

Calculation of Maxwell's Equations Using MATLAB

Subhi Abdalazim Aljily Osman¹, Abdel Radi Abdel Rahman Abdel Gadir Abdel Rahman², Abubaker Abdalla Mustafa³, Musa Adam Abdullah⁴

¹Department of Mathematics, Faculty of Computer Science and Information Technology, University of ALBUTANA, Sudan.

²Department of Mathematics, Faculty of Education, Omdurman Islamic University, Omdurman, Sudan.

³Department of Mathematics, Faculty of Education, Dalanj University, Sudan

⁴Department of Mathematics, College of Computer Science and Information Technology, University of the Holy Quran and Tassel of Science, Sudan

ARTICLE INFO

Published Online:
05 October 2021

Corresponding Author:
Dr. R. Sivaraman

ABSTRACT

Maxwell's equations describe electromagnetic Phenomena. This includes micro- , radio and radar waves .The Maxwell equations are discussed in more detail Faraday's and Amperes laws constitute a first - order hyperbolic system of equations .Matlab is one of the most famous mathematical programs in calculating mathematical problems .The aims of this study is to calculate Maxwell's equations using Matlab .We followed the applied mathematical method by using Matlab .We found that the solution of Matlab is more accuracy and speed than the analytical solution.

KEYWORDS: Calculation , Maxwell's Equations , electromagnetic Phenomena , Matlab

1. INTRODUCTION

Maxwell's equations are a system of vector partial differential equations describing how electricity and magnetism interact and behave in a material. Developed in the 19th century by physicist James Clerk Maxwell, they are the culmination of hundreds (if not thousands) of years' study, experimentation and thought on the relationship between the phenomena of electricity and magnetism. First considered the broadcast attacks against the lattice-based Public-key cryptosystems and also gave some heuristic attacks. However, they showed that NBTU may resist their broadcast attacks, since half of its "message" is random. Very recently, Ding proposed an ingenious algorithm to solve the problem with bounded errors in polynomial time. We found that it can be easily used to save broadcast attack against as we know, it is the first broadcast attack against. We have to point out that some of her lattice-based cryptosystems such as [12] PP337. Matlab stands for Matrix laboratory and was written originally to provide easy access to matrix software developed by the linpack (linear system package) and (Eigen System Package) projects [2] PP 1.

2. PARTIAL DIFFERENTIAL EQUATIONS MODELS

Partial Differential Equations describe many of the fundamental natural phenomena. Examples include Cludelalace's equation for steady state heat conduction, the advection-diffusion equation for pollutant transport, Maxwell's equation for electromagnetic waves, the Navier-Stokes equations for fluid flow and many, many more. The authors' main interest is in solving partial Differential Equations for fluid flow problems and details.[3] pp11.

3. SOLUTION OF A PARTIAL DIFFERENTIAL EQUATIONS

Calculating a Partial Differential Equation means finding the unknown function u . An analytical solution of a partial Differential Equation is a function that satisfies the partial Differential Equations and also satisfies any boundary and/or initial conditions given with the partial Differential Equation. Most partial Differential Equations of interest do not have analytical solutions; a numerical procedure must be used to find an approximate solution. The approximation is made at discrete values of the independent variables and other terms in the partial Differential Equation by approximations. After some

manipulation la finite difference scheme is created from which the approximate solution is obtained [3] PP10.

4. MAXWELL'S EQUATIONS

Maxwell's equations provide a mathematical toolkit through which the response of material to electric and magnetic fields can be understood , modeled and predicted - virtually all electronic technology developed today owes its existence to this set of four equations from radio transmissions to computing Astronomy , engineering , Computer science and physics are just four examples of entire fields that use Maxwell's equations to help understand electromagnetic wave propagation in different frequency ranges . In short, they are a cornerstone of classical physics.

