CA2 - mathematical engineering

Mostafa Kermaninia

810101575

نکته : توضیحات ریاضیاتی در گزارش آمده است، اما توضیحات و جزییات برنامه نویسی مربوطه، خط به خط بصورت کامنت در داخل کد ها نوشته شده است

1) حل معادله حرارت

1.1) فرم کلی معادله در MATLAB

با توجه به صورت سوال، براحتی با جایگذاری این ثوابت:

```
c = 50;
f = DuDx;
s = 0;
```

میتوان به فرم نهایی مدنظر سوال، یعنی این:

$$\frac{\partial^2 u}{\partial x^2} - \frac{1}{c^2}\frac{\partial u}{\partial t} = 0$$

رسید. پس داریم:(فایل Q1.mlx)

```
function [c, f, s] = Equation(x, t, u, DuDx)
    c = 50;     % Coefficient for the time derivative term, scaled by 50
    f = DuDx;   % Flux term representing the spatial derivative of u
    s = 0;      % Source term is zero (no additional source)
end
```

1.2) شرایط اولیه

طبق گفته ی سوال، داریم:(فایل Q1.mlx)

```
function value = Init(x)
    value = 2 .* exp(x);  % Initial condition: u(x, 0) = 2 * exp(x)
end
```

1.3) شرایط مرزی

با توجه به فرم کلی دیفالت در متلب، یعنی این:

$$p(x,t,u)+q(x,t)f\left(x,t,u,\frac{\partial u}{\partial x}\right)=0$$

و شرایط مرزی ای که در سوالمان داریم، کافیست بگذاریم:(فایل **Q1.mlx**)

```
function [pl, ql, pr, qr] = BC(xl, ul, xr, ur, t)
    pl = ul;      % Left boundary condition at x = 0
    ql = 0;       % Coefficient for the left boundary
    pr = ur - 50; % Right boundary condition at x = 2
    qr = 0;       % Coefficient for the right boundary
end
```

1.4) حل معادله

همچنان طبق صورت سوال بازه ها را هم میگذاریم و معادله را طبق آنها حل کرده و داریم:(فایل **Q1.mlx**)

```
x = linspace(0, 2, 200);  % Discretize the spatial domain from 0 to 2 with 200 points
t = linspace(0, 10, 201); % Discretize the time domain from 0 to 10 with 201 points

sol = pdepe(0, @Equation, @Init, @BC, x, t);  % Solve the PDE using pdepe
```

1.5) خواسته ها

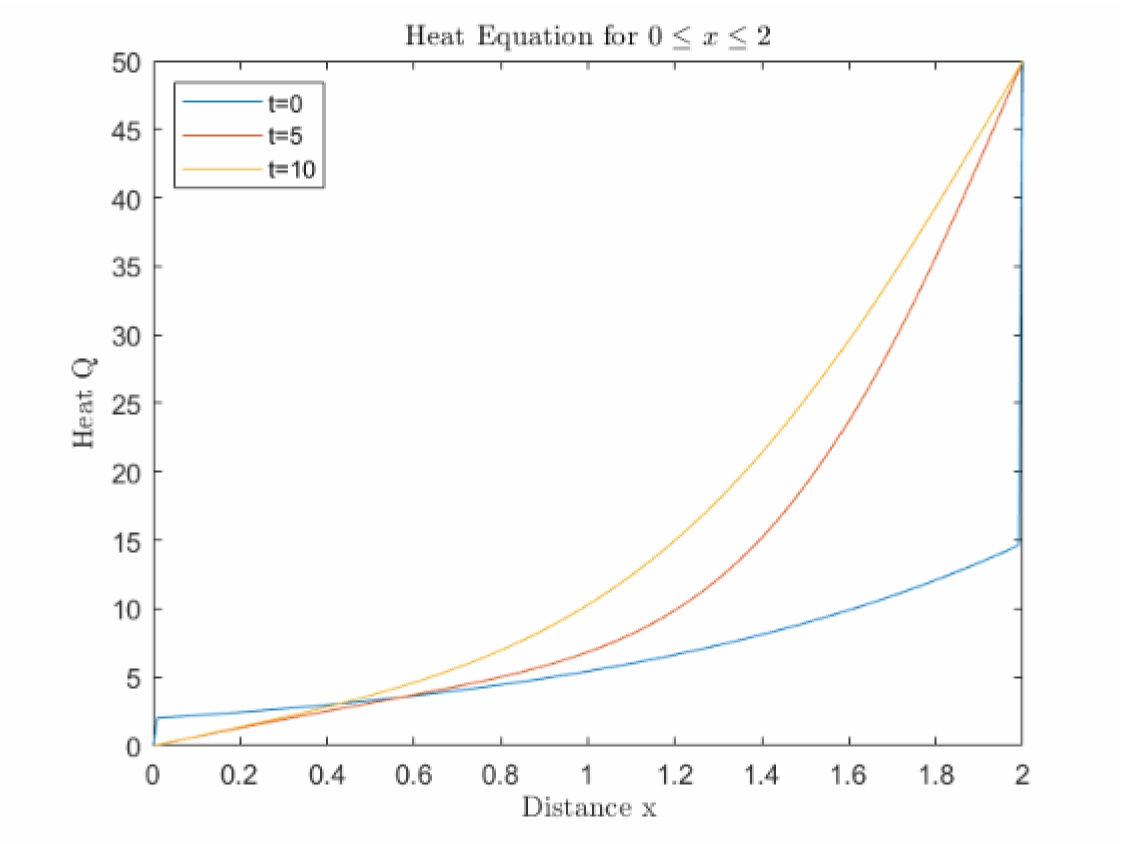
– توابع معرفی شده برای معادله ، شرایط مرزی و شرایط اولیه را پیاده سازی کردم و در بخش های قبلی توضیحات مربوطه را نوشتم

–حالا معادله را به کمک دستور معرفی شده حل کرده و پس از حل معادله به نمایش دمای میله در زمان های **t = 0 ,5 10** ,در یک نمودار پرداخته و تحلیل میکنم:

کد نمودار ها: (فایل **Q1.mlx**)

```
% Plot the solution at different times
plot(x, sol(1,:))       % Plot for t = 0
hold on
plot(x, sol(101,:))     % Plot for t = 5
plot(x, sol(201,:))     % Plot for t = 10
hold off
legend('t=0', 't=5', 't=10', 'Location', 'northwest')  % Add legend
title('Heat Equation for $0 \le x \le 2$', 'interpreter', 'latex')  % Title of the plot
xlabel('Distance x', 'interpreter', 'latex')  % X-axis label
ylabel('Heat Q', 'interpreter', 'latex')      % Y-axis label
```

نمودار نهایی شامل هر 3 نمودار، به این شکل است:



تحلیل:

همانطور که انتظار میرفت، به مرور زمان تاثیر شرایط اولیه از بین می رود و نمودار بیشتر تحت تاثیر صورت معادله و شرایط مرزی خواهد بود. یعنی در t=0 اختلاف بین گرما(دما)ی دو طرف خیلی زیاد نیست اما با گذشت زمان، اختلاف گرما(دما) بیشتر و بیشتر میشود.

