حل معادلات مشتقات جزئی
با روش FDM
همراه با Matlab

علی شهیدی صادقی
دانشگاه صنعتی امیرکبیر

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

موارد مورد بررسی

- مقدمه
- مروری اجمالی بر حل معادله هلم هلتز به روش تفاضل متناهی
- حل معادله هلم هولتز با matlab
- مروری اجمالی بر حل معادله حرارت یک بعدی به روش تفاضل متناهی
- حل معادله حرارت یک بعدی با matlab
- حل معادله حرارت دو بعدی به روش تفاضل متناهی
- حل معادله حرارت دو بعدی با matlab
- مروری اجمالی بر حل معادله موج یک بعدی به روش تفاضل متناهی
- حل معادله موج یک بعدی در matlab
- حل معادله موج دو بعدی به روش تفاضل متناهی
- حل معادله موج دو بعدی با matlab

مقدمه

- یک معادله درجه ۲ PDE با دو متغیر مستقل x و y در نظر بگیرید.

$$A(x,y)\frac{\partial^2 u}{\partial x^2}+B(x,y)\frac{\partial^2 u}{\partial x\partial y}+C(x,y)\frac{\partial^2 u}{\partial y^2}=f\left(x,y,u,\frac{\partial u}{\partial x},\frac{\partial u}{\partial y}\right)$$

$x_0 \le x \le x_f$ و $y_0 \le y \le y_f$

- و شرایط مرزی :

$$u(x,y_0)=b_{y0}(x)\quad,\qquad u(x,y_f)=b_{yf}(x).$$

$$u(x_0,y)=b_{x0}(y)\quad,\qquad u(x_f,y)=b_{xf}(y).$$

مقدمه

- این معادلات رو می توانیم به سه دسته زیر تقسیم کنیم.
- معادلات بیضوی: $B^2 - 4AC < 0$
- معادلات سهموی: $B^2 - 4AC = 0$
- معادلات هذلولی: $B^2 - 4AC > 0$

معادلات بیضوی

- به صورت خاص به بررسی معادله هلم هولتز ناهمگن می پردازیم.

$$\nabla^2 u(x,y) + g(x,y)u(x,y) = f(x,y)$$

$$\frac{\partial^2 u(x,y)}{\partial x^2} + \frac{\partial^2 u(x,y)}{\partial y^2} + g(x,y)u(x,y) = f(x,y)$$

$x_0 \le x \le x_f$ و $y_0 \le y \le y_f$

- شرایط مرزی

$$u(x,y_0) = b_{y0}(x) \quad , \qquad u(x,y_f) = b_{yf}(x).$$

$$u(x_0,y) = b_{x0}(y) \quad , \qquad u(x_f,y) = b_{xf}(y).$$

مروری اجمالی به حل معادله هلم هولتز

- دقت داشته باشید :

$if \quad g(x,y) = 0$ ⟶ معادله پواسن

$if \quad g(x,y) = 0 \quad , \quad f(x,y) = 0$ ⟶ معادله لاپلاس

- برای بدست آوردن فاصله بین هر دو نقطه ما دامنه های x و y را به ترتیب بر M_x و M_y تقسیم می کنیم.

$$\Delta x = \frac{x_f - x_0}{M_x} \quad , \quad \Delta y = \frac{y_f - y_0}{M_y}$$

مروری اجمالی به حل معادله هلم هولتز

- به این ترتیب برای هر نقطه دلخواه خواهیم داشت:

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta x^2} + \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta y^2} + g_{i,j}u_{i,j} = f_{i,j}$$

- که در آن :

$$u_{i,j} = u(x_j, y_i) \quad f_{i,j} = f(x_j, y_i) \quad g_{i,j} = g(x_i, y_j)$$

- شرایط مرزی:

$$u_{0,j} = b_{y0}(x_j) \quad , \qquad u_{My,j} = b_{yf}(x_j).$$

$$u_{i,0} = b_{x0}(y_i) \quad , \qquad u_{i,Mx} = b_{xf}(y_i).$$

مروری اجمالی به حل معادله هلم هولتز

- با ساده کردن معادله داریم:

$$u_{i,j} = r_y\left[u_{i,j+1} + u_{i,j-1}\right] + r_x\left[u_{i+1,j} + u_{i-1,j}\right] + r_{xy}\left[g_{i,j}u_{i,j} - f_{i,j}\right]$$

- که در آن داریم:

$$r_y = \frac{\Delta y^2}{2(\Delta y^2 + \Delta x^2)} \qquad r_x = \frac{\Delta x^2}{2(\Delta y^2 + \Delta x^2)} \qquad r_{xy} = \frac{\Delta y^2 \Delta x^2}{2(\Delta y^2 + \Delta x^2)}$$

حل معادله هلم هولتز با کمک matlab

- پارامتر های ورودی
- مش بندی
- اعمال شرایط مرزی
- تعیین توابع f و g که تعیین کننده نوع معادله پواسن و لاپلاس و هلم هولتز است.
- تعیین مقادیر مش بندی

حل معادله هلم هولتز با کمک matlab

- برای توابع ورودی می توانیم از function و یا از دستور input استفاده کنیم.

```
function [u,x,y] = poisson(f,g,bx0,bxf,by0,byf,D,Mx,My,tol,MaxIter)
```

