INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

The topological derivative method and artificial neural networks for numerical solution of shape inverse problems

Lidia Jackowska-Strumiłło , Jan Sokołowski , Antoni Żochowski

No 3739

Juin 1999

THÈME 4

Rapport de recherche

ISSN 0249-6399

The topological derivative method and artificial neural networks for numerical solution of shape inverse problems

Lidia Jackowska-Strumiłło *, Jan Sokołowski †, Antoni Żochowski‡

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet Numath

Rapport de recherche n ° 3739 — Juin 1999 — 18 pages

Abstract: The new method is proposed for the numerical solution of a class of shape inverse problems. The size and the location of a small opening in the domain of integration of an elliptic equation is identified on the basis of a observation. The observation includes the finite number of shape functionals. The approximation of the shape functionals by using the so-called topological derivatives is used to perform the learning process of an artificial neural network. The results of computations for $2D$ examples show that the method allows to determine an approximation of the global solution to the inverse problem, sufficiently closed to the exact solution. The proposed method can be extended to the problems with an opening of general shape and to the identification problems of small inclusions.

Key-words: Topological derivative, artificial neural network, shape inverse problem

(Résumé : tsvp)

* Computer Engineering Department, Technical University of Lodz, Al. Politechniki 11, 90-924 Lodz, Poland; e-mail: lidia_js@kis.p.lodz.pl

† IECN/Inria-Lorraine, BP 239, 54506 Vandoeuvre lès Nancy, France; e-mail: sokolows@iecn.u-nancy.fr

‡ Systems Research Institute of the Polish Academy of Sciences, ul. Newelska 6, 01-447 Warszawa, Poland; e-mail: zochowsk@ibspan.waw.pl

Unité de recherche INRIA Lorraine
Technopôle de Nancy-Brabois, Campus scientifique,
615 rue de Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY (France)
Téléphone : 03 83 59 30 30 - International : +33 3 3 83 59 30 30
Télécopie : 03 83 27 83 19 - International : +33 3 83 27 83 19
Antenne de Metz, technopôle de Metz 2000, 4 rue Marconi, 55070 METZ

La méthode de la dérivée topologique et des réseaux de neurones dans la recherche de solutions numériques de problèmes inverses

Résumé : On propose cette nouvelle méthode afin de déterminer la solution numérique d'une classe de problèmes inverses. La taille et la position d'une petite ouverture dans le domaine d'intégration d'une équation elliptique sont identifiées sur la base d'une observation comportant un nombre fini de fonctionnelles. L'approximation des fonctionnelles, au moyen des dérivées topologiques, est utilisée pour engager le processus d'apprentissage d'un réseau de neurones. Les résultats de calculs pour des exemples 2D montrent que la méthode permet de déterminer une approximation de la solution globale du problème inverse, suffisamment proche de la solution exacte. La méthode proposée peut être étendue à des problèmes avec une ouverture de forme quelconque et à des problèmes d'identification de petites inclusions.

Mots-clé : dérivée topologique, réseau de neurones, problème inverse

1 Introduction

The topological derivative, denoted by $\mathcal{T}_\Omega \mathcal{J} = \mathcal{T}_\Omega$, of a shape functional $\mathcal{J}(\Omega)$ is introduced in [14] in order to characterize the infinitesimal variation of $\mathcal{J}(\Omega)$ with respect to the infinitesimal variation of the topology of the domain Ω. The topological derivative allows us to derive the new optimality condition for the shape optimization problem:

$$\mathcal{J}(\Omega^*) = \inf_\Omega \mathcal{J}(\Omega) \ .$$

Let us recall that an optimal domain Ω^* is characterized by the first order necessary optimality conditions [13] defined on the boundary of the optimal domain Ω^*,

$$dJ(\Omega^*; V) \geq 0$$

for all admissible vector fields V used in the speed method, and in addition by the following new optimality condition defined in the interior of the domain Ω^*,

$$\mathcal{T}_{\Omega^*}(y) \geq 0 \quad \text{for } y \in \Omega^* \ .$$

The other use of the topological derivative is connected with approximating the influence of the openings (holes) in the domain Ω on the values of integral functionals depending on solutions to elliptic equations defined in perturbed domains Ω_ρ, see the definition below, what allows us to solve a class of shape inverse problems.

Let us consider the following model problem. Suppose, that the shape functionals have the form

$$\mathcal{J}_i(\Omega_\rho) = \int_{\Omega_\rho} F_i(x, u^i_\rho(x), \nabla u^i_\rho(x))dx \qquad i = 1, ..., N,$$

where $\Omega_\rho = \Omega \setminus \overline{B_\rho(y)}$, $\Omega \subset R^2$ is a given domain, $B_\rho(y)$ is a ball with centre $y \in \Omega$ and of a sufficiently small radius $\rho > 0$.

Let the functions $u^i_\rho \in H^1(\Omega_\rho)$ satisfy the following boundary value problems:

$$\Delta u^i_\rho = f^i \quad \text{in} \quad \Omega_\rho \tag{1}$$

$$u^i_\rho = g^i \quad \text{on} \quad \Gamma^i_1 \tag{2}$$

$$\frac{\partial u^i_\rho}{\partial n} = h^i \quad \text{on} \quad \Gamma^i_2 \tag{3}$$

$$\frac{\partial u^i_\rho}{\partial n} = 0 \quad \text{on} \quad \Gamma_\rho = \partial B_\rho(y) \tag{4}$$

We want to identify the location and the size of a small inclusion $B_\rho(y) \subset \Omega$. To this end we use the approximation of the mapping

$$\mathcal{G} : R^3 \ni (\rho, y) \mapsto \{\mathcal{J}_1(\Omega_\rho), ..., \mathcal{J}_N(\Omega_\rho)\} \in R^N$$

of the form

$$\mathcal{G}_i(\rho, y) \cong \mathcal{J}_i(\Omega) + \frac{\rho^2}{2}\mathcal{T}^i_\Omega(y),$$

where $\mathcal{T}^i_\Omega(y)$ denotes the topological derivative of the shape functional $\mathcal{J}_i(\Omega)$ evaluated at $y \in \Omega$.

The inverse mapping $\mathcal{G}^{-1}$, which is difficult to calculate from the mathematical relations, is modelled using artificial neural networks. Feedforward multilayer perceptron networks are applied to solve this mapping problem. Numerical computations that are based on the topological derivative have provided data both for network training and testing procedures. The latter one is required for validation of the network true generalization capabilities. The standard backpropagation error correcting rule is used for network training in which the sum square error cost function is minimized. Results will be provided that demonstrate the validity of employing artificial neural network for this complex nonlinear mapping problem.

The topological derivatives $\mathcal{T}^i_\Omega(y)$ are determined by numerical solutions of an auxiliary system of partial differential equations [14], [16]. Therefore the approximation of $\mathcal{G}$ is relatively easily obtained, making preparation of training data possible. The results of computations for numerical solution of the inverse problem under consideration are provided.

The identification of small openings using classical Newton method is considered in [10].

2 Topological derivatives of shape functionals

We recall the definition and some properties of the topological derivatives of shape functionals. We refer the reader to [14] for the detailed study of topological derivatives for elliptic equations in $2D$.

Let us consider the shape functional

$$\mathcal{J}(\Omega_\rho) = \int_{\Omega_\rho} \mathcal{F}(x, u_\rho(x), \nabla u_\rho(x))dx,$$

where $\Omega_\rho = \Omega \setminus \overline{B_\rho(y)}$. For simplicity we assume that $\mathcal{F}(x,u,q) = F(u) + G(q)$, but the analysis can be carried on in the general case as well. Therefore, the shape functional $\mathcal{J}(\Omega_\rho) = I(\rho)$ written as a function of small parameter $\rho > 0$ takes on the form

$$I(\rho) = \int_{\Omega_\rho} [F(u_\rho) + G(\nabla u_\rho)] \ .$$

The equations for $u_\rho \in H^1(\Omega_\rho)$ are given by (1)–(4), hence the weak solution $u_\rho \in H^1_g(\Omega_\rho)$ satisfies the following integral identity

$$\int_{\Omega_\rho} \nabla u_\rho \cdot \nabla \phi \, d\Omega = \int_{\Gamma_2} h\phi \, dS - \int_{\Omega_\rho} f\phi \, d\Omega, \quad \forall \phi \in H^1_{\Gamma_1}(\Omega_\rho) \ , \tag{5}$$

where we use the standard notation for the Sobolev spaces,

$$H^1_g(\Omega_\rho) = \{\psi \in H^1(\Omega_\rho) \mid \psi = g \quad \text{on} \quad \Gamma_1\} \ ,$$

$$H^1_{\Gamma_1}(\Omega_\rho) = \{\psi \in H^1(\Omega_\rho) \mid \psi = 0 \quad \text{on} \quad \Gamma_1\} \ ,$$

and the convention that the restriction to Ω_ρ of a function $\phi \in H^1_{\Gamma_1}(\Omega)$ is denoted also by ϕ.

We are interested in asymptotic expansion of the functional $I(\rho)$ at the point $\rho = 0^+$ of the form

$$I(\rho) = I(0^+) + \rho I'(0^+) + \frac{1}{2}\rho^2 I''(0^+) + o(\rho^2) \ .$$

We are going to show that for the problems under considerations $I'(0^+) = 0$ and the topological derivative of $\mathcal{J}(\Omega)$ is defined by

$$\mathcal{T}_\Omega \mathcal{J}(y) = \lim_{\rho \downarrow 0} \frac{dI(\rho)}{d(|B_\rho(y)|)} \ ,$$

and in our case $\mathcal{T}_\Omega \mathcal{J} = 2\pi I''(0^+)$. It is important for applications that the topological derivative $\mathcal{T}_\Omega$ can be determined by solving the equation for u and the equation for an appropriate adjoint state v only once, in the unperturbed domain Ω.

It is well known [13], that the solution u_ρ is differentiable with respect to ρ, for $\rho > 0$ and ρ small enough, so the ball $B_\rho(y) \subset \Omega$. The shape derivative denoted by u'_ρ satisfies the following integral identity [13],

$$u'_\rho \in H^1_{\Gamma_1}(\Omega_\rho) \ : \quad \int_{\Omega_\rho} \nabla u'_\rho \cdot \nabla \phi \, d\Omega - \int_{\Gamma_\rho} \frac{\partial u_\rho}{\partial \tau} \cdot \frac{\partial \phi}{\partial \tau} \, dS = \int_{\Gamma_\rho} f \phi \, dS, \tag{6}$$

for all test functions $\phi \in H^1_{\Gamma_1}(\Omega_\rho) \cap H^2(\Omega_\rho)$, $\frac{\partial \phi}{\partial \tau}$ denotes the tangential component of the gradient $\nabla \phi = \frac{\partial \phi}{\partial \tau} + \frac{\partial \phi}{\partial n} n$ on the boundary Γ_ρ. In our notation $\frac{\partial \phi}{\partial \tau}$ is a vector function, the normal derivative on the boundary $\frac{\partial \phi}{\partial n}$ is a scalar function, and $n = (n_1, n_2)$ is the unit normal vector on the boundary of the domain of integration Ω_ρ.

In the first step we need the forms of the first and second order derivatives of the shape functional $\mathcal{J}(\Omega_\rho)$ with respect to ρ. It means that we evaluate the derivatives of the function $I(\rho)$ for sufficiently small $\rho > 0$. It can be shown, that the function $I(\rho)$ is smooth. Using the general formulae [13] for shape derivatives of integral functionals we have

$$I'(\rho) = \int_{\Omega_\rho} \left[F_u(u_\rho) u'_\rho + G_q(\nabla u_\rho) \cdot \nabla u'_\rho \right] - \int_{\Gamma_\rho} \left[F(u_\rho) + G\left(\frac{\partial u_\rho}{\partial \tau} \right) \right] . \tag{7}$$

Next we introduce the adjoint state $v_\rho \in H^1_{\Gamma_1}(\Omega_\rho)$, which solves the following integral identity,

$$- \int_{\Omega_\rho} \left[\nabla v_\rho \cdot \nabla \phi \right] \, d\Omega = \int_{\Omega_\rho} \left[F_u(u_\rho) \phi + G_q(\nabla u_\rho) \cdot \nabla \phi \right] \, d\Omega \tag{8}$$

Here $F_u(u) = \partial F(u)/\partial u$, $G_q(q) = \partial G(q)/\partial q$. From (6), with the test function ϕ replaced by v_ρ, and using (8) with the test function ϕ replaced by u'_ρ, it follows that the first order shape derivative of $I'(\rho)$ takes on the form

$$I'(\rho) = - \int_{\Gamma_\rho} \left[F(u_\rho) + G\left(\frac{\partial u_\rho}{\partial \tau} \right) + f v_\rho + \frac{\partial u_\rho}{\partial \tau} \cdot \frac{\partial v_\rho}{\partial \tau} \right] \, dS . \tag{9}$$

In order to obtain the form of the second order shape derivative $I''(\rho)$ we need the form of the shape derivative $v'_\rho \in H^1_{\Gamma_1}(\Omega_\rho)$ of the adjoint state v_ρ, which solves the integral identity,

$$\begin{aligned} - \int_{\Omega_\rho} \nabla v'_\rho \cdot \nabla \phi \, d\Omega + \int_{\Gamma_\rho} \frac{\partial v_\rho}{\partial \tau} \cdot \frac{\partial \phi}{\partial \tau} \, dS = \int_{\Omega_\rho} F_{uu}(u_\rho) u'_\rho \phi \, d\Omega \\ - \int_{\Gamma_\rho} F_u(u_\rho) \phi \, dS - \int_{\Gamma_\rho} G_q\left(\frac{\partial u_\rho}{\partial \tau} \right) \frac{\partial u_\rho}{\partial \tau} \cdot \frac{\partial \phi}{\partial \tau} \, dS \\ + \int_{\Omega_\rho} G_q(\nabla u_\rho)(\nabla u'_\rho \cdot \nabla \phi) \, d\Omega \\ + \int_{\Omega_\rho} G_{qq}(\nabla u_\rho)(\nabla u'_\rho \cdot \nabla u_\rho)(\nabla u_\rho \cdot \nabla \phi) \, d\Omega \end{aligned} \tag{10}$$

for all test functions $\phi \in H^1_{\Gamma_1}(\Omega_\rho) \cap H^2(\Omega_\rho)$.

Since all integrands in (9) are bounded, we have the following asymptotic behaviour of the first order derivative:

$$\lim_{\rho \to 0+} I'(\rho) = 0 . \tag{11}$$

Using the known formulae for the second order shape derivatives of shape functionals [13], it follows that the second order derivative of the function $I(\rho)$ is given by the following formula

$$\begin{aligned} I''(\rho) &= \int_{\Gamma_\rho}\left[\frac{\partial\{F(u_\rho)+G(\nabla u_\rho)\}}{\partial n}+\frac{\partial(fv_\rho)}{\partial n}+\frac{\partial}{\partial n}\left(\frac{\partial u_\rho}{\partial\tau}\cdot\frac{\partial v_\rho}{\partial\tau}\right)\right]dS \\ &- \int_{\Gamma_\rho}\left[F_u(u_\rho)u'_\rho+G_q(\nabla u_\rho)\cdot\nabla u'_\rho+fv'_\rho+\left(\frac{\partial u_\rho}{\partial\tau}\cdot\frac{\partial v_\rho}{\partial\tau}\right)'\right]dS \\ &- \frac{1}{\rho}\int_{\Gamma_\rho}[F(u_\rho)+G\left(\frac{\partial u_\rho}{\partial\tau}\right)+fv_\rho+\frac{\partial u_\rho}{\partial\tau}\cdot\frac{\partial v_\rho}{\partial\tau}]\,dS \\ &= I_1(\rho)+I_2(\rho)+I_3(\rho). \end{aligned} \tag{12}$$

Observe, that $\frac{\partial}{\partial n}=-\frac{\partial}{\partial r}$ on Γ_ρ, where r is the polar coordinate. The first and the second terms in (12) vanish in the limit,

$$\lim_{\rho\to0+}[I_1(\rho)+I_2(\rho)]=0\ .$$

There remains to evaluate the limit of the last term for $\rho\to0^+$.

2.1 Asymptotic expansion

In order to perform the passage to the limit $\rho\to0^+$ in the expression

$$\frac{dI(\rho)}{d(|B_\rho(y)|)}=-\frac{1}{\rho}\int_{\Gamma_\rho}\left[F(u_\rho)+G\left(\frac{\partial u_\rho}{\partial\tau}\right)+fv_\rho+\frac{\partial u_\rho}{\partial\tau}\frac{\partial v_\rho}{\partial\tau}\right]dS$$

we use the following asymptotic expansion for the solution u_ρ to the elliptic equation in the domain $\Omega_\rho,\ \rho>0$. We refer the reader to the forthcoming paper [15] for a simple proof of the asymptotic expansions given below.

Let

$$\nabla u(y)=[a,b]^T\ ,$$

denote the point value of the gradient of the solution to the elliptic equation in Ω, and consider the polar coordinate system with the centre at y, which coincides with the centre of the ball $B_\rho(y)$.

The solution u_ρ as a function of polar coordinates (r,θ) in the neighbourhood $\rho<r<2\rho$ of the ball $B_\rho(y)$, for ρ sufficiently small, can be expressed for $r\ge\rho$ as follows:

$$u_\rho=u+a\frac{\rho^2}{r}\cos\theta+b\frac{\rho^2}{r}\sin\theta+\mathcal{R}$$

where

$$\mathcal{R}=\rho^2\,[O(\frac{\rho}{r})+l(\rho,r)],$$

and $l(\rho,r)$ may contain finite powers of $\ln\rho,\ln r$. Hence $\mathcal{R}=O(\rho^{2-\epsilon})$ for any $\epsilon>0$. Therefore, in the ring $\rho\le r\le2\rho$, taking into account the regularity of $u=u_0$ in the neighbourhood of $y\in\Omega$ and using the Taylor expansion for u, we have the following expansion for u_ρ,

$$u_\rho=u(y)+a(\frac{\rho^2}{r}+r)\cos\theta+b(\frac{\rho^2}{r}+r)\sin\theta+O(\rho^{2-\epsilon}),$$

where $u(y)$ denotes the value at y of the solution to the elliptic equation in the domain Ω, ie., in the full domain without hole.

The above formulae are given in the polar coordinate system with the centre at y, which coincides with the centre of the ball. In particular, from the above expansion it follows that, we refer the reader to [14] for a proof, the norm of the tangential derivative can be determined from the following formula

$$\frac{\partial u_\rho}{\partial \tau} \cdot \tau = \frac{1}{\rho}\frac{\partial u_\rho}{\partial \theta}|_{r=\rho} = 2(-a\sin\theta + b\cos\theta) + O(\rho^{1-\epsilon}).$$

Using these expansions, we obtain the following result:

Theorem 2.1 *The topological derivative of the functional $\mathcal{J}(\Omega) = \int_\Omega [F(u) + G(\nabla u)]$ is given by the following formula*

$$\mathcal{T}\mathcal{J}_\Omega(y) = -\frac{1}{2\pi}[2\pi F(u(y)) + g(\nabla u(y)) + 2\pi f(y)v(y) + 4\pi \nabla u(y)\cdot\nabla v(y)] \ ,$$

where

$$g(\nabla u(y)) = -\int_0^{2\pi} G\left(-2\frac{\partial u}{\partial x_1}(y)\sin\theta, 2\frac{\partial u}{\partial x_2}(y)\cos\theta\right)d\theta = -\int_0^{2\pi} G\left(-2a\sin\theta, 2b\cos\theta\right)d\theta.$$

3 The first eigenvalue

We complete the results on the topological differentiability of shape functionals with the formula for the topological derivative of the first eigenvalue. We refer to [9] for the asymptotic expansions of the first eigenvalue of the second order elliptic problems defined in the domains with small openings.

Let us consider consider the following eigenvalue problem, where λ_ρ denotes the first eigenvalue and u_ρ corresponding eigenfunction. For $\rho = 0$ we denote the eigenvalue and eigenfunction by λ, u, respectively.

$$\Delta u_\rho + \lambda_\rho u_\rho = 0 \quad \text{in} \quad \Omega_\rho, \tag{13}$$
$$u_\rho = 0 \quad \text{on} \quad \Gamma, \tag{14}$$
$$\frac{\partial u_\rho}{\partial n} = 0 \quad \text{on} \quad \Gamma_\rho = \partial B_\rho(y). \tag{15}$$

The following result is easily obtained from the asymptotic expansions for the first eigenvalue derived in [9].

Theorem 3.1 *The topological derivative Λ of the first eigenvalue is given by the following formula*

$$\Lambda(x) = \pi\lambda[u(x)]^2 - m|\nabla u(x)|^2 \ ,$$

where $m = -2\pi$.

Remark 3.1 *For an arbitrary inclusion of the form $\omega_\rho = \{x \mid \frac{x}{\rho} \in \omega\}$, $0 \in \omega\cap\Omega$, with $\Omega_\rho = \Omega \setminus \overline{\omega_\rho}$ it follows that the topological derivative takes on the form*

$$\Lambda(0) = \lambda[u(0)]^2|\omega| - \nabla u(0)\cdot\mathcal{M}(\omega)\cdot\nabla u(0) \ .$$

The matrix $\mathcal{M}(\omega)$ is the so-called mass matrix associated with the domain ω [11]. In the formula given by the theorem, $\mathcal{M} = -2\pi I$, I denotes the identity matrix.

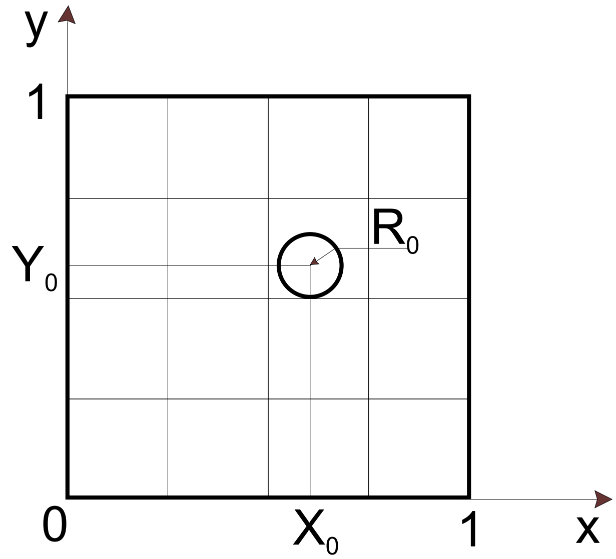

Figure 1: The parameters describing opening (inclusion).

4 Numerical example of shape functionals

We consider the following test examples.

We consider four boundary value problems defined in the same domain $\Omega = (0,1)\times(0,1)$. It means, that for $i = 1,2,3,4$

$$\Delta u_i = 0 \quad \text{in} \;\; \Omega.$$

These problems differ with respect to the boundary conditions. For $i = 1$ they have the form

$$u_1 = 1 \quad \text{on } \{0\} \times \left(\frac{1}{3}, \frac{2}{3}\right); \quad u_1 = 0 \quad \text{on } \{1\} \times (0,1); \quad \frac{\partial u_1}{\partial n} = 0 \quad \text{otherwise.}$$

For $i = 2,3,4$ they are obtained from the above conditions applying the succesive rotation by the angle $\pi/2$.

The shape functionals $\mathcal{J}_j = \mathcal{J}_j(\Omega)$ are defined as follows: for $j = 1,...,12$, $i = 1,2,3,4$.

$$\mathcal{J}_{\{1+3(i-1)\}} = \int_\Omega u_i^2 \ , \quad \mathcal{J}_{\{2+3(i-1)\}} = \int_\Omega \left(\frac{\partial u_i}{\partial x_1}\right)^2 \ , \quad \mathcal{J}_{\{3+3(i-1)\}} = \int_\Omega \left(\frac{\partial u_i}{\partial x_2}\right)^2$$

In the domain $\Omega_\rho = \Omega \setminus \overline{B_\rho(y)}$, $y = (y_1, y_2)$, we add the homogenous Neumann boundary conditions on the boundary Γ_ρ of the ball $B_\rho(y)$.

For any fixed $i = 1,...,4$, and $u = u_i$, we denote by $J_{gk}(\rho), k = 1,2$, the shape functionals depending on the partial derivatives $\frac{\partial u_i}{\partial x_k}$,

$$J_{gk}(\rho) = \int_\Omega \left(\frac{\partial u}{\partial x_k}\right)^2 \ , \quad k = 1,2 \ .$$

The topological derivatives of these shape functionals are obtained from Theorem 2.1 by direct computation of the function g:

$$\left[J''_{g1}(0^+)\right](y) = -\pi \left[\frac{3}{2}\left(\frac{\partial u}{\partial x_1}\right)^2 + \frac{1}{2}\left(\frac{\partial u}{\partial x_2}\right)^2 + 4(\nabla u \cdot \nabla v_1)\right](y)$$

$$\left[J''_{g2}(0^+)\right](y) = -\pi \left[\frac{1}{2}\left(\frac{\partial u}{\partial x_1}\right)^2 + \frac{3}{2}\left(\frac{\partial u}{\partial x_2}\right)^2 + 4(\nabla u \cdot \nabla v_2)\right](y) \ ,$$

where v_k , $k = 1,2$, is the associated adjoint state.

5 Artificial neural networks

Work on artificial neural networks has been motivated by the recognition that the brain computes in a different way from the conventional digital computer. The brain has the capability of organising neurons so as to perform certain computations many times faster than the fastest digital computer in existence today.

In general form an artificial neural network is a machine that is designed to model the way in which the brain performs a particular task or function of interest. The network is usually implemented using electronic components or simulated in software on a digital computer. More precisely an artificial neural network is a massively parallel distributed processor that has a natural property for storing experimental knowledge and making it available. It resembles the brain in two respects [3]:

- knowledge is acquired by the network through a learning process;
- network structure and interneuron connection strengths known as synaptic weights are used to store the knowledge.

The procedure used to perform the learning process is called a learning algorithm, the function of which is to modify the synaptic weights of the network in an orderly fashion so as to attain a desired design objective.

Artificial neurons are the basic elements of artificial neural networks. An example [4] of an artificial neuron's model is shown in Fig.2.

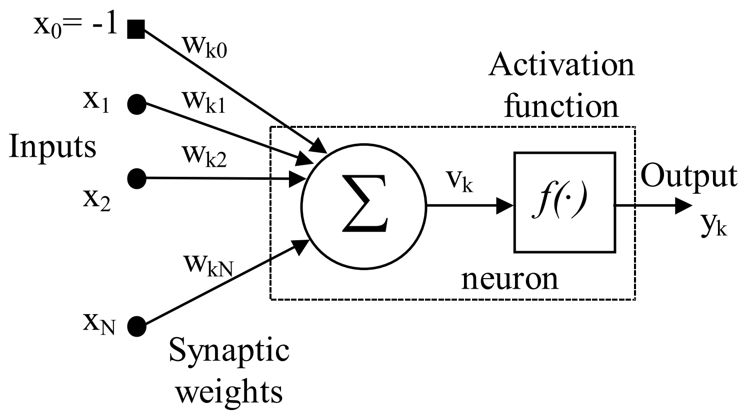

Figure 2: Model of an artificial neuron.

It has a set of inputs $x_1, x_2, ..., x_N$, denoted as the input vector X. Each input signal number k is multiplied by an associated weight $w_{k1}, w_{k2}, ..., w_{kN}$, before it is applied to the summation block $\sum$. Each weight corresponds to the "strength" of a single biological synaptic connection. The summation block, corresponding roughly to the biological cell body, adds all of the weighted inputs algebraically and produces a net signal v_k, which is further processed by an activation function $f(\cdot)$ to obtain the neuron's output signal y_k, given by the equation [4]:

$$y_k = f(\nu_k) = f(\sum_{j=0}^{N} w_{kj}^{\nu} x_j). \tag{16}$$

Activation function $f(\cdot)$ may be a simple linear function or a non-linear function, that more accurately simulates the non-linear transfer characteristics of the biological neuron and permits more general network functions. The most commonly used activation functions are threshold function, sigmoidal function, hyperbolic tangent function and radial basis function. Sigmoidal function is mathematically expressed as [2, 3]:

$$f(z) = \frac{1}{1+e^{-z}}. \tag{17}$$

An artificial neuron with the sigmoidal activation function is called *perceptron*.

A single neuron can perform only certain simple functions. The power of neural computations comes from connecting neurons into networks. Structure and size of the designed neural network depends on the complexity of the problem, which has to be solved by the network. A great variety of network structures is known [3].

The **Multi-Layer Perceptron** (MLP) is the most commonly used network structure, which was also applied in the presented approach. An example of MLP network is shown in Fig. 3. MLP network

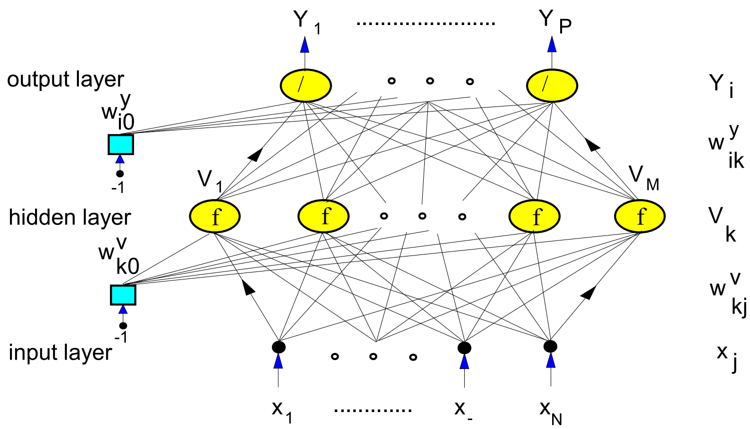

Figure 3: A structure of multilayer perceptron network with one hidden layer.

is built from perceptrons grouped in layers [3]. This is a feedforward network and input signals applied to the network are transmitted in one direction from the network input nodes to the output layer. The middle perceptron layers are called hidden layers. The network presented in Fig. 3 has an input layer and two perceptron layers: the hidden layer and the output layer and is called a two-layer perceptron network. The signal of the i-th network output is given by the equation [3]:

$$Y_i = f(y_i) = f(\sum_{k=0}^{M} w^y_{ik} V_k) = f(\sum_{k=0}^{M} w^y_{ik} f(\sum_{j=0}^{N} w^\nu_{ik} x_j)), \tag{18}$$

where $i = 1, \dots P,\ k = 1, \dots M,\ j = 1, \dots N$.
It has been proved, that feedforward multilayer perceptron networks are universal approximators [6] and therefore this network structure was chosen to solve the problem presented in this paper.

5.1 Application of artificial neural networks for inverse problem solving

Application of Artificial Neural Networks (ANN), instead of analytical calculations, offers a novel and powerful tool for inverse problem solving. The inverse mapping G^{-1}, which allows for identification of inclusion presented in Fig. 1, is difficult to calculate from the mathematical relations and therefore was modelled using artificial neural networks. Similarly as in the classical approach, the inverse mapping G^{-1}, shown in Fig. 4, may be determined unambiguously only when the transformation G has the property, that each input vector X_0, Y_0, R_0 is transformed into a different values output vector $J_1, \dots J_n$ (one to one mapping). ANN-based inverse model is built on the basis of relations between the network input and output vectors. The knowledge about the inverse mapping is saved within the network

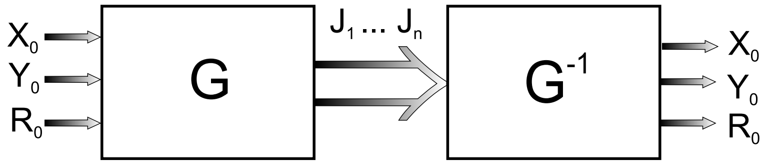

Figure 4: An inverse mapping problem.

structure and network connection weights. Feedforward multilayer perceptron networks, which are universal approximators [6], were used for inverse problem mapping. Twelve values of functionals $J_1, \dots J_{12}$, which ensure the unicity solution had been calculated by the use of topological derivative method for the square with the inclusion and introduced to the network input. The inclusion's radius R_0 and position X_0, Y_0 were approximated at the network output. An unknown mapping of the input vector to the output vector was approximated in an iterative procedure known as neural network training [3].

