Solving Laplace's Equation for Various Transmission Line Structures Using Finite Element Method

Xiaojun Zhu

November 12, 2021

Abstract

In this project, we solved Laplace's equation for three naturally or artificially shielded transmission line structures that support TEM or quasi-TEM modes using the finite element method (FEM). All the simulations were performed in 2D and the triangular mesh cells generated using Matlab tool were used in the FEM. The FEM was validated by comparing the numerical potential distribution in the coaxial line to the corresponding analytical solution. The equipotential lines and static electric field distribution from the presented FEM were plotted and compared to that from the numerical solver in Matlab. Though limitations exist, the good agreement between the numerical and analytical results (or results from another numerical solver in Matlab) shows that the implementation of the FEM was successful and the simulation was accurate enough.

1 Introduction

Transverse electromagnetic (TEM) waves refer to the modes without transverse fields. TEM waves can only exist in a transmission line that has two or more conductors [1]. Typical transmission lines that offer TEM or quasi-TEM modes are coaxial line, microstrip line, stripline, coplanar waveguide, grounded coplanar waveguide, slotline, etc. For these transmission lines, the fields may be derived from a 2D static potential distribution. For example, in a coaxial line where the inner conductor is at a potential of V and the outer conductor is grounded, the fields can be derived from a scalar potential distribution which is a solution to Laplace's equation [1].

Although an analytical solution for the 2D Laplace's equation may be easily obtained for some structures (i.e. coaxial line), for certain structures (i.e. microstrip) evaluating the potential and field distribution may require numerical methods. Both the finite difference method (FDM) and the finite element method (FEM) can be used for solving Laplace's equation. Unlike FDM causing a stair-casing error on curved surfaces, FEM provides more flexibility when dealing with complicated geometries

for using triangular/tetrahedral mesh [2]. In this work, we used FEM to solve 2D Laplace equation for evaluating the potential and electric field distributions in three transmission line structures that support TEM or quasi-TEM modes. The three structures are coaxial line, microstrip, and stripline.

For the microstrip and stripline considered in this work, we applied a homogeneous Neumann boundary condition to enforce a symmetry plane to reduce simulation region/time. In addition, homogeneous Dirichlet boundary conditions were used as artificial shields to reduce the simulation region for the two structures that is not naturally "shielded". These Dirichlet boundaries were placed far enough away from the trace with high potential to bring negligible effect on the solution. The FEM used the mesh generated from *pdetool* in Matlab for each structure in 2D. The implementation of the FEM based on the generated mesh, solving matrix, and plotting results were all coded using Matlab.

Section 2 introduces the finite element analysis (FEA) for solving Laplace's equation and the FEM implementation. Section 3 gives the results and discussions including the simulated potential, field distribution, and result validation. Limitations of the FEM implementation in this work are also discussed. The last section gives the conclusions and possible future work.

2 Methods

The finite element analysis for Laplace's equation used in this work is mainly based on [2], while the FEM implementation is similar to another work in [3] due to the mesh generated from a Matlab tool. In this section, we only introduce the key features of the FEA and the procedures of the FEM implementation and how the mesh data generated from Matlab is used in the FEM.

2.1 Finite element analysis for Laplace's equation

For a lossless transmission line, the Laplace equation can be given by

$$\nabla \cdot \varepsilon \nabla \varphi = 0 \tag{1}$$

where φ is the electric potential and ε is the permittivity of the dielectric of the transmission lines. The Dirichlet boundary condition may be given as

$$\varphi = V \text{ on } \Gamma_D \tag{2}$$

where V is the voltage on Γ_D. To enforce a symmetry plane for certain structure (i.e. microstrip line), a homogeneous Neumann boundary condition may be applied additionally which is given by

$$\hat{n} \cdot (\varepsilon \nabla \varphi) = \kappa_N = 0 \text{ on } \Gamma_N. \tag{3}$$

For a structure without applying symmetry plane (i.e. coaxial transmission line), the simulation boundary is $\Gamma = \Gamma_D$; while for a structure with a symmetry plane the boundary is $\Gamma = \Gamma_D \cup \Gamma_N$. Considering the case with Neumann boundary condition for the symmetry plane, the weak-form representation of the above boundary-value problem may be written as [2]

$$\int_\Omega w_i [\nabla \cdot (\varepsilon \nabla \varphi)] \, d\Omega = 0 \tag{4}$$

where w_i is the weighting/testing functions and Ω is the simulation domain. Using the vector identity:

$$w_i [\nabla \cdot (\varepsilon \nabla \varphi)] = \nabla \cdot (w_i \varepsilon \nabla \varphi) - \varepsilon \nabla \varphi \cdot \nabla w_i \tag{5}$$

and Gauss' theorem:

$$\int_\Omega \nabla \cdot (w_i \varepsilon \nabla \varphi) \, d\Omega = \oint_\Gamma \hat{n} \cdot (w_i \varepsilon \nabla \varphi) \, d\Gamma, \tag{6}$$

then (4) can be written as

$$\int_\Omega \varepsilon \nabla w_i \cdot \nabla \varphi \, d\Omega = \int_{\Gamma_D} \hat{n} \cdot (w_i \varepsilon \nabla \varphi) \, d\Gamma + \int_{\Gamma_N} \kappa_N w_i \, d\Gamma. \tag{7}$$

Since $\kappa_N = 0$, the last integral on the right hand side of (7) is zero. Due to the symmetry of (7), we may apply Galerkin method here [2]. To approximate the potential φ in a discrete set of variables, the potential φ can be expanded as

$$\varphi \approx \sum_{j=1}^{N} N_j \varphi_j + \sum_{j=1}^{N^D} N_j^D \varphi_j^D \tag{8}$$

where N is the total number of nodes at which the potential is unknown, φ_j is the corresponding unknown potential, and N_j is the corresponding basis function at node j; while N^D is the number of nodes at the boundary Γ_D with known potential, φ_j^D is the corresponding known potentials at Γ_D, and N_j^D is the corresponding basis function at node j. Note that the integral over Γ_D disappears since N_i vanishes on the two boundaries. Using the linear interpolation function N_j for the weighting function w_i, (7) then reduce to

$$\int_\Omega \varepsilon \nabla N_i \cdot \nabla \varphi \, d\Omega = 0. \tag{9}$$

Substituting (8) into (9) would yield:

$$\sum_{j=1}^{N} \varphi_j \int_\Omega \varepsilon \nabla N_i \cdot \nabla N_j \, d\Omega = -\sum_{j=1}^{N^D} \varphi_j^D \int_\Omega \varepsilon \nabla N_i \cdot \nabla N_j^D \, d\Omega. \tag{10}$$

One may repeat this process for all the testing functions to obtain the matrix equation given by

$$[\mathbf{K}]\{\varphi\} = \{\mathbf{b}\} \tag{11}$$

where

$$[\mathbf{K}]_{ij} = \int_\Omega \varepsilon \nabla N_i \cdot \nabla N_j \, d\Omega, \tag{12}$$

$$\{\mathbf{b}\}_i = -\sum_{j=1}^{N^D} \varphi_j^D \int_\Omega \varepsilon \nabla N_i \cdot \nabla N_j^D \, d\Omega. \tag{13}$$

In the next subsection, we will introduce the assembly process and the implementation procedures of the FEM for solving $\{\varphi\}$ in Matlab.

2.2 FEM implementation

The explicit expressions of N_i and N_j may be difficult to find since nodes may be connected to different numbers of elements resulting different shape for elements [2]. It would be better to rewrite (12) and (13) as:

$$[\mathbf{K}]_{ij} = \sum_{e=1}^{M} \int_{\Omega^{(e)}} \varepsilon \nabla N_i \cdot \nabla N_j \, d\Omega \quad (9.2.27) \tag{14}$$

$$\{\mathbf{b}\}_i = -\sum_{j=1}^{N^D} \varphi_j^D \sum_{e=1}^{M} \int_{\Omega^{(e)}} \varepsilon \nabla N_i \cdot \nabla N_j^D \, d\Omega \tag{15}$$

where $\Omega^{(e)}$ represents the domain of element e, and M is the total number of elements in Ω. Here, we use the assembly process presented in [2]. We introduce the connectivity array which can be defined as $n(l, e)$ which is assigned as the node number of the lth node in element e. The nodes of each element are numbered counterclockwise. Then, defining $i = n(l, e)$ and $j = n(k, e)$ where l, $k = 1$, 2, 3 for a triangular element except when node i, or node j, or both are associated with the boundary, (14) and (15) can be written as:

$$[\mathbf{K}]_{ij} = \sum_{e=1}^{M} \int_{\Omega^{(e)}} \varepsilon \nabla N_l^{(e)} \cdot \nabla N_k^{(e)} \, d\Omega \tag{16}$$

$$\{\mathbf{b}\}_i = -\sum_{j=1}^{N^{Dh}} \varphi_j^{Dh} \sum_{e=1}^{M} \int_{\Omega^{(e)}} \varepsilon \nabla N_l^{(e)} \cdot \nabla N_k^{Dh^{(e)}} \, d\Omega. \tag{17}$$

where $N_l^{(e)}$, $N_k^{(e)}$ and $N_k^{Dh^{(e)}}$ are the interpolation functions for element e which can be, in Cartesian coordinates, evaluated by

$$N_l^{(e)}(x, y) = \frac{1}{2\Delta^{(e)}}(a_l^{(e)} + b_l^{(e)} x + c_l^{(e)} y) \tag{18}$$

in which

$$\begin{aligned}
a_1^{(e)} &= x_2^{(e)} y_3^{(e)} - x_3^{(e)} y_2^{(e)}, \quad b_1^{(e)} = y_2^{(e)} - y_3^{(e)}, \quad c_1^{(e)} = x_3^{(e)} - x_2^{(e)}, \\
a_2^{(e)} &= x_3^{(e)} y_1^{(e)} - x_1^{(e)} y_3^{(e)}, \quad b_2^{(e)} = y_3^{(e)} - y_1^{(e)}, \quad c_2^{(e)} = x_1^{(e)} - x_3^{(e)}, \\
a_3^{(e)} &= x_1^{(e)} y_2^{(e)} - x_2^{(e)} y_1^{(e)}, \quad b_3^{(e)} = y_1^{(e)} - y_2^{(e)}, \quad c_2^{(e)} = x_2^{(e)} - x_1^{(e)},
\end{aligned} \tag{19}$$

and

$$\Delta^{(e)} = \frac{1}{2}(b_1^{(e)} c_2^{(e)} - b_2^{(e)} c_1^{(e)}) = \text{area of element } e. \tag{20}$$

With all the known coordinates of each node, the coefficients in (19) and the area of each element in (21) can be evaluated readily. For improving the efficiency of the assembly process, we may rewrite (16) as

$$[\mathbf{K}]_{ij} = \sum_{e=1}^{M} K_{lk}^{(e)} \tag{21}$$

where

$$K_{lk}^{(e)} = \int_{\Omega^{(e)}} \varepsilon \nabla N_l^{(e)} \cdot \nabla N_k^{(e)} \, d\Omega = \frac{\varepsilon^{(e)}}{4\Delta^{(e)}}(b_l^{(e)} b_k^{(e)} + c_l^{(e)} c_k^{(e)}) \tag{22}$$

where $\varepsilon^{(e)}$ is the permittivity for element e that is assigned as a constant for each element. We choose $\varepsilon^{(e)} = 1$ for simplicity. In this way, the assembly process goes through each element. Thus, $[\mathbf{K}]_{ij}$ can be evaluated from the sum of the corresponding $K_{lk}^{(e)}$.

We use the *pdetool* in Matlab for generating triangular mesh cells for each simulation structure. Using *Export Mesh*, one may obtain the mesh data (p - Mesh points, e - Mesh edges, and t - Mesh elements) which include the coordinates of each nodes (in p, as called Mesh points), the boundary edge nodes (in e, as called Mesh edges), and the node numbers for each element (in t, as called Mesh elements) [3]. Figure 1 shows the mesh generated from Matlab for a coaxial line with global node numbers and

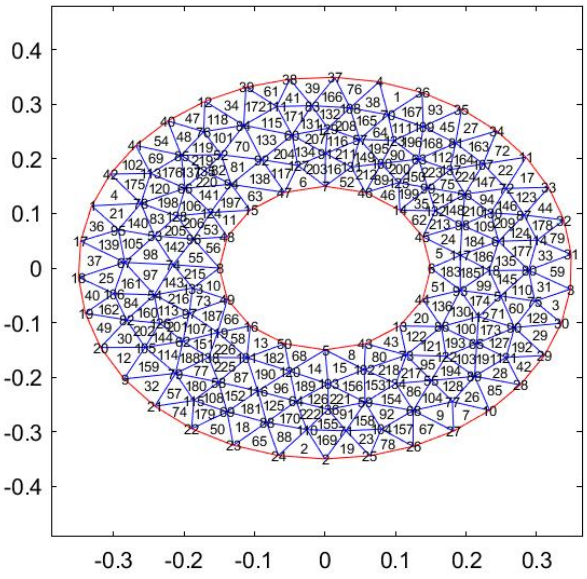


Figure 1: Triangular mesh generated from Matlab for a coaxial line with global node numbers and element numbers.

element numbers. This tool in Matlab generates the node numbers for each element counterclockwise (in t) which exactly matches with the numbering method for the local nodes aforementioned.

Moreover, the mesh edges data file (in e) provides the node numbers on the boundaries. For example, in the coaxial line case it gives the node numbers: 1, 17, 18, 19, 20, ..., 40, 44, 42 on the outer boundary; 5, 43, 13, 44, 6, ..., 49, 16, 50 on the inner boundary, as shown in Figure 1. In fact, it was found that the first 50 nodes in the coordinates file are associated with Γ_D in this case. Though they are not sequentially placed on the outer or inner boundary, but from the node No. 1 to the node No. 50 they are all on the boundaries as shown in Figure 1. This allows us to consider the matrix:

$$\Big[L\Big] = \begin{bmatrix} A & B^T \\ B & K \end{bmatrix} = \begin{bmatrix} \varphi^D \\ \varphi \end{bmatrix} = \begin{bmatrix} b^D \\ b \end{bmatrix} \tag{23}$$

where L (or A, B, and K) can be evaluated using (21) and (22) for the corresponding nodes, K is associated with the nodes with unknown φ, and φ^D is the known potential on Γ_D. A is a M-by-M matrix corresponding to the nodes on Γ_D (this is true only for using the default mesh, and more details can be found in section 3.3), where M is the number of nodes on Γ_D. For the case of microstrip line with Γ_N, as shown in Figure 2, the nodes on Γ_D need to be extracted manually from the mesh edge data file (in e) for assembling the corresponding φ^D, while the nodes on Γ_N (i.e. 46, 47, 48,..., 54 in Figure 2) would be treated as same as the nodes with unknown φ. In another word, the nodes on Γ_N

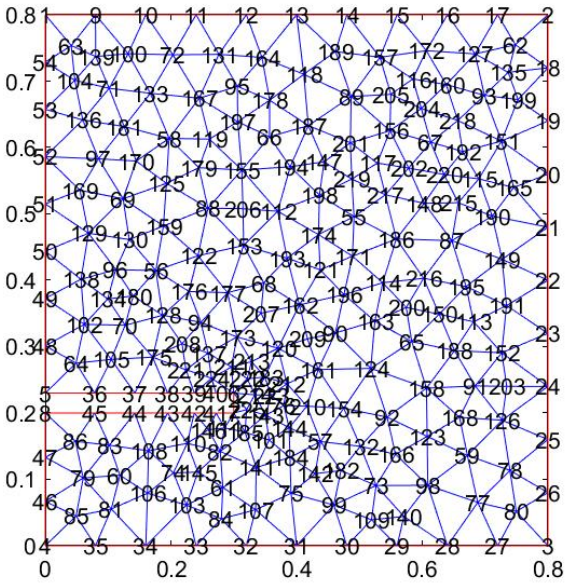


Figure 2: Triangular mesh generated from Matlab for a microstrip with global node numbers.

are associated with the matrix K. For both cases with or without Γ_N, the unknown φ can be solved from the second block of equations from (23) which is given by

$$K\varphi = b - B\varphi^D \tag{24}$$

where b here is a N-by-1 zero matrix in the given weak-form problem in (10). The first block of equations in (23) with the unknown b^D is not of interest to us and it is ignored [3]. It should be noted that manually extracting the nodes on Γ_D is a drawback of the presented implementation. Because the boundary nodes are not well aligned in the the mesh edge data file (in e) from the black-box-like *pdetool* that we apply for generating mesh. We use the default mesh in *pdetool* for a given structure and an interpolation tool in Matlab to obtain a set of *interpolant* φ. This should give us reasonable results for the potential distribution in the three transmission line structures due to their relatively simple geometries.

3 Results and Discussions

For the three structures, we consider homogeneous Dirichlet boundary conditions $\varphi = 1$ kV and $\varphi = 0$ V for the high voltage part/trace and grounded part(s), respectively. We assume the length unit is in decimeter (dm) in the simulation. For example, the inner and outer radii of the dielectric in the coaxial line are 1.5 mm and 3.5 mm, respectively.

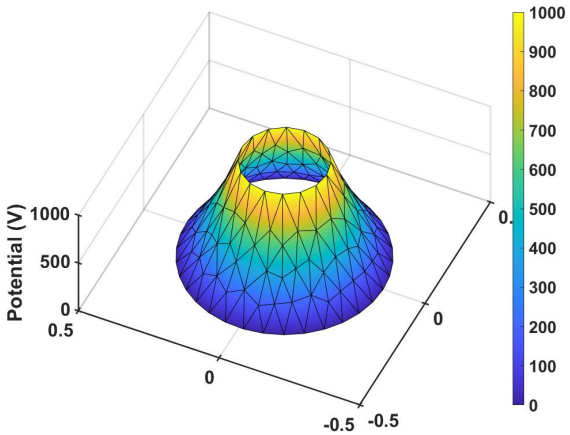


Figure 3: Potential distribution from the FEM for the coaxial line.

3.1 Coaxial transmission line

Figure 3 shows the FEM simulation results for the coaxial transmission line using the triangular surface plot. Since φ is a function of the radial distance r alone due to symmetry, by solving (1) and (2) one may obtain the analytical solution of φ in cylindrical coordinates which is given by

$$\varphi(r) = \frac{V_0 \ln(b/r)}{\ln(b/a)} \tag{25}$$

where $V_0 = 1$ kV is the potential on the inner conductor, $b = 35$ mm and $a = 15$ mm are the outer and inner radii of the dielectric in the coaxial line, respectively. The numerical results can be readily validated by comparing it to the analytical solution in (25). Figure 4 plots both FEM and analytical potential as a function of the radial distance. The good agreement between the two potential distributions suggests that the FEM is successfully implemented and accurate enough. Figure 5 shows the equipotential lines and static electric field distribution. The electric field is in the radial direction and perpendicular to the equipotential lines as expected.

3.2 Microstrip line

A microstrip line may support quasi-TEM modes if the wavelength λ is much larger than the thickness of the dielectrics ($d \ll \lambda$) [1]. A homogeneous Neumann boundary condition is applied in the middle of the trace to enforce a symmetry plane. Homogeneous Dirichlet boundary conditions are used as artificial shields to reduce the simulation region. Figure 6 shows the potential distribution from the FEM for the microstrip line. Figure 7 shows the equipotential lines and static electric field distribution for the microstrip line. Overall, the equipotential lines and static electric field distribution obtained from the FEM are similar to that from the numerical solver in *pdetool* in Matlab, suggesting the presented FEM works reasonably well. At the two homogeneous Dirichlet boundaries (x = 0.8 dm and

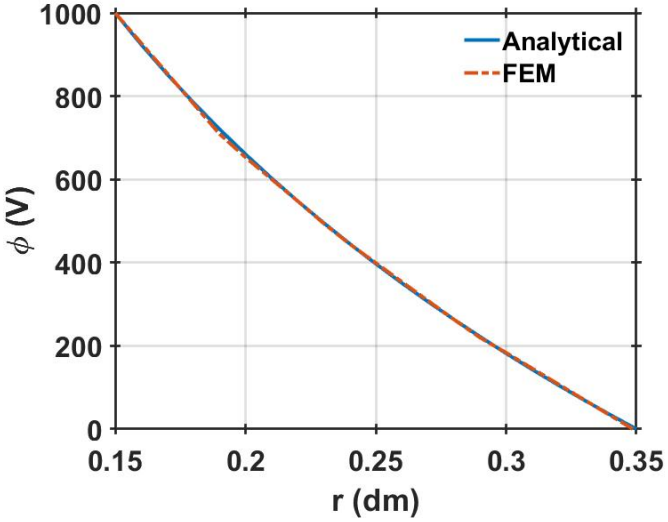


Figure 4: Comparison between the FEM method and analytical solution for the potential distribution as a function of radial distance

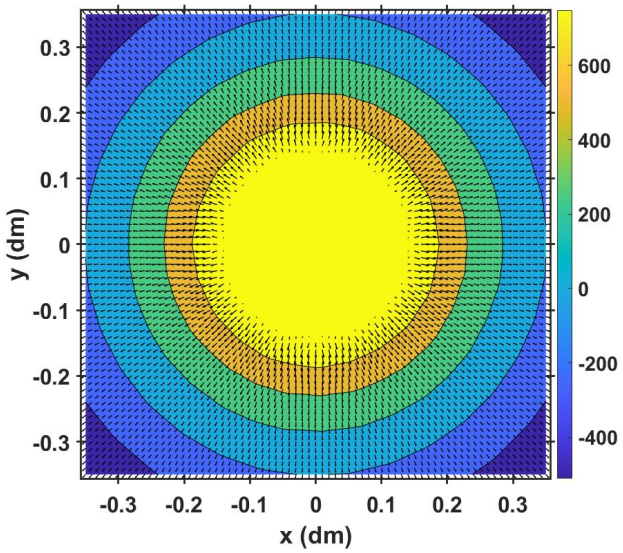


Figure 5: Equipotential lines and static electric field distribution for the coaxial line (black arrows: electric field).

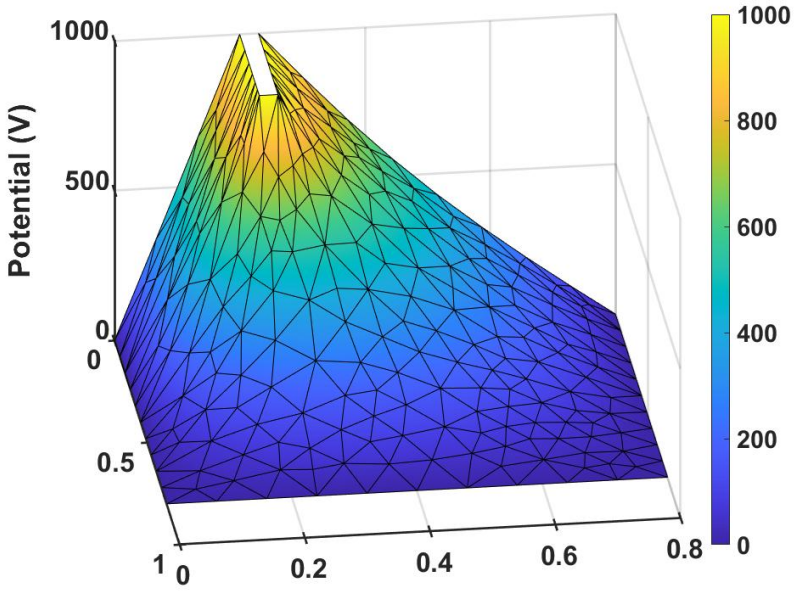


Figure 6: Potential distribution from the FEM for the microstrip line.

y = 0.8 dm) functioning as artificial shields, the potential reduces to zero, as shown in Figure 6 and 7. Thus, the two should not affect the solution in the domain of interest.

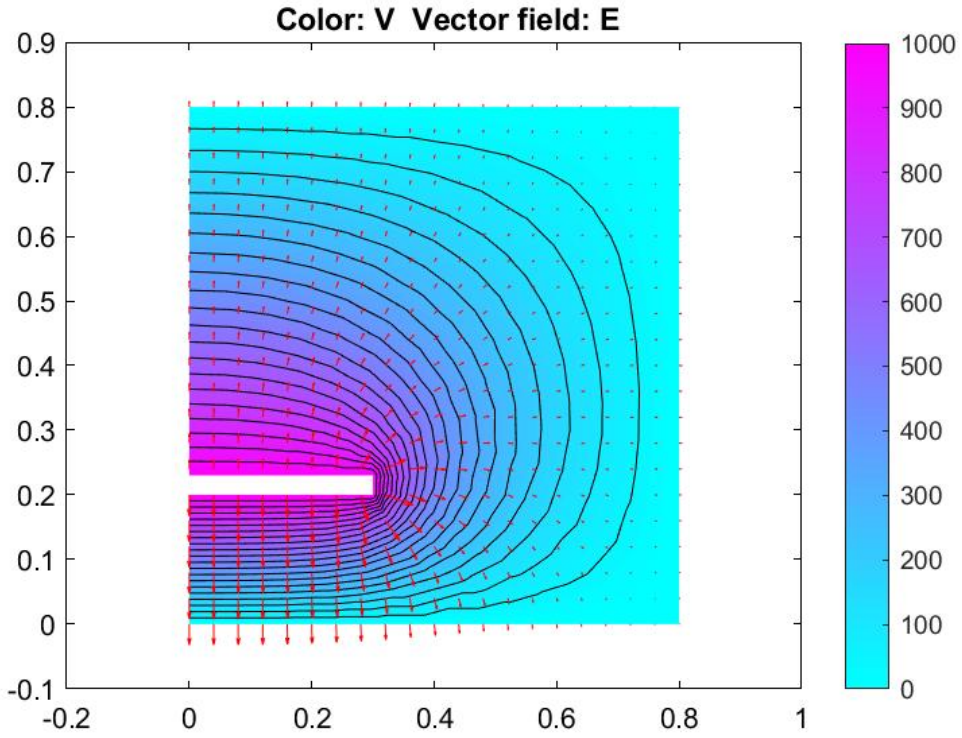
3.3 Stripline

Similar to the microstrip line, the stripline is simulated using the FEM using a homogeneous Neumann and a Dirichlet boundary condition to provide a symmetry plane and an artificial shield, respectively. The only difference is that one Dirichlet boundary serves for another grounding part instead of an artificial shield. Figure 8 shows the potential distribution from the FEM for the stripline. Figure 9 shows the equipotential lines and static electric field distribution for the stripline from both the numerical solver in *pdetool* and the presented FEM in this work. A good agreement between the two methods is also obtained for this structure and the Dirichlet boundary condition acting as an artificial shield should bring no effect on the solution due to zero potential on the boundary (x = 0.8 dm).

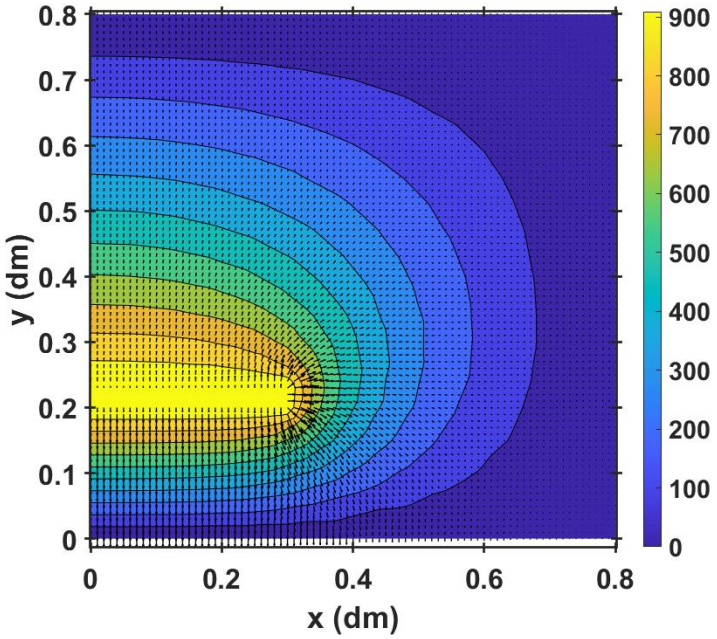
3.4 Limitations of the presented FEM

As we mentioned in section 2.2, the use of the *pdetool* for generating mesh limits the FEM approach in this work. One may need to manually extract the nodes on Γ_D from the mesh edge data file (in e) for assembling the corresponding φ^D, since the nodes in the file are not well aligned. More information or a complete documentation is needed for efficiently using all node data generated by *pdetool*, specifically for file e.

Moreover, it was found that when improving the mesh in the tool the newly added nodes may not



(a)



(b)

Figure 7: Equipotential lines and static electric field distribution for the microstrip line from (a) the *pdetool*, and (b) the presented FEM in this work.

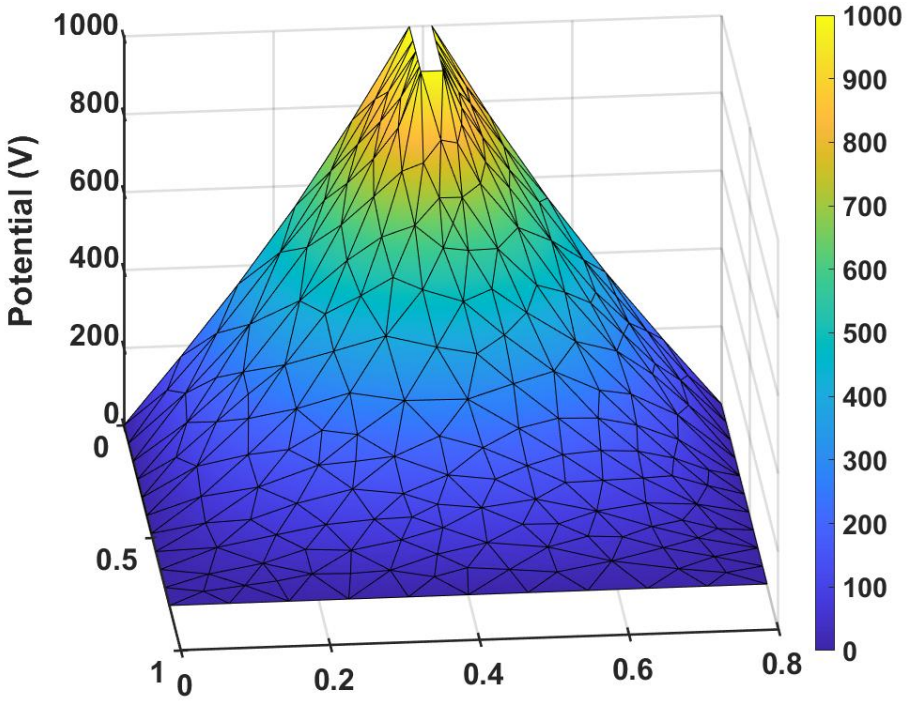


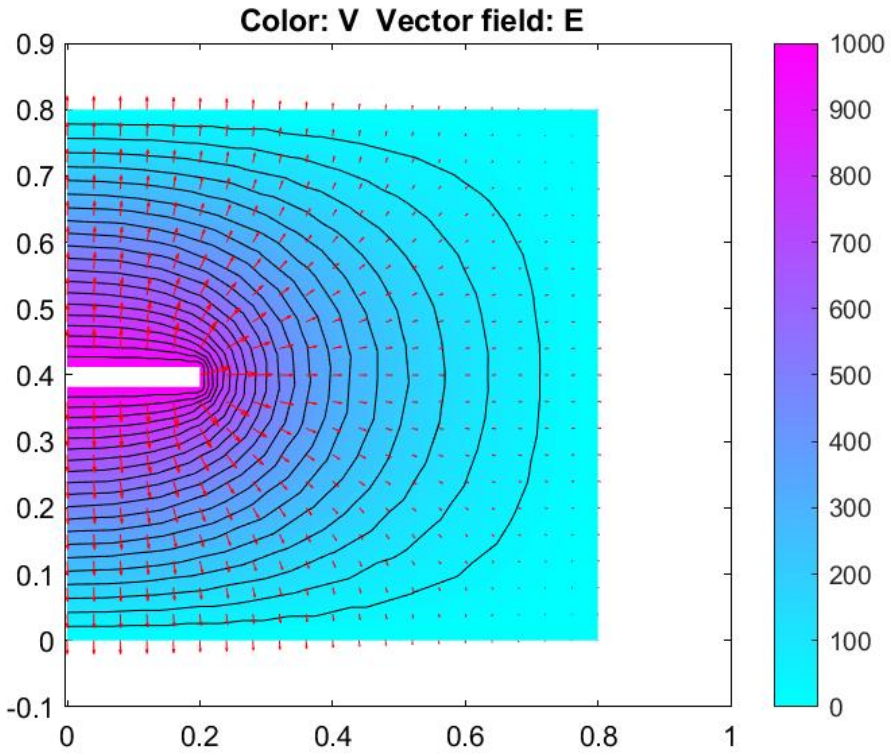
Figure 8: Potential distribution from the FEM for the stripline.

always associated with the matrix A in (23). One may need to rearrange these new edge nodes and their coordinates in file p to make A correspond to all Dirichlet boundary nodes to eventually use (23) and (24). This is quite laborious and inefficient. A robust method for applying the edge nodes may be appealing and necessary for using finer mesh as the node number increases greatly. Or one may develop another mesh generation method which is more compatible to (23) and (24).

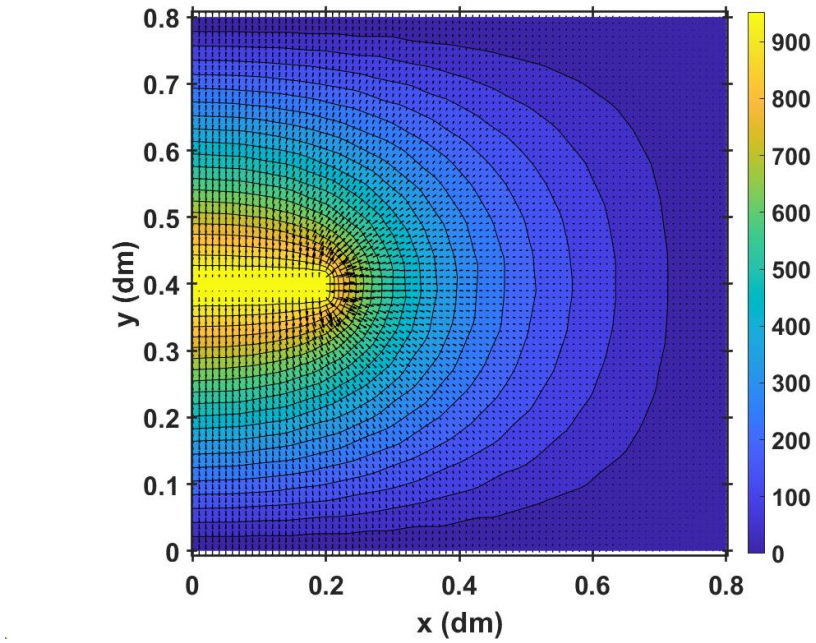
4 Conclusion

In this work, we used FEM to solve 2D Laplace's equation for coaxial line, microstrip line, and stripline. The three structures are either naturally or artificially shielded and support TEM or quasi-TEM modes. Triangular mesh with the node data generated from *pdetool* in Matlab was used for the FEM. We applied homogeneous Neumann boundary condition to enforce a symmetry plane. Homogeneous Dirichlet boundary conditions were used as artificial shields for the microstrip and stripline that are not naturally "shielded". These boundaries were placed far enough away from the high potential trace to bring negligible effect on the solution.

We validated the FEM by comparing the potential obtained from the FEM in the coaxial line to the analytical solution of Laplace's equation in cylindrical coordinates. In addition, the equipotential lines and static electric field distribution from the presented FEM for each structure were plotted and



(a)



(b)

Figure 9: Equipotential lines and static electric field distribution for the stripline from (a) the *pdetool*, and (b) the presented FEM in this work.

compared to that obtained in the numerical solver of *pdetool*. The presented results and comparison suggest a successful use of the FEM.

However, there are few limitations for the presented FEM implementation when using finer mesh. It was found that when using finer mesh the matrix A in (23) may no longer be associated with all Dirichlet boundary nodes. This may require laborious rearrangement of the nodes and coordinates due to the newly added edge nodes. Additional information of data generated by *pdetool* is needed for using all edge node data. Alternatively, one may apply another mesh generation method and develop a robust assembly procedure. These could be future works.

Future studies may also solve 2D Laplace's equation for other structures in 2D, such as coplanar waveguide, grounded coplanar waveguide, slotline, etc, in which support TEM or quasi-TEM modes, and simulate 2D Poisson's equation with certain charge distribution.

5 References

[1] D. M. Pozar, *Microwave Engineering*. John Wiley & Sons, 2011.

[2] J.-M. Jin, *Theory and Computation of Electromagnetic Fields*. John Wiley & Sons, 2011.

[3] *meshToPet*. Accessed on: Nov. 4, 2021, [Online], Available: https://www.mathworks.com/help/pde/ug/pde.femesh.meshtopet.html

[4] J. Alberty, C. Carstensen, and S. A. Funken, "Remarks around 50 lines of Matlab: short finite element implementation," *Numer. Algorithms*, vol. 20, pp. 119-137, 1999.