COMMUNICATIONS IN NUMERICAL METHODS IN ENGINEERING
Commun. Numer. Meth. Engng. 2000; **16**:177–190

An automatic mesh generator for handling small features in open boundary power transmission line problems using artificial neural networks

D. G. Triantafyllidis and D. P. Labridis*

Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, 540 06 Thessaloniki, Greece

SUMMARY

An artificial neural network (ANN) has been trained to automatically predict the mesh density vector of open-boundary faulted power transmission line problems under the presence of small features, such as conductors. In order to produce the data for the training database of the ANN a technique for recursively subdividing the solving area is proposed. A Delaunay-based mesh generator is used to produce the final mesh. Several test cases show that the proposed automatic mesh generator significantly reduces solving time. Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: automatic mesh generation; mesh density prediction; neural networks; power transmission lines; finite element method

1. INTRODUCTION

The finite element method (FEM) has been widely used to solve electromagnetic field problems, mainly due to its ability to handle cases of geometrical complexity, where other methods would probably fail. One of the main disadvantages of the method though, is the need for an experienced user to provide a quality mesh, i.e. a mesh with which the method will converge to an accurate solution quickly.

One way around this problem is the use of adaptive meshing [1, 2]: starting with an initial coarse mesh the problem is solved, then the solution error is estimated for each element and finally the elements with error that exceeds a given threshold are split into smaller elements, thus refining the initial mesh. The procedure is repeated, until the required accuracy is met. This technique will provide very accurate results, but is very memory and time consuming and does not take into account the 'knowledge' obtained from similar previously solved problems.

Another approach would be to use much more elements than necessary, overdiscretizing the solving area. There are two major disadvantages to this approach. First, system resources are not

* Correspondence to: D. P. Labridis, Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, P.O. Box 486, GR-540 06 Thessaloniki, Greece

Received 5 November 1998
Accepted 11 August 1999

efficiently exploited, since more memory and computation time is required. Second, error build-up, due to unnecessary large number of elements, may lead to inaccurate solutions.

The use of artificial neural networks (ANNs) has been proposed in the last years for predicting the mesh density of specific electromagnetic problems [3, 4]. For this purpose a prototype mesh is generated for a magnetic device, which includes all the features expected to be found in future devices to be meshed. The device is sampled at specific sample points and a set of geometric features is associated to the mesh density at the neighbourhood of the given sample point. The data gathered constitute the training database for the ANN. After the completion of the ANN training, mesh density prediction for similar magnetic devices may easily take place, with no significant computational cost, by presenting the device to the ANN.

Once the mesh density has been determined, a density driven mesh generator is used to create the final mesh. A Delaunay process may be used for this purpose [5], while also other techniques [6, 7] based on self-organizing maps [8] or let-it-grow ANNs [9] have been presented. The mesh obtained can either be used as is, or as a good starting point for adaptive meshing, reducing the total number of necessary iterations.

Although the above mesh density prediction methods work well for the magnetic devices that they were designed for, they appear to fail in cases of faulted overhead power transmission line systems (TLS), carrying a zero-sequence current, as this is a case with practical interest. The uniqueness of these problems lies in the disproportion between the conductors sizes and the total solving area. In this paper, an automatic mesh generator based on ANNs is presented, which allows mesh density prediction for open-boundary electromagnetic field problems of overhead power transmission lines [10].

2. PROBLEM DEFINITION AND MODELLING

The faulted power TLS is located over homogenous lossy ground and the total solving area is included in a square 10 km × 10 km. On this remote boundary a Dirichlet condition is applied. The centre of the square is the point at which the middle of the TLS tower touches the ground, leading to a y-axis symmetry. Maxwell's equations are solved by FEM in order to calculate the per-unit length voltage drops of the conductors, when a zero sequence of unit currents is applied [10]. In order to present a power transmission line to the ANN, the solving area has to be sampled at specific sample points. A sample point (+) is the centre of a square, called magnifier [4], of dimensions scale × scale (Figure 1). Mesh density in a magnifier is defined as

$$\text{Mesh density} = \frac{\text{Number of nodes in magnifier}}{\text{scale}} \tag{1}$$

Regarding a sample point, three parameters have been identified to influence mesh density:

(i) the distance r of the sample point from the current carrying conductor,
(ii) the distance h of the sample point from the air-ground boundary and
(iii) the f/ρ ratio, where f is the frequency of the energizing current and ρ is the resistivity of earth.

These parameters are considered as the input vector of the ANN.