- دقت کنید که برای ذخیره کردن m-file خود نام آن را هم نام تابع خود قرار دهید.

```
%solve u_xx + u_yy + g(x,y)u = f(x,y)
%over the region D = [x0,xf,y0,yf] = {(x,y) |x0 <= x <= xf, y0 <= y <= yf}
% with the boundary Conditions:
% u(x0,y) = bx0(y), u(xf,y) = bxf(y)
% u(x,y0) = by0(x), u(x,yf) = byf(x)
%Mx=#ofsubintervals along x axis
%My=#ofsubintervals along y axis
% tol : error tolerance
% MaxIter: the maximum # of iterations
```

حل معادله هلم هولتز با کمک matlab

- برای مش بندی و طی مسیر حل، به محاسبات زیر نیاز است.

```
x0 = D(1); xf = D(2); y0 = D(3); yf = D(4);
dx = (xf - x0)/Mx; x = x0 + [0:Mx]*dx;
dy = (yf - y0)/My; y = y0 + [0:My]*dy;
Mx1=Mx+1;My1=My+1;
```

- همچنین برای محاسبات توابع f و g به صورت عددی از دو حلقه for به صورت زیر استفاده می کنیم.

```
for i = 1:My1
    for j = 1:Mx1
        F(i,j) = f(x(j),y(i)); G(i,j) = g(x(j),y(i));
    end
end
```

حل معادله هلم هولتز با کمک matlab

- برای گذاشتن شرایط مرزی در مش بندی خود از دو حلقه for استفاده می کنیم.

```
for m = 1:My1, u(m,[1 Mx1])=[bx0(y(m)) bxf(y(m))]; end
%left/right side
for n = 1:Mx1, u([1 My1],n) = [byf(x(n)); by0(x(n))]; end
%bottom/top
```

حل معادله هلم هولتز با کمک matlab

- برای ضرایب معادله (r_x و r_y و r_{xy}) به ترتیب زیر دستورات را وارد می کنیم.

```
dx2 = dx*dx; dy2 = dy*dy; dxy2 = 2*(dx2 + dy2);
rx = dx2/dxy2; ry = dy2/dxy2; rxy = rx*dy2;
```

- دستور محاسبه مش بندی مورد نظر به صورت زیر است.

```
for itr = 1:MaxIter
    for j = 2:Mx
        for i = 2:My
            u(i,j) = ry*(u(i,j + 1)+u(i,j - 1)) + rx*(u(i + 1,j)...
            +u(i - 1,j))+ rxy*(G(i,j)*u(i,j)- F(i,j));
        end
    end
    if itr>1&max(max(abs(u - u0))) < tol, break; end
    u0=u;
end
```