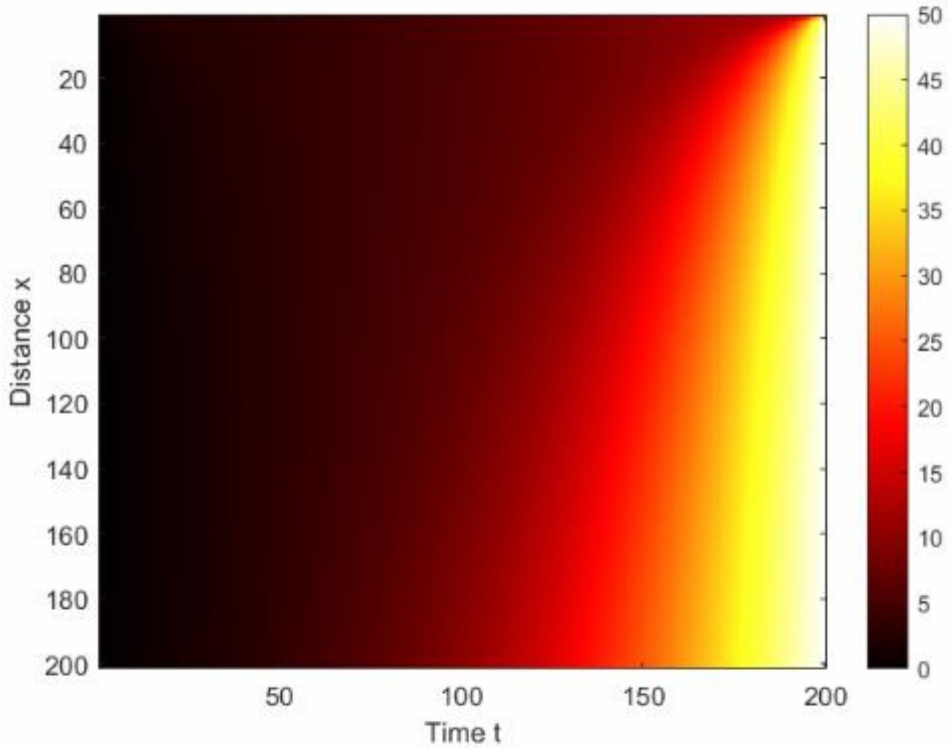
– حالا به کمک دستورات colormap(),imagesc() به صورت دو بعدی نمودار تغییرات دمایی را در طول مکان و زمان رسم و مشاهده کردهو تحلیل میکنیم:

کد نمودارها (فایل Q1.mlx)

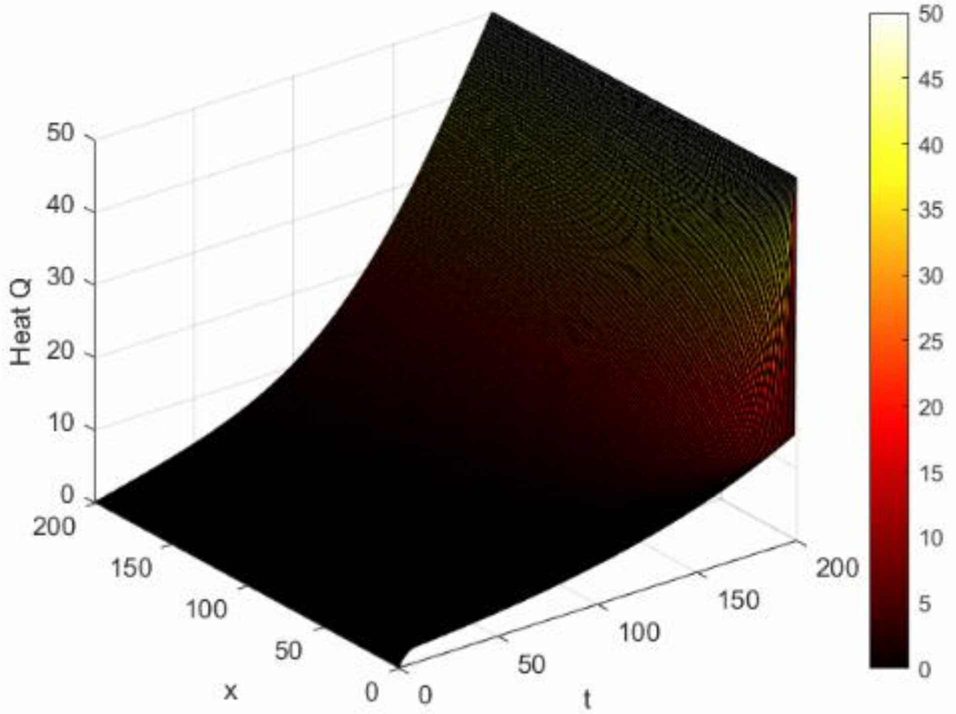
```
imagesc(sol)  % Display the solution as an image
colormap hot  % Apply hot colormap
xlabel('Time t')  % X-axis label
ylabel('Distance x')  % Y-axis label
colorbar  % Add colorbar
```

```
surf(sol)  % Display the solution as a 3D surface plot
xlabel("t")  % X-axis label
ylabel("x")  % Y-axis label
zlabel('Heat Q')  % Z-axis label
colorbar  % Add colorbar
```

نمودارهای دو بعدی و سه بعدی هم مجددا نکات گفته شده را تائید می کنند و آنها را به صورت پیوسته نمایش میدهند.(صفحه ی بعد)



Distance x
Time t
20
40
60
80
100
120
140
160
180
200
50
100
150
200
0
5
10
15
20
25
30
35
40
45
50



Heat Q
x
t
0
10
20
30
40
50
200
150
100
50
0
0
50
100
150
200

در آخر نیز همه کارها را برای بازه های جدید تکرار میکنیم:

کد حل معادله در بازه ی جدید و نمودارها (فایل Q1.mlx)

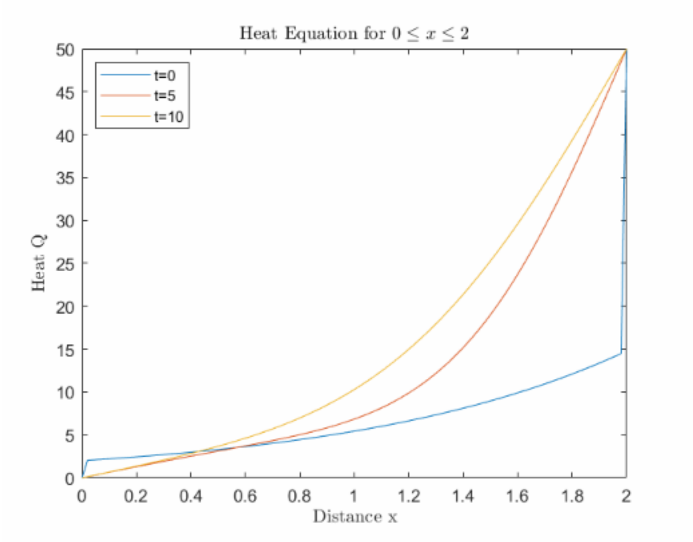
```
% Solve the PDE again with different spatial and time discretization
x = linspace(0, 2, 100);  % Discretize the spatial domain from 0 to 2 with 100 points
t = linspace(0, 10, 101); % Discretize the time domain from 0 to 10 with 101 points

sol = pdepe(0, @Equation, @Init, @BC, x, t);  % Solve the PDE using pdepe
```