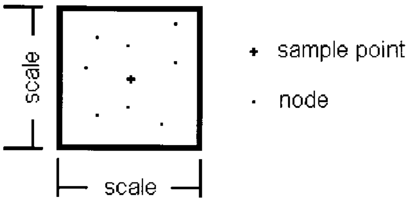

Figure 1. A sample point (+) and its magnifier.

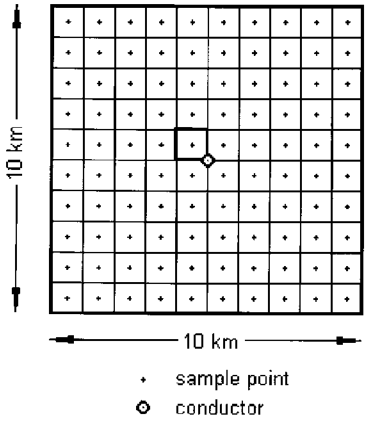

Figure 2. Initial coarse sample point location. One of the four magnifiers, closest to the conductor, is marked bold.

Since the majority of current carrying conductors of TLSs is made of aluminium and ground wires are made of steel, their conductivity is in all cases extremely larger than that of the surrounding air, which is considered to be equal to zero. For this reason, the conductivity of conductors is not included in the input vector of the ANN.

Initially, an arrangement of 10×10 sample points with magnifiers of scale $= 1$ km, centered on the conductor, is generated (Figure 2). This scale for the magnifiers has been proven to be satisfactory for sample points far away from the conductor. Obviously, near the conductor, the scale must become smaller in order to satisfactorily discretize the small features. This is accomplished by the following technique:

(i) Locate the four sample points that are nearest to the conductor.
(ii) Subdivide each of the four magnifiers, corresponding to the sample points of step (i), into four magnifiers of half-scale each.
(iii) Update the sample point list. Discard the sample points located in step (i).
(iv) Repeat steps (i–iii), until the scale of the last magnifiers created is of the same order as the radius of the conductor.

This recursive subdivision scheme produces an arrangement of sample points near the conductor, as can been seen in Figure 3.

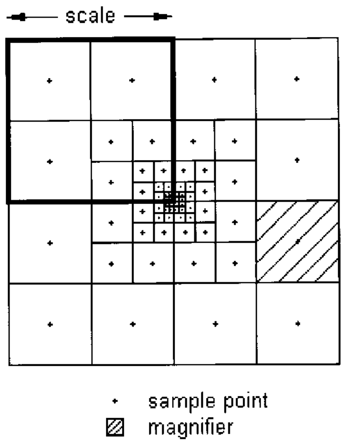

Figure 3. Sample point location near conductor.

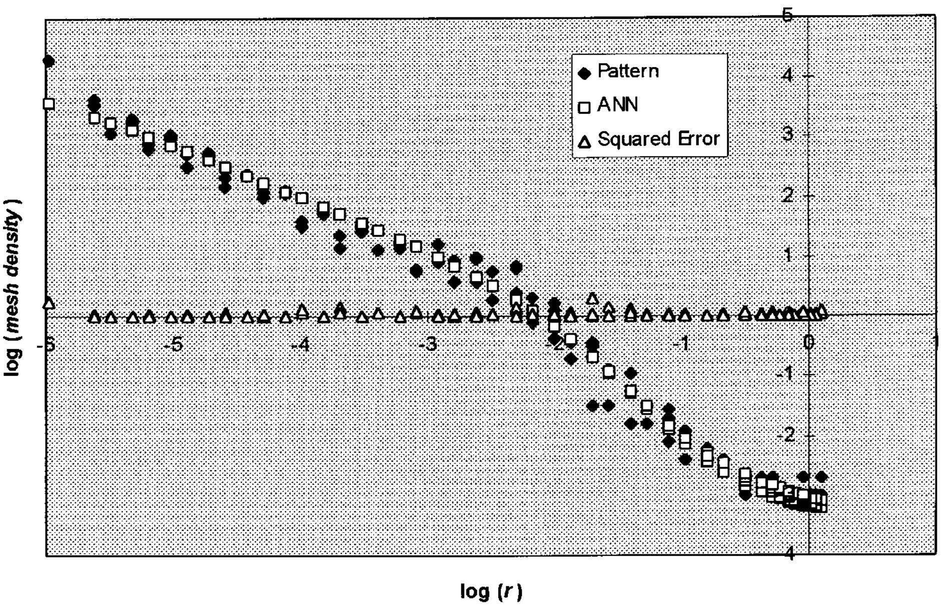

Figure 4. Mesh density versus distance r from conductor (f/ρ = 5000 Hz/Ωm).