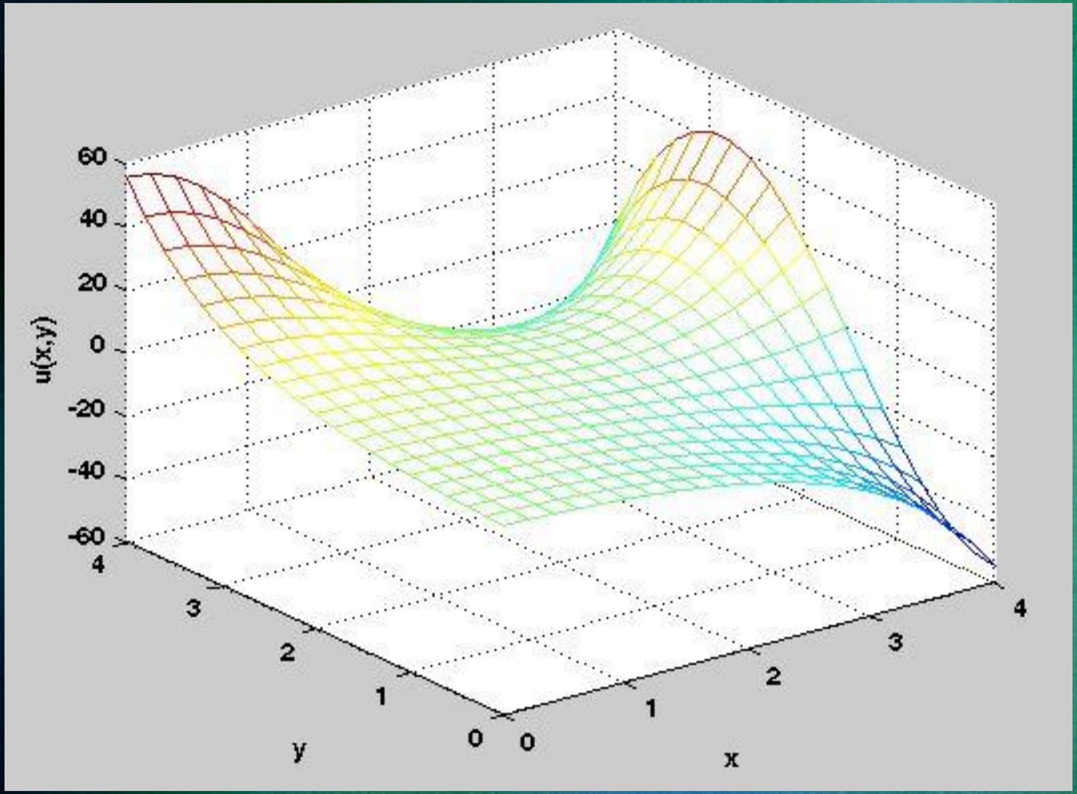
حل معادله هلم هولتز با کمک matlab

- حال به نوشتن برنامه ای که با استفاده از تابع نوشته شده در مراحل قبل، به حل مساله می پردازد؛ می پردازیم.

```
clear all;
close all;
clc;
f = inline('0','x','y'); g = inline('0','x','y');
Mx=20;My=20;
D=[0 4 0 4];
bx0 = inline('exp(y)-cos(y)','y');
bxf = inline('exp(y)*cos(4)-exp(4)*cos(y)','y');
by0 = inline('cos(x)-exp(x)','x');
byf = inline('exp(4)*cos(x)-exp(x)*cos(4)','x');
MaxIter = 5000;
tol = 1e-4;
[u,x,y] = poisson(f,g,bx0,bxf,by0,byf,D,Mx,My,tol,MaxIter);
clf,mesh(x,y,u);disp(u);
xlabel('x'),ylabel('y'),zlabel('u(x,y)');
```

حل معادله هلم هولتز با کمک matlab

- برای جواب نهایی داریم:



معادلات سهموی

- به عنوان معادلات سهموی به بررسی معادله حرارت یک بعدی می پردازیم.

$$A\frac{\partial^2 u(x,t)}{\partial x^2}=\frac{\partial u(x,t)}{\partial t} \qquad 0\le x\le x_f \qquad 0\le t\le T$$

- برای شرایط مرزی و مقدار اولیه داریم:

$$\begin{cases} u(0,t)=b_0(t) \\ u(x_f,t)=b_{xf}(t) \end{cases} \qquad \begin{cases} u(x,0)=i_0(t) \end{cases}$$

مروری اجمالی به حل معادله حرارت

- برای اعمال روش تفاضل متناهی ما فضای $[0,x_f]$ را به M قسمت تقسیم می کنیم.

$$\Delta x = \frac{x_f}{M}$$

- برای اعمال روش تفاضل متناهی ما فضای $[0,T]$ را به N قسمت تقسیم می کنیم.

$$\Delta t = \frac{T}{N}$$

- در نتیجه با بکارگیری روش اویلر داریم:

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

مروری اجمالی به حل معادله حرارت

- پس از ساده سازی معادله داریم:

$$u_i^{k+1} = r\left[u_{i+1}^k + u_{i-1}^k\right] + (1-2r)u_i^k$$

- که در آن:

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

- توجه به این نکته لازم است که برای پایدار ماندن معادله باید $r \leq 0.5$ باشد.

حل معادله حرارت به کمک matlab

- پارامتر های ورودی
- مش بندی
- تعیین مقدار اولیه
- تعیین شرایط مرزی
- تعیین مقادیر مش بندی

حل معادله حرارت به کمک matlab

- برای توابع ورودی می توانیم از function و یا از دستور input استفاده کنیم.

```
%give your input
a=input('cofficient of eq a=');
xf=input('domin xf=');
T=input('time T=');
it0=input('intital eq it0=');
bx0=input('boundry condition in 0 bx0=');
bxf=input('boundry condition in xf bxf=');
M=input('sub interval for x M=');
N=input('sub interval for t N=');
```

حل معادله حرارت به کمک matlab

- برای مش بندی و تعیین مقدار اولیه داریم:

```
dx=xf/M;
dt=T/N;
x=0:dx:xf;
t=0:dt:T;
%initial condition
for i=1:M+1
    u(i,1)=it0(x(i));
end
```

حل معادله حرارت به کمک matlab

- دستورات مربوط به شرایط مرزی به صورت زیر می باشد

```
%boundary condition
for k=1:N+1
    u([1 M+1],k)=[bx0(t(k));bxf(t(k))];
end
```

حل معادله حرارت به کمک matlab

- دستور محاسبه سلول های مش بندی مورد نظر به صورت زیر است.

```
r=a*(dt/(dx)^2);
if (r<=0.5)
    r1=1-2*r;
    for k=1:N
        for i=2:M
            u(i,k+1)=r*(u(i+1,k)+u(i-1,k))+r1*u(i,k);
        end
    end
else
    disp('stablity error');
end
```

حل معادله حرارت به کمک matlab

- پس از اجرای برنامه، اطلاعات خواسته شده را می دهیم به عنوان مثال داریم:

```
cofficient of eq a=1
domin xf=1
time T=0.05
intital eq it0=inline('100','x')
boundry condition in 0 bx0=inline('0','t')
boundry condition in xf bxf=inline('0','t')
sub interval for x M=5
sub interval for t N=5
```