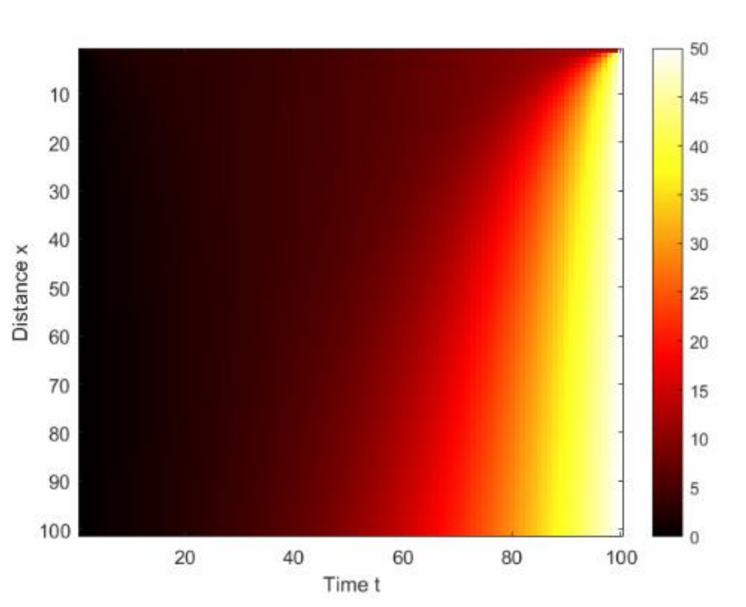
```
% Plot the solution at different times
plot(x, sol(1,:))       % Plot for t = 0
hold on
plot(x, sol(51,:))      % Plot for t = 5
plot(x, sol(101,:))     % Plot for t = 10
hold off
legend('t=0', 't=5', 't=10', 'Location', 'northwest')  % Add legend
title('Heat Equation for $0 \le x \le 2$', 'interpreter', 'latex')  % Title of the plot
xlabel('Distance x', 'interpreter', 'latex')  % X-axis label
ylabel('Heat Q', 'interpreter', 'latex')      % Y-axis label
```

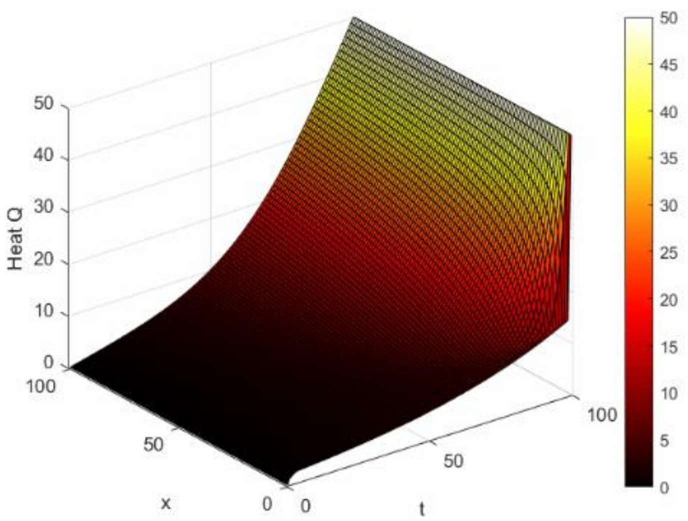
```
imagesc(sol)  % Display the solution as an image
colormap hot  % Apply hot colormap
xlabel('Time t')  % X-axis label
ylabel('Distance x')  % Y-axis label
colorbar  % Add colorbar
```

```
surf(sol)  % Display the solution as a 3D surface plot
xlabel("t")  % X-axis label
ylabel("x")  % Y-axis label
zlabel('Heat Q')  % Z-axis label
colorbar  % Add colorbar
```

خود نمودار ها:







تنها تفاوت این است که گامهای بلندتری بر میداریم و اینکار باعث میشود کمی حالت روان بودن نمودارها کاهش بیابد و در اصل، accuracy نمودار ها کاهش یافته است.

2) حل معادله هلمهلتز

فرم کلی معادله:

$$\nabla^2 A + k^2 A = 0, \quad k : Wave\ number, \quad A : Amplitude$$

برای استخراج این معادله از معادله ی موج، داریم:

The three-dimensional wave equation for a field $u(x, y, z, t)$ is given by:

$$\frac{\partial^2 u}{\partial t^2} = c^2 \nabla^2 u$$

where c is the wave speed and ∇^2 is the Laplacian operator:

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$$

To derive the Helmholtz equation from the wave equation, we assume a simple traveling wave solution of the form:

$$u(x, y, z, t) = A(x, y, z)e^{-i\omega t}$$

where:

- $A(x, y, z)$ is the amplitude of the wave.
- ω is the angular frequency.
- t is time.
- i is the imaginary unit.

Substituting $u(x, y, z, t)$ into the wave equation, we get:

$$\frac{\partial^2}{\partial t^2}(A(x, y, z)e^{-i\omega t}) = c^2\nabla^2(A(x, y, z)e^{-i\omega t})$$

Taking the partial derivatives with respect to time, we have:

$$\frac{\partial}{\partial t}(A(x, y, z)e^{-i\omega t}) = -i\omega A(x, y, z)e^{-i\omega t}$$

$$\frac{\partial^2}{\partial t^2}(A(x, y, z)e^{-i\omega t}) = -\omega^2 A(x, y, z)e^{-i\omega t}$$

And for the right-hand side:

$$c^2\nabla^2(A(x, y, z)e^{-i\omega t}) = c^2(\nabla^2 A(x, y, z))e^{-i\omega t}$$

Substituting these derivatives back into the wave equation, we get:

$$-\omega^2 A(x, y, z)e^{-i\omega t} = c^2(\nabla^2 A(x, y, z))e^{-i\omega t}$$

Dividing both sides by $e^{-i\omega t}$, we obtain:

$$-\omega^2 A(x, y, z) = c^2\nabla^2 A(x, y, z)$$

Rewriting this equation, we get:

$$\nabla^2 A + \left(\frac{\omega}{c}\right)^2 A = 0$$

If we let $k = \frac{\omega}{c}$, the Helmholtz equation is:

$$\nabla^2 A + k^2 A = 0$$

روش‌های حل معادله هلمهولتز:

1. روش جداسازی متغیرها :این روش برای حل معادله در دامنه‌های ساده و با شرایط مرزی مناسب استفاده می‌شود. ابتدا فرض می‌شود که راه‌حل به صورت حاصل ضرب چند تابع تک‌متغیره است و سپس با جداسازی متغیرها، چندین معادله دیفرانسیل معمولی به دست می‌آید.
2. روش طیفی :(Spectral Methods) این روش شامل تبدیل فوریه یا تبدیل‌های مشابه برای تبدیل معادله دیفرانسیل به دامنه فرکانسی است. در این دامنه، معادله تبدیل به معادله‌ای جبری می‌شود که حل آن ساده‌تر است.
3. روش عددی :از روش‌های عددی مانند روش تفاضل محدود (Finite Difference Method)یا روش المان محدود (Finite Element Method) برای حل معادله هلمهولتز در دامنه‌های پیچیده‌تر استفاده می‌شود.