Figure 4 presents among others the mesh density defined in (1) versus the distance r of a sample point from the conductor.

In order to obtain suitable data for training the ANN, a power line consisting of only one conductor is considered (Figure 5). The line is energized by a unit current, for an f/ρ ratio ranging from 0.005 to 5000 Hz/Ωm, with a step increasing this ratio by 10. The above range is sufficient for any practical power system problem of this kind. Initially, a simple coarse mesh was created.

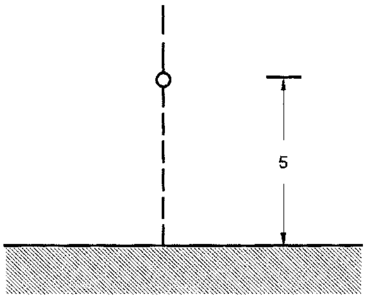

Figure 5. Cross-section of a power transmission line for ANN training purposes.

Adaptive meshing [1, 2] was used, until the solution obtained by FEM converged to a constant value for the per-unit voltage drop of the conductor. Usually, up to six iterations were necessary, producing a final mesh of about 5000 nodes. The configuration was sampled and the data produced made up the training database of the ANN.

TLSs used in practice always have more than one conductor. Therefore, the above sampling procedure must be generalized for performance purposes. This can be done under the assumption that the field far away from a TLS carrying only zero-sequence currents is practically independent from the exact position and number of conductors. Thus, a mesh density prediction takes place initially for each of the N current carrying conductors, assuming that no other conductors are present. A cloud of points on the plane is generated for each conductor, according to the respective mesh density prediction. For this purpose, a uniform distribution of points is generated within each magnifier. The number of points is proportional to the predicted mesh density, keeping the same proportion factor for all magnifiers. At last, the N clouds are overlapped and merged into one, in order to produce the final cloud of points which will drive the mesh density driven Delaunay-based mesh generator, producing the final mesh [5] (Figure 6). Should the number of points within each magnifier be as close as possible to the number of points resulting from the training procedure, then the predicted mesh density should be initially divided by N, else only the relative mesh density of one magnifier to another is preserved.

3. ANN ARCHITECTURE AND TRAINING

The method of backpropagation [11] has been used to train a three-layered feedforward artificial neural network (Figure 7), the characteristics of which are presented in Table I. For simplicity the hyperbolic tangent tanh( ), which is already available in most programming languages, was used as the non-linear transfer function of the ANN. The number of neurons in the hidden layer was selected according to Kolmogorov's theorem [12]:

$$p = 2n + 1 = 2.3 + 1 = 7 \tag{2}$$

Before training starts, the input variables r and h are normalized by dividing them with the half-length of the square of the solving area (i.e. 5 km in our case), in order to make the training database scale invariant. Also, the logarithms $\log(r)$ and $\log(f/\rho)$ of the input variables r and f/ρ,

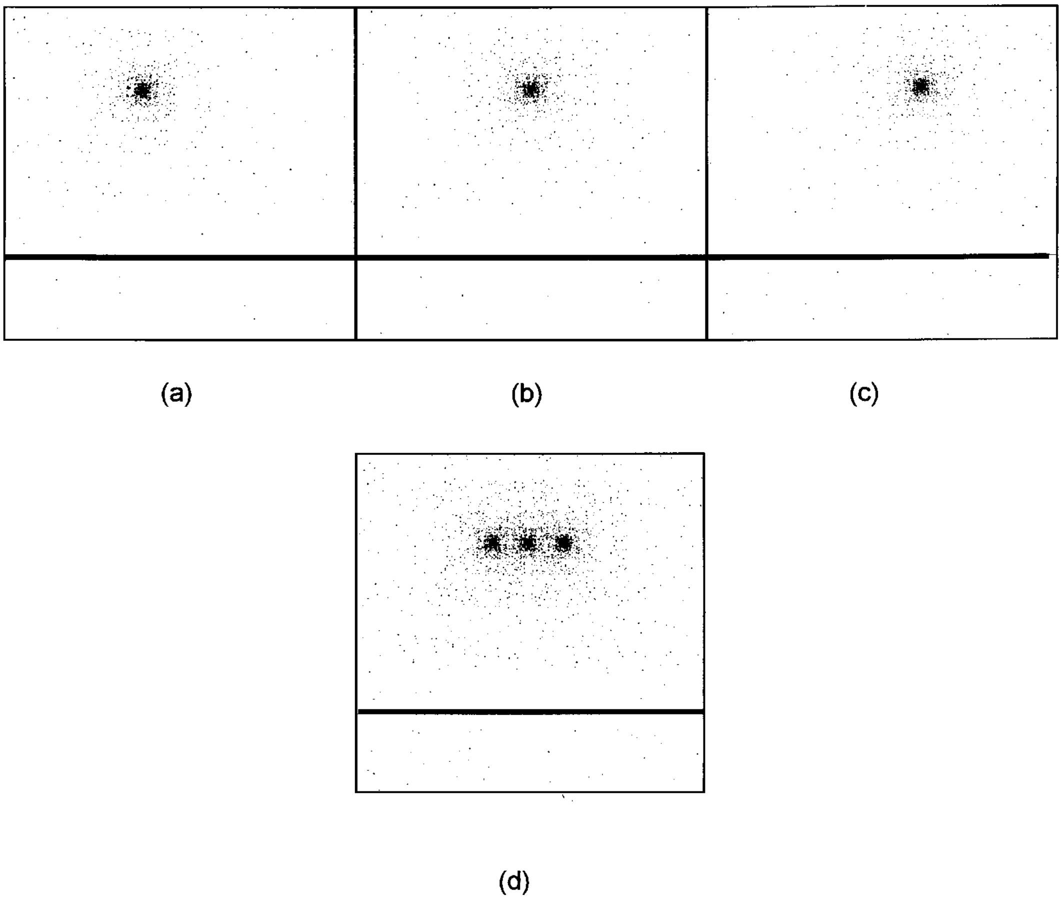

Figure 6. Predicted point clouds for (a) left conductor, (b) middle conductor, (c) right conductor of test case *a* (Figure 8(a)) and (d) merged point cloud. ($f/\rho = 0.1$ Hz/Ωm).

respectively, are used, because of their great dispersion. The input, as well as the output pattern variables, are also normalized in the region $[-0.9, 0.9]$. Since the problem solved has symmetry in the y-axis, only patterns from sample points with $y \leqslant 0$ are embodied in the training database.

Figure 4 finally shows mesh density defined in (1) versus distance r, comparing results from the trained ANN with the corresponding patterns from the training database, as well as the squared error for every distance r.

4. RESULTS

Once the ANN has been trained, it can be used to automatically predict mesh density for other similar problems. In Figure 8 three different overhead power TLSs are shown. Test case *a* consists

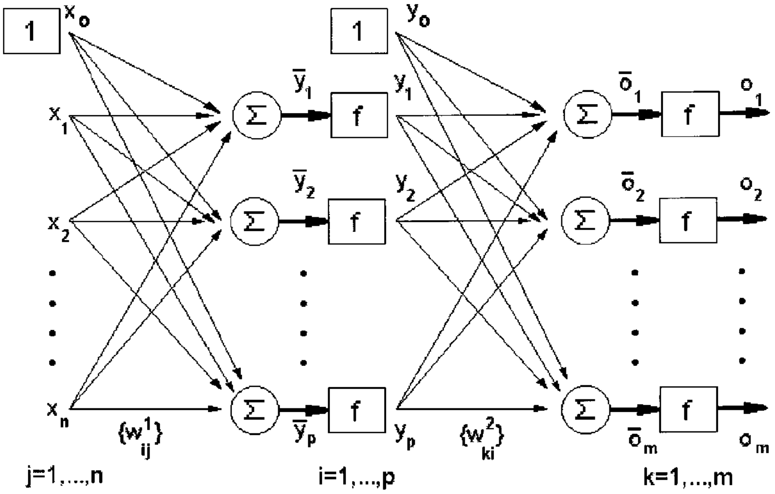

Figure 7. ANN architecture.

Table I. Architecture and training characteristics of the ANN shown in Figure 7.

Total number of layers	3
Number of neurons in input layer n	3
Number of neurons in hidden layer p	7
Number of neurons in output layer m	1
Learning rate η_1 for first layer	0.8
Learning rate η_2 for second layer	0.3
Momentum μ	0.95
tanh coefficient a	0.5
Final mean squared error	0.023743
Normalization region for all inputs	$-0.9\ldots0.9$
Normalization region for output	$-0.9\ldots0.9$
Number of patterns	1064
Number of iterations	22031

of three conductors, being a typical medium-voltage line. Test case b is a typical 150 kV line of the Greek power system, while test case c is a double-circuit three-phase transmission line [13]. Additionally, two more test cases, based on the transmission line of Figure 5 for which the ANN was trained, were also examined. Table II presents the characteristics of the five test cases.

Figures 9 and 10 compare meshes for the transmission lines examined, resulting from the adaptive meshing procedure and the trained ANN, for two different f/ρ ratios. Figure 11 shows the ANN mesh for test case b, in the conductors neighbourhood.

Table III presents performance data on the cases examined, while Table IV shows in detail a comparison between the adaptive meshing procedure for test case c and the proposed ANN

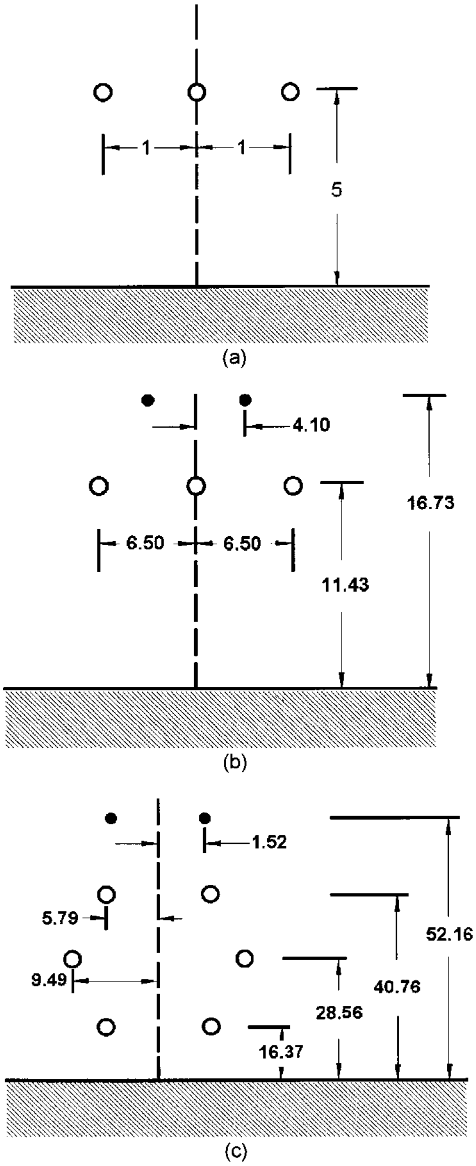

Figure 8. Test case *a*. Cross-section of a typical 20 kV distribution line; (b) Test case *b*. Cross-section of a 150 kV overhead transmission line of the greek power system; (c) Test case *c*. Cross-section of a 750 kV double-circuit line. Only the left circuit is energized; All distances indicated are in meters.

Table II. Test cases data.

	20 kV single circuit line (Test case *a*)	150 kV single circuit line (Test case *b*)	735 kV double circuit line (Test case *c*)	Single conductor-earth return line (Test case *d*)	Single conductor-earth return line (Test case *e*)
Phase conductors					
Conductor type	Solid	ACSR	ACSR	Solid	Solid
Outside diameter (mm)	5.6419	25.146	35.103	5.6419	5.6419
Inside diameter (mm)		9.71	23.364		
dc resistance (Ω/km)	0.2833	0.09136	0.04965	0.2833	0.2833
Ground wires					
Conductor type	—	Solid St conductor	Alumoweld-strand	—	—
Outside diameter (mm)		0.9525	0.9779		
dc resistance (Ω/km)		3.4431	1.4913		
f/ρ ratio (Hz/Ωm)	1	100	1000	0.01	10

mesh. The complex voltage drop $\bar{V}$ per unit length of the upper left conductor of each TLS has been chosen as a measure of comparison, which is made in all cases for a comparable number of nodes. Since one additional iteration will in most cases approximately double the number of nodes, some deviation from this rule is acceptable. The given CPU time is for a 200 MHz Pentium-based PC, running the Linux operating system.

From these tables it can be concluded that the ANN meshes of all test cases need significantly less CPU time, while they compute all TLS operational parameters (such as the shown complex voltage $\bar{V}$), with satisfactory precision.

5. CONCLUSIONS

An automatic mesh generator, implementing a technique capable of creating a suitable training database for using ANNs in mesh density prediction in cases of open boundary electromagnetic field problems, where small features are present, has been developed. The mesh generator has been applied in cases of faulted transmission line systems. The trained ANN has performed well in automatically predicting mesh density in numerous cases examined. A Delaunay-based mesh generator has been used in order to create the FEM mesh. In all cases examined the mesh produced acceptable results from the first iteration itself, as opposed to the adaptive mesh grading procedure, where several iterations were required.

The proposed mesh generator does not predict mesh density for no-current carrying conductors, because this has not been proven necessary for the problems examined. However, it would be useful to implement such a technique using ANNs, in order to deal with more complicated power system problems (i.e. buried pipelines). The authors believe that this may be feasible in a future work.

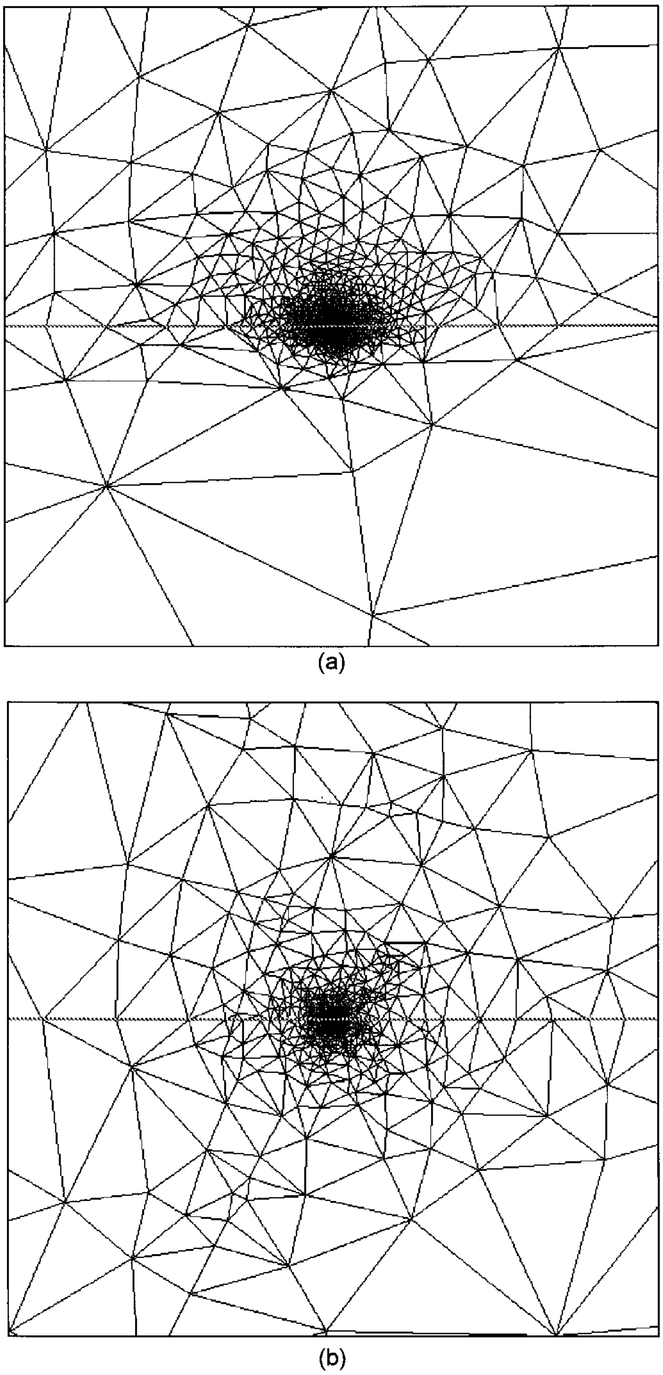

Figure 9. (a) Adaptive mesh of fifth iteration for test case *b* ($f/\rho = 100$ Hz/Ωm). (b) ANN initial mesh for test case *b* ($f/\rho = 100$ Hz/Ωm).

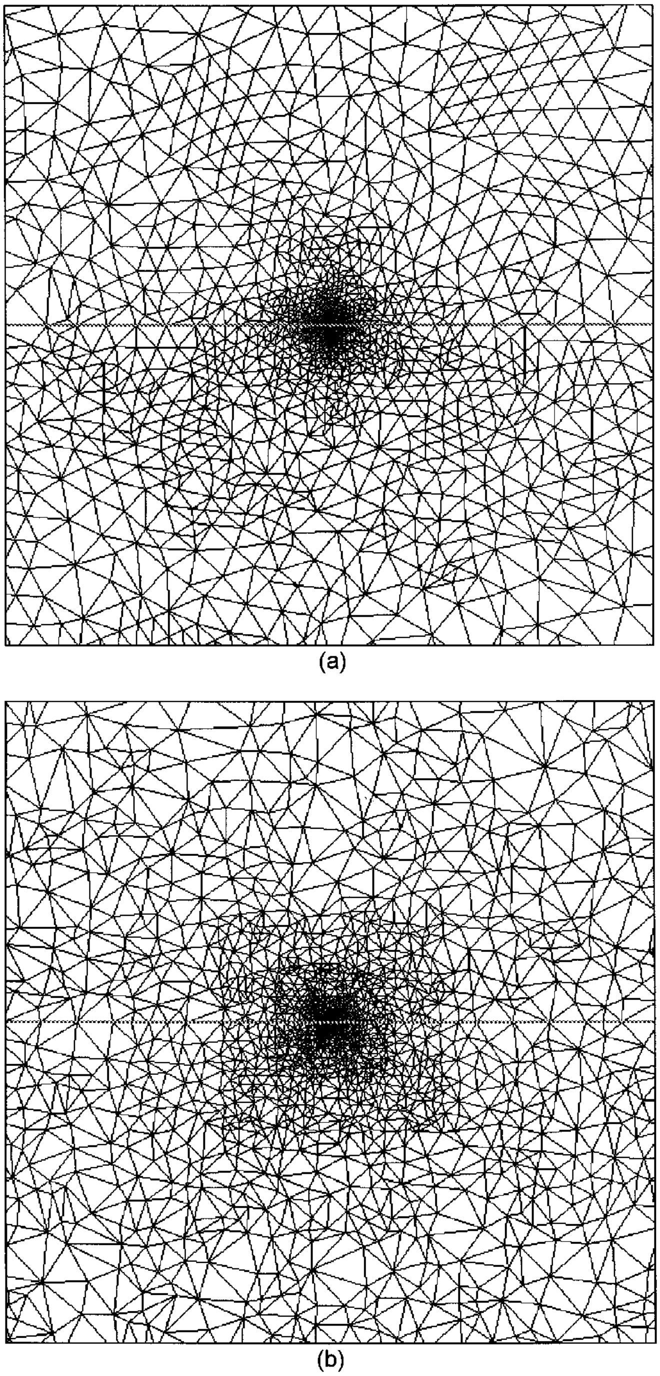

Figure 10. (a) Adaptive mesh of fifth iteration for test case *d* ($f/\rho = 0.01$ Hz/Ωm). (b) ANN initial mesh for test case *d* ($f/\rho = 0.01$ Hz/Ωm).

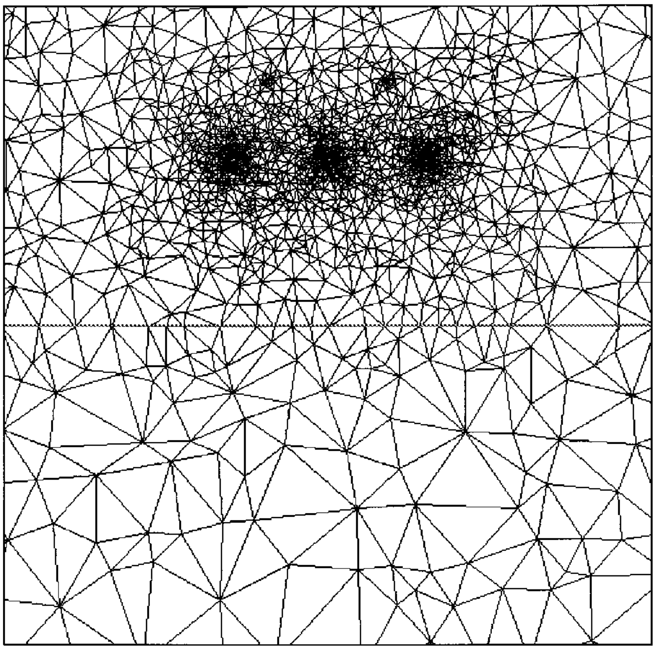

Figure 11. ANN initial mesh of test case *b* in the conductors neighbourhood.

Table III(a). Results for adaptive meshes.

Test case	f/ρ (Hz/Ωm)	Iterations	CPU time (min:s)	Nodes	Elements	$\bar{V}$ (V/m)
a	1	5	1:50	3348	6672	0.303 + j0.03
b	100	5	4:08	6924	13833	0.408 + j1.57
c	1000	5	5:28	7779	15543	0.722 + j13.0
d	0.01	6	2:21	4108	8160	0.297 + j0.0174
e	10	6	2:52	5892	11768	0.385 + j1.353

Table III(b). Results for ANN meshes.

Test case	f/ρ (Hz/Ωm)	Iterations	CPU time (min:s)	Nodes	Elements	$\bar{V}$ (V/m)
a	1	1	0:59	4163	8286	0.303 + j0.03
b	100	1	1:20	5046	10073	0.411 + j1.57
c	1000	1	1:39	5598	11181	0.739 + 12.9
d	0.01	1	0:57	4671	9253	0.3 + j0.0174
e	10	1	0:50	4891	9727	0.395 + j1.353

Table IV(a). Adaptive mesh grading iterations for test case *c*.

Iteration	Total CPU Time (min:s)	Nodes	Elements	Memory usage (KB)	$\bar{V}$ (V/m)
1	0:25	965	1915	999 520	0.7475390 + j11.64703
2	0:49	1388	2761	1 763 600	0.7434661 + j12.38945
3	1:24	2285	4555	3 651 088	0.7349815 + j12.75508
4	2:34	4076	8137	8 498 752	0.7273584 + j12.90095
5	5:28	7779	15543	20 338 994	0.7218519 + j13.00763

Table IV(b). Comparison with ANN mesh.

Iteration	Total CPU Time (min:s)	Nodes	Elements	Memory usage (KB)	$\bar{V}$ (V/m)
1	1:39	5598	11181	11 784 544	0.7387412 + j12.90135

ACKNOWLEDGEMENTS

The authors would like to thank Ass. Prof. Jonathan R. Shewchuk for his excellent mesh generator 'triangle' [14, 15], which has been widely used in the present paper. Also, the above work was partially supported by the Research Committee of the Aristotle University of Thessaloniki, Greece.

REFERENCES

1. Cendes ZJ, Shenton DN. Adaptive mesh refinement in the finite element computation of magnetic fields. *IEEE Transactions on Magnetics* 1985; **21**(5):1811–1816.
2. Golias NA, Tsiboukis TD. An approach to refining three-dimensional tetrahedral meshes based on Delaunay transformations. *International Journal for Numerical Methods in Engineering* 1994; **37**:793–812.
3. Dyck DN, Lowther DA. Determining an approximate finite element mesh density using neural network techniques. *IEEE Transactions on Magnetics* 1992; **28**(2):1767–1770.
4. Chedid R, Najjar N. Automatic finite-element mesh generation using artificial neural networks—Part I: prediction of mesh density. *IEEE Transactions on Magnetics* 1996; **32**(5):5173–5178.
5. Lowther DA, Dyck DN. A density driven mesh generator guided by a neural network. *IEEE Transactions on Magnetics* 1993; **29**(2):1927–1930.
6. Ahn C-H *et al*. A self-organizing neural network approach for automatic mesh generation. *IEEE Transactions on Magnetics* 1991; **27**(5):4201–4204.
7. Jeong B-S, Lee S-Y, Ahn C-H. Automatic mesh generator based on self-organizing finite-element tesselation for electromagnetic field problems. *IEEE Transactions on Magnetics* 1995; **31**(3):1757–1760.
8. Kohonen T. The self-organizing map. *IEEE Proceedings* 1990; **78**(9):1464–1480.
9. Alfonzetti S *et al*. Automatic mesh generation by the let-it-grow neural network. *IEEE Transactions on Magnetics* 1996; **32**(3):1349–1352.
10. Triantafyllidis DG, Papagiannis GP, Labridis D. Calculation of overhead transmission line impedances—a finite element approach. *IEEE Transactions on Power Delivery* 1999; **14**(1):287–293.
11. Narenda KS, Parthasarathy K. A diagrammatic representation of back propagation. *Technical Report 8815*, *Center for System Science*, *Department of Electrical Engineering*, *Yale University*, 1988.
12. Maren A, Harston C, Pap R. *Handbook of Neural Computing Applications*. Academic Press: San Diego, California, 1990.

13. Magnuson PC, Traveling waves on multi-conductor open-wire lines: A numerical survey of the effects of frequency dependence on modal decomposition. *IEEE Transactions on Power Apparatus and Systems* 1973; **92**(3):999–1008.
14. Shewchuk JR. Delaunay refinement mesh generation. *Ph.D. Thesis*, *Technical Report CMU-CS-97-137*, *School of Computer Science*, *Carnegie Mellon University*, *Pittsburgh*, *Pennsylvania*, 1997.
15. Shewchuk JR. Triangle: Engineering a 2D quality mesh generator and delaunay triangulator. *First Workshop on Applied Computational Geometry*, Philadelphia, Pennsylvania, Association for Computing Machinery, 1996; 124–133.