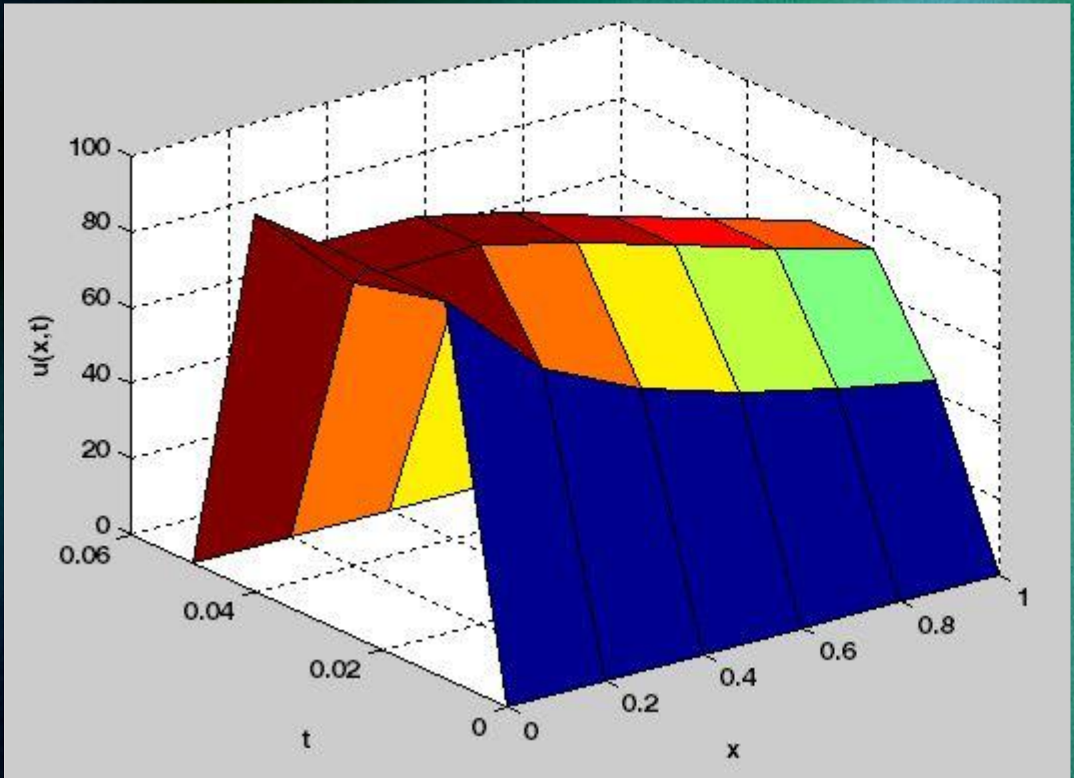
حل معادله حرارت به کمک matlab

- نتایج به دست آمده بعد از اجرای برنامه:

0	0	0	0	0	0
100.0000	75.0000	62.5000	54.6875	48.8281	43.9453
100.0000	100.0000	93.7500	85.9375	78.1250	70.8008
100.0000	100.0000	93.7500	85.9375	78.1250	70.8008
100.0000	75.0000	62.5000	54.6875	48.8281	43.9453
0	0	0	0	0	0

حل معادله حرارت به کمک matlab

- نتایج به دست آمده بعد از اجرای برنامه:



حل معادله حرارت دو بعدی به روش تفاضل متناهی

- مثال دیگری از معادله سهمی گون معادله حرارت دو بعدی است.که به صورت زیر توصیف می شود.

$$A\left(\frac{\partial^2 u(x,y,t)}{\partial x^2}+\frac{\partial^2 u(x,y,t)}{\partial y^2}\right)=\frac{\partial u(x,y,t)}{\partial t}$$

$$x_0 \le x \le x_f \qquad y_0 \le y \le y_f \qquad 0 \le t \le T$$

- با شرایط مرزی

$$u(x_0,y,t)=b_{x0}(y,t) \qquad , \qquad u\left(x_f,y,t\right)=b_{xf}(y,t)$$

$$u(x,y_0,t)=b_{y0}(x,t) \qquad , \qquad u\left(x,y_f,t\right)=b_{yf}(x,t)$$

حل معادله حرارت دو بعدی به روش تفاضل متناهی

- همچنین برای شرایط اولیه داریم:

$$u(x, y, 0) = i_0(x, y)$$

- برای معادله حرارت می توان نوشت:

$$A\left(\frac{u_{i,j+1}^{k} - 2u_{i,j}^{k} + u_{i,j-1}^{k}}{\Delta x^2} + \frac{u_{i+1,j}^{k} - 2u_{i,j}^{k} + u_{i-1,j}^{k}}{\Delta y^2}\right) = \frac{u_{i,j}^{k+1} - u_{i,j}^{k}}{\Delta t}$$

حل معادله حرارت دو بعدی به روش تفاضل متناهی

- نهایتا با ساده سازی داریم:

$$r_x\left(u_{i,j+1}^k - 2u_{i,j}^k + u_{i,j-1}^k\right) + r_y\left(u_{i+1,j}^k - 2u_{i,j}^k + u_{i-1,j}^k\right) + u_{i,j}^k = u_{i,j}^{k+1}$$

- که در آن داریم:

$$r_x = A\Delta t/\Delta x^2 \qquad r_y = A\Delta t/\Delta y^2$$

$$\Delta x = x_f - x_0/M_x \qquad \Delta y = y_f - y_0/M_y \qquad \Delta t = T/N$$

حل معادله حرارت دو بعدی با matlab

- پارامتر های ورودی
- مش بندی
- تعیین مقدار اولیه
- تعیین شرایط مرزی
- تعیین مقادیر مش بندی

حل معادله حرارت دو بعدی با matlab

- برای تعیین پارامتر های ورودی دستورات زیر را وارد می کنیم:

```
a = 1e-4;
ixy0 = inline('10','x','y');
bx0 = inline('exp(y)-cos(y)','y','t');
bxf = inline('exp(y)*cos(4)-exp(4)*cos(y)','y','t');
by0 = inline('cos(x)-exp(x)','x','t');
byf = inline('exp(4)*cos(x)-exp(x)*cos(4)','x','t');
D = [0 4 0 4];
T = 5000;
Mx = 20;
My = 20;
N = 500;
```

