www.NoavaranGermi.ir

به نام خدا

گزارش کار هیدرولیک محاسباتی

عنوان پروژه:حل معادله موج دو بعدی به روش لیپ فراگ

(Solved equation wave 2D by metod leap frog)

نام و نام خانوادگی:سهیل محمودی

شماره دانشجویی:9414135016

نام استاد: جناب دکتر جباری

پاییز 95

www.NoavaranGermi.ir

فهرست

مقدمه.....4

معادله موج دو بعدی..... 4

معادله موج مرتبه اول دو بعدی به روش لیپ فراگ..... 5

خطای برشی معادله موج مرتبه اول دو بعدی به روش لیپ فراگ..... 5

معادله بهسازی شده معادله موج مرتبه اول دو بعدی به روش لیپ فراگ.....7

مزایا روش لیپ فراگ..... 7

معایب روش لیپ فراگ..... 7

نکته مهم..... 8

شرح پروژه..... 9

شرایط اولیه در زمان n=1 9

شرایط اولیه در زمان n=2 استفاده از روش تفاضل بالا دست..... 10

www.NoavaranGermi.ir

اعمال شرایط مرزی..... 10

شرایط مرزی دیریشله..... 10

شرایط مرزی نیومان..... 11

کد نویسی متلب..... 11

عنوان یک مثال دو بعدی..... 15

نتایج خروجی از متلب..... 15

بررسی پایداری..... 16

نتایج خروجی از متلب برای پایداری در دستگاه دکارتی..... 18

نتایج خروجی از متلب برای پایداری در دستگاه قطبی.....18

صحت سنجی به روش اول(دستی).....19

صحت سنجی برای زمان n=1 19

www.NoavaranGermi.ir

نتایج برای زمان $n=1$ با خروجی *matla* 20

صحت سنجی برای زمان n=2 20

نتایج برای زمان $n=2$ با خروجی *matlab*..... 23

صحت سنجی برای زمان n=3 24

نتایج برای زمان $n=3$ با خروجی *matlab* 25

صحت سنجی به روش دوم(مقایسه روش لیپ فراگ با *ftcs*)..... 26

کد متلب به روش *ftcs*..... 26

مقایسه به روش گرافیکی..... 29

مقایسه به روش عددی..... 30

www.NoavaranGermi.ir

مقدمه:

در حالت کلی معادله موج مرتبه اول یک بعدی به صورت زیر تعریف میشود:

$$\frac{\partial u}{\partial t} = -c\frac{\partial u}{\partial x}$$

که در معاده بالا c سرعت موج می باشد. برای چنین معادله خطوط مشخصه خطهای مستقیمی با معادله($x\text{-}at=conest$) هستند.متغیر u در امتداد این خطوط با مقدار ثابت C جابجا می شود.روشهای مختلف تقریب تفاضل محدود این معادله موج مرتبه اول را پژوهشگران مختلف بررسی کردند. این روش ها ضمنی وصریح فرمول بندی شده است.

ما در این پروژه قصد داریم معادل موج را با روش صریح لیپ فراگ مورد بررسی قرار دهیم.

معرفی معادله موج دو بعدی:

حال اگر بخواهیم معادله موج دردو بعدی مرتبه اول را تعریف کنیم به صورت زیر تعریف میشود که با فرض سرعت موج در هر دو جهت x,y با هم برابر است:

$$\frac{\partial \mathrm{u}}{\partial \mathrm{t}} + cx\frac{\partial u}{\partial x} + cy\frac{\partial u}{\partial y} = 0$$

به ازای مقدار ثابت cx و cy یک معادله خطی است. برای چنین معادله ساده ای خطوط مشخصه، خط های مستقیمی هستند. متغیر u در امتداد این خطوط با مقدار ثابت cx و cy جابجا می شود. روش های مختلف تقریب تفاضل محدود این معادله ی موج مرتبه ی اول را پژوهشگران مختلف بررسی کرده اند. این تقریب ها در شکلهای صریح و ضمنی فرمولبندی شده اند که ما روش صریح لیپ فراگ را در این قسمت ارایه می کنیم.

معادله موج مرتبه اول دوبعدی به روش لیپ فراگ:

وقتی که روش لیپ فراگ را بر معادله موج مرتبه دوم اعمال میکنیم یک روش صریح با سه سطع زمانی بدست می اید که معادله گسترش آن به صورت زیر می باشد:

$$\frac{U_{i,j}^{n+1}-U_{i,j}^{n-1}}{2\Delta t}=-c_x\left(\frac{U_{i+1,j}^{n}-U_{i-1,j}^{N}}{2\Delta x}\right)-c_y(\frac{U_{i,j+1}^{n}-U_{i,j-1}^{n}}{2\Delta y})$$

در روش لیپ فراگ هم برای مشتق های زمانی هم برای مشتق های مکانی از تفاضل مرکزی استفاده شده است.روش لیپ فراگ را روش با سه سطح زمانی می نامند زیرا برای یافتن U در زمان n+1 نیاز به معلوم بودن U در زمان های n ,n-1 است.

خطای برشی معادله موج به روش لیپ فراگ:

$$\frac{u_{i,j}^{n+1}-u_{i,j}^{n-1}}{2\Delta t}+c\frac{u_{i,j+1}^{n}-u_{i,j-1}^{n}}{2\Delta x}+c\frac{u_{i+1,j}^{n}-u_{i-1,j}^{n}}{2\Delta y}=0$$

$$u_{i,j}^{n+1}=u_{i,j}^{n}+\frac{\partial \mathrm{u}}{\partial \mathrm{t}}\Delta t+\frac{\partial^2 u}{\partial t^2}\frac{\Delta t^2}{2}+\frac{\partial^3 u}{\partial t^3}\frac{\Delta t^3}{6}+\cdots$$

$$u_{i,j}^{n-1}=u_{i,j}^{n}-\frac{\partial \mathrm{u}}{\partial \mathrm{t}}\Delta t+\frac{\partial^2 u}{\partial t^2}\frac{\Delta t^2}{2}-\frac{\partial^3 u}{\partial t^3}\frac{\Delta t^3}{6}+\cdots$$

دو رابطه را از هم کم می کنیم:

$$u_{i,j}^{n+1}-u_{i,j}^{n-1}=2\frac{\partial \mathrm{u}}{\partial \mathrm{t}}\Delta t+2\frac{\partial^3 u}{\partial t^3}\frac{\Delta t^3}{6}+\cdots\rightarrow\frac{\partial \mathrm{u}}{\partial \mathrm{t}}=\frac{u_{i,j}^{n+1}-u_{i,j}^{n-1}}{2\Delta t}+\frac{\partial^3 u}{\partial t^3}\frac{\Delta t^2}{6}+..$$

$$u_{i,j+1}^{n}=u_{i,j}^{n}+\frac{\partial \mathrm{u}}{\partial x}\Delta x+\frac{\partial^2 u}{\partial x^2}\frac{\Delta x^2}{2}+\frac{\partial^3 u}{\partial x^3}\frac{\Delta x^3}{6}+\cdots$$

$$u_{i,j-1}^{n} = u_{i,j}^{n} - \frac{\partial \mathrm{u}}{\partial x}\Delta x + \frac{\partial^2 u}{\partial x^2}\frac{\Delta x^2}{2} - \frac{\partial^3 u}{\partial x^3}\frac{\Delta x^3}{6} + \cdots$$

دو رابطه را از هم کم می کنیم:

$$u_{i,j+1}^{n} - u_{i,j-1}^{n} = 2\frac{\partial \mathrm{u}}{\partial x}\Delta x + 2\frac{\partial^3 u}{\partial x^3}\frac{\Delta x^3}{6} + \cdots \rightarrow \frac{\partial \mathrm{u}}{\partial x} = \frac{u_{i,j+1}^{n} - u_{i,j-1}^{n}}{2\Delta x} + \frac{\partial^3 u}{\partial x^3}\frac{\Delta x^2}{6} + ..$$

$$u_{i+1,j}^{n} = u_{i,j}^{n} + \frac{\partial \mathrm{u}}{\partial y}\Delta y + \frac{\partial^2 u}{\partial y^2}\frac{\Delta y^2}{2} + \frac{\partial^3 u}{\partial y^3}\frac{\Delta y^3}{6} + \cdots$$

$$u_{i-1,j}^{n} = u_{i,j}^{n} - \frac{\partial \mathrm{u}}{\partial y}\Delta y + \frac{\partial^2 u}{\partial y^2}\frac{\Delta y^2}{2} - \frac{\partial^3 u}{\partial y^3}\frac{\Delta y^3}{6} + \cdots$$

دو رابطه را از هم کم می کنیم:

$$u_{i+1,j}^{n} - u_{i-1,j}^{n} = 2\frac{\partial \mathrm{u}}{\partial y}\Delta x + 2\frac{\partial^3 u}{\partial y^3}\frac{\Delta y^3}{6} + \cdots \rightarrow \frac{\partial \mathrm{u}}{\partial y} = \frac{u_{i+1,j}^{n} - u_{i-1,j}^{n}}{2\Delta y} + \frac{\partial^3 u}{\partial y^3}\frac{\Delta y^2}{6} + ..$$

$$\frac{\partial \mathrm{u}}{\partial \mathrm{t}} + c\frac{\partial \mathrm{u}}{\partial x} + c\frac{\partial \mathrm{u}}{\partial y} = 0$$

$$\frac{u_{i,j}^{n+1} - u_{i,j}^{n-1}}{2\Delta t} + c\frac{u_{i,j+1}^{n} - u_{i,j-1}^{n}}{2\Delta x} + c\frac{u_{i+1,j}^{n} - u_{i-1,j}^{n}}{2\Delta y} + c\frac{\partial^3 u}{\partial x^3}\frac{\Delta x^2}{6} + \frac{\partial^3 u}{\partial t^3}\frac{\Delta t^2}{6} + c\frac{\partial^3 u}{\partial y^3}\frac{\Delta y^2}{6} \ldots = 0$$

$$\mathbf{T.E = O(\Delta t^2 + c\Delta x^2 + c\Delta y^2)}$$

همانطور که ملاحظه می کنید عبارت غالب در خطای برشی مشتق فرد u_{xxx} , u_{yyy} , u_{yyy} می باشد. بنا براین جواب عمدتا" دارای خطای فاز خواهد بود. این روش از دقت مرتبه دو است. در چنین حالتی جمله های مشتق زوج در معادله بهسازی شده وجود ندارد لذا خطای استهلاک

وجود ندارد. بنابراین روش لیپ فراگ به طور طبیعی پایدار است. و خطاهایی که که در اثر غیر دقیق بودن شرایط مرزی یا خطای گرد کردن توسط کامپیوتر انجام می شود، از بین نمی رود.

معادله بهسازی معادله موج به روش لیپ فراگ:

$$U_t + cU_x = \frac{c(\Delta x^2)}{6}(\gamma^2 - 1)U_{xxx} - \frac{c(\Delta x^4)}{120}(9\gamma^4 - 10\gamma^2 + 1)U_{xxxxx} + \cdots.$$

مزایا روش لیپ فراگ :

عبارت غالب در خطای برشی شامل مشتق فرد U_{xxx} است .بنابراین جواب عمدتا دارای خطای فاز است و از دقت مرتبه دوم است در چنین حالتی جملات مشتق زوج در معادله بهسازی شده وجود ندارد پس خطای استهلاک ندارد. ولی باتوجه به اینکه این روش از دقت مرتبه دوم است و خطای استهلاک ندارد .

معایب روش لیپ فراگ:

- ✓ شرایط اولیه در دو گام زمانیn,n-1 باید معلوم باشد .
- ✓ افزایش حجم ذخیره سازی کامپیوتر در صورت عدم دقت در برنامه نویسی

نکته مهم:

از آنجایی که محور افقی صفحه دو بعدی را با j=1 شروع و شمارنده محور افقی با j است ومحور قائم را با i=1 شروع میشودو شمارنده محور قائم i نشان میدهیم . بنابراین در صفحه دو بعدی محورافقی ار j=1 شروع میشود و تا j=jx ادامه پیدا میکند و محور قائم از i=1 وتا i=jy ادامه پیدا میکند.پس معادله موج دو بعدی مرتبه اول گسترش یافته به روش لیپ فراگ به صورت زیر تغییر میکند:

$$\frac{U_{i,j}^{n+1}-U_{i,j}^{n-1}}{2\Delta t}=-c_x\left(\frac{U_{i,j+1}^{n}-U_{i,j-1}^{n}}{2\Delta x}\right)-c_y\left(\frac{U_{i+1,j}^{n}-U_{i-1,j}^{n}}{2\Delta y}\right)$$

شرح پروژه:

پروژه یک صفحه دو بعدی

شروع محور افقی با : $X1$

شروع زمان : $t1$

پایان محور افقی با :$X2$

پایان زمان : $t2$

شروع محور قائم با : $y1$

گام مکانی در جهت افقی : D_X

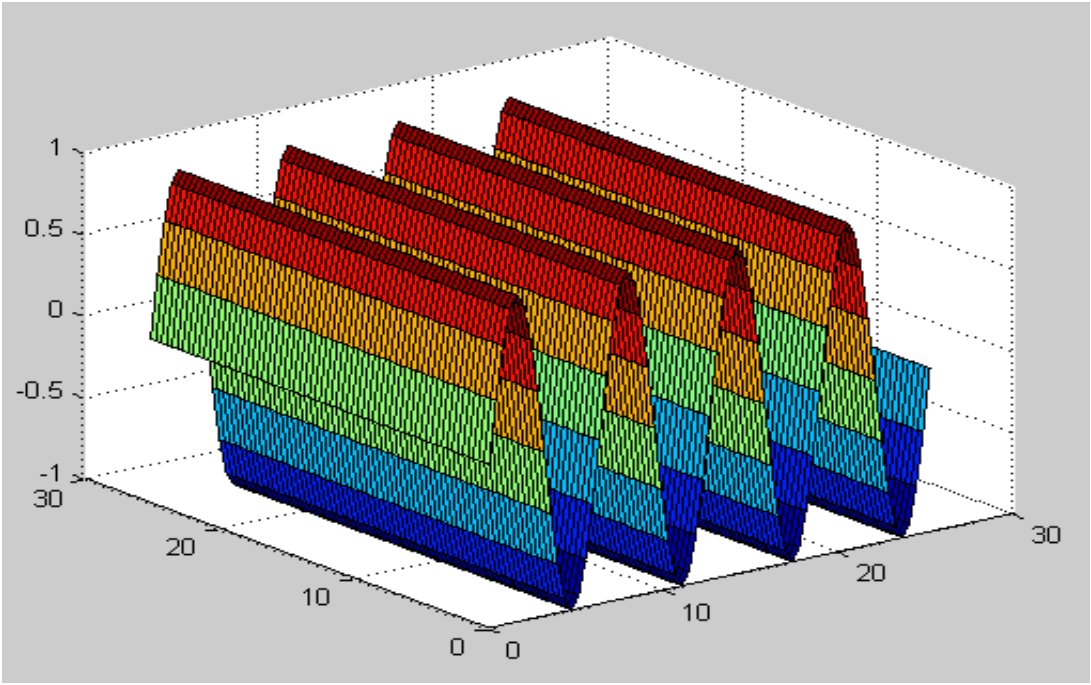
پایان محور قائم با :$y2$

گام مکانی در جهت قائم : D_Y

شرایط اولیه در زمان $n=1$

ما در این پروژه نیاز به یک شرایط اولیه داریم شرایط اولیه را در زمان اول به صورت زیر است

با فرض اینکه یک موج سینوسی در زمان $n=1$ درجهت محور افقی وارد شده است.



