Neural-Network Methods for Boundary Value Problems with Irregular Boundaries

Isaac Elias Lagaris, Aristidis C. Likas, *Member, IEEE*, and Dimitrios G. Papageorgiou

***Abstract*—Partial differential equations (PDEs) with boundary conditions (Dirichlet or Neumann) defined on boundaries with simple geometry have been successfully treated using sigmoidal multilayer perceptrons in previous works. This article deals with the case of complex boundary geometry, where the boundary is determined by a number of points that belong to it and are closely located, so as to offer a reasonable representation. Two networks are employed: a multilayer perceptron and a radial basis function network. The later is used to account for the exact satisfaction of the boundary conditions. The method has been successfully tested on two-dimensional and three-dimensional PDEs and has yielded accurate results.**

***Index Terms*—Boundary value problems, engineering problems,**

In this way, the problem is transformed into the following constrained minimization problem:

$$\min_p E(p) = \sum_{i=1}^{K} (L\Psi_M(\vec{x}_i, p) - f(\vec{x}_i))^2 \tag{5}$$

subject to the constraints imposed by the BCs

$$\Psi_M(\vec{r}_i, p) = b_i, \quad i = 1, 2, \ldots, M \quad \text{(Dirichlet)} \tag{6}$$

or

$$\vec{n}_i \cdot \nabla\Psi_M(\vec{r}_i, p) = c_i, \quad i = 1, 2, \ldots, M \quad \text{(Neumann).} \tag{7}$$

The constrained optimization problem above may be tackled in a number of ways.

1) Devise a model $\Psi_M(\vec{x}, p)$, such that the constraints are exactly satisfied by construction and hence use unconstrained optimization techniques.
2) Use a suitable constrained optimization method for nonlinear constraints. For instance: Lagrange multipliers, active set methods, or a penalty function approach [11].

A model suitable for the first approach is based on the synergy of two feedforward neural networks of different type, and it can be written as

$$\Psi_M(\vec{x}, p) = N(\vec{x}, p) + \sum_{l=1}^{M} q_l e^{-\lambda|\vec{x} - \alpha\vec{r}_l + \vec{h}|^2} \tag{8}$$

where $|\cdot|$ denotes the Euclidean norm and $N(\vec{x}, p)$ is a multilayer perceptron (MLP) with p denoting the set of its weights and biases. The sum in the above equation represents an RBF network with M hidden units that all share a common exponential factor λ. Here α is a scalar constant and $\vec{h}$ is a constant vector. The parameters λ, α and $\vec{h}$ are chosen appropriately so as to ease the numerical task.

For a given set p of MLP parameters, the coefficients q_l are uniquely determined by requiring that the boundary conditions are satisfied, i.e.,

$$b_i - N(\vec{r}_i, p) = \sum_{l=1}^{M} q_l e^{-\lambda|\vec{r}_i - \alpha\vec{r}_l + \vec{h}|^2} \quad \text{(Dirichlet)} \tag{9}$$

or

$$\begin{aligned} c_i - \vec{n}_i \cdot \nabla N(\vec{r}_i, p) = {} & -2\lambda \sum_{l=1}^{M} q_l e^{-\lambda|\vec{r}_i - \alpha\vec{r}_l + \vec{h}|^2} \vec{n}_i \\ & \cdot (\vec{r}_i - \alpha\vec{r}_l + \vec{h}) \quad \text{(Neumann)} \end{aligned} \tag{10}$$

for $i = 1, 2, \ldots, M$. Therefore, in order to obtain the parameters q_i that satisfy the BCs, one has to solve a linear system, which for the Dirichlet case reads

$$Aq = \gamma, \quad A_{ij} = e^{-\lambda|\vec{r}_i - \alpha\vec{r}_j + \vec{h}|^2}, \quad \gamma_i = b_i - N(\vec{r}_i, p) \tag{11}$$

while for the Neumann case

$$Kq = \delta, \quad K_{ij} = -$$

listed as follows:

$$\frac{\partial \Psi_M(\vec{x},p)}{\partial p} = \frac{\partial N(\vec{x},p)}{\partial p} + \sum_{l=1}^{M} \frac{\partial q_l}{\partial p} e^{-\lambda(\vec{x}-\alpha\vec{r}_l+\vec{h})^2}. \qquad (16)$$

Since for the case of Dirichlet BCs (11)

$$q_l = \sum_{i=1}^{M} A_{li}^{-1}(b_i - N(\vec{r}_i,p)) \qquad (17)$$

and for the case of Neumann BCs (12)

$$q_l = \sum_{i=1}^{M} K_{li}^{-1}(c_i - \vec{n}_i \cdot \nabla N(\vec{r}_i,p)) \qquad (18)$$

we correspondingly get

$$\frac{\partial q_l}{\partial p} = -\sum_{i=1}^{M} A_{li}^{-1} \frac{\partial N(\vec{r}_i,p)}{\partial p} \qquad (19)$$

and

$$\frac{\partial q_l}{\partial p} = -\sum_{i=1}^{M} K_{li}^{-1} \vec{n}_i \cdot \nabla \frac{\partial N(\vec{r}_i,p)}{\partial p} \qquad (20)$$

i.e., one has to solve as many $M \times M$ linear systems as the number of the parameters p. Derivatives of the MLP with respect to either the parameters p or the input variables $\vec{x}$ can be easily derived and are given in [1], [2].

In order to apply the proposed method, the values of λ, α and $\vec{h}$ must be specified. These define the linear systems (matrices A and K). In our experiments the linear systems were solved using standard LU decomposition. We did not use any special methods for sparse linear systems nor any parallel programming techniques.

For the Dirichlet case $\alpha = 1$ and $\vec{h} = 0$ were adequate to produce a nonsingular well-behaved matrix A. For large values of λ, the Gaussian terms in the RBF are all highly localized so that they affect the model only in the neighborhood of the boundary points. In other words, the RBF contributes a "correction" that accounts for the BCs. For small values of λ, the matrix looses rank and becomes singular. So λ must be selected with caution. A good choice is found to be: $\lambda \approx 1/d$, where d is the minimum distance between any two points on the boundary, i.e., $d = \min_{i,j}[|\vec{r}_i - \vec{r}_j|]$, where $i,j = 1,2,\ldots,M$. Note that different values λ_i may also be used instead of a common one. In that case the corresponding d_i would be the distance of the closest boundary neighbor to point $\vec{r}_i$, i.e.,

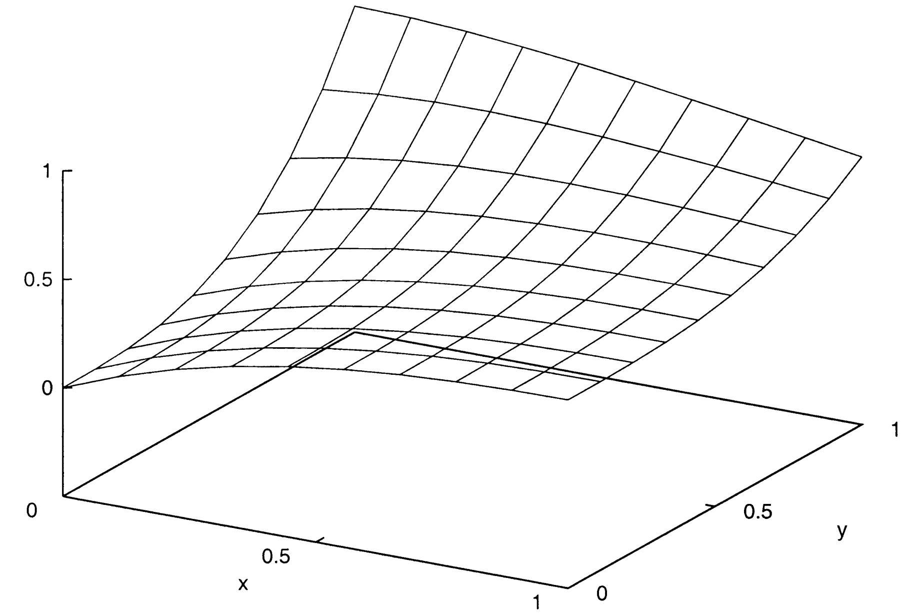

Fig. 1. Exact solution of problem 1.

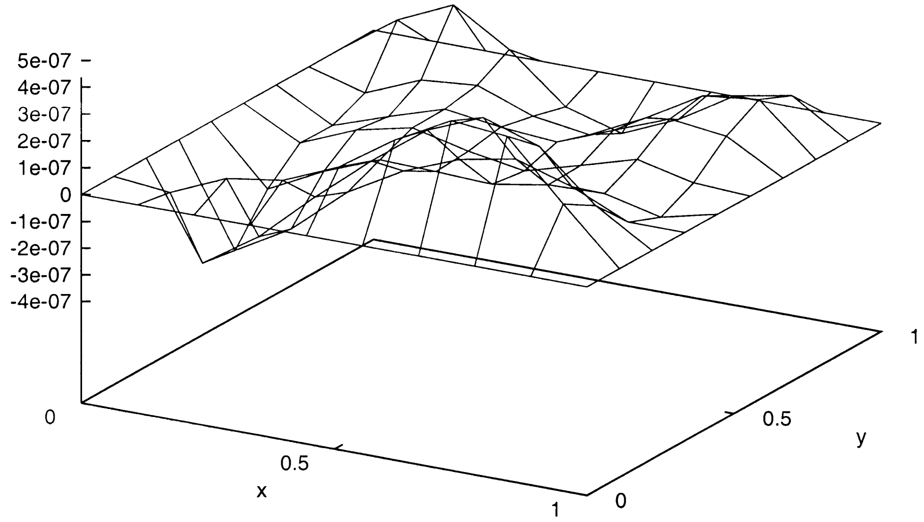

Fig. 2. Accuracy of obtained solution for problem 1 with Dirichlet BCs.

A. 2-D Problems

Problem 1: Consider the linear problem

$$\nabla^2\Psi(x,y) = e^{-x}(x-2+y^3+6y), \quad x,y \in [0,1] \tag{21}$$

with Dirichlet boundary conditions

$$\Psi(0,y) = y^3, \quad \Psi(1,y) = \frac{1+y^3}{e}, \quad \Psi(x,0) = xe^{-x}$$
$$\Psi(x,1) = e^{-x}(1+x). \tag{22}$$

The analytic solution is: $\Psi_a(x,y) = e^{-x}(x+y^3)$. This example has been treated in [1] by a simpler neural-network model and by the Galerkin FEM. According to the results reported in [1], the neural-network approach seems to have certain advantages both in efficiency and in, interpolation accuracy.

For comparison purposes, the same problem is treated here by *picking points on the square boundary as if it were an irregular shape*. More specifically, we consider points (x,y) on the boundary, by dividing the interval [0, 1] on the x axis and y axis, respectively, using equidistant points. The total number of points taken on the boundary is $M = 36$. Inside the definition domain we pick points on a rectangular grid by subdividing the [0, 1] interval in ten equal subintervals that correspond to nine points in each direction. Thus a total of $K = 81$ points are selected. The analytic solution is presented in Fig. 1, while the accuracy $|\Psi_M(x,y) - \Psi_a(x,y)|$ of the obtained solution using an MLP with 20 hidden units is presented in Fig. 2. Comparing this solution with the one obtained in [1] we can conclude that

