دانشگاه شهید بهشتی

دانشکده مهندسی مکانیک و انرژی

پروژه حل عددی جریان کوئت

درس دینامیک سیالات محاسباتی

استاد :دکتر رسام

نیم سال اول 1400-99

اعضای گروه :

اتابک بهادرنیا

امیرعلی حسینی

کامیار بخشی پور

پیام بقایی

روز و تاریخ تحویل گزارش :یکشنبه 99/7/27

با استفاده از معادله ناویر-استوکس و اعمال فرضیات ساده کننده بر روی آن،میتوان معادله نهایی دیفرانسیلی را برای جریان کوئت بدست آورد. بر این اساس میتوان نوشت:

$$Assumptions: v = w = 0$$
$$u = u(y)$$
$$Unsteady\ and\ incompressible\ flow$$
$$Negligible\ pressure\ gradient$$
$$Body\ forces\ are\ zero\ in\ x\ direction$$

$$for\ x\ component\ of\ momentum: \quad \rho\frac{Du}{Dt} = -\nabla P + \mu\nabla^2 u + f_x$$

$$\rho\left(\frac{\partial u}{\partial t} + \cancel{u\frac{du}{dx}} + \cancel{v\frac{du}{dy}} + \cancel{w\frac{du}{dz}}\right) = \cancel{-\nabla P} + \mu\left(\cancel{\frac{\partial u^2}{\partial^2 \mathrm{x}}} + \frac{\partial u^2}{\partial^2 y} + \cancel{\frac{\partial u^2}{\partial^2 z}}\right) + \cancel{f_x}$$

$$\rho\frac{\partial u}{\partial t} = \mu\frac{\partial u^2}{\partial^2 y} \xrightarrow{\text{در نتیجه}} \frac{\partial u}{\partial t} = \nu\frac{\partial u^2}{\partial^2 y}$$

معادله بالا یک PDE غیر خطی مرتبه دو است و نمیتوان آن را به روش عادی حل کرد. یکی از روش های موجود برای برای حل عددی این معادله، روش صریح است. در این روش گسسته سازی در time level “n” صورت میگیرد.با استفاده از گسسته سازی زمانی با تقریب تفاضل محدود پیشرو مرتبه یک و گسسته سازی مکانی با تقریب تفاضل محدود مرکزی مرتبه 2 ، معادله بالا به شکل زیر گسسته سازی میشود:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} = \nu\frac{u_{i+1}^n - u_i^n + u_{i-1}^n}{\Delta x^2}$$

بعبارتی:

$$u_i^{n+1} = \frac{\nu\Delta t}{\Delta x^2}\left(u_{i+1}^n - u_i^n + u_{i-1}^n\right) + u_i^n$$

با استفاده از معادله بالا میتوان در زمان پیشروی کرد و مقادیر سرعت را در زمان ها و سرعت های مجهول بدست آورد.

این روش تنها درصورتی پایدار است که شرط زیر ارضا شود:

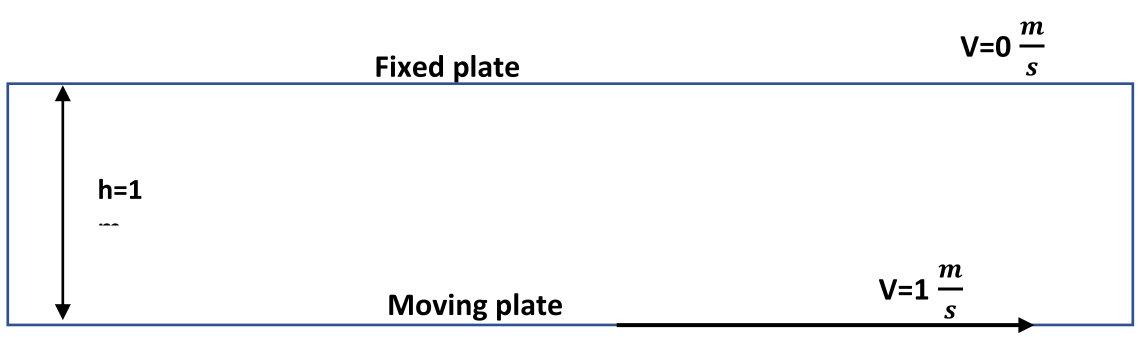
$$\frac{\nu \Delta t}{\Delta x^2} \leq \frac{1}{2}$$

با استفاده از برنامه نویسی میتوان دستگاه معادلات نقاط مختلف در زمان های مختلف را نوشت و با اعمال شروط اولیه و مرزی آن را حل کرد. به این ترتیب پاسخ حل عددی بدست خواهد آمد. شروط مرزی و اولیه این مسئله عبارتند از:

$$initial\ condition: \quad u(0,1) = 1\frac{m}{s}$$
$$boundry\ condition: \quad u(t,0) = 1\frac{m}{s} \quad , \quad u(t,1) = 0\frac{m}{s}$$

پارامتر های سیستم عبارتند از:

$$h = 1 : distance\ of\ two\ plates$$
$$\nu = 2.17 * 10^{-4}\frac{m^2}{s} : kinematic\ viscosity$$



با استفاده از نرم افزار متلب دستگاه معادلات نقاط و زمان های مختلف را تشکیل میدهیم. برای اینکار تعداد گام های مکانی را برابر 30 و فاصله زمانی حل را حداکثر مقدار ممکن در نظر میگیریم در نظر میگیریم. شرط پایداری را بررسی میکنیم:

$$\frac{\nu \Delta t}{(\Delta x)^2} \leq \frac{1}{2} \xrightarrow{yields} (\Delta t)_{max} = \frac{(\Delta x)^2}{2\nu}$$