حل معادله حرارت دو بعدی با matlab

- برای مش بندی و تعیین مقدار اولیه دستورات زیر را وارد می کنیم:

```
dx = (D(2) - D(1))/Mx; x = D(1)+[0:Mx]*dx;
dy = (D(4) - D(3))/My; y = D(3)+[0:My]*dy;
dt = T/N; t = [0:N]*dt;
for j=1:Mx+1
    for i=1:My+1
        u(i,j) = ixy0(x(j),y(i));
    end
end
rx = a*dt/(dx*dx); ry = a*dt/(dy*dy);
```

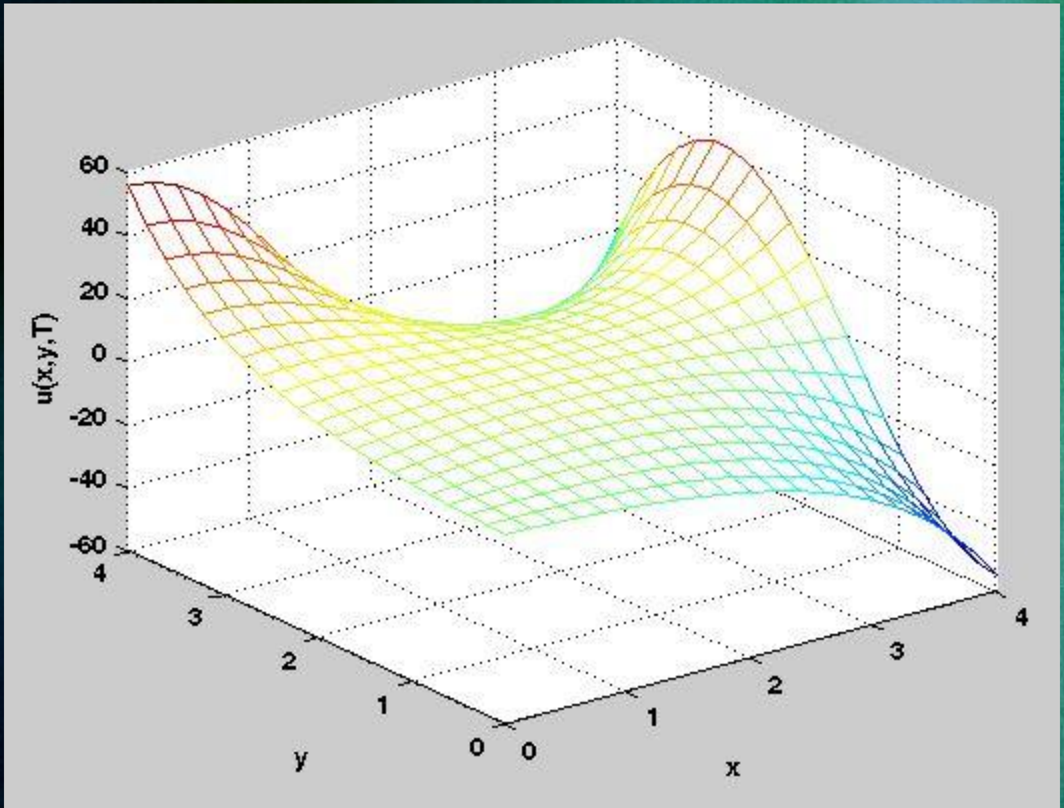
حل معادله حرارت دو بعدی با matlab

- برای تعیین شرایط مرزی و تعیین مقادیر سلول های محاسباتی دستورات زیر را وارد می کنیم:

```
for k=1:N
    for i = 1:My+1
        u(i,[1,Mx+1],k) = [bx0(y(i),t(k));bxf(y(i),t(k))];
    end
    for j = 1:Mx+1
        u([1 My+1],j,k)=[by0(x(j),t(k));byf(x(j),t(k))];
    end
    for j=2:Mx
        for i=2:My
            u(j,i,k+1)=rx.*[u(j+1,i,k)-2.*u(j,i,k)+u(j-1,i,k)]+...
                ry.*[u(j,i+1,k)-2.*u(j,i,k)+u(j,i-1,k)]+u(j,i,k);
        end
    end
end
```

حل معادله حرارت دو بعدی با matlab

- با اجرای برنامه به نتایج زیر می رسیم.



معادلات هذلولی

- به عنوان مثال برای معادلات هذلولی به معادله موج یک بعدی می پردازیم.

$$A\frac{\partial^2 u(x,t)}{\partial x^2}=\frac{\partial^2 u(x,t)}{\partial t^2}$$

$$0 \le x \le xf \qquad 0 \le t \le T$$

- برای شرایط مرزی و اولیه داریم:

$$\begin{cases} u(0,t)=b_0(t) \\ u\left(x_f,t\right)=b_{xf}(t) \end{cases}$$

$$\begin{cases} u(x,0)=i_0(x) \\ \dfrac{\partial u(x,0)}{\partial t}=i\acute{}_0(t) \end{cases}$$

مروری اجمالی بر حل معادله موج

- برای حل به کمک روش تفاضل متناهی داریم:

$$A\frac{u_{i+1}^{k}-2u_{i}^{k}+u_{i-1}^{k}}{\Delta x^{2}}=\frac{u_{i}^{k+1}-2u_{i}^{k}+u_{i}^{k-1}}{\Delta t^{2}}$$

- که در آن داریم:

$$\Delta x=\frac{x_f}{M}$$

$$\Delta t=\frac{T}{N}$$

مروری اجمالی بر حل معادله موج

- نتیجتا با ساده سازی داریم:

$$u_i^{k+1} = r\left(u_{i+1}^k + u_{i-1}^k\right) + 2(1-r)u_i^k - u_i^{k-1}$$

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

- برای بدست آوردن u_i^1 ، $k = 0$ را در معادله جایگذاری می کنیم. که به علت وجود $u_i^{-1} = u(x, -\Delta t)$ ، نمی توانیم آن را بدست آوریم.
- نتیجتا با تقریب شرایط اولیه داریم:

$$\frac{u_i^1 - u_i^{-1}}{2\Delta t} = \acute{i}_0(x_i)$$

مروری اجمالی بر حل معادله موج

- با جایگذاری در معادله اصلی داریم:

$$u_i^1 = \frac{1}{2}\mathrm{r}\left(u_{i+1}^0 + u_{i-1}^0\right) + (1-r)u_i^0 + \acute{\imath}_0(x_i)\Delta t$$

- همچنین برای پایداری مسئله $r \leq 1$ می باشد.

حل معادله موج به کمک matlab

- تعیین پارامتر های ورودی
- مش بندی
- تعیین مقدار اولیه
- تعیین شرایط مرزی
- حل مساله برای u_i^1
- حل مساله برای بقیه مقادیر

حل معادله موج به کمک matlab

- برای تعیین پارامتر های ورودی کد دستورات زیر را وارد می کنیم. در این مثال از Function استفاده می کنیم.
- دقت کنید که برای ذخیره کردن m-file خود نام آن را هم نام تابع خود قرار دهید.

```
function [u,x,t] = wave(a,xf,T,it0,i1t0,bx0,bxf,M,N)
%solve a u_xx = u_tt for 0<=x<=xf, 0<=t<=T
% Initial Condition: u(x,0) = it0(x), u_t(x,0) = i1t0(x)
% Boundary Condition: u(0,t)= bx0(t), u(xf,t) = bxf(t)
%M=# of subintervals along x axis
%N=# of subintervals along t axis
```

حل معادله موج به کمک matlab

- برای مش بندی و تعیین مقادیر اولیه از کد دستورات زیر استفاده می کنیم.

```
dx = xf/M; x = 0:dx:xf;
dt = T/N; t = 0:dt:T;
r = a*(dt/dx)^ 2; r1 = r/2; r2 = 2*(1 - r);

for i = 1:M+1
    u(i,1) = it0(x(i)) ;
end
```

حل معادله موج به کمک matlab

- برای تعیین شرایط مرزی از کد دستورات زیر استفاده می کنیم.

```
for k = 1:N+1
    u([1 M+1],k) = [bx0(t(k));bxf(t(k))];
end
```

حل معادله موج به کمک matlab

- برای حل مساله برای u_i^1 کد دستورات زیر را وارد می کنیم.

```
for i = 2:M
    u(i,2) = r1*[u(i+1,1)+u(i-1,1)] + (1-r)*u(i,1) + dt*i1t0(x(i));
end
```

حل معادله موج به کمک matlab

- برای حل مساله برای بقیه مقادیر کد دستورات زیر را وارد می کنیم.

```
for k = 2:N
    for i = 2:M
        u(i,k+1) = r*[u(i+1,k)+u(i-1,k)] + r2*u(i,k) - u(i,k-1);
    end
end
```