شرایط اولیه در زمان $n=2$ استفاده از روش تفاضل بالا دست

همانطور که قبلا گفته شد یکی از معایب رو ش لیپ فراگ این است نیاز به دو گام زمانی دارد و چون هم برای مشتق های مکانی و هم برای مشتق های زمانی از تفاضل مرکزی استفاده میکند.پس برای رفع این مشکل برای زمان دوم از روش تفاضل بالا دست مرتبه اول استفاده می شود.

معرفی روش تفاضل بالا دست مرتبه اول برای گام زمانی $n=2$

معادله روش تفاضل بالا دست مرتبه اول به صورت زیر است که برای مشتق های مکانی از تفاظل مرکزی و برای مشتق های زمانی تفاضل پیشرو استفاده شده است.

$$\frac{U_{i,j}^{n+1}-U_{i,j}^{n}}{\Delta t}=-C_x\left(\frac{U_{i,j}^{n}-U_{i,j-1}^{n}}{\Delta x}\right)-C_Y\left(\frac{U_{i,j}^{N}-U_{i-1,j}^{n}}{\Delta y}\right)$$

اعمال شرایط مرزی:

در این پروژه برای صفحه دوبعدی شرایط مرزی دیریشله و شرایط مرزی نیومان نیز اعمال شده است.

الف– شرایط مرزی دیریشله (*Dirichlet Boundary condition*):

- ✓ نقاط واقع درمحور قائم درسمت چپ: شرایط مرزی این نقاط تحت عنوان *Uv1* نمایش داده شده است .
- ✓ نقاط واقع درمحور افقی در پایین صفحه: شرایط مرزی این نقاط تحت عنوان *Uh1* نمایش شده است.
- ✓ نقاط واقع درمحور افقی دربالای صفحه: شرایط مرزی این نقاط تحت عنوان *Uh2* نمایش داده شده است.

ب– شرایط مرزی نیومان (*neumann Boundary condition*):

نقاط واقع در محور قائم سمت راست:شرایط مرزی نیومان این نقاط تحت عنوان *Uv2* نمایش داده شده است.همانطور که میدانیم شرایط مرزی نیومان مشتق عمودی متغیر وابسته در مرزها می باشد.

$$U_{i,j}^{n+1} = U_{i,j}^{n-1} - c_x D_t(Unewm) - C_Y(U_{i+1,j}^{n} - U_{i-1,j}^{n})$$

کد نویسی با متلب(*Matlab*):

```
%soheil mahmoodi 9414135016%

% solve metod lipfrog wave 2D%
clc
clear all
close all
x1=input('start x1=');
x2=input('end x2=');
y1=input('start y1=');
y2=input('end y2=');
t1=input('start time=');
t2=input('end time=');
Dx=input('step makani x=');
Dy=input('step makani y=');
Dt=input('step time=');
M=input('mesh dar che time mored nazar ast=');
%####################################%


%input boundary counditions%
Uv1=input('dirichlet boundary counditions Uv1=');
Uh1=input('dirichlet boundary counditions Uh1=');
```

```
Uh2=input('dirichlet boundary counditions Uh2=');
Uv2newm=input('newman boundary counditions Uv2newm=');

%wave velocity%
cx=1; cy=1;

%calculate number currant x and number currant y%

Cx=cx*Dt/Dx;
Cy=cy*Dt/Dy;
disp(sprintf('number currant x: %0.2f',Cx));
disp(sprintf('number currant y: %0.2f',Cy));

%calculate vectors x,y,t and number step x,y,t and calculate
matris U%

X=[x1:Dx:x2];
Y=[y1:Dy:y2];
t=t1:Dt:t2;
N=fix((t2-t1)/Dt)+1;
jx=fix((x2-x1)/Dx)+1;
jy=fix((y2-y1)/Dy)+1;
U=zeros(jx,jy,N);
%##################################%


%initial condition for lipfrag%
%step time n=1%
for i=1:jy; j=1:jx;
    U(i,j,1)=sin(X(j));
end
%step time n=2%
    for j=1; i=1:jy;
        U(i,j,2)=Uv1;
    end
    for j=2:jx; i=1;
        U(i,j,2)=Uh1;
    end
    for j=jx; i=2:jy-1;
        U(i,j,2)=U(i,j,1)-(cx)*Dt*(Uv2newm)-(Cy)*(U(i,j,1)-U(i-
1,j,1));
    end
    for  j=2:jx; i=jy;
        U(i,j,2)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
```

```
        U(i,j,2)=U(i,j,1)-(Cx)*(U(i,j,1)-U(i,j-1,1))-
(Cy)*(U(i,j,1)-U(i-1,j,1));
    end
%###########################%


%load boundary counditions%
for n=2:N-1;
    for j=1; i=1:jy;
        U(i,j,n+1)=Uv1;
    end
    for j=2:jx; i=1;
        U(i,j,n+1)=Uh1;
    end
    for j=jx; i=2:jy-1;
        U(i,j,n+1)=U(i,j,n-1)-(cx)*Dt*(Uv2newm)-
(Cy)*(U(i+1,j,n)-U(i-1,j,n));
    end
    for  j=2:jx; i=jy;
        U(i,j,n+1)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
        U(i,j,n+1)=U(i,j,n-1)-(Cx)*(U(i,j+1,n)-U(i,j-1,n))-
(Cy)*(U(i+1,j,n)-U(i-1,j,n));
    end
end
%##################################%


%%graph 3D solution for Leapfrog method%
figure(1)
meshc(X,Y,U(:,:,M));
xlabel('x(m)')
ylabel('y(m)')
zlabel('U(m)')
axis([x1 x2 y1 y2 -5 5]);
drawnow
title('leapfrog mesh 2D');
grid
hold on
figure(2)
surfc(X,Y,U(:,:,M))
xlabel('x(m)')
ylabel('y(m)')
zlabel('U(m)')
axis([x1 x2 y1 y2 -5 5]);
grid
```

```
%SHARAYET PAYDARI%

G1=complex(-1*(power((power((Cx*sin(X)+Cy*sin(Y))+1,2)),0.5)),-
1*(Cx*sin(X)+Cy*sin(Y)));
G2=complex(1*(power((power((Cx*sin(X)+Cy*sin(Y))+1,2)),0.5)),-
1*(Cx*sin(X)+Cy*sin(Y)));
figure(3)
plot(X,G1,'b.-',X,G2,'r.-')
xlabel('(x,Y)(m)')
ylabel('G(ZARIB BOZORG NAMAI)')
title('diagram decarti paydari')
legend('G1','G2')
grid
figure(4)
polar(X,G2)
title('diagram ghotbi paydari');
figure(5)
plot3(X,Y,G1,'b',X,Y,G2,'r')
xlabel('X(m)')
ylabel('Y(m)')
zlabel('G(m)')
axis([x1 x2 y1 y2 -2 2])
title('diagram 3D paydari')
legend('G1','G2')
grid
%##############################################%
```

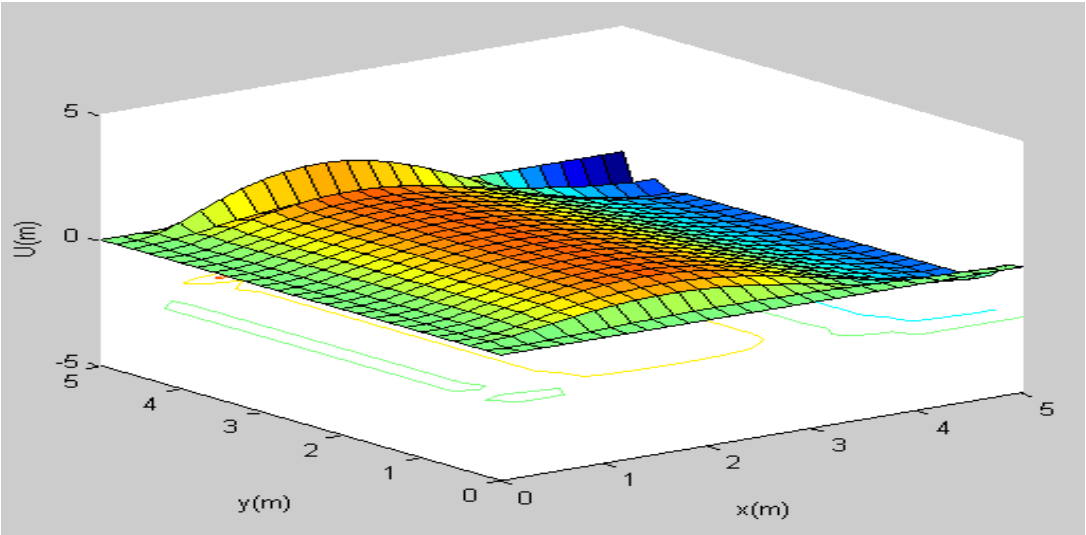
عنوان مساله دو بعدی:

این پروژه را برای مثال زیر حل میکنیم:

شروع محور افقی	$X1=0$
پایان محور افقی	$X2=5$
شروع محمور قائم	$Y1=0$
پایان محور قائم	$Y2=5$
شروع زمان	$T1=0$
پایان زمان	$T2=5$
گام مکانی در جهت افقی	$Dx=0.2$
گام مکانی در جهت قائم	$Dy=0.2$
گام های زمانی	$Dt=0.1$

نتایج خروجی از متلب:

برای مثال ما قصد داریم نتایج را در زمان $n=6$ بدست اوریم که نتایج به صورت زیر است.



بررسی پایداری به روش ون نیومن :

معادله موج دوبعدی مرتبه اول به روش لیپ فراگ به صورت زیر است:

$$\frac{U_{i,j}^{n+1}-U_{i,j}^{n-1}}{2\Delta t}=-c_x\left(\frac{U_{i,j+1}^{n}-U_{i,j-1}^{n}}{2\Delta x}\right)-c_y(\frac{U_{i+1,j}^{n}-U_{i-1,j}^{n}}{2\Delta y})$$

حالا طرفین را در $2\Delta t$ ضرب میکنیم سپس $C_x=\frac{\Delta t\times c_x}{\Delta x}$ عدد کورانت در جهت X و $C_y=$ $\frac{\Delta t\times c_y}{\Delta x}$عدد کورانت در جهت Y میباشد را جایگذاری می میکنیم که معادله به صورت زیر میشود:

$$U_{i,j}^{n+1}-U_{i,j}^{n-1}=-C_x\left(U_{i,j+1}^{n}-U_{i,j-1}^{n}\right)-C_y(U_{i+1,j}^{n}-U_{i-1,j}^{n})$$

حالا برای تحلیل پایداری به روش نیومن مولفه از سری فوریه را با U_i^n به صورت زیر را در نظر بگیرید

$$U_i^n=U^n e^{I\theta i}$$

که در معادله فوق $I=\sqrt{-1}$, θ زاویه فاز موج ودر اخر Gضریب بزرگ نمایی را به صورت زیر اعمال میکنیم.

$$\frac{U^{n+1}}{U^n}=G$$

$$\frac{U^{n-1}}{U^n}=\frac{1}{G}$$

که در آخر معادله به شکل زیر میشود:

$$U^{n+1}e^{I\theta j}e^{I\beta i}-U^{n-1}e^{I\theta j}e^{I\beta i}=-C_x(U^n e^{I\beta i}e^{I\theta(j+1)}-U^n e^{I\beta i}e^{I\theta(j-1)})-C_y(U^n e^{I\beta(i+1)}e^{I\theta(j)}-U^n e^{I\beta(i-1)}e^{I\theta(j)})$$

با ساده سازی و اعمال G داریم:

$$\mathrm{G}-\frac{1}{G}=-C_x\left(e^{I\theta}-e^{-I\theta}\right)-C_y(e^{I\beta}-e^{-I\beta})$$

$$\frac{e^{I\theta}-e^{-I\theta}}{2I}=\sin(\theta) \qquad \frac{e^{I\beta}-e^{-I\beta}}{2I}=\sin(\beta)$$

$$G-\frac{1}{G}=-C_x\big(2Isin(\theta)\big)-C_y\big(2Isin(\beta)\big)$$
$$=-C_x\big(2Isin(\theta)\big)-C_y(2Isin(\beta))$$

$$G-\frac{1}{G}=\lambda \qquad\qquad G^2-\lambda G-1=0$$

$$G_1=\frac{\lambda+\sqrt{\lambda^2+4}}{2} \qquad\qquad G_2=\frac{\lambda-\sqrt{\lambda^2+4}}{2}$$

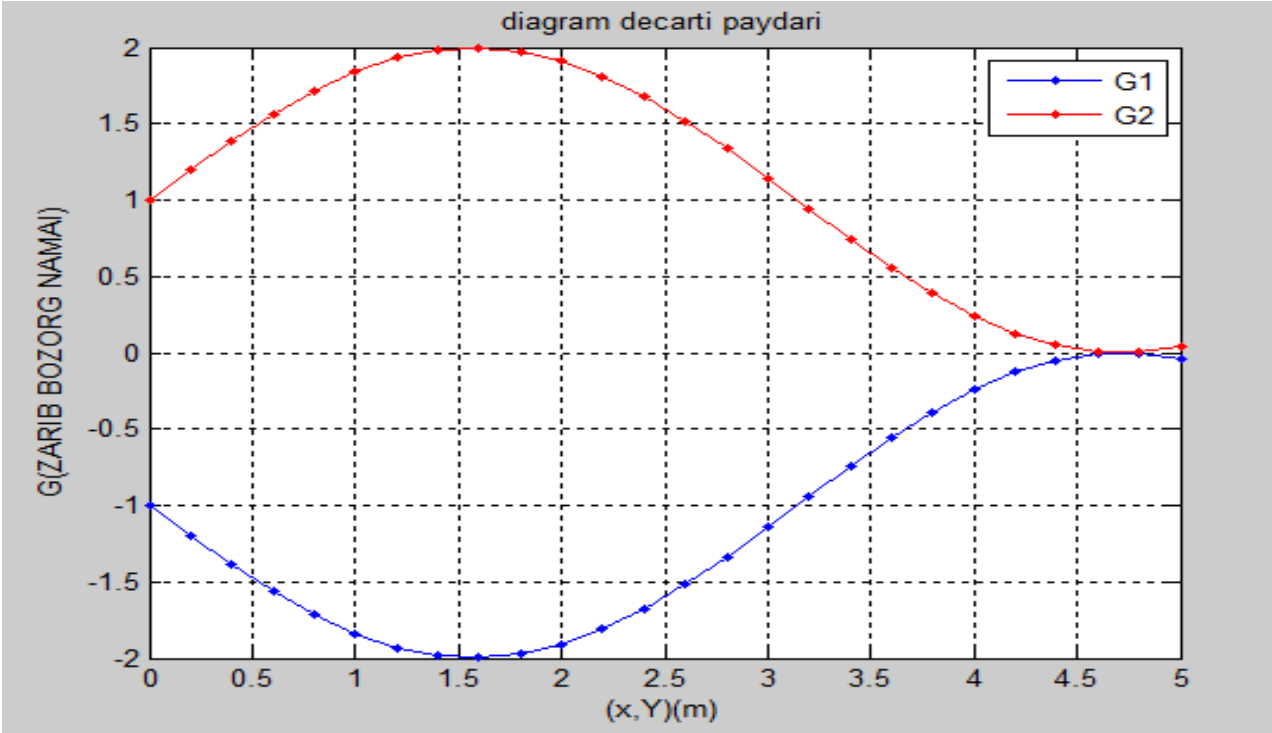
شرط پایداری $|G|^2\leq 1$

$$G_1=-I\big(C_x\sin(\theta)+C_y\sin(\beta)\big)+\sqrt{(C_x\sin(\theta)+C_y\sin(\beta))^2+1}$$

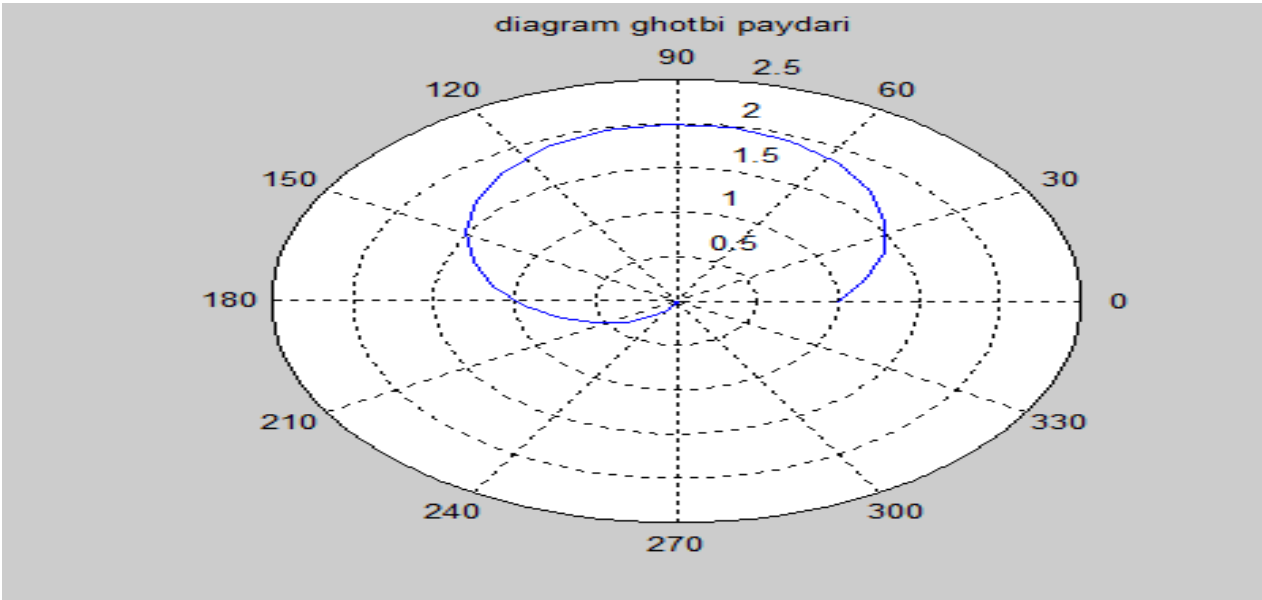
$$G_2=-I\big(C_x\sin(\theta)+C_y\sin(\beta)\big)-\sqrt{(C_x\sin(\theta)+C_y\sin(\beta))^2+1}$$

حالا نمودارها را دستگاه دکارتی و قطبی رسم میکنیم.

نتایج خروجی از متلب برای پایداری در دستگاه دکارتی:



نتایج خروجی از متلب برای پایداری دستگاه قطبی:



صحت سنجی به روش اول (دستی):

برای صحت سنجی یک مثال را مورد انالیز قرا می دهیم و جواب را با برنامه ***matlab*** مقایسه میکنیم.

اطلاعات این مثال به صورت زیر است:

$$X1=1 \quad X2=3 \quad Y1=1 \quad Y2=3$$

$$t1=1 \quad t2=3 \quad D_x=1 \quad D_y=1 \quad D_t=1 \quad C_x=1 \quad C_y=1$$

صحت سنجی برای زمان *n=1* :

همانطور که گفته شد در زمان *n=1* یک موج سینوسی در جهت افقی بر صفحه اعمال شده است.یعنی در زمان *n=1* با اطلاعات بالا یک ماتریس 3*3 داریم که هرمقدار موج به صورت زیر بدست می اید.

```
%initial condition for leapfrog%
%step time n=1%
for i=1:jy; j=1:jx;
    U(i,j,1)=sin((j));
end
```

```
X =

     1     2     3
```

```
Y =

     1     2     3
```

$U(1,1,1) = \sin(1) = 0.84147$ $\quad U(1,2,1) = \sin(2) = 0.9093$ $\quad U(1,3,1) = \sin(3) = 0.1411$

$U(2,1,1) = \sin(1) = 0.84147$ $\quad U(2,2,1) = \sin(2) = 0.9093$ $\quad U(2,3,1) = \sin(3) = 0.1411$

$U(3,1,1) = \sin(1) = 0.84147$ $\quad U(3,2,1) = \sin(2) = 0.9093$ $\quad U(3,3,1) = \sin(3) = 0.1411$

نتایج برای زمان $n=1$ با خروجی *matlab* :

نتایج بعد از انالیز به صورت زیر است.حالا این نتایج را با نتایج متلب برای زمان $n=1$ مقایسه میکنیم که متوجه می شویم نتایج کاملا یکسان است.

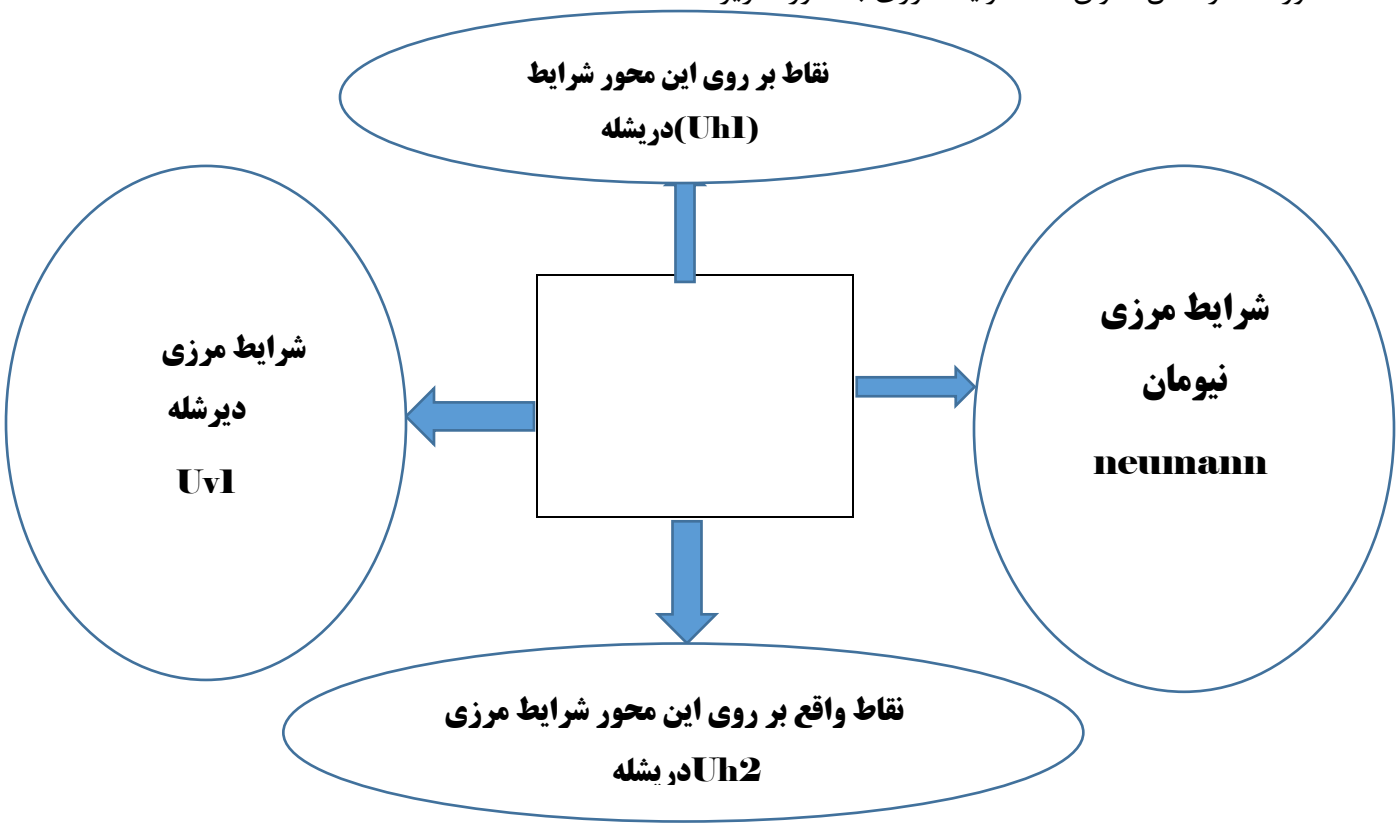
```
U(:,:,1) =

    0.8415    0.9093    0.1411
    0.8415    0.9093    0.1411
    0.8415    0.9093    0.1411
```