In the 150+ years since their discovery, computing technology has advanced quite considerably - computational electromagnetic is a field that combines Physics, mathematics and computer Science to use computational techniques to solve problems described by Maxwell's equations. Doing so involves developing and using highly efficient classes of approximations to Maxwell's equations since in most cases it is impossible to find exact solutions. These classes of methods are called numerical methods, and their study is the field of Numerical Analysis .Two such numerical methods, finite difference methods and finite element methods, be examined in this paper [5] PP7.

5. COMPUTATIONAL ELECTROMAGNETIC

The Maxwell equations were first formulated by James Cleris Maxwell .They are:

$$\begin{aligned} .D &= \rho \text{ (Gauss law)} \\ \nabla \cdot \mathbf{B} &= 0 \text{ (Gauss law)} \\ \partial \mathbf{B} / \partial t &= - \nabla \times \mathbf{E} \text{ (faraday's law)} \\ \partial \mathbf{D} / \partial t &= \nabla \times \mathbf{H} - \mathbf{J}_e \text{ (Ampere's law)} \end{aligned}$$

where E is the electric Field , D is the electric flux density , H is the magnetic field.

B is the magnetic flux density Jo the electric current density and p is the charge density .The two Gauss laws can be derived from Faraday’s and Ampere’s laws Provided that the initial conditions fulfill the Gauss ' laws .These equations are linear and it may hence appear to be rather easy to solve them analytically . However boundary and interface conditions malse the Maxwell equation hard to solve analytically , they can be solved analytically only for a few very simple shapes such as a sphere or an infinite circular cylinder - Hence one has to rely on a mix of experiments and approximative and or numerical methods . Numerical methods for the Maxwell equations are usually referred to as computational electromagnetic .Experiments and CEM complement each other when developing a product . An advantage of numerical methods is that they make it possible to test a large number of different constructions without actually

building them .Is also useful when experiments are difficult and /or dangerous to perform - such an example is a lightning strike on an aricraft in flight .the fast variations in the electromagnetic fields make it a challenge to construct numerical methods for the Maxwell equations. [11] PP1.

6. MAXWELL'S EQUATIONS IN LINEAR DIELECTRICS

Maxwell's Equations in a material $\Omega \subset \mathbb{R}^3$ are a system of vector Partial differential equations we gives. Maxwell 's E equations

$$\begin{aligned} \text{Faraday's law } \nabla \times \mathbf{E} &= - \partial \mathbf{B} / \partial t \quad (1) \\ \text{Maxwell-Ampere law } \nabla \times \mathbf{H} &= \mathbf{J} + \partial \mathbf{D} / \partial t \quad (2) \\ \text{Gauss's law } \nabla \cdot \mathbf{D} &= \rho \quad (3) \\ \text{Gauss -Poisson law } \nabla \cdot \mathbf{E} &= \rho / \epsilon \quad (4) \end{aligned}$$

In system(3-D,D) represents the electric Flux density E represents the electric Field, B represents the magnetic Flux density and H represents the magnetic field .

All field variables are functions of time t and space $\mathbf{X}=(x,y,c) \in \mathbb{R}^3$ - the current density J and the charge density pare related by the continuity equation.

$$\nabla \cdot \mathbf{J} + \partial \rho / \partial t = 0 \quad (5)$$

These classical equations provide a total of twelve unknown equations but only eight equations in which to calculate for Them, meaning that to find concrete calculation we must append to Maxwell's equations several other Physical laws governing the response of the material, called the. constitutive laws .Constitutive laws govern the behavior of electromagnetic waves in different media i for Linear dielectric media, such as a vacuum, the constitutive laws that Pertain to Maxwell's equations are

$$\begin{aligned} \mathbf{D} &= \epsilon \mathbf{E} \quad (6) \\ \mathbf{H} &= \mu \mathbf{J} \quad (7) \end{aligned}$$

with the material- dependent Parameters E, the frequency independent electric permittivity and s the frequency-independent magnetic Permeability. we consider linear dielectrics in which ϵ and μ are constauts- In metamaterials, these are not constants, but instead functions of the angular frequency of aware Propagating Through the material. When substituted into Equations the resulting pair of equations relates E and H directly as seen. Through the equations :

$$\begin{aligned} \text{Faraday's law } \nabla \times \mathbf{H} &= \mathbf{J} + \partial \mathbf{D} / \partial t \quad (8) \\ \text{Maxwell's Ampere law } \nabla \times \mathbf{E} &= - \partial \mathbf{B} / \partial t \end{aligned}$$

