۱

<< برنامه FCTS >> هیدرولیک محاسباتی - (۱۴۰۰/۰۴/۰۲) جلسه پنجم

<< برنامه‌نویسی تمرین کلاسی جلسه قبل >>

```
clc;
clear
close all

% determine of Input parameters

X=1.0
T=1.0
dx= 0.1
dT=0.001
Alpha=1.0
```

$nx = \frac{X}{dx} + 1.0 \longrightarrow$ تعداد گره های مکانی

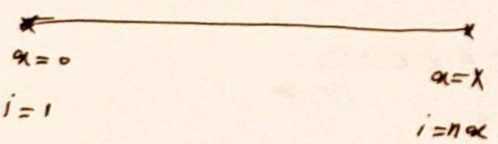
$nt = \frac{T}{dt} + 1.0 \longrightarrow$ تعداد گره های زمانی

```
% End of this section

u=zeros(nx,nt)
```

$\longrightarrow$ *تعریف ماتریس*

```
% Boundary condition
```



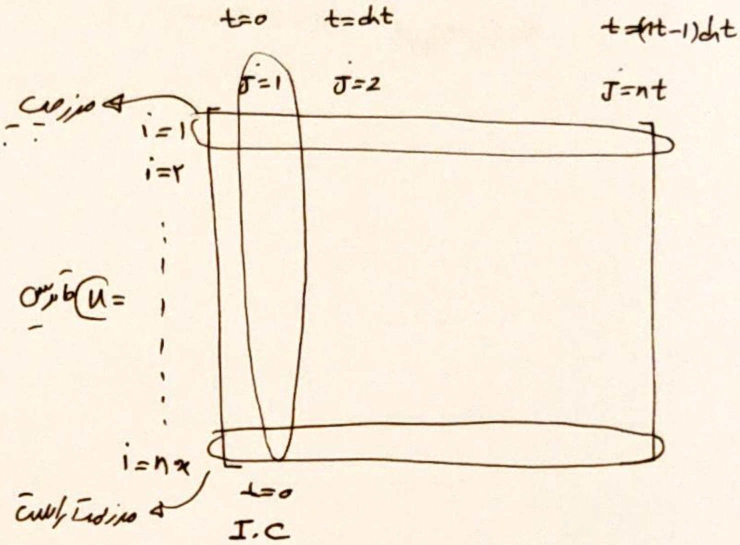

```
For J=1,nt
  u(1,J)=0
end
```

مرز چپ

```
For J=1,nt
  u(nx,J)=0
end
```

مرز راست

<< هر ستون معادل یک گام زمانی >>



%. Initial condition $\quad C(x,0)=C_0$

بیان ریاضی

Initial condition

معادل شرط اول همان شرایط ابتدا

$$C(x,0)\begin{cases} 2x & x\le \frac{1}{2} \\ 2(1-x) & \frac{1}{2}\le x\le 1.0 \end{cases}$$

روش ۱

```
For i=1,nx
  c(i,1)=
  xj=(i-1)dx
  if xj<0.5
    c(i,1)=2xj
  else
    c(i,1)=2x(1-xj)
end
```

روش ۲

```
For i=1,nx/2
  c(i,1)=2x(i-1)dx
           x
  end
For i=nx/2,nx
  c(i,1)=2x(1-(i-1)dx)
```

معادله دیفرانسیل گسسته سازی به عنوان برای بدست آوردن ماتریس ها یا همان C

روش FTCS برای محاسبه مقادیر c(i,j) مورد استفاده قرار میگیریم

$$\boxed{C_i^{n+1}=rC_{i+1}^{n}+(1-2r)C_i^{n}+rC_{i-1}^{n}}$$

%. main Body of program

```
For j=2,nt
  i=2,nx
  c(i,j)= rc(i+1,j-1) + (1-2r)c(i,j-1) + rc(i-1,j-1)
          (C_{i+1}^n)     (C_i^n)          (C_{i-1}^n)
  end
end
```

۲

ترسیم گراف و نمودار

برای ترسیم گراف ها می توان ماتریس u را در اکسل کپی نموده و هر ستون آن جداگانه ترسیم نمود و همچنین می توان در خود متلب نیز این گراف ها را ترسیم نمود.

```
For j=1:nt
plot ( u(:,j) )
```

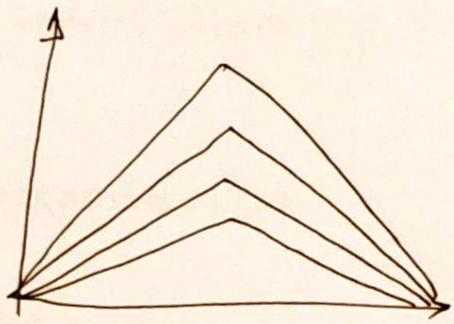
→ «هر دفعه j می باشد»

i=1,nx

j=1 u(1,1), u(2,1), u(3,1),....

```
title(' Ali ')
xlable('x(m)')
ylable('c(gr/lit)')
hold on
```

→ * کلیه نمودارها در زمانهای مختلف را در یک شکل نمایش می دهد.



```
end
```

هیدرولیک محاسباتی - (۱۴۰۰/۰۲/۰۲) جلسه پنجم

(Ex-1

$$c_i^{n+1} = r c_{i+1}^{n} + (1-2r) c_i^{n} + r c_{i-1}^{n} \quad \text{«۱»}$$

در روش FTCS که یک روش صریح می‌باشد به کمک یک معادله یا همان معادله ① توانستیم غلظت یا سرعت در هر گره را محاسبه نمائیم به این نوع روش‌ها، روش‌های صریح یا explicit گفته می‌شود و نیاز به حل دستگاه معادلات نمی‌باشد.

اما مشکل روش‌های صریح آن است که دارای محدودیت پایداری می‌باشند، در جلسه قبل گفتیم که محدودیت پایداری این روش‌ها $r<0.5$ مطرح می‌گردد و این محدودیت در انتخاب گام زمانی اعمال می‌گردد.

$$\boxed{\begin{array}{c} r=\alpha\dfrac{\Delta t}{\Delta x^2} < 0.5 \longrightarrow \Delta t < \dfrac{0.5\,\Delta x^2}{\alpha} \\ \Delta x = 0.1 \\ \alpha = 1.0 \end{array}} \longrightarrow \Delta t < 0.5 \times 0.01 = 0.005$$

» برای رفع این محدودیت سراغ روش‌های ضمنی می‌رویم:
implicit

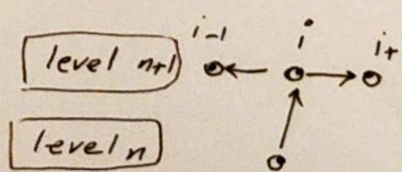
① روش BTCS (Backward time center space):

«معادله diffusion» $\dfrac{\partial c}{\partial t} = \alpha \dfrac{\partial^2 c}{\partial x^2}$

$$\underbrace{\frac{c_i^{n} - c_i^{n-1}}{\Delta t}}_{\boxed{B.T}} = \alpha \underbrace{\frac{c_{i-1}^{n} - 2c_i^{n} + c_{i-1}^{n}}{\Delta x^2}}_{\boxed{C.S}}$$

برای درک بهتر معلومات و مجهولات در رابطه فوق، ($n \longrightarrow n+1$) جایگزین می‌کنیم:

$$\frac{\overset{\text{مجهول}}{c_i^{n+1}} - \overset{\text{معلوم}}{c_i^{n}}}{\Delta t} = \alpha \frac{\overset{\text{مجهول}}{c_{i+1}^{n+1}} - 2\overset{\text{مجهول}}{c_i^{n+1}} + \overset{\text{مجهول}}{c_{i-1}^{n+1}}}{\Delta x^2}$$



$$\frac{\alpha \Delta t}{\Delta x^2} = r$$

$$r c_{i+1}^{n+1} - (1+2r) c_i^{n+1} + r c_{i-1}^{n+1} = -c_i^{n} \quad (2)$$

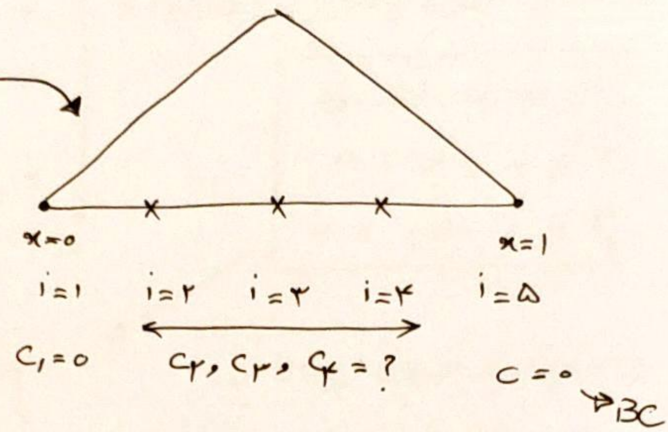
همانطور که ملاحظه می‌گردد از معادله ② بصورت صریح و ساده نمی‌توان مقدار c_i^{n+1} را محاسبه نمود.

و c_i^{n+1} به مقادیر c_{i+1}^{n+1} و c_{i-1}^{n+1} وابسته شده است و بایستی یک دستگاه معادلات حل گردد.

لذا از روش های ضمنی اگرچه محدودیت و مشکل پایداری ندارند ولی از لحاظ حجم محاسبات از پیچیدگی بیشتری برخوردارند:

<< فرض نمائید همان مسأله قبل را به روش BTCS با $r=0/5$

را می خواهیم حل نمائیم معادله ② را برای $i=2$ ، $i=3$ و $i=4$



خواهیم داشت <<

$\boxed{i=2}$

(با توجه به B.C) ⓪

$$rC_3^{n+1}-(1+2r)C_2^{n+1}+\cancel{rC_1^{n+1}}=-C_2^n$$

$$\boxed{rC_3^{n+1}-(1+2r)C_2^{n+1}=-C_2^n} \quad (3)$$

$\boxed{i=3}$

$$\boxed{rC_4^{n+1}-(1+2r)C_3^{n+1}+rC_2^{n+1}=-C_3^n} \quad (4)$$

$\boxed{i=4}$

(با توجه به B.C) 0

$$\cancel{rC_5^{n+1}}-(1+2r)C_4^{n+1}+rC_3^{n+1}=-C_4^n$$

$$\boxed{-(1+2r)C_4^{n+1}+rC_3^{n+1}=-C_4^n} \quad (5)$$

معادلات ۳ ، ۴ و ۵ را می توان بصورت یک دستگاه ۳ معادله ، ۳ مجهول نوشت:

$$\begin{bmatrix} -(1+2r) & r & 0 \\ r & -(1+2r) & r \\ 0 & r & -(1+2r) \end{bmatrix} \begin{bmatrix} C_2^{n+1} \\ C_3^{n+1} \\ C_4^{n+1} \end{bmatrix} = \begin{bmatrix} -C_2^n \\ -C_3^n \\ -C_4^n \end{bmatrix}$$

ماتریس ضرایب $[A]$ ، $[u_i^{n+1}]$ بردار مجهولات $=$ $[R]$ بردار پاسخ

* در روش ضمنی یک دستگاه معادلات بصورت $[A][u]=[R]$ بایستی حل گردد.

۳

درصورتیکه تعداد گره‌ها افزایش یابد ماتریس A همواره یک ماتریس ۳ قطری باقی می‌ماند

$$A=\begin{bmatrix} -(1+2r) & r & 0 & 0 & 0 & 0 & 0 \\ r & -(1+2r) & r & 0 & 0 & 0 & 0 \\ 0 & r & -(1+2r) & r & 0 & 0 & 0 \end{bmatrix}$$

<< ماتریس A همواره یک ماتریس ۳ قطری می‌باشد

1) یک عنصر روی قطر اصلی
2) یک " پایین " "
3) " " بالای " "

$$A=\begin{bmatrix} b_1 & c_1 & 0 & 0 & 0 & 0 & 0 \\ a_2 & b_2 & c_2 & 0 & 0 & 0 & 0 \\ 0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\ & & & b_i & c_i & & \\ & & & & & a_n & b_n \end{bmatrix}$$

قطر اصلی

صفر

صفر

بالای قطر اصلی

پایین قطر اصلی

<< ماتریس ضرایب در روش BTCS نیز ماتریس ۳ قطری می‌باشد. >>

$b_i=-(1+2r)$

$c_i=r$

$a_i=r$

برای حل دستگاه معادلات $[A][u]=[R]$ می‌توان از روش مستقیم استفاده نمود:

<< طرفین را ضرب در $[A^{-1}]$ می‌کنیم. >>

$$[A]^{-1}[A][u]=[A]^{-1}[R] \longrightarrow \boxed{u=[A]^{-1}[R]}$$

اما این روش، به هیچ عنوان برای حل‌های عددی توصیه نمی‌گردد و به دست آوردن $[A]^{-1}$ درصورتیکه تعداد گره‌ها زیاد باشد بسیار زمان‌بر و غیراقتصادی است. لذا از روش‌های با راندمان بالاتر برای حل دستگاه معادلات استفاده می‌کنیم.