صحت سنجی برای زمان $n=2$:

همانطور که قبلا گفته شد برای زمان $n=2$ از روش تفاضل بالا دست مرتبه اول استفاده شده است که نتایج ان در انالیز به صورت زیر است:

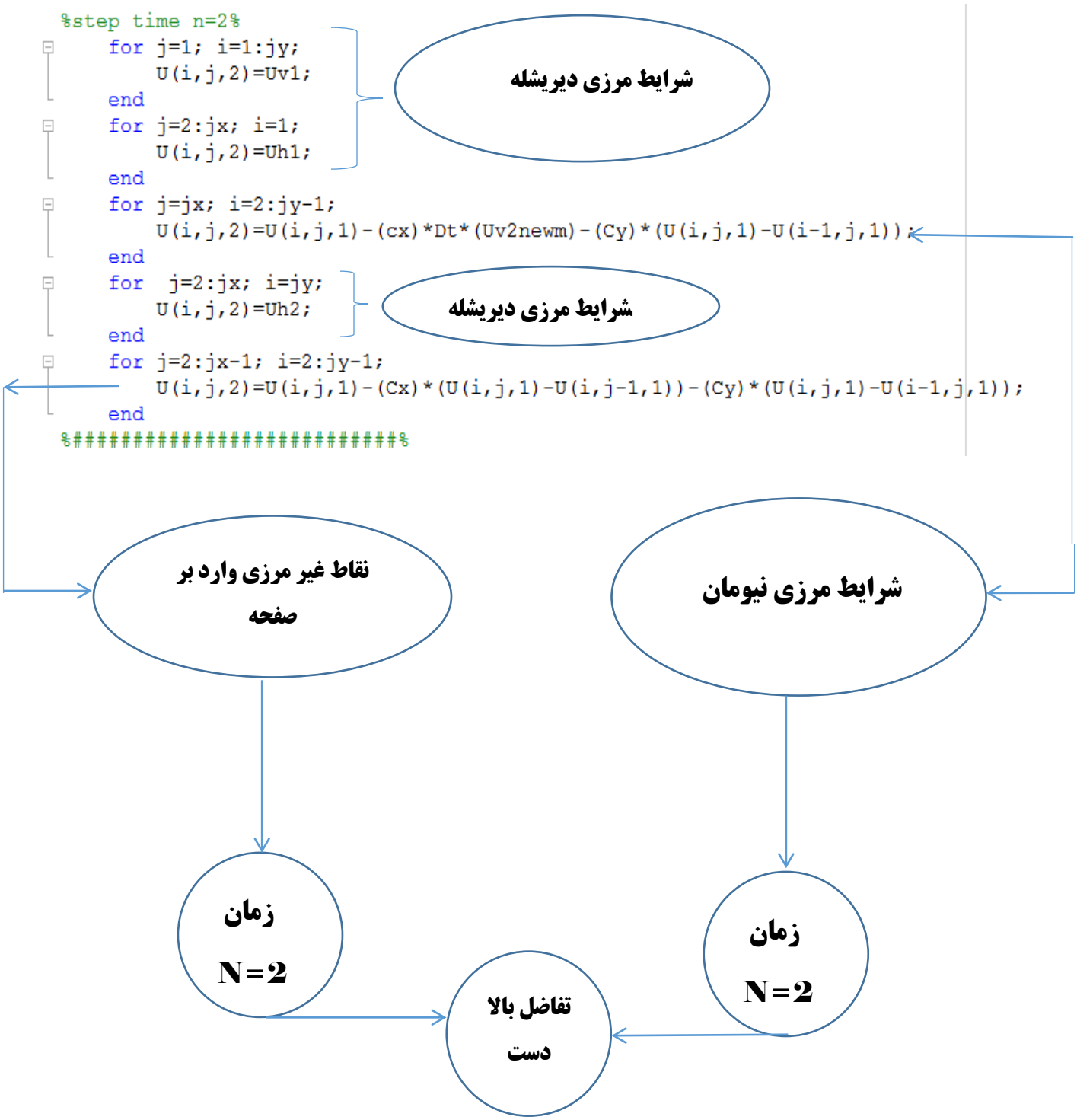
همانطور که در مثال عنوان شد شرایط مرزی به صورت زیر است:



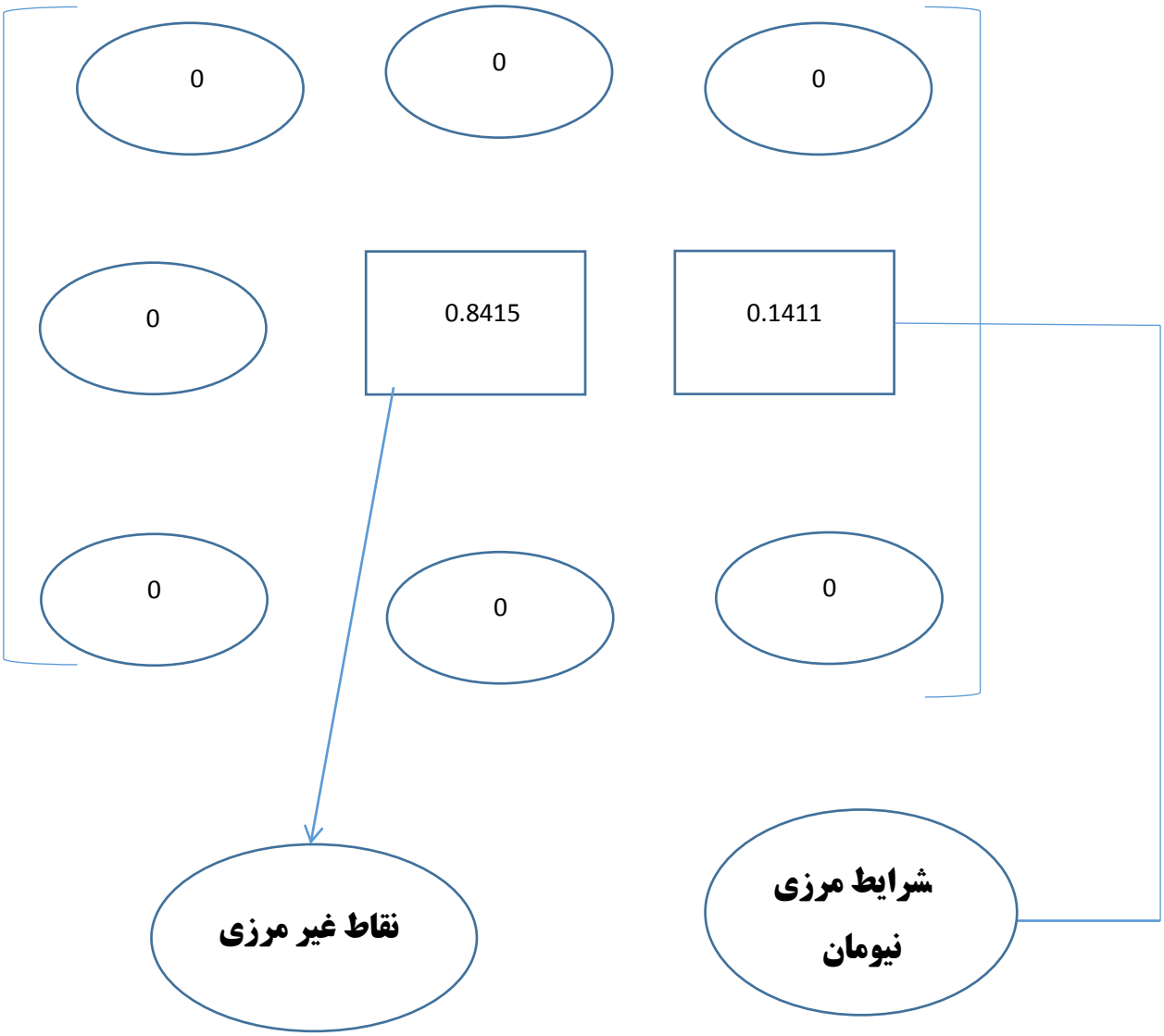
چون ماتریس **3*3** دارم برای این مثال به صورت زیر عمل میکنیم:

برای نقاط بر روی شرایط نیومان با استفاده از فرمول تفاضل بالادست مرتبه اول به صورت زیر عمل میکنیم.

```
%step time n=2%
    for j=1; i=1:jy;
        U(i,j,2)=Uv1;
    end
    for j=2:jx; i=1;
        U(i,j,2)=Uh1;
    end
    for j=jx; i=2:jy-1;
        U(i,j,2)=U(i,j,1)-(cx)*Dt*(Uv2newm)-(Cy)*(U(i,j,1)-U(i-1,j,1));
    end
    for  j=2:jx; i=jy;
        U(i,j,2)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
        U(i,j,2)=U(i,j,1)-(Cx)*(U(i,j,1)-U(i,j-1,1))-(Cy)*(U(i,j,1)-U(i-1,j,1));
    end
%###########################%
```



چون طبق مثال یک ماتریس **3*3** در زمان $n=2$ پس در این زمان محاسبات مانند شکل زیر است.



شرایط مرزی نیومان به روش تفاضل بالا دست:

$$U_{i,j}^{2} = U_{i,j}^{1} - c_x * D_t * (Unewman) - C_y(U_{i,j}^{1} - U_{i-1,j}^{1})$$

$$U_{2,3}^{2} = U_{2,3}^{1} - 0 - 1 * (U_{2,3}^{1} - U_{1,3}^{1})$$

$$U_{2,3}^{2} = 0.1411 - 1 * 0 - 1(0.1411 - 0.1411) = 0.1411$$

نقاط میانی به روش تفاضل بالادست:

$$U_{i,j}^{2} = U_{i,j}^{1} - C_X\left(U_{i,j}^{1} - U_{i,j-1}^{1}\right) - C_y(U_{i,j}^{1} - U_{i-1,j}^{1})$$

$$U_{2,2}^{2} = 0.9093 - 1 * (0.9093 - 0.9093) - 1 * (0.9093 - 0.8415) = 0.8415$$

نتایج متلب برای n=2 :

```
U(:,:,2) =

     0         0         0
     0    0.8415    0.1411
     0         0         0
```

صحت سنجی برای زمان n=3:

برای این زمان از فرمول های روش لیپ فراگ استفاده میشود که محاسبات آن به صورت زیر است:

```
%load boundary counditions%
for n=2:N-1;
    for j=1; i=1:jy;
        U(i,j,n+1)=Uv1;
    end
    for j=2:jx; i=1;
        U(i,j,n+1)=Uh1;
    end
    for j=jx; i=2:jy-1;
        U(i,j,n+1)=U(i,j,n-1)-(cx)*Dt*(Uv2newm)-(Cy)*(U(i+1,j,n)-U(i-1,j,n));
    end
    for  j=2:jx; i=jy;
        U(i,j,n+1)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
        U(i,j,n+1)=U(i,j,n-1)-(Cx)*(U(i,j+1,n)-U(i,j-1,n))-(Cy)*(U(i+1,j,n)-U(i-1,j,n));
    end
end
%################################%
```

$$U_{2,2}^{3} = U_{2,2}^{1} - 1 * \left(U_{2,3}^{2} - U_{2,1}^{2}\right) - 1 * \left(U_{3,2}^{2} - U_{1,2}^{2}\right)$$

$$U_{2,2}^{3} = 0.9093 - 0.1411 = 0.7682$$

شرایط مرزی نیومان به روش لیپ فراگ:

$$U_{2,3}^{3} = U_{2,3}^{1} - 1 * 0 - 1 * \left(U_{3,3}^{2} - U_{1,3}^{2}\right)$$

$$U_{2,3}^{3} = 0.1411 - 0 = 0.1411$$

نتایج خروجی از متلب برای زمان n=3 :

```
U(:,:,3) =

         0         0         0
         0    0.7682    0.1411
         0         0         0
```

همانطور که دیده شد نتایج یکسان است پس صحت سنجی درست است.

صحت سنجی به روش دوم(مقایسه روش لیپ فراگ با روش FTCS):

در این روش ابتدا کد روش **FTCS** را می نویسیم سپس در یک زمان دلخواه نتایج را با هم مقایسه میکنیم.

کد متلب به روش FTCS :

```
%soheil mahmoodi 9414135016%
% solve metod lipfrog wave 2D%
clc
clear all
close all
x1=input('start x1=');
x2=input('end x2=');
y1=input('start y1=');
y2=input('end y2=');
t1=input('start time=');
t2=input('end time=');
Dx=input('step makani x=');
Dy=input('step makani y=');
Dt=input('step time=');
M=input('mesh dar che time mored nazar ast=');
%##################################%
```

```
%input boundary counditions%
Uv1=input('dirichlet boundary counditions Uv1=');
Uh1=input('dirichlet boundary counditions Uh1=');
Uh2=input('dirichlet boundary counditions Uh2=');
Uv2newman=input('dirichlet boundary counditions Uv2newman=');

%wave velocity%
cx=1; cy=1;

%calculate number currant x and number currant y%
Cx=cx*Dt/Dx;
Cy=cy*Dt/Dy;
disp(sprintf('number currant x: %0.2f',Cx));
disp(sprintf('number currant y: %0.2f',Cy));
%calculate vectors x,y,t and number step x,y,t and calculate
matris U%
X=[x1:Dx:x2];
Y=[y1:Dy:y2];
t=[t1:Dt:t2];
N=fix((t2-t1)/Dt)+1;
jx=fix((x2-x1)/Dx)+1;
jy=fix((y2-y1)/Dy)+1;
T=zeros(jx,jy,N);
U=zeros(jx,jy,N);
%##################################%

%initial condition for leapfrog%
%step time n=1%
for i=1:jy; j=1:jx;
    T(i,j,1)=sin((j));
end
%step time n=2 metod ftcs%
    for j=1; i=1:jy;
        T(i,j,2)=Uv1;
    end
    for j=2:jx; i=1;
        T(i,j,2)=Uh1;
    end
    for j=jx; i=2:jy-1;
        T(i,j,2)=T(i,j,1)-(cx)*Dt*(0.5)*(Uv2newman)-
(Cy)*(0.5)*(T(i+1,j,1)-T(i-1,j,1));
    end
    for  j=2:jx; i=jy;
        T(i,j,2)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
```