These are called the curl equations of the Maxwell system form we can also glean two conditions that must be satisfied for these adapted forms , namely that .

$$\begin{aligned} \nabla \cdot \mathbf{E} &= \rho \\ \nabla \cdot \mathbf{H} &= 0 \end{aligned}$$

where is some given charge density .These conditions , combined with system are then in a form with which we can describe the behavior of the electric and magnetic fields in a

linear dielectric when ϵ and μ are constants. Given some initial conditions ϵ and μ and additional boundary conditions, we can then simulate the changes in the system from the initial state as it moves through time, modeling the propagation of electromagnetic waves in the dielectric.

In the case when ϵ and μ are frequency dependent additional equations have to be added to the Maxwell system. This happens in the case of met materials and is discussed we consider problems in one spatial dimensions. Sines rise to two separate decoupled systems of three equations each called the transverse Electric and Transverse Magnetic modes [5]PP 9.

7. MATLAB

A Function in its Graph y ():

This function is used to set initial values for the appropriate screen mode for graphic mode. Seams the screen where there are two modes (graphich).

For the screen, which are graphics mode and text Mode. The general formula for this function is:
Void initgraph (cint far & graph driver , int for a graph modechar far * Pathto driver):

Get Drive Name Function ()

This function is used to return the name of the currently loaded trigger, meaning a pointer to the current drawing operator returns the general formula for this function.

char * get drive name (void) :

Example (7.1):

If we wanted to know what kind of device's current drawing driver.

Solution :

We write the following sentences .
char * driver name;
initgraph (&gd,&gm"C: //TC //BGI ") ;
drivename = get driver name ;
outtextxy (200 , 200 , driver name) ;

Graph Result Function ():

This function is used to get the error code after calling the initsgraph function and the graph errors function the graph result function returns the error code for any drawing process that has an error and the graph error msg function takes the error code.

If no error accrues in the drawing process.
gokwasreturned.

In order to get an error, delete the BGI: file.
from the current directory or BGI folder from the settings Path , we get an error message(EGAVGA - BGI) device drive file not found.
interrorcode = graphresult ();
if(errorcode !=gok){
printf (graph errormsg (error code) ;
getch();
return0; } [9] PP 3

8. CALCULATION OF MAXWELL'S EQUATIONS USING MATLAB

i. Maxwell's Law Example Code .m:

clearall

The parameters for this function are get address ism address , and the path where the graphic files are located on the hard disks of your device to start the drawing system you must call the initial values for the drawing system with the graphics.h graphics library file and changes them to the drawing screen by leading the drawing player graphic drive from disks and put the system in graphic mode and BGIfile in the program folder - it works to return all graphic settings to their default position (color , current location , Palette , view Point) - Normally setting graph result to Zero , which is the first step we need in order to write programs draw.

We need to pass the variables gd ,gm to the function and usually the following values are used gd= VGA and gm = VGAHL for drawing . The coordinates of this type of screen G40 X 480 pixels.

