

Fuzzy Logic for Scaling Finite Element Solutions of Electromagnetic Fields

Kostas J. Satsios, *Student Member, IEEE*, Dimitris P. Labridis, *Member, IEEE*, and Petros S. Dokopoulos, *Member, IEEE*

Abstract—Artificial intelligence (AI) has been used to determine the quasi-stationary two-dimensional electromagnetic fields within rectangular boundaries. Amplitude and phase of magnetic vector potential have been calculated in an iron slot with an embedded current carrying conductor. A suitable fuzzy neural network (FNN) for scaling finite elements electromagnetic field calculations has been developed. FNN has been trained, using finite elements calculations within rectangular boundaries. Then, FNN has been used to calculate the field in a new geometry differing significantly from the geometries used for training. It was concluded that FNN may be used to scale results from one geometry to another with negligible errors.

Index Terms—Artificial intelligence, eddy currents, finite element methods, fuzzy neural networks.

I. INTRODUCTION

THE FINITE element method (FEM) is a successful numerical method for the solution of complex electromagnetic field problems. The method always provides a stable sequential methodology to define, discretize, assemble, and solve for models with complex geometries. The original problem is transformed to a numerical problem, and the computing time increases with the number of the discretization nodes [1].

A method of scaling the results from one geometry to another may be of interest if it needs shorter computing time than an additional FEM calculation. This scaling may be used in the often encountered case when someone has to present a parametrical result analysis and especially when the boundaries of the geometry are changing. If instead of scaling we use a FEM solver, the FEM solver has to rediscritize every new domain and to solve for the original and usually large number of discretization nodes. Therefore, it is useful to develop a method capable of obtaining a solution of a complex problem by solving with FEM only in a few cases and to have a scaling law for determining the missing cases with acceptable error.

Artificial intelligence (AI) seems to be an efficient method to create systems which are capable of learning relationships and then use this knowledge for further calculations. Fuzzy logic is a research area of AI which is receiving increased attention lately. Fuzzy systems have been successfully applied in system control [2], system identification [3], [4], power

systems transient stability [5], optimal load flow [6], and short-term load forecasting [7].

The purpose of the present work is to investigate whether AI may be also used for the above-mentioned scaling problem. Therefore, a fuzzy system has been developed in this paper for the scaling of the FEM solution from one geometry to another of an electromagnetic problem. This fuzzy system combines the fuzzy inference principles [8] with the neural network structure and learning abilities into an integrated neural network-based fuzzy decision system called a fuzzy neural network (FNN) [7].

The method has been applied in the two-dimensional (2-D) steady-state field-diffusion problem of a rectangular conductor embedded in a slot of an electrical machine, since it has been already treated by other authors using analytical and finite elements methods, [9] and, therefore, a comparison becomes easy.

II. THE METHODS USED

A. Finite Elements Calculations

1) *Electromagnetic Diffusion and Boundary Conditions:* The 2-D electromagnetic diffusion problem for the z direction components of the magnetic vector potential (MVP) A_z and of the total current density vector J_z is described by the system of equations

$$\frac{1}{\mu_0\mu_r} \left[\frac{\partial^2 A_z}{\partial x^2} + \frac{\partial^2 A_z}{\partial y^2} \right] - j\omega\sigma A_z + J_{sz} = 0$$

$$-j\omega\sigma A_z + J_{sz} = J_z \quad (1a)$$

the boundary conditions

$$A_z|_C = A_0(x, y) \quad (1b)$$

and the integral form

$$\iint_S J_z ds = I_{rms} \quad (1c)$$

where μ_0 is the magnetic permeability of vacuum, μ_r is the relative magnetic permeability, J_{sz} is the z direction component of the uniformly distributed source current density, and I_{rms} is the rms of the current flowing through each conductor. The unknowns in the system of (1a) are A_z and J_{sz} , while the values of A_z at the limit C of region S are specified by the Dirichlet condition (1b), and the total current density J_z is specified in the integral form (1c).

Manuscript received December 17, 1995; revised October 10, 1996.

The authors are with the Department of Electrical and Computer Engineering, Power Systems Laboratory, Aristotle University of Thessaloniki, GR-54006 Thessaloniki, Greece.

Publisher Item Identifier S 0018-9464(97)02086-4.

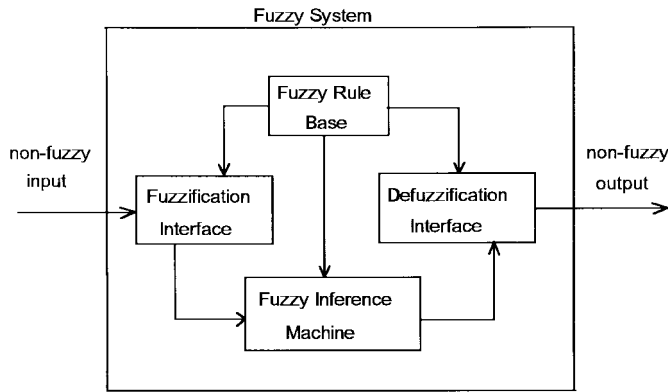


Fig. 1. Basic configuration of fuzzy system.

2) *Finite Elements Formulation*: Applying the Galerkin method to the system of equations (1) and assembling the element contributions in the usual way [10], the following matrix equation is obtained:

$$\begin{bmatrix} \frac{1}{\mu_0 \mu_r} \mathbf{S}_F + j\omega\sigma \mathbf{T}_F & -j\omega\sigma \mathbf{Q} \\ -j\omega\sigma \mathbf{Q}^t & j\omega\sigma \mathbf{W} \end{bmatrix} \begin{bmatrix} \mathbf{A} \\ \mathbf{G} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}. \quad (2)$$

In (2), \mathbf{S}_F and \mathbf{T}_F are the usual finite element matrices encountered in the solution of eddy current problems [1], while the vectors \mathbf{Q} , \mathbf{I} , and \mathbf{G} , and the diagonal matrix \mathbf{W} are defined in [11] for the multiconductor finite element formulation.

B. Description of Fuzzy Neural Networks

We consider the so-called multi-input single-output fuzzy systems. In our problem, inputs are the coordinates x, y from the space $U(x, y \in U)$, and output is the MVP in each node. They have to be transformed and interpreted to and from fuzzy variables in order to use fuzzy logic to solve our problem. So the basic configuration of the fuzzy system used in this paper is shown in Fig. 1, [4] and comprises four principal components: a *fuzzification interface*, a *fuzzy rule base*, a *fuzzy inference machine*, and a *defuzzification interface*.

The *fuzzification interface* performs a scale mapping that transfers the observed nonfuzzy input space $U \subseteq R^n$ to the fuzzy sets defined in U . Hence, the fuzzification interface provides a link between the nonfuzzy outside world and the fuzzy system framework. A fuzzy set [8] defined in U is characterized by a membership function. There are in general many fuzzy sets defined in U .

The *fuzzy rule base* is a set of m rules in general linguistic or conditional statements in the form of: "IF a set of conditions is satisfied, THEN a set of consequences are inferred."

The *fuzzy inference machine* is the decision making logic [3] which employs fuzzy rules from the fuzzy rule base to determine fuzzy outputs of a fuzzy system corresponding to its fuzzified inputs. In this paper, a *fuzzy inference machine* of the form suggested by Takagi–Sugeno [3] is employed, where fuzzy sets are involved only in the premise part (IF-part) of the rules, while the consequent part (THEN-part) is described by a nonfuzzy function of the input variables.

The j th rule, R^j , in our case may be described as follows:

$$\begin{aligned} R^j: & \text{ IF } x \text{ and } y \text{ belong to the } j\text{th membership} \\ & \text{ functions } \mu_x^j \text{ and } \mu_y^j \text{ correspondingly} \\ & \text{ THEN } A^j = \lambda_0^j + \lambda_x^j x + \lambda_y^j y \end{aligned} \quad (3)$$

where R_j ($j = 1, \dots, m$) are the fuzzy rules, x, y are the input variables to the fuzzy system, A^j is the MVP proposed by the j th rule, and μ_x^j, μ_y^j are the membership functions which characterize the j th rule fuzzy sets defined in the space of the variables of x, y coordinates. These membership functions in our case have been chosen as follows:

$$\mu_x^j(x) = \exp \left[-\frac{1}{2} \left(\frac{x - \bar{\alpha}_x^j}{\sigma_x^j} \right)^2 \right] \quad (4a)$$

$$\mu_y^j(y) = \exp \left[-\frac{1}{2} \left(\frac{y - \bar{\alpha}_y^j}{\sigma_y^j} \right)^2 \right] \quad (4b)$$

where $\bar{\alpha}_x^j, \bar{\alpha}_y^j$, and σ_x^j, σ_y^j are the mean values and the standard deviations of the membership distributions, respectively. The factors λ_0^j, λ_x^j , and λ_y^j of the consequent part of the j th rule are to be suitably determined.

The *defuzzification interface* defuzzifies the fuzzy outputs of the fuzzy inference machine and generates a nonfuzzy (crisp) output, which is the actual output of the fuzzy system. The weighted average defuzzification [2], the most commonly used method, is used here.

The crisp output of the fuzzy system, defined above, (also in Fig. 1) is given by

$$A(x, y) = \frac{\sum_{j=1}^m A^j \mu^j}{\sum_{j=1}^m \mu^j} \quad (5)$$

where

$$\mu^j = \mu_x^j(x) \mu_y^j(y) \quad (6)$$

gives the degree of fulfillment of rule R^j by the input vector (x, y) and A^j is given in (3).

The fuzzy system whose crisp output is defined by (5) can be represented by a three layer network as shown in Fig. 2. Due to the similarity of such a fuzzy system to a three layer neural network, it is called an FNN. Another similarity of FNN's with neural networks is their training, described next.

C. Gradient Training of the Fuzzy Neural Network

FEM results of different geometries of the same problem are used to create a suitable training data base (TDB) for the FNN. Using this TDB, it is possible to construct the fuzzy rule base. Fuzzy rule base parameters are determined by a training process, so that the output of the FNN adequately matches the FEM MVP results. Once trained, the FNN may be used to calculate the electromagnetic field for other cases.

The parameters of the FNN to be adjusted, i.e., the tuning parameters, through its training are $\bar{\alpha}_\varepsilon^j, \sigma_\varepsilon^j$ (for $\varepsilon = x, y$ and

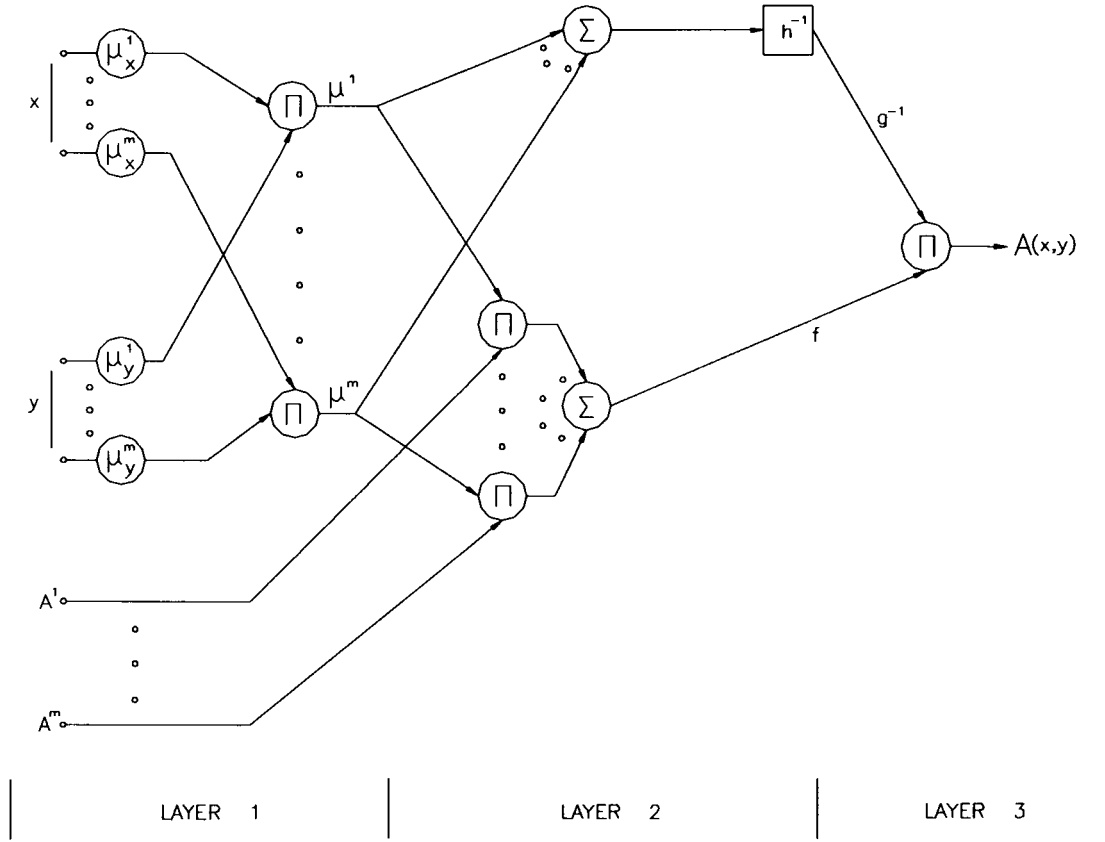


Fig. 2. Network representation of Fuzzy System. Π : product operator. Σ : sum operator. h^{-1} : inversion operator. A^1, \dots, A^m : MVP proposed by the j th ($j = 1, \dots, m$) rule. $A(x, y)$: MVP result.

$j = 1, \dots, m$) and λ_ζ^j (for $\zeta = 0, x, y$ and $j = 1, \dots, m$). Let z denote the vector of the tuning parameters. Initially, it is assumed that the number of rules m is fixed.

The FNN is trained by presenting it with a set of q input/desired output pairs $(x^p, y^p/A_t^p), p = 1, \dots, q$ (q is the number of nodes). A gradient algorithm is then used to tune the FNN, so as to minimize the FNN mean square error

$$J(z) = \frac{1}{q} \sum_{p=1}^q J^p \tag{7}$$

with

$$J^p = \frac{1}{2} [A^p(x, y) - A_t^p(x, y)]^2 \tag{8}$$

where $A^p(x, y)$ and $A_t^p(x, y)$ are the calculate values of MVP at the node p from FNN and FEM, respectively.

Given an input/desired output pair $(x^p, y^p/A_t^p)$, the gradients of the square error J^p with respect to the system parameters are

$$\nabla_{\alpha_\varepsilon^j} J^p = \frac{\mu^j}{m} (A^p(x, y) - A_t^p(x, y)) (A^j - A^p(x, y)) \sum_{j=1}^m \mu^j \cdot \frac{\varepsilon^p - \bar{\alpha}_\varepsilon^j}{\sigma_\varepsilon^{j^2}} \tag{9a}$$

$$\nabla_{\sigma_\varepsilon^j} J^p = \frac{\mu^j}{m} (A^p(x, y) - A_t^p(x, y)) (A^j - A^p(x, y)) \cdot \frac{\varepsilon^p - \bar{\alpha}_\varepsilon^j}{\sigma_\varepsilon^{j^3}} \tag{9b}$$

$$\nabla_{\lambda_\zeta^j} J^p = \frac{\mu^j}{m} (A^p(x, y) - A_t^p(x, y)) \kappa^p \sum_{j=1}^m \mu^j \tag{9c}$$

where $\varepsilon = x, y, \zeta = 0, x, y$, and $\kappa = 1, x, y$.

The minimization of $J(z)$ in (7) through a gradient algorithm leads to the learning rule which is expressed by the following formula:

$$z(\nu + 1) = z(\nu) - n \nabla_z J \tag{10a}$$

$$\nabla_z J = \frac{1}{q} \sum_{p=1}^q \nabla_z J^p \tag{10b}$$

where n is an acceleration factor, ν is the iteration index, and the gradient $\nabla_z J^p$ is computed using (9a)–(9c).

D. Rule Creation

1) *Initialization of the FNN Rules:* The number of rules m can be arbitrarily determined. This may in general lead to long training time and large errors. To improve the training time and reduce the errors, m is determined by a sequential procedure.

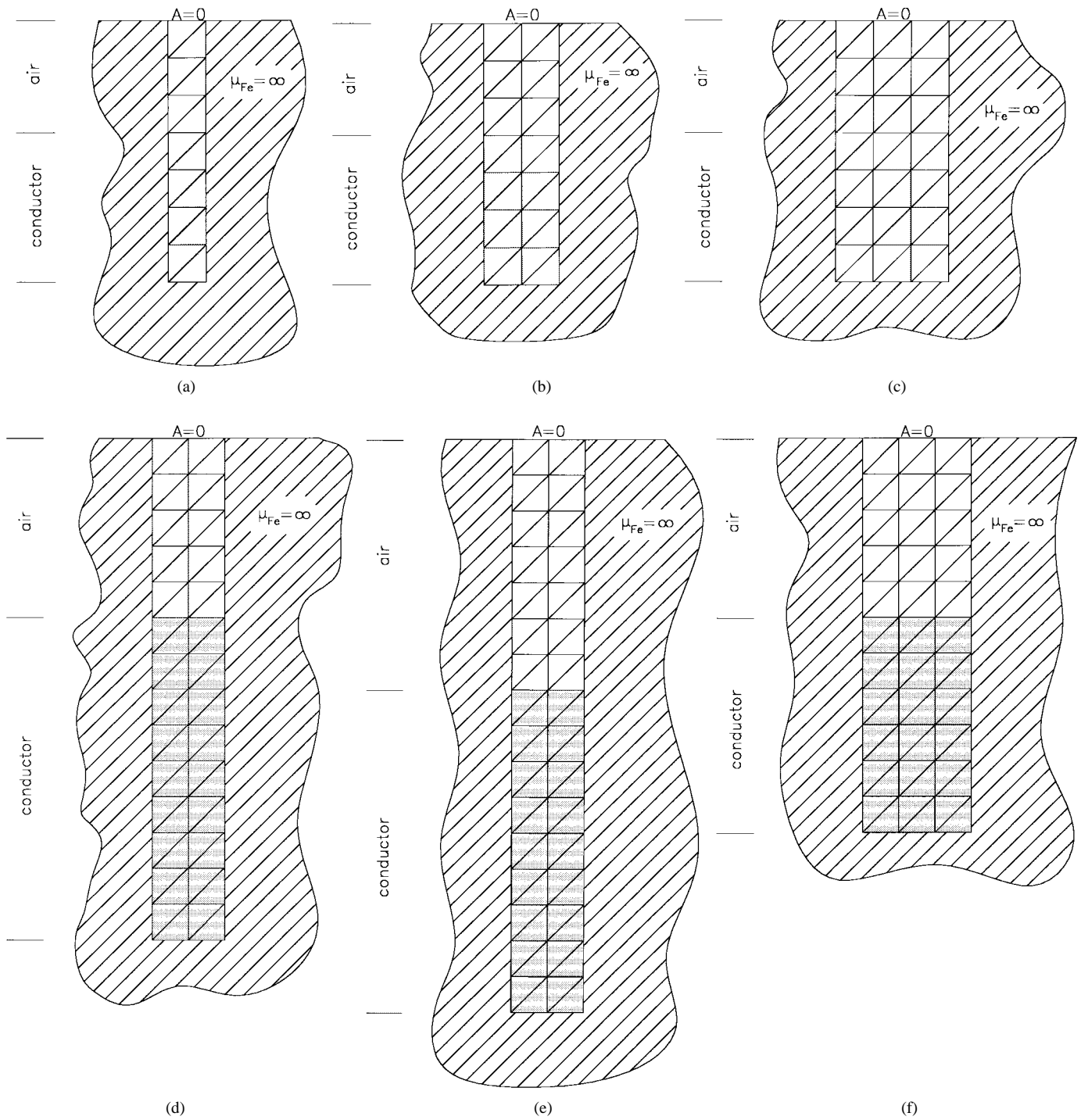


Fig. 3. (a) Geometry 1. (b) Geometry 2. (c) Geometry 3. (d) Geometry 4. (e) Geometry 5. (f) Geometry 6.

We start with a certain initialization of rules with a single rule ($m = 1$), and we use a rule base adaptation procedure [7]. The parameters of $m = 1$ fuzzy rule are initialized on the basis of the first input/desired output sample pair $(x^1, y^1/A_t^1)$ as follows [4]:

$$\begin{aligned} \bar{\alpha}_\varepsilon^1 &= \varepsilon^1 \\ \sigma_\varepsilon^1 &= \frac{1}{2m} [\max(\varepsilon^p) - \min(\varepsilon^p)] \\ \lambda_0^1 &= A_t^1, \lambda_\varepsilon^1 = 0, \quad (\text{for } \varepsilon = x, y; p = 1, \dots, q). \end{aligned} \quad (11)$$

The choosing method for the parameters described above performs the function of fuzzification that converts input data into suitable membership values, which may be viewed as labels of fuzzy sets. The mean values of the memberships are centered directly at point (x^1, y^1) , while the membership deviations reflect the degree of fuzzification and are selected in such a way that allows overlaps of membership functions μ_x^1, μ_y^1 .

2) *Rule Base Adaptation:* This procedure starts with the initialization of the first ($m = 1$) rule. The gradient training

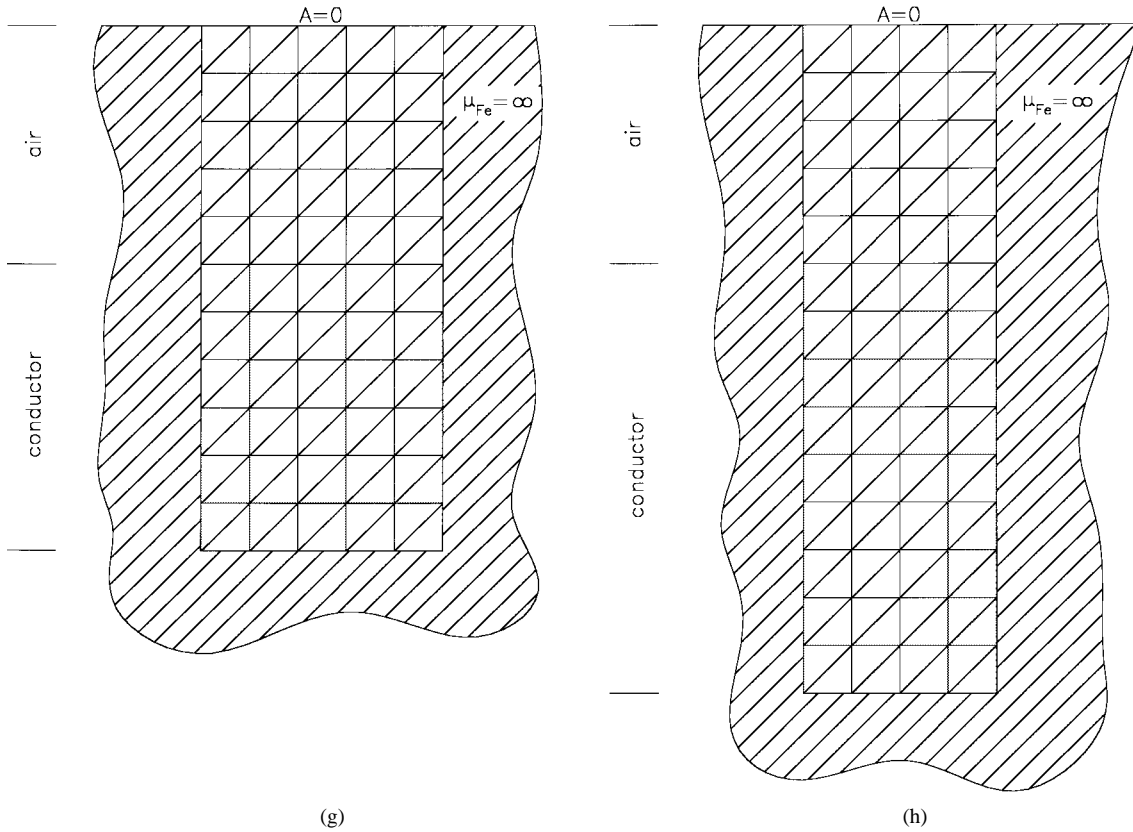


Fig. 3. (Continued). (g) Geometry 7. (h) Geometry 8.

algorithm described by (9a)–(9c) is used to train the FNN based on input/desired output p pairs. When the procedure has reached m rules, we consider an additional new training pattern $(x^p, y^p/A_t^p)$.

Let us denote the *firing strength* of the fuzzy rule base as

$$S_\mu(x^p, y^p) = \sum_{j=1}^m \mu^j. \quad (12)$$

We also define a threshold β as the *least acceptable firing strength* of the fuzzy rule base. If $S_\mu(x^p, y^p) < \beta$, a new rule R^{m+1} must be added to the rule base. Let $\mu_\varepsilon^{m+1}(\bar{\alpha}_\varepsilon^{m+1}, \sigma_\varepsilon^{m+1})$ denote the new membership in the ε th premise axis. Then, the parameters of μ_ε^{m+1} are selected as follows:

$$\bar{\alpha}_\varepsilon^{m+1} = \varepsilon^p \quad (13a)$$

$$\sigma_\varepsilon^{m+1} = \gamma(\varepsilon^p - \bar{\alpha}_\varepsilon^{\text{nearest}}) \quad (13b)$$

where $\varepsilon = x, y, \bar{\alpha}_\varepsilon^{\text{nearest}}$ is the mean value of an existing membership closest to the incoming pattern vector ε^p and γ is an overlapping factor which has been chosen equal to two from computer experiments.

The generation of new rules establishes the rule base adaptation mechanism which is described by the following steps.

Step 1) Feed forward the new pattern x^p through the FNN and compute the corresponding firing factor $S_\mu(x^p, y^p)$.

Step 2) If $S_\mu(x^p, y^p) \geq \beta$, then leave the rule base unchanged and perform Gradient Training to match the new sample pair.

TABLE I
GEOMETRIES USED

GEOMETRY	NUMBER OF NODES	NUMBER OF ELEMENTS	HEIGHT	WIDTH
1	16	14	7	1
2	24	28	7	2
3	32	42	7	3
4	45	56	14	2
5	51	64	16	2
6	48	66	11	3
7	72	110	11	5
8	75	112	14	4

Step 3) If $S_\mu(x^p, y^p) < \beta$, then create a new fuzzy rule R^{m+1} , select parameters according to (13), and perform Gradient training on the expanded fuzzy rule base.

To summarize, the overall FNN training procedure comprises three major parts: the rule base initialization, the rule base adaptation, and the gradient learning algorithm. The proposed training scheme offers the advantage of including just the necessary fuzzy rules within the premise space leading to a minimum of FNN parameters for training.

III. APPLICATION OF THE METHOD

The 2-D steady-state field diffusion problem of a slot-embedded conductor with the data defined in [9] has been used to test the proposed method. This problem has been selected because there is an analytical solution in the corresponding one-dimensional case. The FEM solution has been compared to the analytical one [9], [10], and the differences were negligible. So, FEM may be considered as a reference for the FNN calculated results and the corresponding error estimations.

TABLE II
MVP AMPLITUDE IN THE CONDUCTOR OF THE GEOMETRY SHOWN IN FIG. 3(h) AS OBTAINED BY FEM AND FNN

COORDINATES		MVP AMPLITUDE (Wb/m)		$\left \frac{FEM - FNN}{FEM} \right \times 100$
X	Y	FEM	FNN	
0.00	0.00	6.7077	6.7040	0.0555
1.00	0.00	6.7078	6.7050	0.0421
2.00	0.00	6.7081	6.7069	0.0183
3.00	0.00	6.7084	6.7081	0.0045
4.00	0.00	6.7087	6.7096	0.0133
0.00	1.00	6.7063	6.7111	0.0719
1.00	1.00	6.7070	6.7114	0.0659
2.00	1.00	6.7076	6.7122	0.0679
3.00	1.00	6.7082	6.7180	0.1460
4.00	1.00	6.7088	6.7194	0.1576
0.00	2.00	6.7080	6.7530	0.6706
1.00	2.00	6.7093	6.7535	0.6591
2.00	2.00	6.7104	6.7562	0.6830
3.00	2.00	6.7115	6.7632	0.7700
4.00	2.00	6.7125	6.7657	0.7930
0.00	3.00	6.7268	6.7826	0.8299
1.00	3.00	6.7276	6.7778	0.7456
2.00	3.00	6.7288	6.7762	0.7051
3.00	3.00	6.7299	6.7902	0.8961
4.00	3.00	6.7304	6.8003	1.0387
0.00	4.00	6.7783	6.8382	0.8837
1.00	4.00	6.7765	6.8326	0.8280
2.00	4.00	6.7762	6.8306	0.8031
3.00	4.00	6.7758	6.8363	0.8932
4.00	4.00	6.7738	6.8428	1.0189
0.00	5.00	6.8563	6.8944	0.5554
1.00	5.00	6.8489	6.8923	0.6344
2.00	5.00	6.8454	6.8902	0.6549
3.00	5.00	6.8419	6.8847	0.6256
4.00	5.00	6.8348	6.8705	0.5229
0.00	6.00	6.8872	6.8655	0.3158
1.00	6.00	6.8743	6.8493	0.3640
2.00	6.00	6.8683	6.8370	0.4555
3.00	6.00	6.8623	6.8345	0.4053
4.00	6.00	6.8501	6.8418	0.1218
0.00	7.00	6.6850	6.6841	0.0128
1.00	7.00	6.6767	6.6778	0.0164
2.00	7.00	6.6740	6.6734	0.0084
3.00	7.00	6.6716	6.6724	0.0120
4.00	7.00	6.6634	6.6735	0.1517

The slot is embedded in infinitely permeable iron. The permeabilities of both conductor and air are equal to $\mu = 1$. The sheet conductivity is $\sigma = 1$, and the conductor carries a total measurable sheet current $I = 4$ A at a frequency of $\omega = 1$. The boundary condition on the MVP is that $A = 0$ at the top of the air gap.

Our main purpose is to train a FNN in different rectangular geometries of this problem, keeping constant permeabilities, conductivities of materials, and total current of the conductor so that the MVP distribution in a new geometry can be calculated.

A. Selection of the FNN Input and Output Variables

A two-input single-output FNN has been used. The inputs and the output of the FNN are as follows:

$$\begin{aligned}
 \text{inputs:} & \quad x = \text{coordinate } x(p), \quad y = \text{coordinate } y(p) \\
 \text{output:} & \quad A = \text{amplitude of magnetic vector potential} \\
 & \quad A(p) \text{ or} \\
 \text{output:} & \quad \varphi = \text{phase of magnetic vector potential } A(p)
 \end{aligned}
 \tag{14}$$

where $p = 1, \dots, q$, q is the node index and q is the number of nodes of discretization.

The MVP of the steady state problem is expressed using complex phasors, and, therefore, it consists of two parts: the amplitude and the phase. Since the FNN method has a single output, two different FNN are required to calculate MVP nodal values. The first FNN must be trained in order to match the amplitude and the second one in order to match the phase of $A(p)$, i.e., node p with $x(p), y(p)$ coordinates.

The FNN would in general need information of the relative position the conductor occupies within the slot. So, if we decide to use a single FNN for the total solution region, we have to introduce as an input the height of the conductor. Instead of doing this, two FNN have been used, one for the conductor region and the other for the air region. The first FNN has been used for scaling results in the conductor and the other in the air. The results have shown that this multiple-FNN method is more efficient than a single-FNN method. The physical explanation of the above may be that the separation in two different regions materials helped the FNN to understand better the material limits as well as the material interfaces.

TABLE III
MVP AMPLITUDE IN THE AIR OF THE GEOMETRY SHOWN IN FIG. 3(h) AS OBTAINED BY FEM AND FNN

COORDINATES		MVP AMPLITUDE (Wb/m)		$\left \frac{FEM - FNN}{FEM} \right \times 100$
X	Y	FEM	FNN	
0.00	8.00	5.970	5.833	2.283
1.00	8.00	5.992	5.843	2.479
2.00	8.00	6.000	5.888	1.871
3.00	8.00	6.008	5.911	1.627
4.00	8.00	6.031	5.930	1.661
0.00	9.00	4.988	4.825	3.263
1.00	9.00	4.994	4.819	3.517
2.00	9.00	5.000	4.867	2.667
3.00	9.00	5.006	4.857	2.969
4.00	9.00	5.012	4.872	2.795
0.00	10.00	3.995	3.923	1.800
1.00	10.00	3.997	3.918	1.985
2.00	10.00	4.000	3.930	1.738
3.00	10.00	4.003	3.876	3.166
4.00	10.00	4.005	3.738	6.677
0.00	11.00	2.998	2.864	4.445
1.00	11.00	2.999	2.866	4.430
2.00	11.00	3.000	2.857	4.758
3.00	11.00	3.002	2.873	4.270
4.00	11.00	3.002	2.975	0.923
0.00	12.00	1.999	1.909	4.524
1.00	12.00	1.999	1.907	4.635
2.00	12.00	2.000	1.926	3.677
3.00	12.00	2.001	1.937	3.196
4.00	12.00	2.001	2.024	1.140
0.00	13.00	1.000	1.001	0.139
1.00	13.00	1.000	0.997	0.222
2.00	13.00	1.000	1.014	1.430
3.00	13.00	1.000	0.995	0.552
4.00	13.00	1.000	0.998	0.211

So a total of $2 \times 2 = 4$ FNN's is required. After their training, two of them are used to calculate MVP, amplitude and phase, respectively, in the conductor discretization nodes and the other two in the air discretization nodes.

B. Training of the FNN

In order to test the proposed method, eight different geometries shown in Fig. 3(a)–(h) are considered with their data defined in Table I. These geometries are used to train and test the performance of the FNN. Applying the FEM in these geometries, suitable results are obtained to construct TDB's for FNN's. The training procedure of each FNN is described as follows.

The first geometry is randomly chosen between the eight constructed geometries. The rule base of the FNN is initialized so that it contains only one rule ($m = 1$), defined by the first input–output pair of the corresponding data set to this geometry. The FNN is then trained using this data set, which consists of input-output data pairs with the form of (14). Additional rules are created during the training process as described before. The training was made with a mean absolute error of 0.5% or less. At the end of the FNN training, the rule base is stored. Then, a second geometry is randomly chosen from the set of geometries. The FNN is trained using the existing rule base, and the corresponding data is set to this geometry. The same procedure is followed to train the FNN totally in the seven out of the eight geometries. At the end of the training procedure, the FNN rule base contains ten

rules on the average. The performance of the FNN is tested in the additional eighth geometry, for which FNN is not trained. The corresponding FEM solution is also obtained in order to compare the FNN results. The method has been also tested additionally in the seven other cases. For each case, seven other geometries have been used to train the FNN, and the remaining geometry has been used as a test object.

IV. TEST RESULTS

The method has been tested in eight different cases as previously mentioned. In general, the deviations between FEM and FNN calculations are nearly the same in all cases. In the following, results from only one case, shown in Fig. 3(h) and the most pessimistic one, are presented separately for the amplitude and the phase of MVP and for the conductor and the air regions.

Table II summarizes the results for the amplitude. The results of our method are compared with results obtained using the FEM described in Section II. The amplitude of nodal MVP using FEM and FNN, as well as the corresponding relative errors between FEM and FNN, are being reported. For this case, the average absolute error between FEM and FNN is 0.44%. MVP amplitude estimation using FNN in the air of the geometry shown in Fig. 3(h) is given in Table III, which has the same structure with Table II. In this case, the average absolute error is 2.63%.

MVP phase calculations in the conductor and in the air are given in Tables IV and V, respectively. The angle of MVP

TABLE IV
MVP PHASE IN THE CONDUCTOR OF THE GEOMETRY SHOWN IN FIG. 3(h) AS OBTAINED BY FEM AND FNN

COORDINATES		MVP PHASE (DEGREES)		<i>FEM - FNN</i>
X	Y	FEM	FNN	
0.000	0.000	-6.235	-6.565	0.330
1.000	0.000	-6.243	-6.565	0.322
2.000	0.000	-6.249	-6.565	0.317
3.000	0.000	-6.254	-6.564	0.311
4.000	0.000	-6.260	-6.558	0.298
0.000	1.000	-6.282	-6.565	0.284
1.000	1.000	-6.283	-6.565	0.282
2.000	1.000	-6.286	-6.565	0.279
3.000	1.000	-6.289	-6.565	0.276
4.000	1.000	-6.290	-6.562	0.272
0.000	2.000	-6.407	-6.565	0.158
1.000	2.000	-6.401	-6.565	0.164
2.000	2.000	-6.398	-6.565	0.167
3.000	2.000	-6.394	-6.565	0.171
4.000	2.000	-6.388	-6.564	0.176
0.000	3.000	-6.578	-6.560	0.019
1.000	3.000	-6.558	-6.560	0.002
2.000	3.000	-6.542	-6.559	0.017
3.000	3.000	-6.526	-6.558	0.032
4.000	3.000	-6.507	-6.557	0.050
0.000	4.000	-6.609	-6.253	0.357
1.000	4.000	-6.573	-6.246	0.327
2.000	4.000	-6.543	-6.229	0.314
3.000	4.000	-6.513	-6.199	0.314
4.000	4.000	-6.482	-6.160	0.322
0.000	5.000	-6.082	-5.139	0.943
1.000	5.000	-6.053	-5.145	0.908
2.000	5.000	-6.020	-5.139	0.880
3.000	5.000	-5.987	-5.110	0.876
4.000	5.000	-5.962	-5.085	0.877
0.000	6.000	-4.484	-3.612	0.872
1.000	6.000	-4.512	-3.642	0.870
2.000	6.000	-4.502	-3.651	0.851
3.000	6.000	-4.491	-3.573	0.917
4.000	6.000	-4.519	-3.496	1.022
0.000	7.000	-1.801	-1.786	0.015
1.000	7.000	-1.917	-1.858	0.059
2.000	7.000	-1.939	-1.945	0.006
3.000	7.000	-1.962	-1.973	0.010
4.000	7.000	-2.069	-1.979	0.091