شرایط مرزی:

شرایط مرزی معادله هلمهولتز به صورت زیر می‌تواند باشد:

1. شرایط دیریکله :(Dirichlet Boundary Conditions) مقدار تابع AAAدر مرزهای دامنه مشخص است

$$A|_{\partial\Omega} = g$$

2. شرایط نویمان :(Neumann Boundary Conditions) مقدار مشتق نرمال تابع AAA در مرزهای دامنه مشخص است

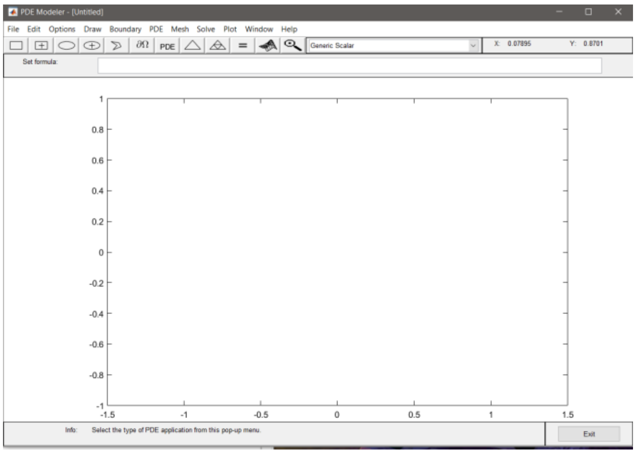
$$\left.\frac{\partial A}{\partial n}\right|_{\partial\Omega} = h$$

3. شرایط مخلوط :(Mixed Boundary Conditions) ترکیبی از شرایط دیریکله و نویمان در مرزهای مختلف دامنه.

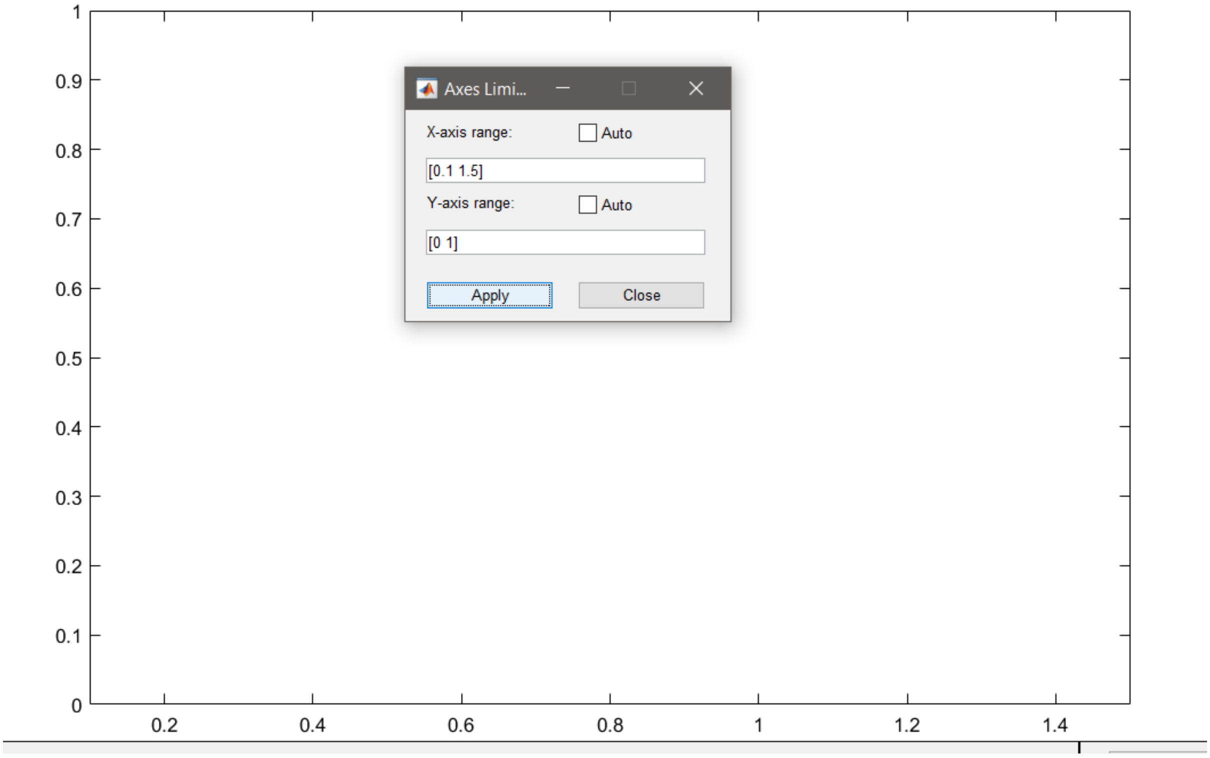
4. شرایط رادییشن **(Radiation Boundary Conditions):** برای مسائل در فضاهای باز (مانند امواج رادیویی)، شرایط رادییشن استفاده می‌شود که شامل فروکش کردن موج در بی‌نهایت است.