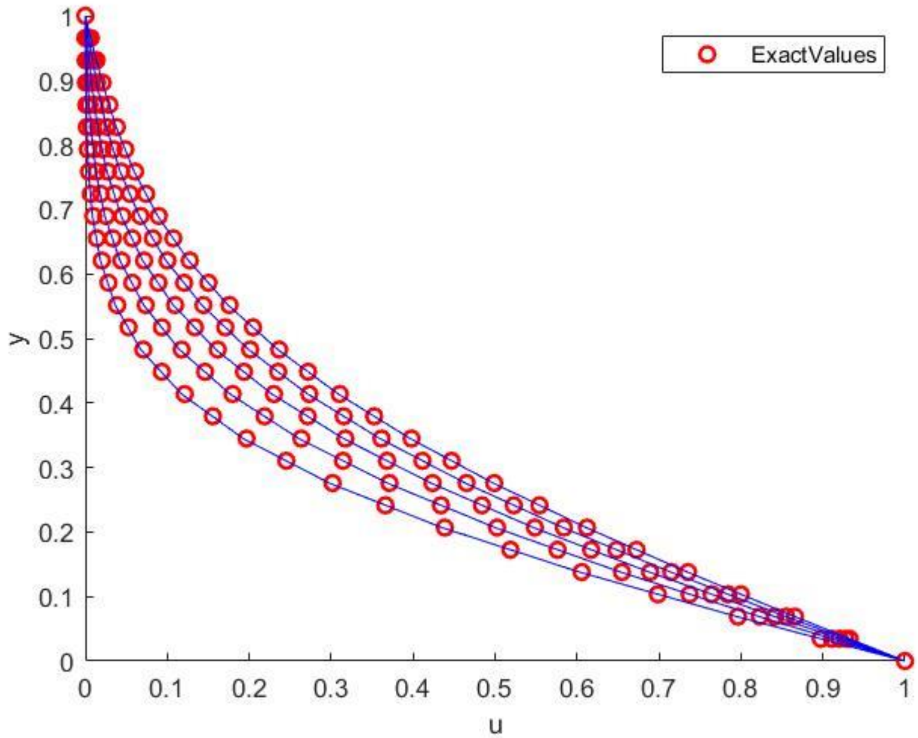
کد متلب :

```
clc;
clear all;
%----------------------System Parameters--------------------
----------------
    h= 1;   %distance of two plates
    U=1;    %velocity
    ny= 30; % space nodes number
    u_numerical=zeros(ny,1); %velocity vector definition
    u_numerical(1,1)=U;   %boundary condition
    u_numerical(ny,1)=0;  %boundary condition
    ntstart=input("input the first time level number?\n");
    dy= h/(ny-1);             %space step size for y
direction
    v= 2.17e-4;               %kinematic viscosity (momentum
diffusivity)
    dt=(dy^2)/(2*v);          % maximum time step size for a
stable solve
    NUM_diffusivity= (v*dt)/(dy^2); %Numerical_diffusivity
    TimeLevelDistance=input("distance between  every two
tandem time levels.\n");   %distance of two time levels
which we would like to plot
    steps= TimeLevelDistance*input("how many time levels
you need to plot?\n");
%----------------Plotting exact and numerical solution-----
----------------
figure(1)
hold on;
tic;
for numt=ntstart-1:ntstart+(steps-1)-1 %plotting time range
if (mod(numt,TimeLevelDistance)==0)  %an auxiliary condtion
for a better plotting
        uexact=exactsolution(numt,dt,ny,dy,U,h,v);
        x = 0:dy:(ny-1)*dy;
        A=plot(uexact,x,'ro','linewidth',1.5);
        ylabel('y')
        xlabel('u')
        hold on;
end
end
for numt=2:ntstart+steps-1;
u_numerical=FTCS(u_numerical,ny,U,NUM_diffusivity);
```

```
        if ((numt>=ntstart &&
mod(numt,TimeLevelDistance)==0))
        x = 0:dy:(ny-1)*dy;
        B=plot(u_numerical,x,'b-')
        hold on;
        end
end
        legend('ExactValues')
%--------------------------Error calculator----------------
----------------
figure(2)
hold on;
nyinitial=9;
nyfinal=14;
for ny=nyinitial:nyfinal
dy= h/(ny-1);
u_numerical=zeros(ny,1); %velocity vector definition
u_numerical(1,1)=U;   %boundary condition
u_numerical(ny,1)=0;  %boundary condition
dt=10; %time step fixed.
            numt=ny;
            uexact=exactsolution(numt,dt,ny,dy,U,h,v);
        for numt=2:(ny+1);

u_numerical=FTCS(u_numerical,ny,U,NUM_diffusivity);
            end
        e(ny-nyinitial+1,1)=err(uexact,u_numerical,ny);
        delta(ny-nyinitial+1,1)=dy;
end
loglog(delta,e,'k-o','linewidth',2)
toc;
set(gca,'fontsize',14)
legend('FTCS','Location','southeast')
```

نمودار تطبیق مقادیر حل دقیق و حل عددی برای پنج لحظه مختلف:



نمودار مقادیر خطای روش عددی FTCS به ازای گام های مکانی مختلف: