

Final Exam for Group B (20 Marks)

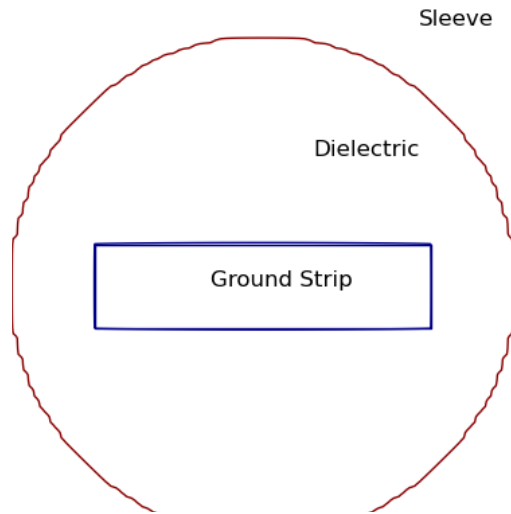
Nov 10, 2015

Dept. of Electrical Engineering, IIT Madras

CADLAB July 2015 session

- ▷ **Time duration of exam is two hours**
- ▷ **vector operations are a must or lose lots of marks!!**
- ▷ **Label all plots. Add legends. Make the plots professional looking.**
- ▷ **Comments are not optional. They are required. Note that I expect L_AT_EX discussion and python comments.**
- ▷ **pseudocode should be readable and neatly formatted.**
- ▷ **code should be written as part of a L_AT_EX file.**
- ▷ **Please download the blank L_AT_EX file with noweb enabled. This is compatible with earlier L_AT_EX versions, and should work on all machines.**
- ▷ **L_AT_EX file should be named *your-roll-number.lyx***
- ▷ **I should be able to extract the python code using the view menu.**
- ▷ **Include the plots in the lyx file and generate a pdf.**
- ▷ **Internet will be turned off at the beginning of the exam.**
- ▷ **Zip the L_AT_EX file and the pdf file together and name it as *your-roll-number.zip*. Leave it in the home directory of your machine and show the same to the TA before leaving.**
- ▷ **Late submission will result in reduced marks.**

A circular copper grounding strip is connected to a copper sleeve as shown in the figure. The region between the two is a dielectric whose conductivity is zero. We wish to determine the potential between the strip and the sleeve if a voltage of 1 volt is applied to the sleeve.



The grounding strip is 1 cm wide and 0.25 cm thick. The sleeve has a radius of 0.75cm. The equation to solve is

$$\nabla^2 \phi = 0$$

The finite difference equation for this problem is

$$\phi_{mn} = \frac{1}{4} (\phi_{m+1,n} + \phi_{m-1,n} + \phi_{m,n-1} + \phi_{m,n+1})$$

We simplify the model to a square block that is the sleeve with a circular hollow region. The sleeve is held at 1 volt. In the middle is the ground strip, held at 0 volts.

1. Define your simulation region, and the number of grid points, N_x and N_y , you are choosing along each direction. Initialize ϕ to 0.5 volts Note: Use $N_x = 31$ by $N_y = 31$ grid for simulation with the grounding strip being 16 grids long and 4 grids in thickness. The radius of the sleeve is 12 grids.

2. Define x and y coordinates with the origin at $(15,15)$. Use `meshgrid` to create arrays X and Y .
3. Use the `where` command to find the indices corresponding to 1 volt. Call the list of indices `ii`. Set those grid points to 1 volt. Hint: use X and Y in your `where` command.
4. Use the `where` command to find the indices corresponding to 0 volts. Call list of indices `jj`. Set those grid points to zero volts.
5. In a separate figure, create a contour plot of the initial potential, with 11 levels. Label appropriately.
6. Create a function that takes a number N and iterates the solution for N iterations. In each iteration, iterate the entire square region, and then impose the two electrode voltages. Vectorise the iteration loop, and obtain the error at each iteration. Run initially for 200 iterations.
7. Plot the error vs iteration number. Fit the error to an exponential fit and determine the residual error. Predict where to stop the iteration when you have 0.1% residual error.
8. Run the code for the predicted number of iterations and save the solution in `phi1`. Run the code for 5 times longer and save the solution in `phi2`. Assume that `phi2` is exact, and obtain the error in `phi1`. Is it 10^{-3} ? i.e., Have you estimated the initial error and the slope of the error curve correctly in q7?
9. Plot the contours of the error potential, given by

$$\Delta\phi = \text{phi1} - \text{phi2}$$

after the predicted number of iterations in earlier contour figure. Where is the error maximum? Discuss why the error is maximum there. Compare to the lab resistor problem.

Notes:

- 1. In L_YX, write pseudocode showing how you will solve the problem. The individual pieces of code can be under each line of pseudocode.**
 - (a) What will your data structures be?**
 - (b) How will you find the sleeve points and the strip points?**
 - (c) Iteration loop**
 - (d) Error estimation to find stopping index**
- 2. Enter the python code in L_YX and verify that you can extract it from L_YX. If this does not work, include the python code as L_YX-Code and mention at the top that python extraction did not work. Instead include the python code itself in the zip file.**
- 3. Include the figures generated by the python program, with appropriate captions.**
- 4. Have a discussion section where you discuss the results.**

Useful Python Commands (use “?” to get help on these from ipython)

```
from pylab import *
import system-function as name
Note: lstsq is found as scipy.linalg.lstsq
ones(List)
zeros(List)
range(N0,N1,Nstep)
arange(N0,N1,Nstep)
linspace(a,b,N)
logspace(log10(a),log10(b),N)
X,Y=meshgrid(x,y)
where(condition)
where(condition & condition)
where(condition | condition)
a=b.copy()
lstsq(A,b) to fit  $Ax=b$ 
A.max() to find max value of numpy array
A.astype(type) to convert a numpy array to another type (eg int)
def func(args):
    ...
    return List
matrix=c_[vector,vector,...] to create a matrix from vectors
figure(n) to switch to, or start a new figure labelled n
plot(x,y,style,...,lw=...)
semilogx(x,y,style,...,lw=...)
semilogy(x,y,style,...,lw=...)
contour(x,y,matrix,levels...)
xlabel(label,size=)
ylabel(label,size=)
title(label,size=)
legend(List) to create a list of strings in plot
annotate(str,pos,blpos,...) to create annotation in plot
grid(Boolean)
show()
```