<< یکی از این روش‌ها، روش حذفی گوس می‌باشد که ضرایب ماتریس را بالا مثلثی یا پایین مثلثی می‌کند:

$$[A]\longrightarrow[\bar{A}]=\begin{bmatrix} \bar{b}_1 & \bar{c}_1 & 0 & 0 & 0 & & \\ 0 & \bar{b}_2 & \bar{c}_2 & 0 & 0 & & \\ & & \bar{b}_3 & & & & \\ & & & \ddots & & & \\ & & & & b_i & & \\ & & & & & \ddots & \\ & & & & & \bar{b}_{n-1} & \bar{c}_{n-1} \\ & & & & & & b_n \end{bmatrix}\begin{bmatrix} u_1^{n+1} \\ \vdots \\ \vdots \\ u_{n-1}^{n+1} \\ u_n^{n+1} \end{bmatrix}=\begin{bmatrix} R_1^{n+1} \\ R_2^{n+1} \\ \vdots \\ \bar{R}_n \end{bmatrix}$$

$$[\bar{A}]\underset{\text{بردار}}{[u]}=\underset{\text{بردار}}{[\bar{R}]}$$

<< بعد از بالا مثلثی یا پایین مثلثی بودن ماتریس [A] می‌توان مجهولات را از آخرین گره شروع به تعیین نمود >>

ضرایب

۴

گره آخر

«محاسبه می‌شود»

گره اول

$$u_n^{n+1} = \frac{\bar{R}_n}{\bar{b}_n} \longrightarrow \text{آخرین گره}$$

$$u_{n-1}^{n+1} = \frac{\bar{R}_{n-1} - \bar{c}_{n-1}\, u_n^{n+1}}{\bar{b}_{n-1}}$$

«از گره آخر به سمت گره اول مجهولات همگی محاسبه می‌شود»

* در جلسه آینده:

Ex

۱- ارسال برنامه FTCS تا روز یکشنبه برای مثال حل شده در کلاس

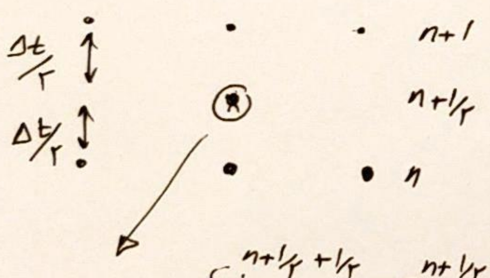
۲- ارسال برنامه FTCS برای مسئله پیاز

۳- بررسی نحوه‌ی حل دستگاه معادلات سه قطری به روش حذفی گوس و روش توماس، تفاوت آنها

«آموزه: الگوریتم و برنامه حذفی گوس برای حل دستگاه معادلات را برای شما برنامه نویسی می‌کنیم.»

روش ضمنی (Implicit) Cranck-Nickelson

«روش ضمنی BTCS دارای دقت (Δx^2 و Δt) بوده و در روش کرانک-نیکلسون دقت حل عددی به (Δx^2 و Δt^2) افزایش یابد. برای این منظور گسسته سازی مشتق زمانی در یک زمان مجازی $n+1/2$ به صورت زیر انجام می‌شود



$$\frac{\partial c}{\partial t} = \frac{c_i^{n+1/2+1/2} - c_i^{n+1/2-1/2}}{\left(\frac{2\Delta t}{2}\right)} + T.E(\Delta t)^2$$

گسسته سازی مشتق زمانی

$$\frac{\partial c}{\partial t} = \frac{(c_i^{n+1} - c_i^n)}{\Delta t}$$

$$\alpha \frac{\partial^2 c}{\partial x} = \frac{\alpha}{2}\left(\frac{c_{i+1}^{n} - 2c_i^{n} + c_{i-1}^{n}}{\Delta x^2} + \frac{c_{i+1}^{n+1} - 2c_i^{n+1} + c_{i-1}^{n+1}}{\Delta x^2}\right)$$

گسسته سازی مشتق مکانی

$$\frac{\partial c}{\partial t} = \alpha \frac{\partial^2 c}{\partial x^2} \longrightarrow [A]$$

$$\begin{bmatrix} b_i & c_i \\ a_i & \\ & \end{bmatrix}$$

Ex-۲) برای روش کرانک نیکلسون مقدار ماتریس $[A]$ و $[R]$ را بدست آورید.

$$[A][u^{n+1}]=[R]$$