2.1) مراحل حل معادله

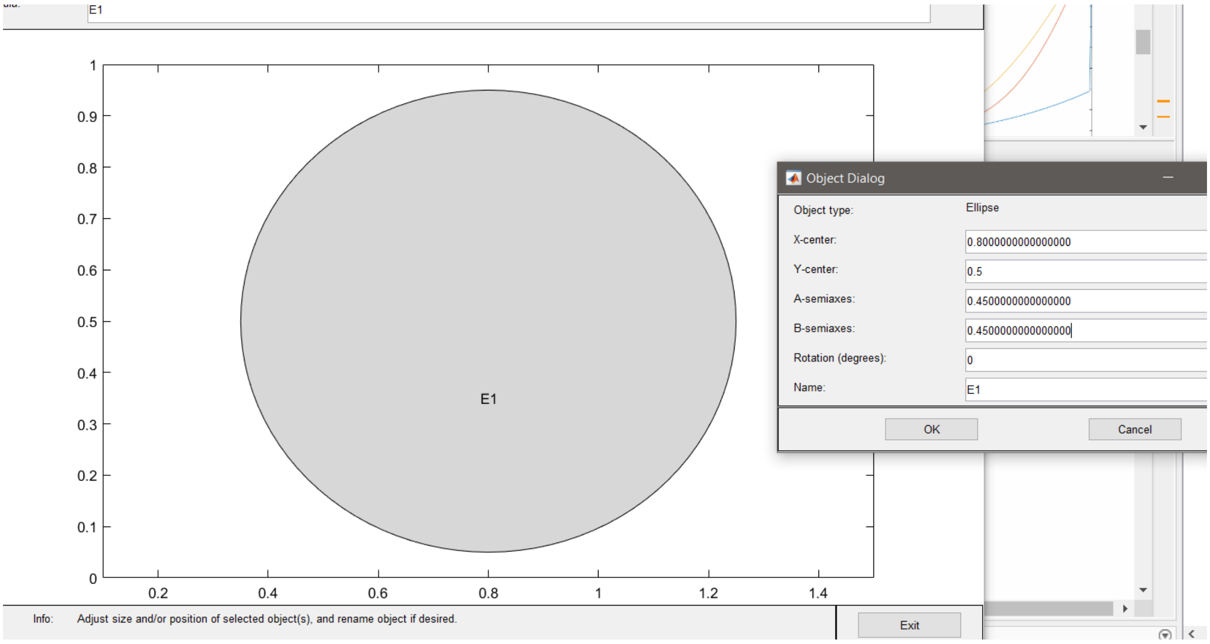
- ابتدا **Modeler PDE** را با نوشتن عبارت **pdeModeler** در **CMD** نرم افزار **MATLAB** باز کردم:



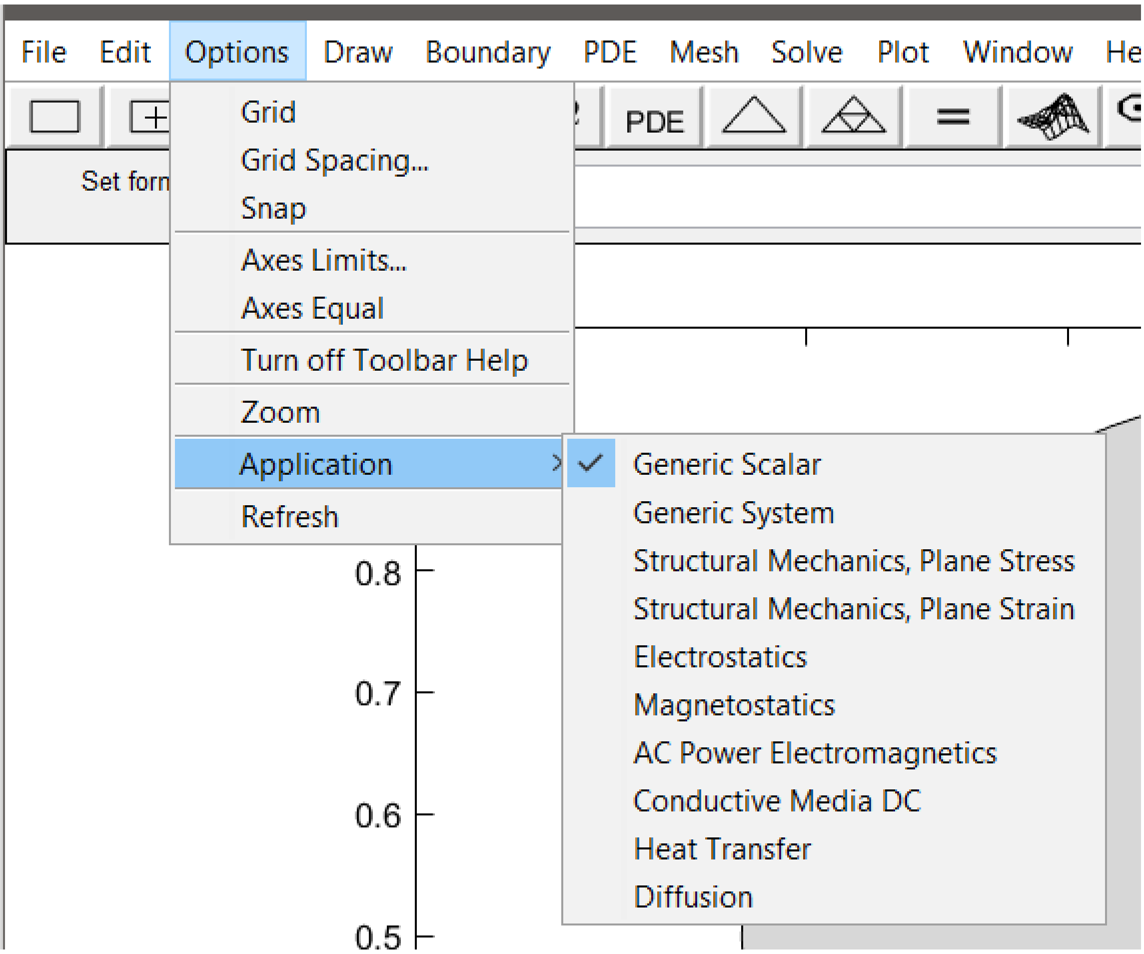
-محدوده محور های **x** و **y** را به ترتیب [**0.1 1.5**] و [**0 1**] در نظر گرفتم.(برای انجام اینکار به قسمت **limits Axes-Options** مراجعه کردم):



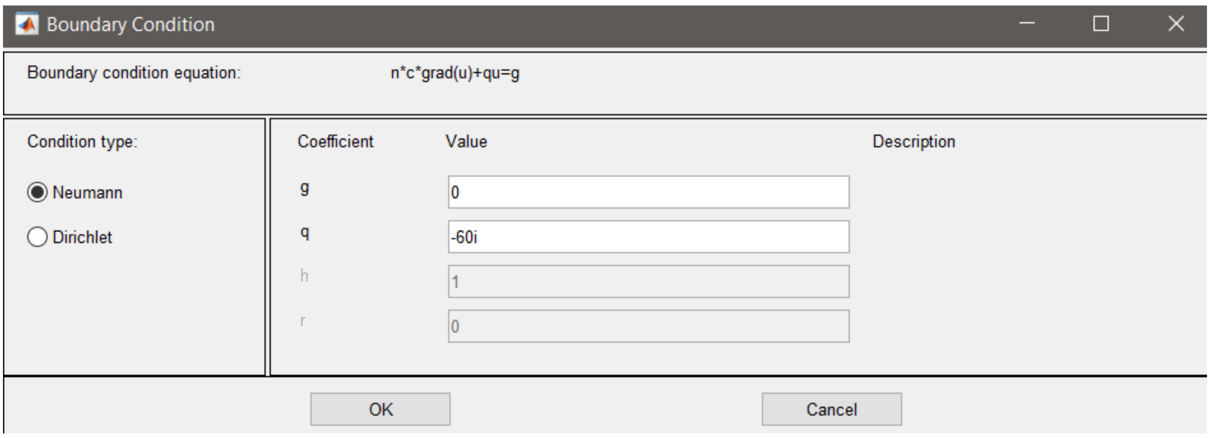
-یک دایره به شعاع **0.45** و مرکز (**0.8 0.5**) رسم کردم:



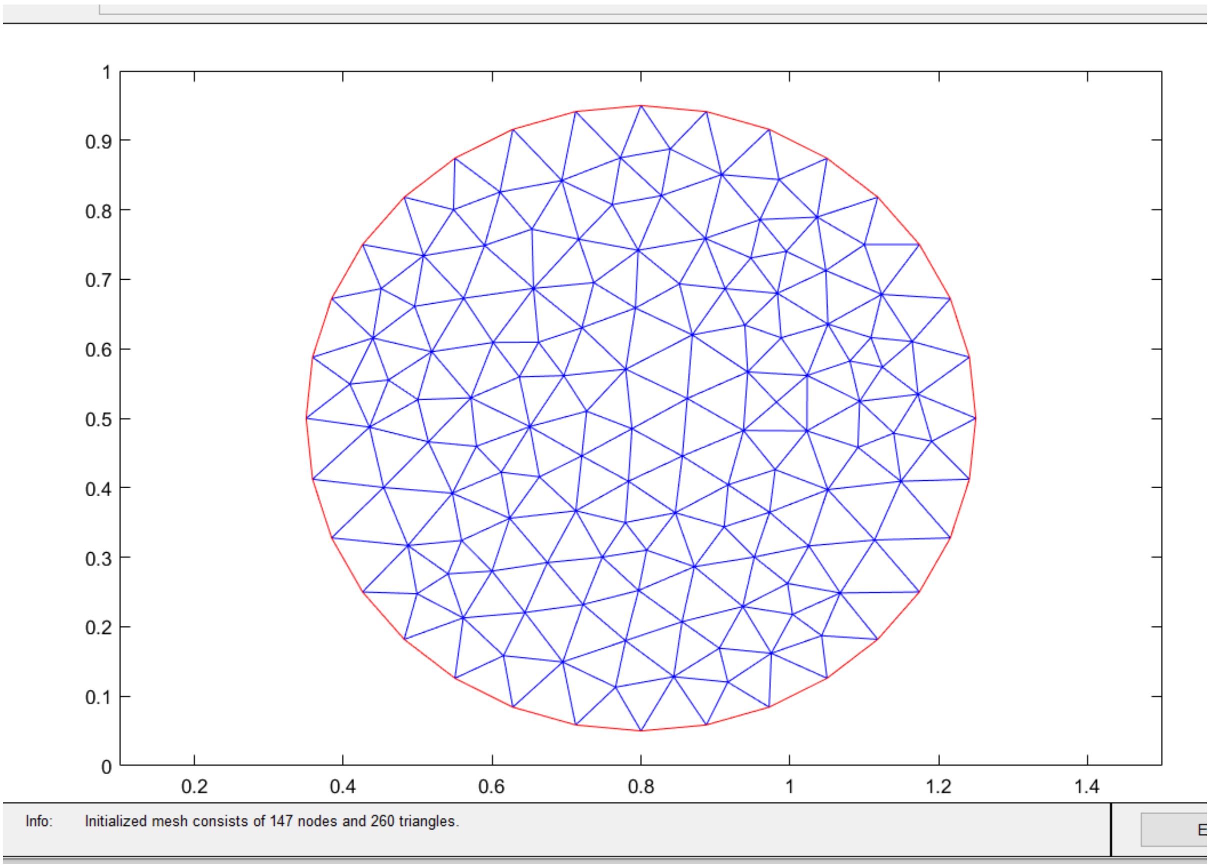
-مود کاری را با مراجعه به بخش **Application mode** به **Scalar Generic** تنظیم کردم:



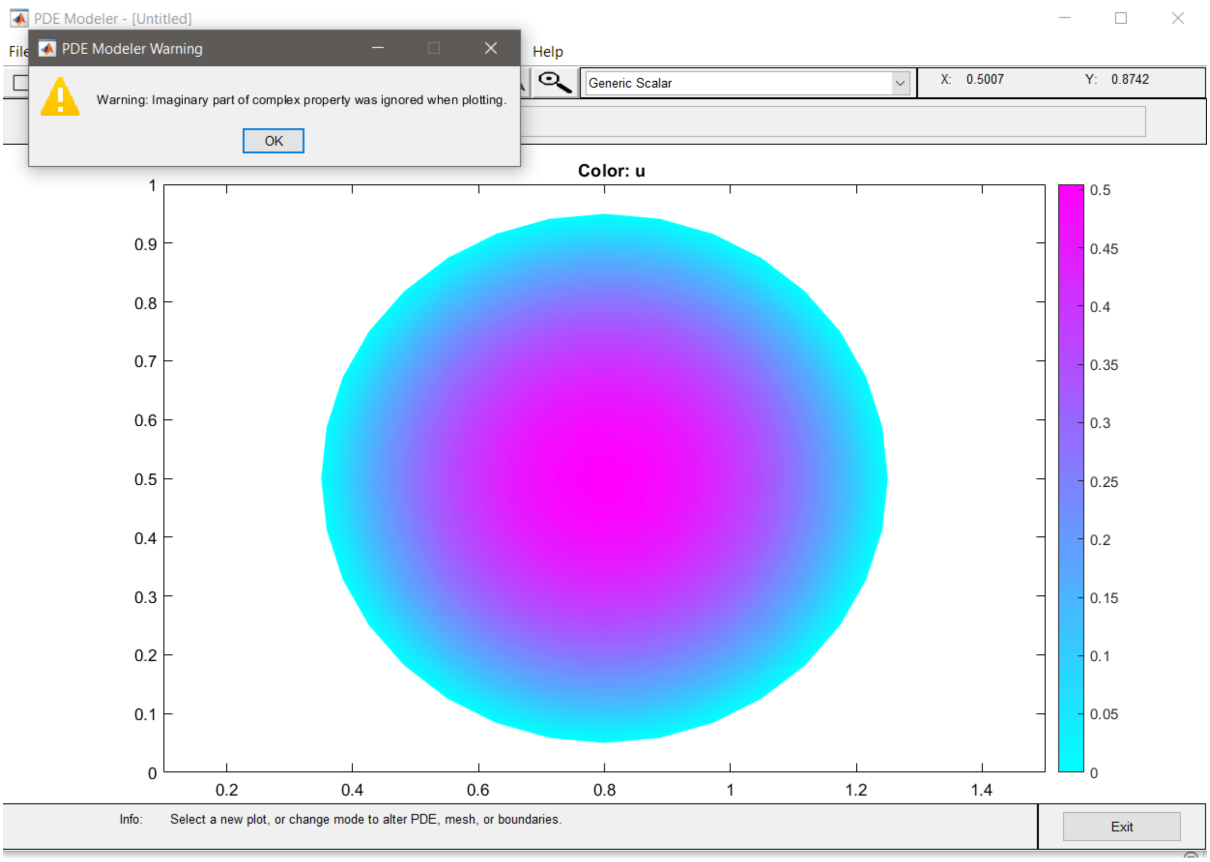
-شرایط مرزی را با مراجعه به قسمت **Boundary Specify- Conditions** مشخص کردم. دقت کردم که شرایط مرزی از شرط **Neumann** با **q = i−60** و **0 = g** استفاده کنم:



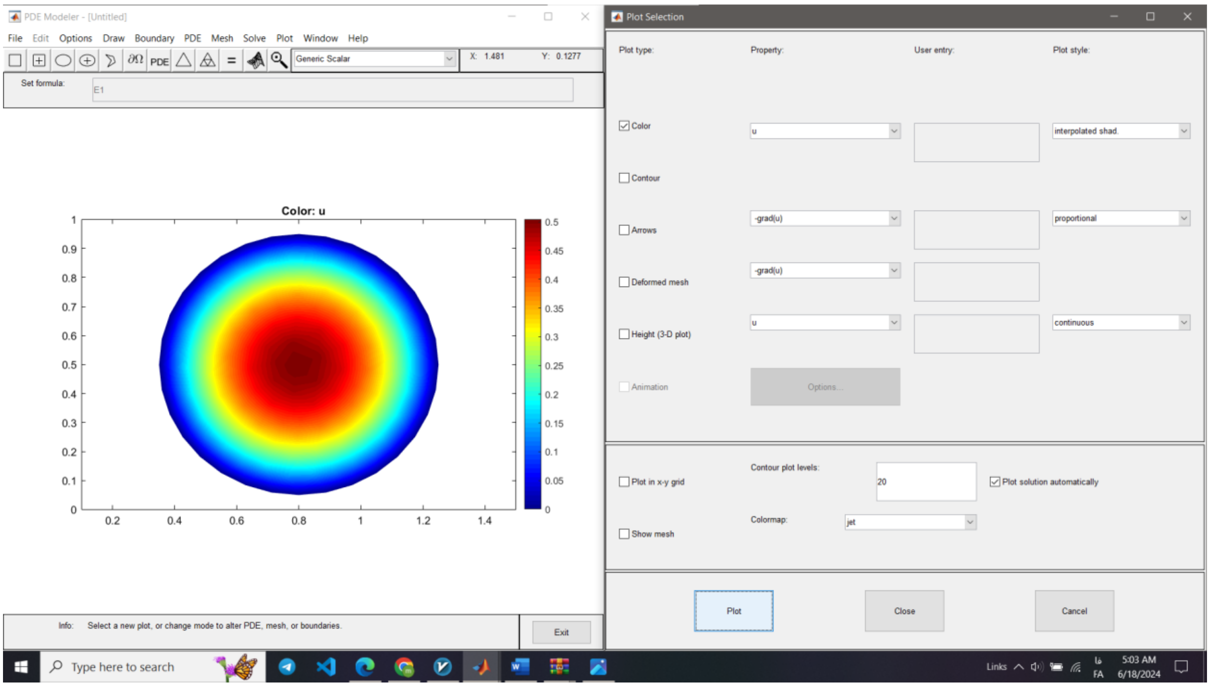
-با مراجعه به بخش **Mesh Initialize-Mesh** شرایط اولیه مش را تعیین کردم:



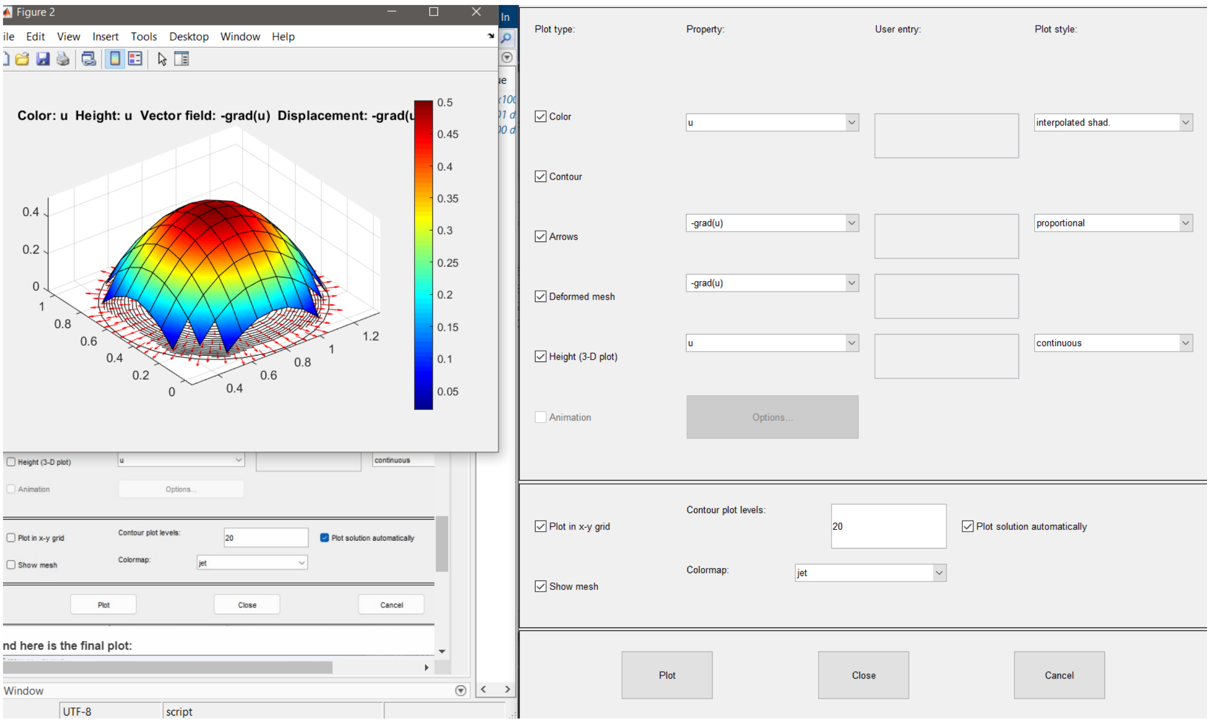
-حال معادله را حل کردم و شکل موج را رسم کردم:



و با رنگ های دیگر:



و یا سه بعدی و با نمایش مش ها و این تنظیمات:

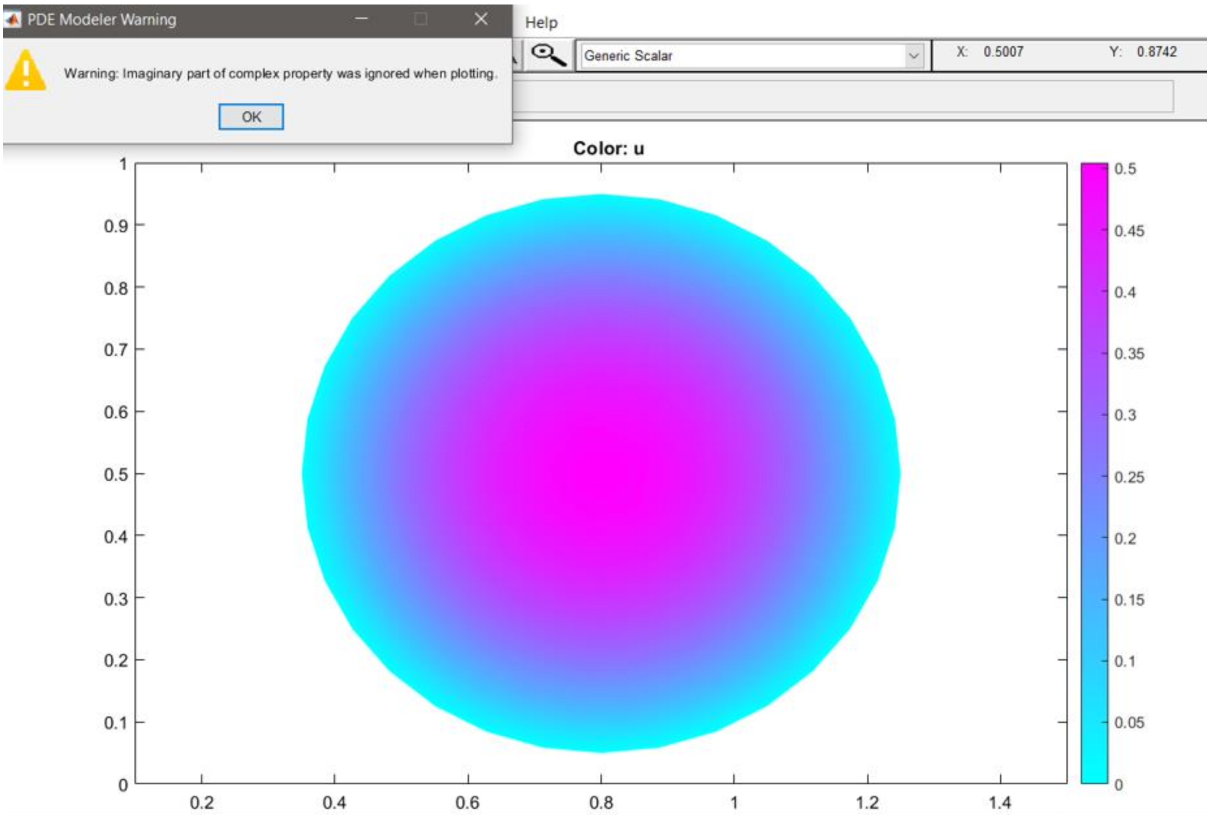


– همچنین رفتار موج را توجیه میکنیم:

این نمودار، پاسخ معادله ی انتشار صوت را نشان می دهد و همانطور که مشاهده میکنید در شکل ها، در نزدیکی منبع صوت (مرکز دایره) شدت آن زیاد است و وقتی از منبع دور میشویم شدت آن کم و کمتر میشود.

توضیحات شکل نهایی:

- دایره‌ی رسم شده :این دایره ناحیه‌ی حل معادله هلمهولتز را مشخص می‌کند. شرایط مرزی نیومن روی این دایره اعمال شده است.
- شرایط مرزی :شرایط مرزی نیومن به این معنی است که مشتق uنسبت به نرمال سطح دایره ثابت است. این شرایط معمولاً نشان‌دهنده‌ی بازتاب موج از مرز است.
- پاسخ :رنگ‌های مختلف در تصویر نشان‌دهنده‌ی مقادیر مختلف uهستند. همانطور که مشاهده می‌شود، توزیع رنگ‌ها به صورت دایره‌ای و هم‌مرکز است که نشان‌دهنده‌ی موج‌های کروی است که از مرکز دایره به سمت خارج منتشر می‌شوند.
- بخش موهومی u:اخطار نمایش داده شده نشان می‌دهد که بخش موهومی uنادیده گرفته شده است. این موضوع معمولاً به دلیل حل معادله به صورت عدد مختلط است که می‌تواند نشان‌دهنده‌ی موج‌های تضعیف شده یا انتقال یافته باشد.



فایل نهایی دریافت شده از pdeModeler (Q2.m)

```matlab
% This script is written and read by pdetool and should NOT be edited.
% There are two recommended alternatives:
% 1) Export the required variables from pdetool and create a MATLAB script
%    to perform operations on these.
% 2) Define the problem completely using a MATLAB script. See
%    https://www.mathworks.com/help/pde/examples.html for examples
%    of this approach.
function pdemodel
[pde_fig,ax]=pdeinit;
pdetool('appl_cb',1);
set(ax,'DataAspectRatio',[1 1.0714285714285714 1]);
set(ax,'PlotBoxAspectRatio',[1.4999999999999998 1 2.1428571428571428]);
set(ax,'XLim',[0.10000000000000001 1.5]);
set(ax,'YLim',[0 1]);
set(ax,'XTickMode','auto');
set(ax,'YTickMode','auto');

% Geometry description:
pdeellip(0.80000000000000004,0.5,0.45000000000000001,0.45000000000000001,...
0,'E1');
set(findobj(get(pde_fig,'Children'),'Tag','PDEEval'),'String','E1')

% Boundary conditions:
pdetool('changemode',0)
pdesetbd(4,...
'dir',...
1,...
'1',...
'0')
pdesetbd(3,...
'dir',...
1,...
'1',...
'0')
pdesetbd(2,...
'dir',...
1,...
'1',...
'0')
```

```
pdesetbd(1,...
'dir',...
1,...
'1',...
'0')

% Mesh generation:
setappdata(pde_fig,'Hgrad',1.3);
setappdata(pde_fig,'refinemethod','regular');
setappdata(pde_fig,'jiggle',char('on','mean',''));
setappdata(pde_fig,'MesherVersion','preR2013a');
pdetool('initmesh')

% PDE coefficients:
pdeseteq(1,...
'1.0',...
'0.0',...
'10',...
'1.0',...
'0:10',...
'0.0',...
'0.0',...
'[0 100]')
setappdata(pde_fig,'currparam',...
['1.0';...
'0.0';...
'10 ';...
'1.0'])

% Solve parameters:
setappdata(pde_fig,'solveparam',...
char('0','1000','10','pdeadworst',...
'0.5','longest','0','1E-4','','fixed','Inf'))

% Plotflags and user data strings:
setappdata(pde_fig,'plotflags',[1 1 1 1 1 1 1 1 0 0 0 1 1 0 0 0 0 1]);
setappdata(pde_fig,'colstring','');
setappdata(pde_fig,'arrowstring','');
setappdata(pde_fig,'deformstring','');
```

setappdata(pde_fig,'deformstring','');

```
setappdata(pde_fig,'heightstring','');

% Solve PDE:
pdetool('solve')
```