The detectgraph () function can be used to final the values of gmandgd. After that the device is able to receive various drawing commands

“Calculation of Maxwell’s Equations Using Matlab”

```
clc
```

```
x = linspace(-4,4,100);  
y = linspace(-4,4,100);  
z = linspace(-4,4,100);
```

```
[X, Y, Z] = meshgrid(x, y, z);
```

```
Ex = sin(2*pi*(Z/3));
```

```
Ey = 0*X;
```

```
Ez = 0*X;
```

```
[Bx, By, Bz] = curl(X, Y, Z, Ex, Ey, Ez);
```

```
for k=1:100
```

```
E(k) = mean(mean(Ex(:,:,k),1),2);
```

```
B(k) = mean(mean(By(:,:,k),1),2);
```

```
end
```

```
plot3(0*x, y, E, 'red', 'LineWidth', 2); hold on
```

```
quiver3(0*x(1:3:100), y(1:3:100), 0*z(1:3:100), 0*x(1:3:100), 0*y(1:3:100), E(1:3:100), 0, 'r', 'lineWidth', 1.5);
```

```
hold on
```

```
plot3(B, y, 0*z, 'g', 'LineWidth', 2);
```

```
quiver3(0*x(1:3:100), y(1:3:100), 0*z(1:3:100), B(1:3:100), 0*y(1:3:100), 0*z(1:3:100), 0, 'g', 'lineWidth', 1.5);
```

```
grid on, axis square
```

```
set(gca, 'FontSize', 15, 'lineWidth', 2);
```

```
xlabel('x');
```

```
ylabel('y');
```

```
zlabel('z');
```

```
legend('Electric Field','Electric Field Vector','magnetic Field','magnetic Field vector')
```

Graph Results

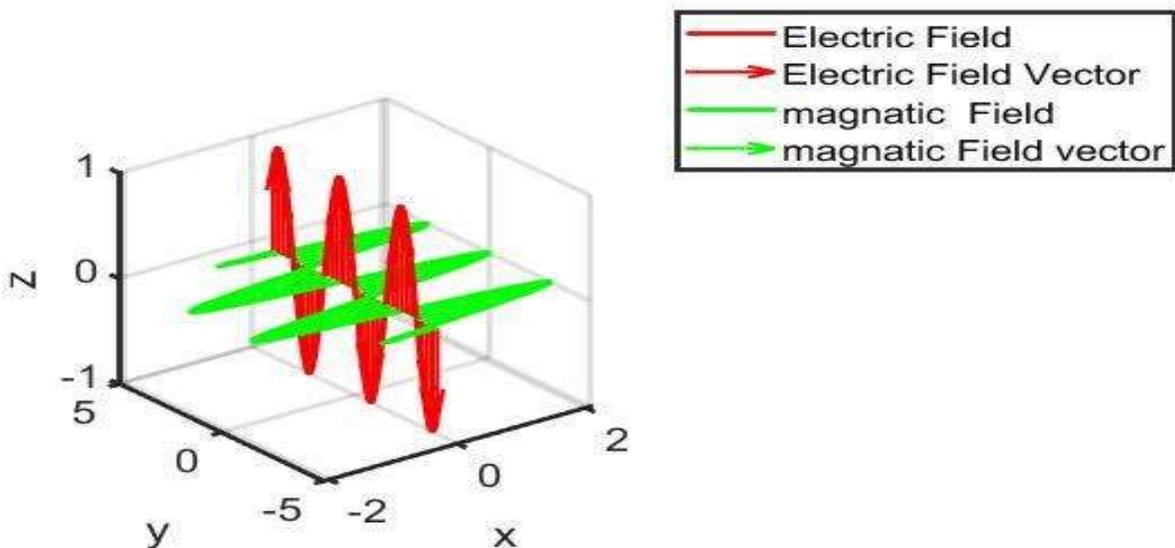


Fig1: Electromagnetic and electrical filed using Maxwell law

ii. Ampere’s Law Example Code .m:

```
mu0 = pi*4E-7;
AmpereB = @(m0u,I,r) (mu0.*I)./(2*pi*r);
radius = linspace(1E-3, 1);
I = 1;
figure(1)
semilogy(radius, AmpereB(mu0,I,radius))
grid
xlabel('r (m)')
ylabel('B')
```

Graph Results:

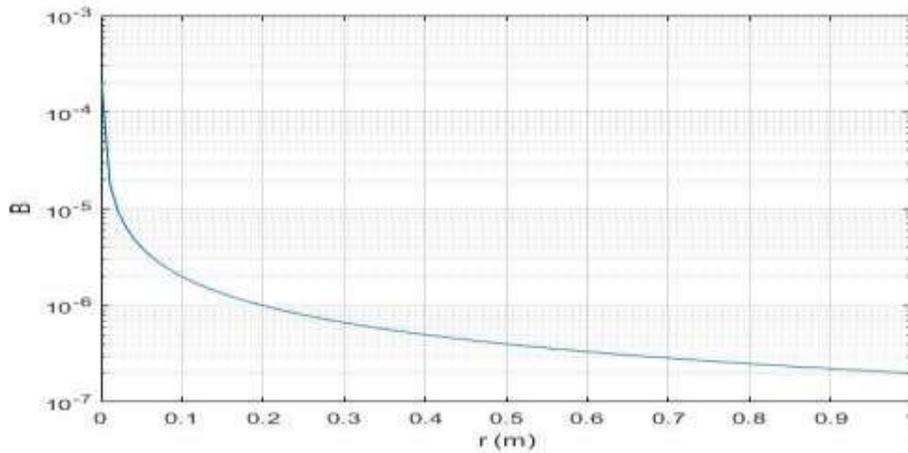


Fig2: Relation between B factor and radius of magnetic field using ampere law

iii. Faraday Law Example Code .m:

```
J = 500;
N = 20;
pi = 4 * atan(1);
f = 1;
omega = 2 * pi * f;
k = 2 * pi;
c = omega / k;
lambda = c / f;
a = 3 * lambda;
dx = a / J;
dt = dx / c;
x = [0:dx:a];
axis("manual",[0,a,-2,2])
t = 0;
u0 = sin(omega*t-k*x);
v0 = omega * cos(omega*t-k*x);
u = sin(omega*t-k*x);
plot(x,[u;u0]);
t = dt;
u1 = u0 + dt * v0;
u = sin(omega*t-k*x);
plot(x,[u;u1]);
```

```
for n=1:N
t = t + dt;
u = sin(omega*t-k*x);

for j=2:J
u2(j) = u1(j+1) + u(j-1) - u0(j);
end;
u2(1) = u1(2) + u(J) - u0(1);
u2(J+1) = u2(1);
plot(x,[u;u2]);
u0 = u1;
u1 = u2;
end;
```

Graph Results

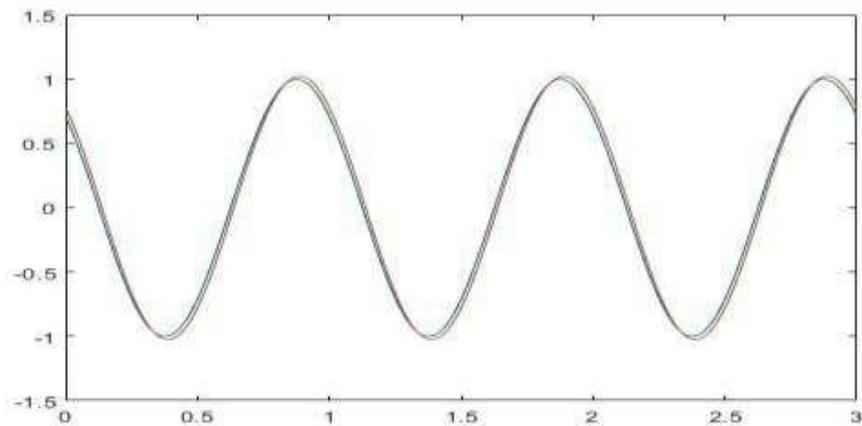


Fig3: Electromagnetic wave using Faraday law

Magnetic Field for One Electrical Cable and Discussion:

With the Matlab script physics.m, the magnetic field in the XY-plane ($z = 0$) in the center of a set of 1 to 4 conductors, which are aligned in the Z-direction, can be determined, the wire goes to zero, the strength of the magnetic field goes towards infinity. This can create problems when using numerical methods to compute the magnetic field and the spatial derivatives of the magnetic field.

The Parameters used in the Modeling were entered in the Input Section of the Code:

XY coordinates and currents for the 4 cables .path for calculating the line integral and for calculating the magnetic flux and XY coordinates for calculating the divergence and curvature at the next grid point.

```
Nwire = 1;
I = [10,0,0,0];
xW = [0.00, 0.005, -0.005, 0.002];
yW = [0, -0.0050, 0.005, 0.008];
minZ = -5; maxZ = 5;
nZ = 4101;
radiusW = 0.001;
```

```
N = 401;
L = 0.005*2;
minX = -L; maxX = L;
minY = -L; maxY = L;
minR = 1e-3;
```

“Calculation of Maxwell’s Equations Using Matlab”

```
L = 3e-3;  
minLx = -L; maxLx = L;  
minLy = -L; maxLy = L;
```

```
xD = 0.05e-3;  
yD = 0.05e-3;
```

A [2D] grid is created using the mesh grid function to calculate the distances of each point on the grid from the origin. The XY positions of each wire are adjusted so that the minimum distance x or y between a grid point and a wire position is not less than $dx / 2$ or $dy / 2$, where dx and dy are the grid distances.

```
mu0 = 4*pi*1e-7;  
K = I .* (mu0/(4*pi));
```

```
Bsat = 2e-3;  
zW = linspace(minZ, maxZ, nZ);  
dLz = zW(2)-zW(1);  
dLx = 0; dLy = 0;  
dL = [dLxdLydLz];
```

```
x = linspace(minX,maxX,N);  
y = linspace(minY, maxY,N);  
dx = x(2)-x(1);  
dy = y(2)-y(1);
```

```
[xG, yG] = meshgrid(x,y);  
dx = x(2)-x(1); dy = y(2)-y(1);  
RG = sqrt(xG.^2 + yG.^2);  
for n1 = 1 : Nwire  
for n2 = 1 : N  
if abs(x(n2) - xW(n1)) < dx/2 ; xW(n1) = x(n2) + dx/2; end  
if abs(y(n2) - yW(n1)) < dy/2 ; yW(n1) = y(n2) + dy/2; end  
end  
end
```

The components B_x, B_y, B_z of the magnetic field are calculated at each point of the grid according to BiotSavart's law. If the strength of the magnetic field is greater than a saturation value, the saturation value is assigned to the magnetic field at this point on the grid.

```
Bx = zeros(N,N); By = zeros(N,N); Bz = zeros(N,N);
```

```
for n = 1 : Nwire
```

```
for m = 1 : nZ
```

```
Rx = xG - xW(n);
```

```
Ry = yG - yW(n);
```

```
Rz = 0 -zW(m);
```

```
R = sqrt(Rx.^2 + Ry.^2 + Rz.^2);
```

```
R3 = R.^3;
```

```
Bx = Bx + K(n) .* (dLy .* Rz - dLz .* Ry) ./ R3;
```

```
By = By + K(n) .* (dLz .* Rx - dLx .* Rz) ./ R3;
```

```
Bz = Bz + K(n) .* (dLx .* Ry - dLy .* Rx) ./ R3;
```

```
end
```

```
end
```

```
B = sqrt(Bx.^2 + By.^2 + Bz.^2);
```

“Calculation of Maxwell’s Equations Using Matlab”

The magnetic field is only calculated in a nearby area and in the center of a finite wire. This gives a good approximation of an infinite extension cord and gives the expected result B_z and near the middle of the cord, B_x and B_y are independent of the Z value.

The XY circulation path is defined from the input parameters, which correspond to a rectangle, so that the line integral of equation 2 is only evaluated in the X or Y direction. The numerical calculation of the circulation results in an estimate of the current enclosed by the rectangular path.

```
X1 = find(x>minLx,1);X2 = find(x>maxLx,1);
Y1 = find(y>minLy,1);Y2 = find(y>maxLy,1);
if mod(X2-X1,2) ~= 0;
    X2 = X2+1; end;
if mod(Y2-Y1,2) ~= 0; Y2 = Y2+1; end;
a = x(X1); b = x(X2); fn = Bx(Y1,X1:X2);
BdL(1) = simpson1d(fn,a,b);
a = y(Y1); b = y(Y2); fn = By(Y1:Y2,X2)';
BdL(2) = simpson1d(fn,a,b);
a = x(X1); b = x(X2); fn = -Bx(Y2,X1:X2);
BdL(3) = simpson1d(fn,a,b);
a = y(Y1); b = y(Y2); fn = By(Y1:Y2,X1)';
BdL(4) = -simpson1d(fn,a,b);
Ienclosed = sum(BdL)/mu0;
a = y(Y1); b = y(Y2); fn = Bx(Y1:Y2,X2)';
phi(1) = simpson1d(fn,a,b);
a = y(Y1); b = y(Y2); fn = -Bx(Y1:Y2,X1)';
phi(2) = simpson1d(fn,a,b);
a = x(X1); b = x(X2); fn = By(Y2,X1:X2);
phi(3) = simpson1d(fn,a,b);
a = x(X1); b = x(X2); fn = -Bx(Y1,X1:X2);
phi(4) = simpson1d(fn,a,b);
phi_total = sum(phi);
% Divergence divB and Curl curlB -----
Nx = find(x > xD,1);
Ny = find(y > yD,1);
divB = divergence(xG,yG,Bx,By);
min_divB = min(min(divB));
max_divB = max(max(divB));
curlB = curl(xG,yG,Bx,By);
min_curlB = min(min(curlB));
max_curlB = max(max(curlB));
dBdx = (Bx(Ny,Nx+1) - Bx(Ny,Nx-1))/(x(Nx+1)-x(Nx-1));
dBdy = (By(Ny+1,Nx) - By(Ny-1,Nx))/(y(Ny+1)-y(Ny-1));
divBp = dBdx+dBdy;
curlB_1 = mu0*I(1)/(dx/2)^2;
```

The integral is calculated with the simpson1d function. When using this function, the function must have an odd number of elements. The Orbit results in a closed area of unit height in the Z-direction, through which the entire magnetic flux is calculated.

The divergence and curvature of the magnetic field are calculated at a point on the grid that is closest to the point X_D, Y_D , indicated in the input section of the code. The calculations may not be accurate for the points near the cables.

Experiment: Number of wires = 1, x location of wires [mm] =0.2, y location of wires [mm] =0.02