phasors in the discretization nodes using FEM and FNN as well as relative errors between FEM and FNN are reported. The average absolute error in the conductor is 0.37 degrees and in the air is 0.013 degrees.

From Tables II-V, it is evident that the resulting MVP from FNN is comparable to the corresponding FEM results. Once FNN is trained, the field in other new geometries can be calculated, and the computing time is negligibly small in comparison with the time needed for FEM calculations of the new geometry. In the reported cases, the training of one FNN requires a time approximately equal to the seven FEM calculations.

The frequency distribution of the relative absolute error of MVP amplitude between FEM and FNN in the conductor and in the air of the geometry shown in Fig. 3(h) are in Fig. 4 and Fig. 5, respectively. From Fig. 4, it can be shown that, in the conductor, the 80% of the errors is less than 0.8%. Similarly, Fig. 5 shows that in the air the 80% of the errors is less than 4%.

Figs. 6 and 7 show a three-dimensional representation of MVP amplitude as a function of the node's x, y coordinates

Frequency (%)

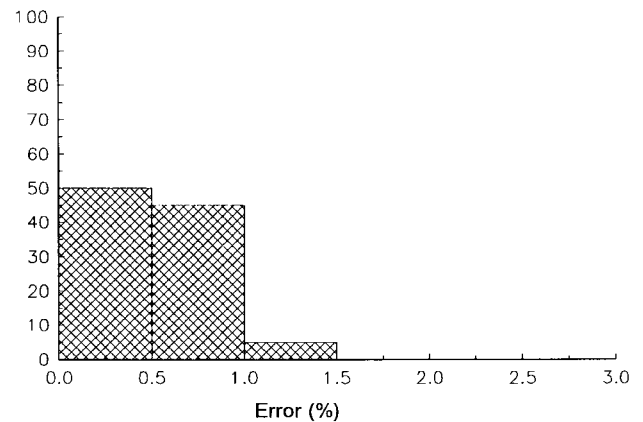


Fig. 4. Frequency distribution of errors between the amplitudes of MVP calculations by FEM and FNN in the conductor of the geometry shown in Fig. 3(h).

in the conductor and in the air of the geometry shown in Fig. 3(h), as it is calculated by FEM and FNN. In both cases, FNN seems to follow satisfactory the results of FEM.

TABLE V
MVP PHASE IN THE AIR OF THE GEOMETRY SHOWN IN FIG. 3(h) AS OBTAINED BY FEM AND FNN

COORDINATES		MVP PHASE (DEGREES)		FEM - FNN
X	Y	FEM	FNN	
0.00	8.00	-0.103	-0.098	0.005
1.00	8.00	-0.016	-0.047	0.031
2.00	8.00	0.002	-0.002	0.004
3.00	8.00	0.017	0.000	0.017
4.00	8.00	0.097	0.005	0.092
0.00	9.00	-0.045	-0.058	0.014
1.00	9.00	-0.018	-0.025	0.006
2.00	9.00	0.001	-0.005	0.006
3.00	9.00	0.019	-0.001	0.019
4.00	9.00	0.043	0.005	0.037
0.00	10.00	-0.023	-0.026	0.003
1.00	10.00	-0.013	-0.019	0.006
2.00	10.00	0.000	-0.009	0.009
3.00	10.00	0.013	-0.001	0.014
4.00	10.00	0.022	0.007	0.015
0.00	11.00	-0.014	-0.020	0.006
1.00	11.00	-0.009	-0.017	0.008
2.00	11.00	0.000	-0.010	0.010
3.00	11.00	0.009	0.000	0.009
4.00	11.00	0.013	0.008	0.005
0.00	12.00	-0.009	-0.018	0.009
1.00	12.00	-0.006	-0.015	0.009
2.00	12.00	0.000	-0.009	0.009
3.00	12.00	0.006	0.001	0.005
4.00	12.00	0.009	0.009	0.000
0.00	13.00	-0.007	-0.018	0.011
1.00	13.00	-0.005	-0.015	0.010
2.00	13.00	0.000	-0.008	0.008
3.00	13.00	0.005	0.002	0.003
4.00	13.00	0.007	0.010	0.003

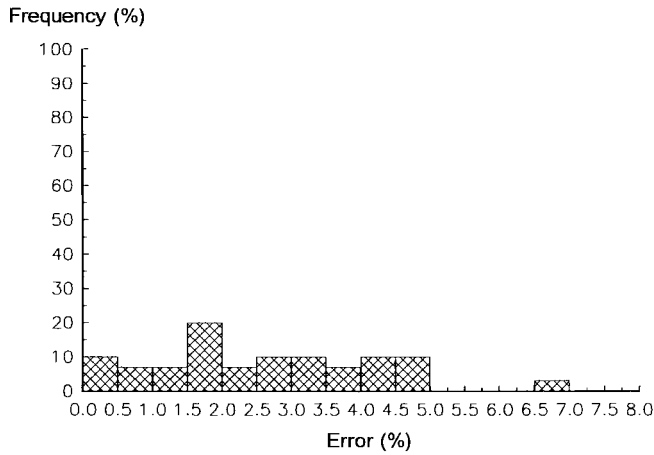


Fig. 5. Frequency distribution of errors between the amplitudes of MVP calculations by FEM and FNN in the air of the geometry shown in Fig. 3(h).

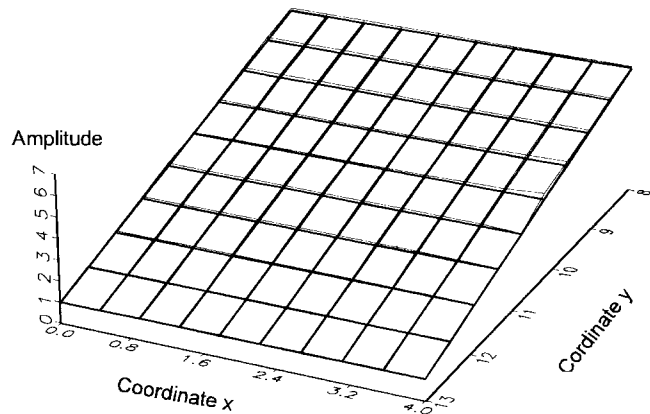


Fig. 7. FEM and FNN MVP amplitude computation (slim and bold line, respectively) versus x, y coordinates in the air of the geometry shown in Fig. 3(h).

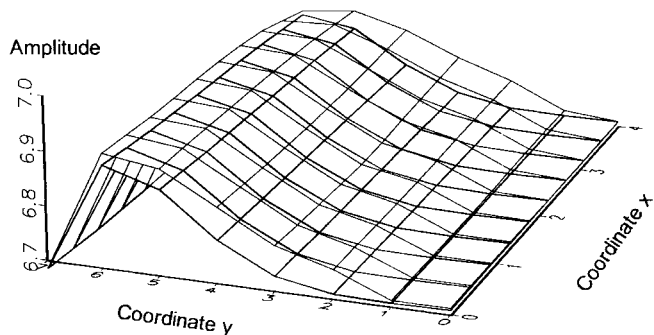


Fig. 6. FEM and FNN MVP amplitude computation (slim and bold line, respectively) versus x, y coordinates in the conductor of the geometry shown in Fig. 3(h).

V. CONCLUSIONS

FNN's may be used for scaling results in electromagnetic field calculations. The proposed FNN has given satisfactory results in the electromagnetic field diffusion problem of the current carrying conductor embedded in an iron slot.

It could be deduced that after suitable training FNN has a comparable accuracy with FEM in the above field diffusion problem, while it needs negligible computing times. The training of the FNN becomes faster if the problem domain is subdivided into two material-defined regions with separate FNN's in each one. The fact that the initialization parameters of the FNN are not randomly chosen but are assigned reasonable values accelerates the training procedure of the FNN.

REFERENCES

- [1] P. Silvester and R. Ferrari, *Finite Elements for Electrical Engineers*. Cambridge: Cambridge Univ. Press, 1983.
- [2] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller—Part I and II," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, no. 2, pp. 404–435, 1990.
- [3] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, pp. 116–132, 1985.
- [4] L. X. Wang and J. M. Mendel, "Back-propagation fuzzy system as nonlinear dynamic system identifiers," in *Proc. FUZZ-IEEE '92*, pp. 1409–1418.
- [5] J. L. Souflis, A. V. Machias, and B. C. Papadias, "An application of fuzzy concepts to transient stability evaluation," *IEEE Trans. Power Syst.*, vol. 4, no. 3, pp. 1003–1009, 1989.
- [6] V. Miranda and J. T. Saraiva, "Fuzzy modeling of power system optimal load flow," *IEEE Trans. Power Syst.*, vol. 7, no. 2, pp. 843–849, 1992.
- [7] A. G. Bakirtzis, J. B. Theocharis, S. J. Kiartzis, and K. J. Satsios, "Short term load forecasting using fuzzy neural networks," *IEEE Trans. Power Syst.*, vol. 10, no. 3, pp. 1518–1524, 1995.
- [8] L. Zadeh, "Outline of a new approach to the analysis of complex systems and decision process," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 1, Jan. 1973.
- [9] A. Konrad, "The numerical solution of steady-state skin effect problems—an integrodifferential approach," *IEEE Trans. Magn.*, vol. MAG-17, no. 1, pp. 1148–1152, 1981.
- [10] J. Weiss and Z. Csendes, "A one-step finite element method for multiconductor skin effect problems," *IEEE Trans. Power App. Syst.*, vol. PAS-101, no. 10, pp. 3796–3803, Oct. 1982.
- [11] D. Labridis and P. Dokopoulos, "Finite element computation of field losses and forces in a three-phase gas cable with nonsymmetrical conductor arrangement," *IEEE Trans. Power Delivery*, vol. 3, no. 4, pp. 1326–1333, Oct. 1988.

Kostas J. Satsios (S'94) was born in Serres, Greece, in May 1971. He received the Dipl. Eng. degree from the Department of Electrical Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1994. Since 1994, he has been a Ph.D. student in the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece.

His research interests are in finite elements and artificial intelligence applications in power systems.

Mr. Satsios is a member of the Society of Professional Engineers of Greece.

Dimitris P. Labridis (S'87–M'90) was born in Thessaloniki, Greece, on July 26, 1958. He received the Dipl.-Eng. and Ph.D. degrees from the Department of Electrical Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1981 and 1989, respectively.

From 1982 to 1989, he worked as a Research Assistant at the Department of Electrical Engineering, Aristotle University of Thessaloniki. Since 1994, he has been an Assistant Professor at the Department of Electrical Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece. His special interests are power system analysis with special emphasis on the simulation of transmission and distribution systems, electromagnetic and thermal field analysis, and numerical methods in engineering.

Petros S. Dokopoulos (M'94) was born in Athens, Greece, in September 1939. He received the Dipl. Eng. degree from the Technical University of Athens, Greece, in 1962 and the Ph.D. degree from the University of Brunswick, Germany, in 1967.

From 1962 to 1967, he was with the High Voltage Laboratory at the University of Brunswick, Germany. He was with the Nuclear Research Center at Jülich, Germany, from 1967 to 1974. From 1974 to 1978, he was with the Joint European Torus. Since 1978 he has been a Professor at the Department of Electrical Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece. He has worked as consultant to Brown Boveri and Cie, Mannheim, Germany; Siemens, Erlangen, Germany; Public Power Corporation, Greece; and the National Telecommunication Organization, Greece. His scientific fields of interest are dielectrics, power switches, generators, power cables, and alternative sources. He has 55 publications and seven patents on these subjects.