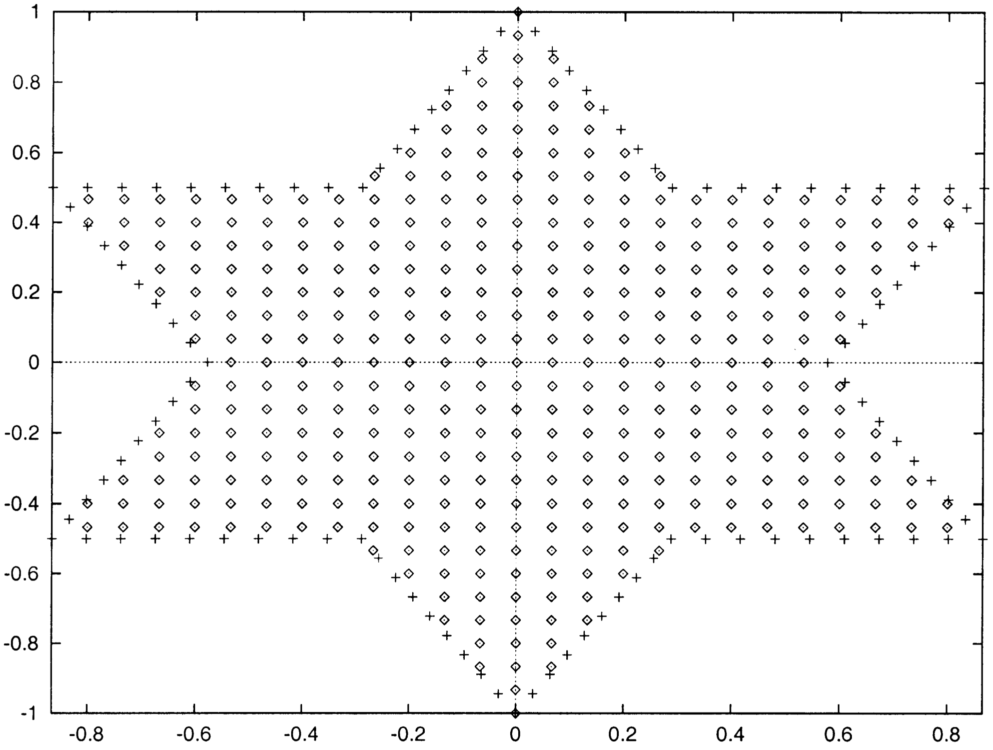

Fig. 3. The star-shaped domain and the boundary points corresponding to problem 2. Boundary points are shown as crosses.

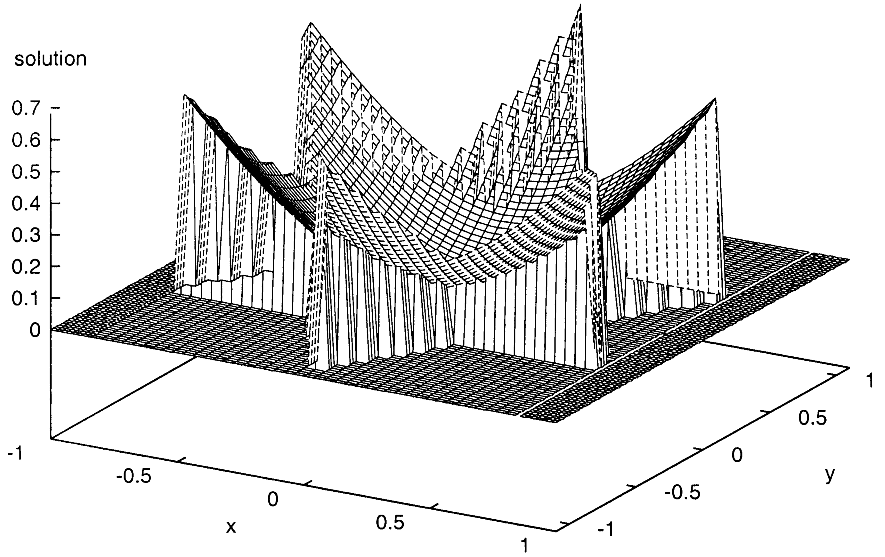

Fig. 4. Exact solution of problem 2.

the proposed method is equally effective and retains its advantages over the Galerkin FEM as well.

Problem 2: The following highly nonlinear problem is considered:

$$\nabla^2\Psi(x,y)+e^{\Psi(x,y)}=1+x^2+y^2+\frac{4}{(1+x^2+y^2)^2} \tag{23}$$

with the *star-shaped* domain displayed in Fig. 3. The analytical solution is $\Psi_a(x,y)=\log(1+x^2+y^2)$ (displayed in Fig. 4) and it has been used to compute the values at the boundary points. We have considered both Dirichlet and Neumann BCs with $M=109$ boundary points and $K=391$ grid points. An MLP with 20 hidden units was used. The accuracy of the obtained solution is displayed in Fig. 5 for the case of Dirichlet BCs, while Fig. 6 displays the output of the RBF network, which contributes as a correction term to satisfy the BCs exactly. Similar results are obtained for the case of Neumann BCs.

To show the interpolative ability of our solutions, the plot in Fig. 5 was made using points belonging to a finer grid (test points), in addition to the collocation (training) points. We found that the accuracy of the solution at these intermediate test points is maintained at the level of the neighboring training ones. This

