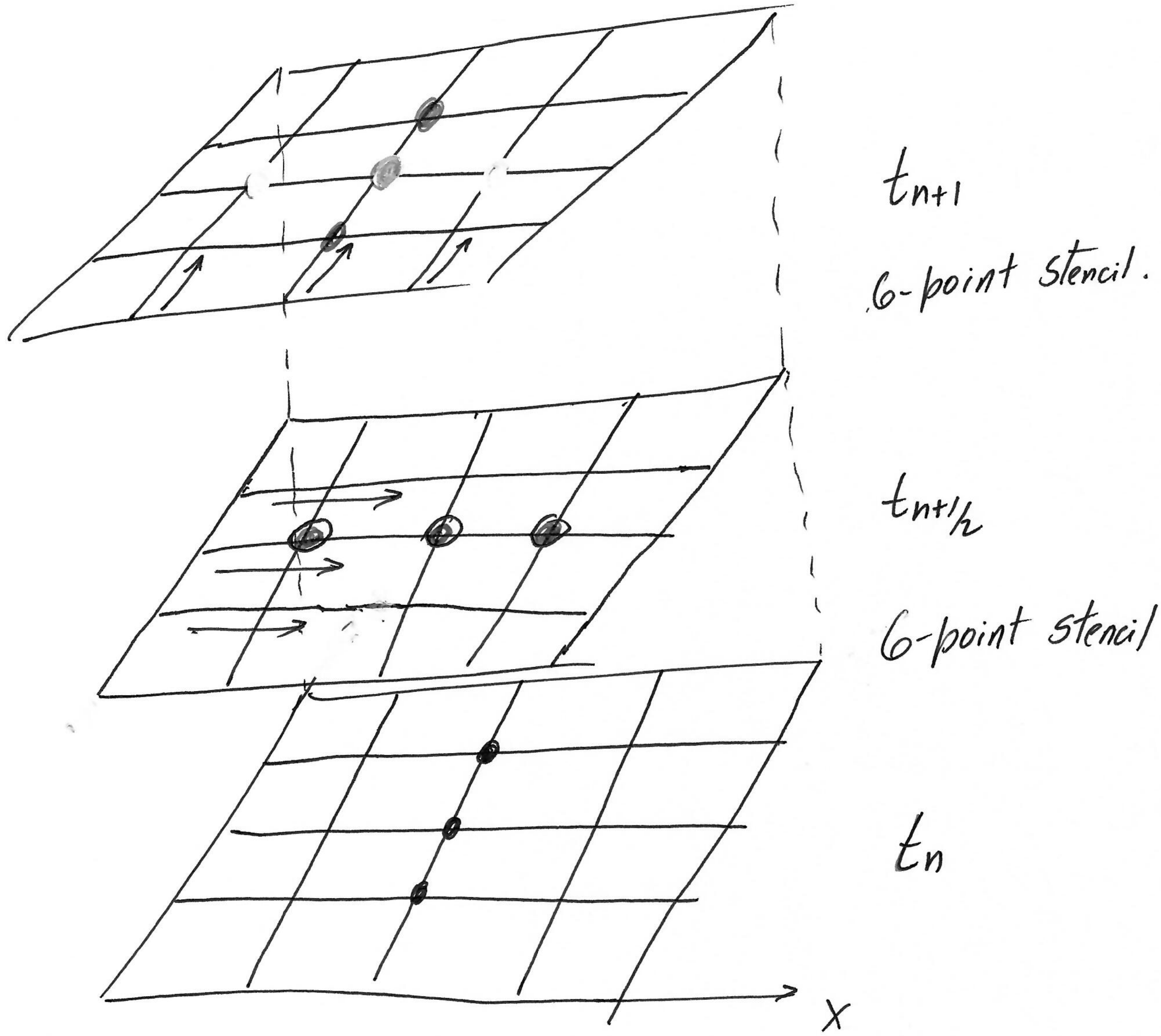
It can also be obtained using split operator:

$\frac{1}{2} u_t = \sigma u_{xx}$	$\frac{1}{2} u_t = \sigma u_{yy}$
FT-CS$_x$ at $(x_j, y_k, t_{n+1/2})$ with $\Delta t/2$	BT-CS$_y$ at (x_j, y_k, t_{n+1}) with $\Delta t/2$

And adding the two contributions.

$$v_{jk}^{n+1} = r_x \delta_x^2 v_{jk}^{n+1/2} + v_{jk}^{n+1/2} \quad (3.2)$$

$$v_{jk}^{n+1} = r_y \delta_y^2 v_{jk}^{n+1} + v_{jk}^{n+1/2} \quad (3.3)$$


t_{n+1}
6-point stencil.
$t_{n+1/2}$
6-point stencil
t_n
x

The two finite difference equations to be solved are

(I) Implicit in the x-direction at $t_{n+1/2}$

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

Predictor.
$k = 1, 2, .., K-1$ y-axis
$j = 1, 2, .., J-1$ x-axis
Implicit in x-direct.