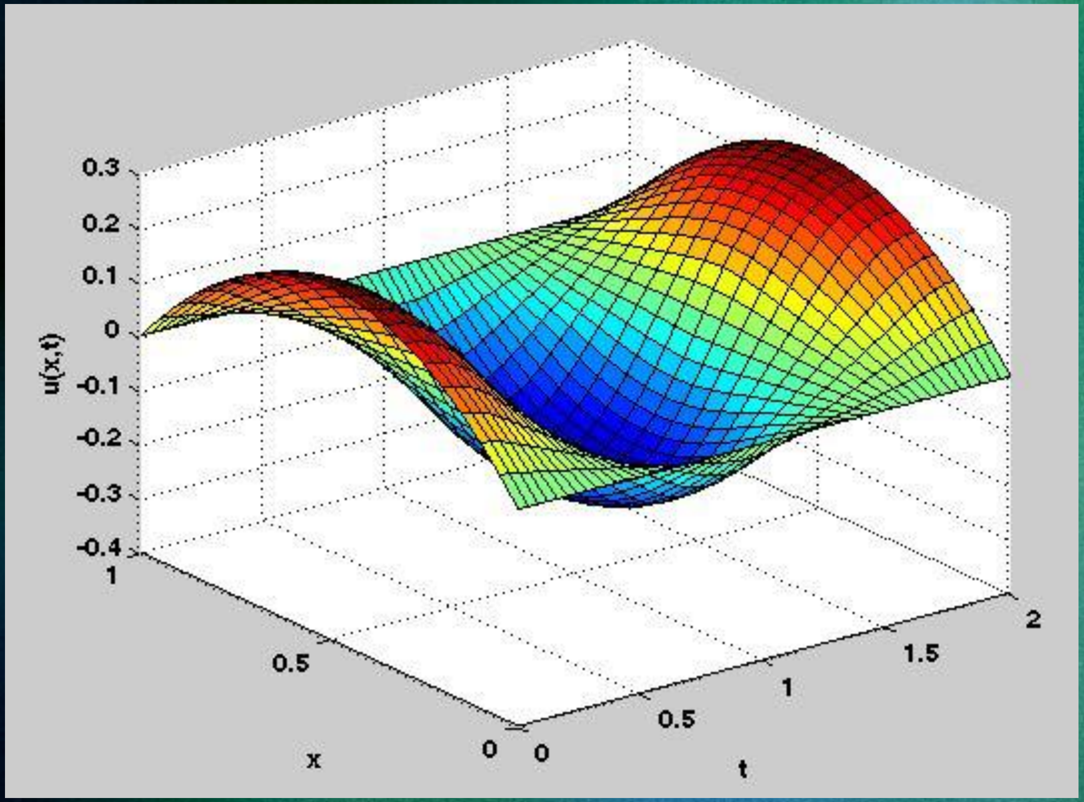
حل معادله موج به کمک matlab

- حال برای Function مربوطه برنامه ای به صورت زیر می نویسیم.

```
clear all;
a=1;
it0 = inline('x*(1-x)','x');
i1t0 = inline('0','x');
bx0 = inline('0','t');
bxf = inline('0','t');
xf=1;M=10; T=1.1;N=11;
[u,x,t] = wave(a,xf,T,it0,i1t0,bx0,bxf,M,N);
if(u ~= 1000)
    disp(u);
    surf(t,x,u);
    xlabel('t'),ylabel('x'),zlabel('u(x,t)');
else
    disp('error')
end
```

حل معادله موج به کمک matlab

- با اجرای برنامه نتایج زیر بدست می آید.



حل معادله موج دو بعدی به روش تفاضل متناهی

- در این قسمت ما به بررسی معادله موج دو بعدی با رابطه زیر می پردازیم.

$$A\left(\frac{\partial^2 u(x,y,t)}{\partial x^2}+\frac{\partial^2 u(x,y,t)}{\partial y^2}\right)=\frac{\partial^2 u(x,y,t)}{\partial t^2}$$

$$0 \le x \le x_f$$

$$0 \le y \le y_f$$

$$0 \le t \le T$$

حل معادله موج دو بعدی به روش تفاضل متناهی

- شرایط مرزی و اولیه معادله به صورت زیر می باشد.

$$\begin{cases} u(0,y,t) = b_{x0}(y,t) \\ u(x_f,y,t) = b_{xf}(y,t) \\ u(x,0,t) = b_{y0}(x,t) \\ u(x,y_f,t) = b_{yf}(x,t) \end{cases}$$

$$\begin{cases} u(x,y,0) = i_0(x,y) \\ \dfrac{\partial u(x,y,0)}{\partial t} = i'_0(x,y) \end{cases}$$

حل معادله موج دو بعدی به روش تفاضل متناهی

- همانند روش یک بعدی ما با تقریب تفاضل مرکزی برای مشتق مرتبه دوم داریم:

$$A\left(\frac{u_{i,j+1}^{k}-2u_{i,j}^{k}+u_{i,j-1}^{k}}{\Delta x^2}+\frac{u_{i+1,j}^{k}-2u_{i,j}^{k}+u_{i-1,j}^{k}}{\Delta y^2}\right)=\frac{u_{i,j}^{k+1}-2u_{i,j}^{k}+u_{i-1,j}^{k}}{\Delta t^2}$$

$$\left\{\begin{array}{l}\Delta x=\dfrac{x_f}{M_x}\\[2ex]\Delta y=\dfrac{y_f}{N_y}\\[2ex]\Delta t=\dfrac{T}{N}\end{array}\right.$$

حل معادله موج دو بعدی به روش تفاضل متناهی

- نهایتا با ساده سازی به معادله زیر می رسیم.

$$u_{i,j}^{k+1} = r_x[u_{i,j+1}^k + u_{i,j-1}^k] + 2(1 - r_x - r_y)u_{i,j}^k + r_y[u_{i+1,j}^k + u_{i-1,j}^k] - u_{i,j}^{k-1}$$

$$r_x = \frac{A\Delta t^2}{\Delta x^2} \qquad r_y = \frac{A\Delta t^2}{\Delta y^2}$$

- برای بدست آوردن $u_{i,j}^1$ ، $k = 0$ را در معادله جایگذاری می کنیم. که به علت وجود $u(x,y,-\Delta t)$ $u_{i,j}^{-1} =$ ، نمی توانیم آن را بدست آوریم.

حل معادله موج دو بعدی به روش تفاضل متناهی

- بنابراین ما با تقریب تفاضل مرکزی برای شرایط اولیه داریم:

$$\frac{u_{i,j}^{1}-u_{i,j}^{-1}}{2\Delta t}=\acute{\iota}_0(x,y)$$

- بنابراین با جایگذاری در معادله اصلی برای $k=0$ داریم:

$$u_{i,j}^{1}=\frac{1}{2}\left[r_x\left(u_{i,j+1}^{0}+u_{i,j-1}^{0}\right)+r_y\left(u_{i+1,j}^{0}+u_{i-1,j}^{0}\right)\right]+\left(1-r_x-r_y\right)u_{i,j}^{0}+\acute{\iota}_0\left(x_j,y_i\right)\Delta t$$

حل معادله موج دو بعدی با matlab

- تعیین پارامتر های ورودی
- مش بندی
- تعیین مقدار اولیه
- تعیین شرایط مرزی
- حل مساله برای $u_{i,j}^{1}$
- حل مساله برای بقیه مقادیر