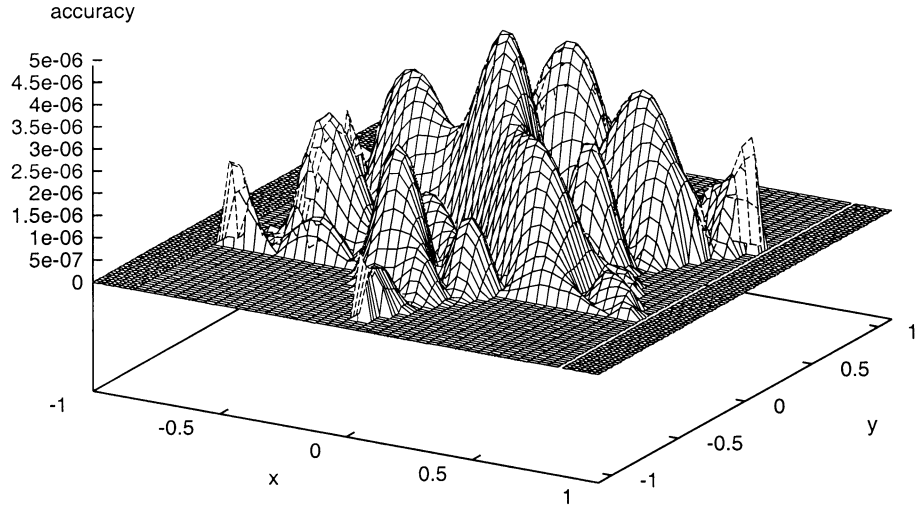

Fig. 5. Accuracy of obtained solution for problem 2 with Dirichlet BCs.

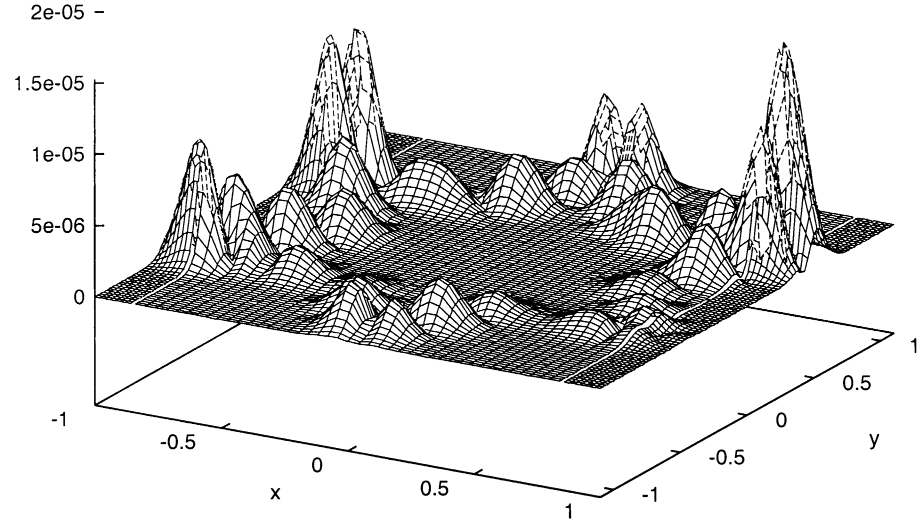

Fig. 6. The output of the RBF network for the domain of problem 2.

is a very desirable feature that is attributed to the MLP interpolation capability (the RBF contributes only around the boundary) and constitutes one of the assets of the proposed method.

Problem 3: We have solved the previous nonlinear PDE considering the cardioid domain displayed in Fig. 7. We have used $M = 100$ boundary points and $K = 500$ grid points displayed in Fig. 6. An MLP with 20 hidden units was used. The accuracy of the obtained solution (with Neumann BCs) at a dense grid of interpolation points is shown in Fig. 8. The results are similar for the case of Dirichlet BCs.

B. Three Dimensional Problem

Problem 4: We considered the problem

$$\nabla^2\Psi(x,y,z) = \Psi^2(x,y,z) + e^x y^2 + z^2\sin(y) - (e^x y^2 + (z^2-2)\sin(y))^2. \tag{24}$$

The domain is most conveniently described in spherical coordinates (r,ϕ,θ) as: $r \in [0.5, 1], \phi \in [0, \pi/2], \theta \in [0, \pi/2]$. The problem, however, is solved using Cartesian coordinates (x, y, z).

The analytical solution is $\Psi_a(x,y,z) = e^x y^2 + (z^2 - 2)\sin(y)$. We considered $M = 176$ boundary points and $K = 729$ grid points and solve the nonlinear equation with both Dirichlet and Neumann BCs. The obtained solutions using an MLP with 40 hidden units are accurate with absolute error value less than 10^{-5}.

C. Convergence and Stability Issues

In order to investigate the convergence properties of the method, we conducted several numerical experiments using the nonlinear example of problem 2 with Dirichlet BCs. Specifically we calculated the approximation error in the max norm for several choices of the number of the hidden MLP units.

This is plotted in Fig. 9. Notice that it is very similar to earlier findings [1], where in addition a comparison to the performance of the finite elements method is provided. We see that the accuracy can be controlled efficiently by varying the number of the

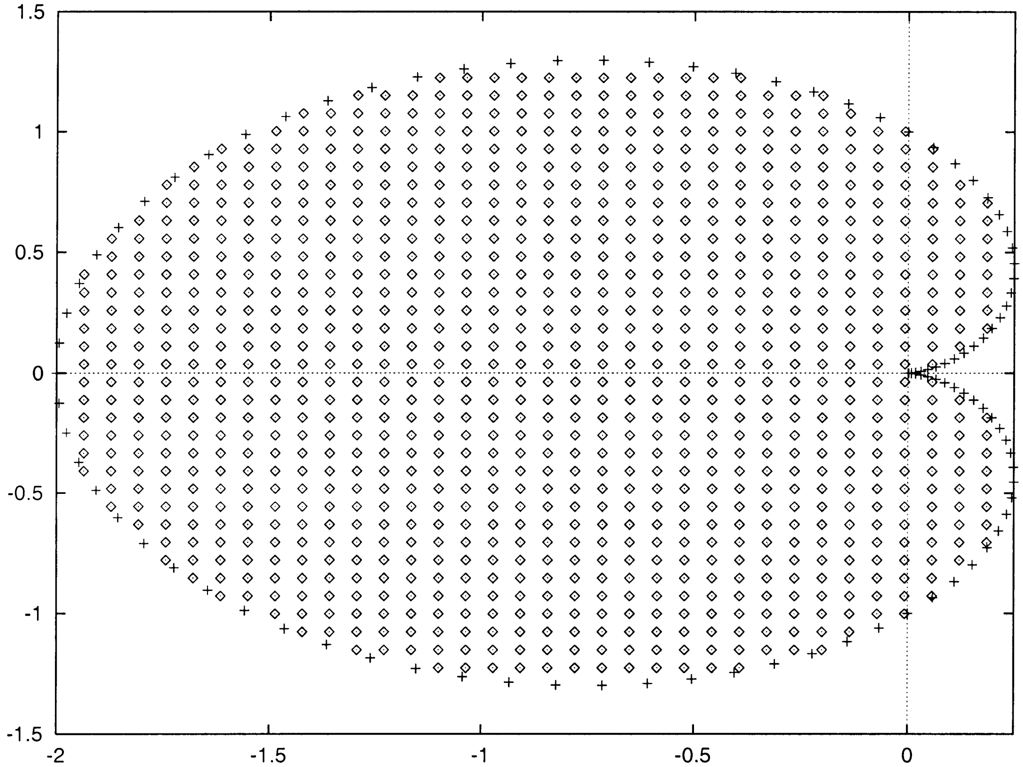

Fig. 7. The domain and the boundary points for problem 3. Boundary points are shown as crosses.

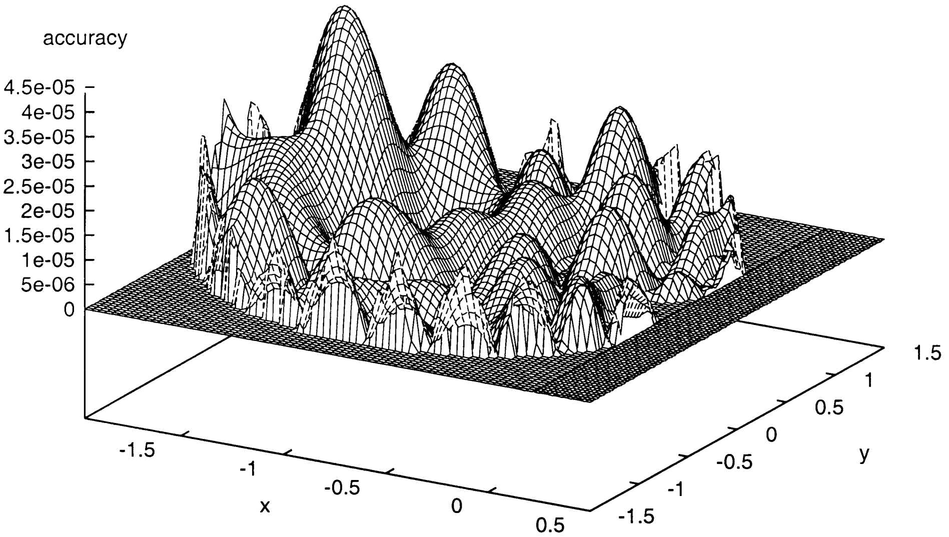

Fig. 8. Accuracy of obtained solution for problem 3 with Neumann BCs.