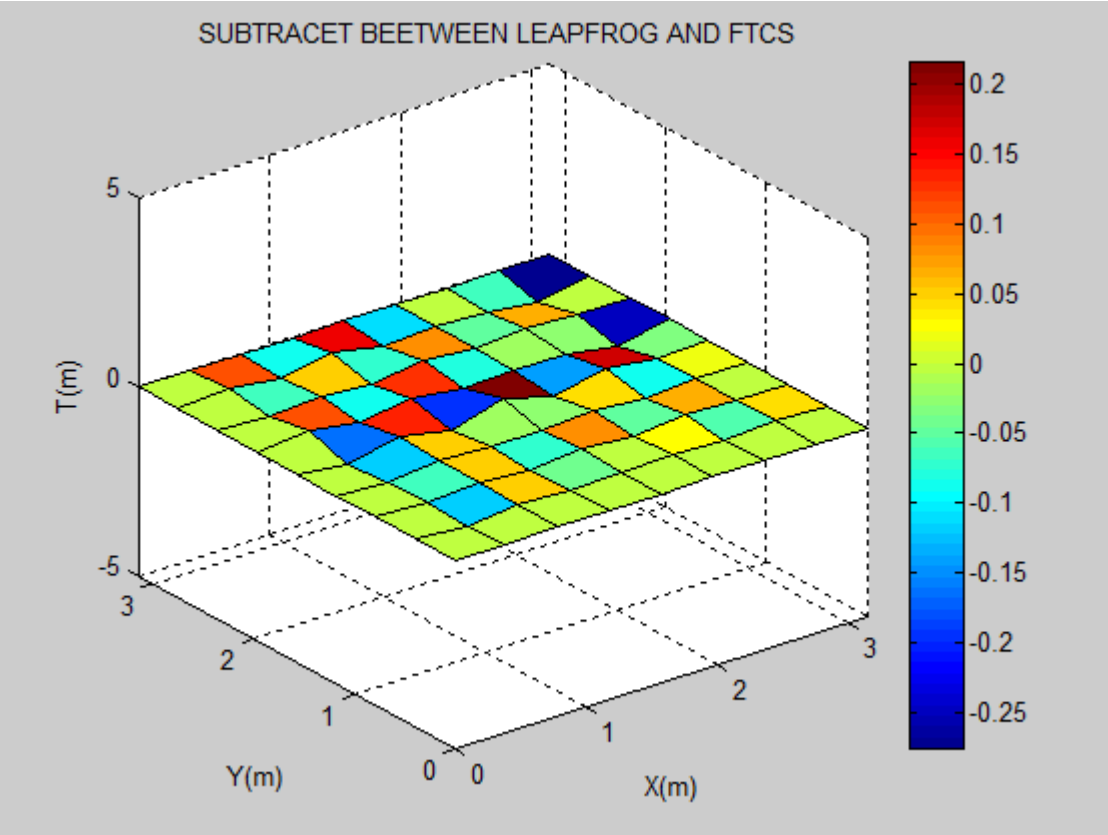
```
        T(i,j,2)=T(i,j,1)-(Cx)*(0.5)*(T(i,j+1,1)-T(i,j-1,1))-
(Cy)*(0.5)*(T(i+1,j,1)-T(i-1,j,1));
    end
%###########################%
%load boundary counditions metod ftcs%
for n=2:N-1;
    for j=1; i=1:jy;
        T(i,j,n+1)=Uv1;
    end
    for j=2:jx; i=1;
        T(i,j,n+1)=Uh1;
    end
    for j=jx; i=2:jy-1;
        T(i,j,n+1)=T(i,j,n-1)-(cx)*Dt*(0.5)*(Uv2newman)-
(Cy)*(0.5)*(T(i+1,j,n)-T(i-1,j,n));
    end
    for  j=2:jx; i=jy;
        T(i,j,n+1)=Uh2;
    end
    for j=2:jx-1; i=2:jy-1;
        T(i,j,n+1)=T(i,j,n-1)-(Cx)*(0.5)*(T(i,j+1,n)-T(i,j-
1,n))-(Cy)*(0.5)*(T(i+1,j,n)-T(i-1,j,n));
    end
end
%##################################%

%%graph 3D solution for Leapfrog method%
for n=2:N-1
figure(1)
meshc(X,Y,T(:,:,n+1));
xlabel('x(m)')
ylabel('y(m)')
zlabel('T(m)')
axis([x1 x2 y1 y2 -5 5]);
drawnow
title('leapfrog mesh 2D');
grid
hold on
figure(2)
surfc(X,Y,T(:,:,n+1))
xlabel('x(m)')
ylabel('y(m)')
zlabel('T(m)')
axis([x1 x2 y1 y2 -5 5]);
grid
end
```

حالا نتایج روش لیپ فراگ را با روش **FTCS** مقایسه میکنیم در یک زمان دلخواه نتایج به صورت زیر است:

مقایسه به صورت گرافیکی:

اختلاف دو روش در زمان **23** به صورت گرافیکی نشان داده شده است که شکل گرافیکی آن به صورت زیر است.



همانطو که دیده می شود اختلاف ان ناچیز است

مقایسه دو روش به صورت عددی:

در زمان n=26 برای روش لیپ فراگ:

0	0	0	0	0	0
0	-0.0398	-0.0001	0.0458	-0.1750	0.2125
0	-0.0731	0.0894	-0.1806	0.3862	-0.4173
0	-0.0114	-0.0434	0.1532	-0.4155	0.4830
0	-0.1679	0.2090	-0.4230	0.8672	-0.9227
0	0.2000	-0.2689	0.5343	-1.0898	1.1923
0	-0.0731	0.0894	-0.1806	0.3862	-0.4173
0	0.2284	-0.3122	0.6417	-1.3304	1.4628
0	0	0	0	0	0

Columns 7 through 9

0	0	0
-0.2473	0.3627	0.0000
0.4568	-0.6445	-0.0000
-0.5442	0.7850	0.0000
1.0175	-1.3742	-0.0000
-1.2723	1.7886	0.0000
0.4568	-0.6445	-0.0000
-1.5692	2.2108	0.0000
0	0	0

در زمان n=26 برای روش ftcs:

Columns 1 through 6

0	0	0	0	0	0
0	0.0798	-0.0512	0.0861	-0.1725	0.1852
0	-0.0118	0.0425	-0.1081	0.3041	-0.3690
0	0.1066	-0.0930	0.1682	-0.3876	0.4415
0	-0.0036	0.0768	-0.2285	0.6515	-0.7831
0	0.0894	-0.1811	0.4036	-1.0071	1.2113
0	-0.0118	0.0425	-0.1081	0.3041	-0.3690
0	0.1162	-0.2229	0.4857	-1.2222	1.4675
0	0	0	0	0	0

Columns 7 through 9

0	0	0
-0.1928	0.3191	0.0000
0.3875	-0.6417	-0.0000
-0.4553	0.7665	0.0000
0.8446	-1.3427	-0.0000
-1.2431	2.0392	0.0000
0.3875	-0.6417	-0.0000
-1.5056	2.4866	0.0000
0	0	0