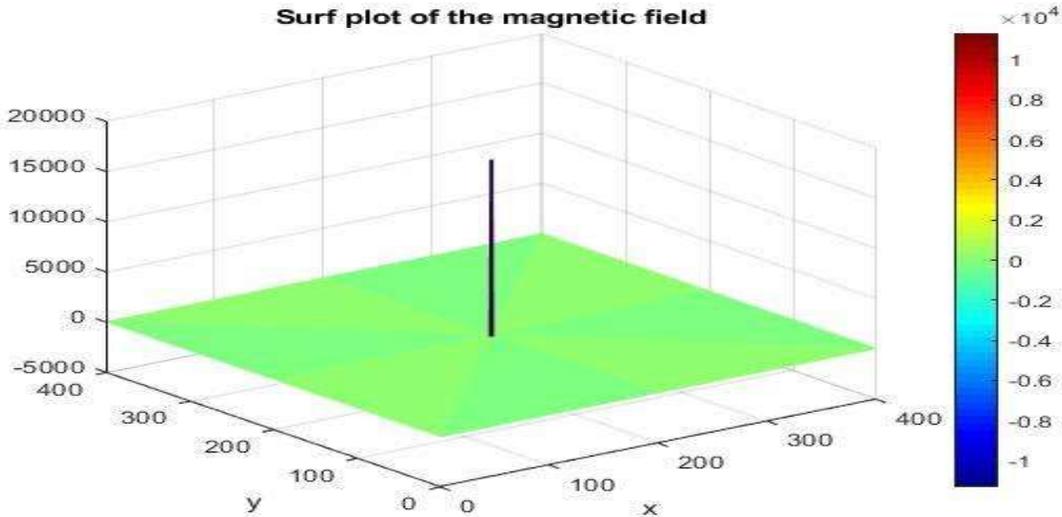


Fig4: The curl of the magnetic field is zero at all points except at the location of the wire

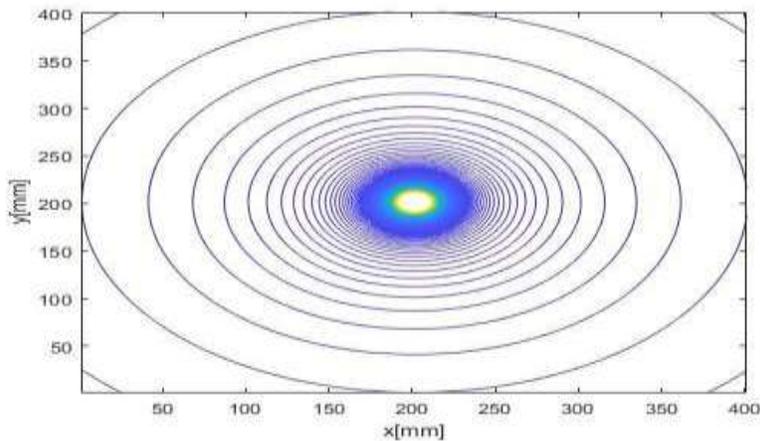


Fig4: Contourf plot of the magnetic field

9. RESULTS

Through paper we found the following results :Possibility of calculating Maxwell's equations by Matlab and we discussed three types of calculating of Maxwell's equations and we indicated to some applications of Maxwell's equations.

REFERENCES

1. DANIEL FIEis'ch, Astudent's Guide to Maxwell equations wittenbery university, First additiom the united states of A merica by cambridgeuniversity, New york, 2008 pp .
2. David Houcque ,introduction to Matlab for EnGineering STUDENTS, North westem university ,version 1.2 August 2005,pp.
3. D-M causonand-caMingham, Introduction finite Differnee Methods for 1Partial Differential equations ventus publishing APS 2010 PP10– 11.
4. Hassan Mustafa, Al- Ouneeidi , Partial Differential equations Part two, Al-Rashed library, Riyadh, king dom of saudi Arabia, (2005), PP1426.
5. Iseacstalleup, Finite Element and finite Difference Methods of Maxwell's Equations is Matamaterial so regon state university, Honors callege, second Edition , March 16 2018 PP7-9.
6. Maha Awwad Al-kubaisi, Parital Differential Eguation somarpagmukhtar university, publications libyalibya 2005 , PP11 .
7. Math work-using Matlab -versin6- 2001 chapter 15 PP84.
8. SorenBotTeher -Solving oDE and pDE in Matla b SESE, 1a12 1 2009 ,PP19.
9. TarigRahale scientific guide for the computer Drawing Course 1495 Al- Guds Open university, (2014-20152 adition, ,346.PP3.

“Calculation of Maxwell’s Equations Using Matlab”

10. T,Plantared and w-susilo- Broadcast attacks against lattice-based cryp- tosystems. (ACNS 2009) 7 addition PP456 .
11. UIF Andersson time- BomainMethodls for the Maxwell Equations –universities –ervice USAB ,stockholm 2001, PP1-2.
12. Y - Pan, ydeng, y-giangand,ANewlattice Based caryptosystem Mixed with aknapsack. Cryptologyeprint Arehive Report (2009 , 337 avariable at hettep//eprint. iacr.orgm,2009,PP337.