hidden units. We also investigated how the solution is affected by considering different boundary point sets, while keeping all other computational parameters unaltered. We conducted our experiments again with problem 2 as above.

Let us denote by M the total number of the points on the boundary. The star shaped boundary has 12 vertices (and 12 sides). On each side equal number of points were considered and care has been taken that the star vertices are always included. We also experimented with various distributions, for instance uniform and several sinusoidal forms. An extreme case is $M = 12$, i.e., only the star vertices are taken as representative boundary points. For $M = 12$ to 48 we observed a slight variation among the obtained solutions. For $M = 48$ and above the obtained solutions are identical. In Table I we list the approximation error in the max norm for several choices of M, using an MLP with 20 hidden units. In addition, we investigated the case where a vertex is intentionally excluded from the set for the extreme case of $M = 11$ and also for the case $M = 47$. The solution for the former case is eminently different only around the omitted vertex and the pointwise approximation error is plotted in Fig. 10. Notice that the approximation error in the area of the missing vertex is of the order 4×10^{-3}. For $M = 47$ (again

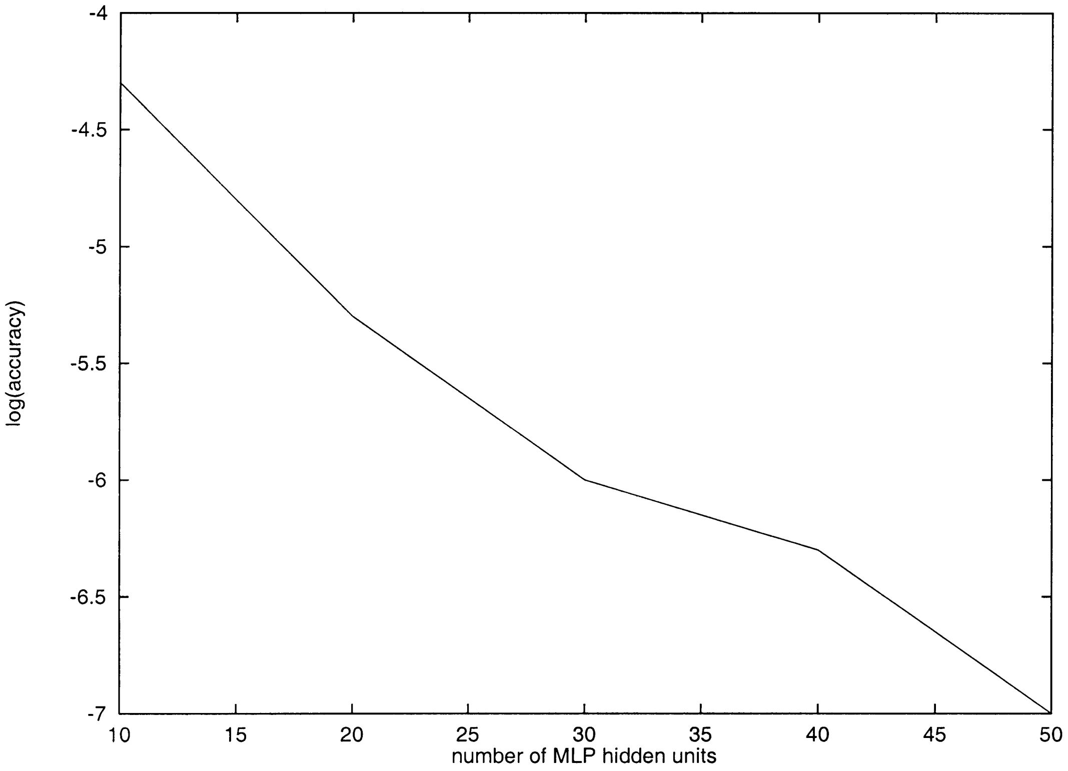

Fig. 9. Plot of the logarithm of the approximation error in the max-norm as a function of the number of MLP hidden units.

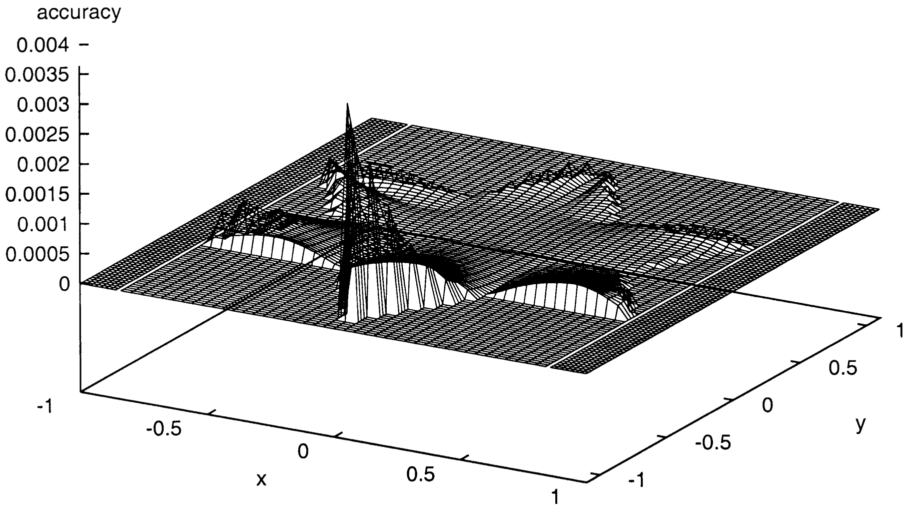

Fig. 10. Accuracy of the obtained solution for problem 2 with Dirichlet BCs when the boundary contains only the star vertices and one of them is missing.

TABLE I
APPROXIMATION ERROR FOR PROBLEM 2 WITH DIRICHLET BCS FOR DIFFERENT CHOICES OF THE BOUNDARY SET

Number of Boundary Points	*Accuracy*
12	5×10^{-5}
24	10^{-5}
48	5×10^{-6}
72	5×10^{-6}
96	5×10^{-6}

with one vertex ommited) this phenomenon, while still being present, is supressed by two orders of magnitude ($5{\times}10^{-5}$). This is both expected since, the boundary is poorly represented when some vertex is ommited, and at the same time desirable, since it demonstrates that the boundary does indeed, as it should, affect the solution. Hence we conclude that the method yields consistent results and therefore is suitable for application to real problems.

V. CONCLUSION

We presented a method capable of solving boundary value problems of the Dirichlet and Neumann types, for boundaries that due to their geometrical complexity can only be described via a set of participating points. The method employs the collocation as well as optimization techniques and is based on the synergy of MLP and RBF artificial neural networks. It provides accurate solutions in a closed analytic form that satisfy the BCs at the selected points exactly. The proposed method is quite general and can be used for a wide class of linear and nonlinear PDEs.

Future work will focus on the application of the method to specific problems, containing real objects with arbitrarily complex boundaries. Interesting problems of this kind arise in many scientific fields. Since the method lends itself to parallel processing, we have a strong interest in implementing the method on both, general purpose parallel hardware and on specialized hardware (neuroprocessors). The latter would reveal the full potential of the proposed approach and may lead to the development of specialized machines that will allow the treatment of difficult and computationally demanding problems.

ACKNOWLEDGMENT

I. E. Lagaris wishes to acknowledge the warm hospitality offered by Prof. Ishii and Prof. Tsoukalas of Purdue University, School of Nuclear Engineering, during his stay at Lafayette, where part of this work was carried out.

REFERENCES

[1] I. E. Lagaris, A. Likas, and D. I. Fotiadis, “Artificial neural networks for solving ordinary and partial differential equations,” *IEEE Trans. Neural Networks*, vol. 9, pp. 987–1000, 1998.

[2] ——, “Artificial neural network methods in quantum mechanics,” *Comput. Phys. Commun.*, vol. 104, pp. 1–14, 1997.

[3] A. Charalambopoulos, G. Dassios, D. I. Fotiadis, and C. Massalas, “Frequency spectrum of the human-neck system,” *Int. J. Eng. Sci.*, vol. 35, pp. 753–768, 1997.