(II) Implicit in the y-direction at t_{n+1}

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

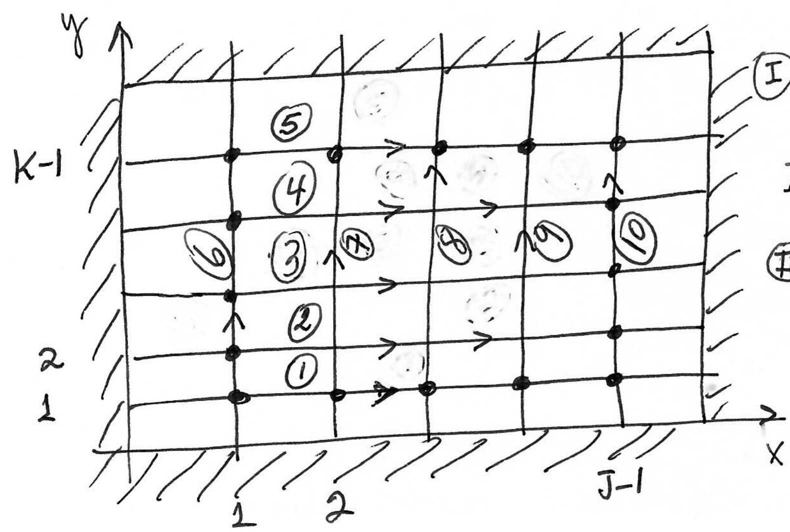
Corrector.
$j = 1, 2, \ldots J-1$
$k = 1, 2, .., K-1$
Implicit in y-direct.

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
Implicit $\vec{U}^{n+1/2}$ from $\vec{U}^n$

(II) Corrector comp.
Column-Column to obtain
$\vec{U}^{n+1}$ from $\vec{U}^{n+1/2}$
Implicit.

(I) Matix Representation of Predictor step.

Row K: $$\boxed{\left[I + C_x\right]\vec{U}_K^{\,n+1/2} = g^n_{y,K}} \qquad (8)$$

where

$$\vec{U}_K^{\,n+1/2} = \begin{bmatrix} U_{1,K}^{n+1/2} \\ U_{2,K}^{n+1/2} \\ \vdots \\ U_{J-1,K}^{n+1/2} \end{bmatrix}, \qquad C_x = \frac{r_x}{2}\begin{bmatrix} 2 & -1 & 0 & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ & \ddots & \ddots & \ddots & \\ & & & -1 & 2 \end{bmatrix}_{(J-1)\times(J-1)}$$

$$g^n_{y,K} = \begin{bmatrix} U_{1,K}^{n} + \frac{r_y}{2}\delta_y^2 U_{1,K}^{n} + \frac{r_x}{2}U_{0,K}^{n+1/2} \\ \vdots \\ U_{j,K}^{n} + \frac{r_y}{2}\delta_y^2 U_{j,K}^{n} \\ \vdots \\ U_{J-1,K}^{n} + \frac{r_y}{2}\delta_y^2 U_{J-1,K}^{n} + \frac{r_x}{2}U_{J,K}^{n+1/2} \end{bmatrix}$$

We need to solve this system for each row $\boxed{K = 1, 2, , K-1}$ 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

Column j: $$\boxed{\left[I + C_y\right]\vec{U}_j^{\,n+1} = g^{n+1/2}_{x,j}}, \qquad j = 1, 2, ., J-1 \qquad (9)$$

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

For K fixed, Comput. along row K.

$j=1$

$$U_{1,k}^{n+1/2} - \frac{r_x}{2}\left[U_{0,k}^{n+1/2} - 2U_{1,k}^{n+1/2} + U_{2,k}^{n+1/2}\right] = U_{1,k}^{n} + \frac{r_y}{2}\delta_y^2 U_{1,k}^{n}$$

$$\Rightarrow \quad \left(1 - \frac{r_x}{2}2\right)U_{1,k}^{n+1/2} - \frac{r_x}{2}U_{2,k}^{n+1/2} = U_{1,k}^{n} + \frac{r_y}{2}\delta_y^2 U_{1,k}^{n} + \frac{r_x}{2}U_{0,k}^{n+1/2}$$

$j=2$

$$-\frac{r_x}{2}U_{1,k}^{n+1/2} + \left(1 - \frac{r_x}{2}2\right)U_{2,k}^{n+1/2} - \frac{r_x}{2}U_{3,k}^{n+1/2} = \left(U_{2,k}\right)$$

$\vdots$

$j=J-1$

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

Of operations (Complexity of algorithm).

PREDICTOR: Each tridiag. System requires $\simeq 5(J-1)$ operations

J-1 unknowns, column

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

⇒ A tridiag. System is solved $\boxed{K-1 \text{ times}}$ then

rows

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

Corrector: J-1 tridiag. Systems require $\simeq 5(K-1)$ operations

K-1: unknws. rows

of K-1 dimension. This is done (J-1) times

columns

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

TOTAL # operations $\simeq$ 10 JK. much less than $\frac{5}{3}KJ^3$.

employed when solving Crane-Nickolson tridiag. block system.

It can be proved (Homework).

a) Local discretization error of ADI is given by

$$\tau_{J,K}^n = (\tau_{J,K}^n)_{C-N} + O(\Delta t^2) = O(\Delta x^2) + O(\Delta y^2) + O(\Delta t^2).$$

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