حل معادله موج دو بعدی با matlab

- برای تعیین پارامتر های ورودی کد دستورات زیر را وارد کنیم.

```
clear all;
close all;
clc;
it0 = inline('0.1*sin(pi*x)*sin(pi*y/2)','x','y');
i1t0 = inline('0','x','y');
bx0 = inline('0','y','t');
bxf = inline('0','y','t');
by0 = inline('0','x','t');
byf = inline('0','x','t');
a = 0.25; D = [0 2 0 2]; T = 1.8 ; Mx = 40; My = 40; N = 40;
```

حل معادله موج دو بعدی با matlab

- برای مش بندی و تعیین مقادیر اولیه از کد دستورات زیر استفاده می کنیم.

```
dx = (D(2)- D(1))/Mx; x = D(1)+[0:Mx]*dx;
dy = (D(4)- D(3))/My; y = D(3)+[0:My]*dy;
dt = T/N; t = [0:N]*dt;
%Initialization
for j = 2:Mx
    for i = 2:My
        u(i,j) = it0(x(j),y(i)); ut(i,j) = i1t0(x(j),y(i));
    end
end

adt2 = a*dt*dt; rx = adt2/(dx*dx); ry = adt2/(dy*dy);
rxy1 = 1-rx-ry ; rxy2 = rxy1*2;
```

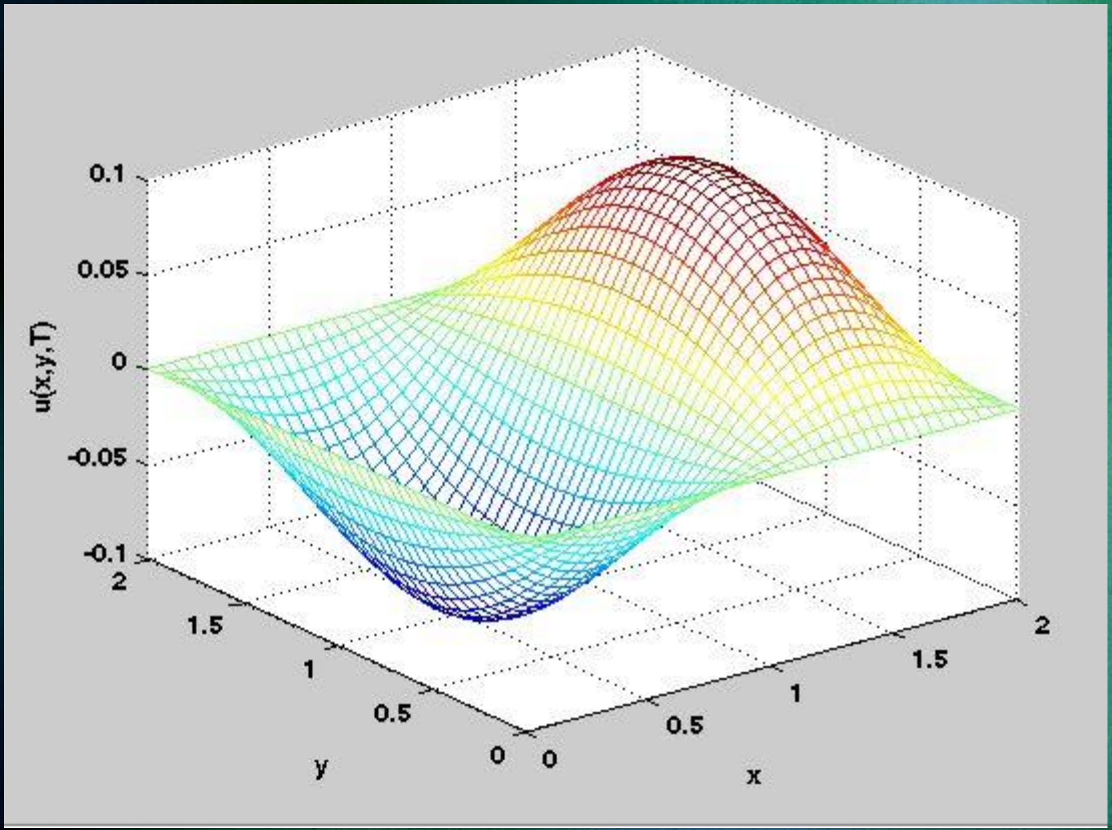
حل معادله موج دو بعدی با matlab

- برای تعیین شرایط مرزی و محاسبه بقیه مقادیر سلول های محاسباتی از کد دستورات زیر استفاده می کنیم.

```
for k = 1:N
    for i = 1:My + 1 %Boundary condition
        u(i,[1 Mx + 1],k) = [bx0(y(i),t(k)); bxf(y(i),t(k))];
    end
    for j=1:Mx+1
        u([1 My + 1],j,k) = [by0(x(j),t(k)); byf(x(j),t(k))];
    end
    if k == 1
        for i = 2:My
            for j = 2:Mx
                u(i,j,k+1) = 0.5*(rx*(u(i,j - 1,k) + u(i,j + 1,k))...
                    + ry*(u(i - 1,j,k)+u(i + 1,j,k))) + rxy1*u(i,j,k) + dt*ut(i,j);
            end
        end
    else
        for i = 2:My
            for j = 2:Mx
                u(i,j,k+1) = rx*(u(i,j - 1,k)+ u(i,j + 1,k))...
                    + ry*(u(i - 1,j,k) + u(i + 1,j,k)) + rxy2*u(i,j,k) -u(i,j,k-1);
            end
        end
    end
end
```

حل معادله موج دو بعدی با matlab

- با اجرای برنامه نتایج زیر بدست می آید.



با تشکر از توجه شما