The objective of the learning algorithm is to adjust the weights of the used ANN on the basis of a given set of input-output pairs for a given cost function to be minimised [3]. The most frequently used sum square error cost function was applied:

$$E(w) = \frac{1}{2} \sum_{k=1}^{N} [y_k(w) - d_k]^2, \tag{19}$$

where: w - vector of network weights, $y_k(w)$ - network output signal for k-th learning sample, d_k - estimator of the output signal for k-th sample calculated by the use of topological derivative method, N - number of samples.

For the considered identification problem the backpropagation error correcting rule with Levenberg - Marquardt optimisation algorithm described bellow was used to minimize the cost function defined in (19).

The change of cost function in neighbourhood of existing weights w may be approximated by the use of Taylor series expansion, as follows [3]:

$$\Delta E(w) = E(w + \Delta w) - E(w) \approx g^T \cdot \Delta w + \frac{1}{2} \Delta w^T \cdot H \cdot \Delta w,$$

where g is the gradient vector and H is the Hessian matrix. Differentiating with respect to Δw, the change $\Delta E(w)$ is minimized when

$$g + H \cdot \Delta w = 0,$$

which yields the optimum value of Δw to be

$$\Delta w = -H^{-1} \cdot g,$$

where H^{-1} is the inverse of the Hessian matrix. Then the solution is obtained by

$$w = w_0 - H^{-1} \cdot g$$

and is the basis of the Newton's method [3].

The Levenberg - Marquardt algorithm approaches second order training speed without having to compute the Hessian matrix [2]. For the cost functoin given in (19) the Hessian can be approximated as

$$H = J^T \cdot J,$$

and the gradient is computed as

$$g = J^T \cdot e,$$

where J is the Jacobian matrix, which contains first derivatives of the network errors with the respect to the weights, and e is a vector of network errors. The Jacobian matrix can be computed through a standard backpropagation technique [2] that is much less complex than computing the Hessian matrix. The Levenberg - Marquardt algorithm uses approximated solution, given by

$$w_{k+1} = w_k - [J^T \cdot J + \mu I]^{-1} \cdot J^T \cdot e, \tag{20}$$

where the scalar μ is a regularisation parameter. When μ equals zero this algorithm is just a Newton's method, using the approximate Hessian matrix. When μ is large, this becomes gradient descent method with a small step size. Newton's method is faster and more accurate near an error minimum, so μ is decreased after each successful step and is increased only when a tentative step would increase the performance function. The Levenberg - Marquardt algorithm allows to train the neural networks with the rate 10 to 100 times faster than the standard gradient descent backpropagation method and is recommended for the MLP networks with the great number of neurons [2]. This algorithm was implemented by the MATLAB calculating packet and applied for the neural network training for the considered inverse problem.

In our particular problem different feedforward network structures with a single hidden layer were tested. Optimal network structure was chosen on the base of error analysis and networks computer simulations. Finally chosen network structure (12-24-3) i.e.: twelve inputs, twenty four processing units with a sigmoidal transfer function in the network hidden layer and three linear unit in the output layer, had 387 weights. Numerical computations that were based on the topological derivative have provided data both for network training and testing procedures. The training and testing data were computed for different inclusion radius values, which were changed from 0,05 to 0,2 and corresponding them position values, which were changed in the way to built the uniform discretisation grid and fulfil the conditions:

$$2R_i < X_i < 1 - 2R_i,$$

$$2R_i < Y_i < 1 - 2R_i.$$

Than the corresponding values of functionals J_0, J_1, J_2 for four configurations described earlier were calculated by the use of topological derivative method for each set of inputs. From the whole number of data sets 1285 were selected for the network training and 205 for the network testing. The latter ones were required for validation of the network true generalization capabilities.

Fig. 5–13 present the result of network testing and the error distribution for the inclusion identification calculated for each of inclusion parameters X_i, Y_i, R_i. Three cases are shown.

In the first, the radius of the inclusion was $r = 0.075$. The figures 5–7 depict the absolute errors of the identified parameters. For radius they are below 5% , for position also about 5% .

In the second, the radius of the inclusion was $r = 0.1$. The figures 8–10 depict the absolute errors of the identified parameters. For radius they are below 5% , for position about 3% .

In the third, the radius of the inclusion was $r = 0.18$. The figures 11–13 depict the absolute errors of the identified parameters. For radius they are below 5% , for position about 2% .

First experiments seem to indicate, that the approach based on using topological derivative for producing training data for neural networks, gives promising results.

References

[1] D. Göhde, *Singuläre Störung von Randvertproblemen durch ein kleines Loch im Gebiet*, Zeitschrift für Analysis und ihre Anvendungen Vol.4(5)(1985), pp. 467–477.

[2] Hagan, M., Menhaj M., *Training feedforward networks with the Marquardt algorithm*, IEEE Trans. on Neural Networks, Vol. 5, No. 6, pp.989-993,1994.

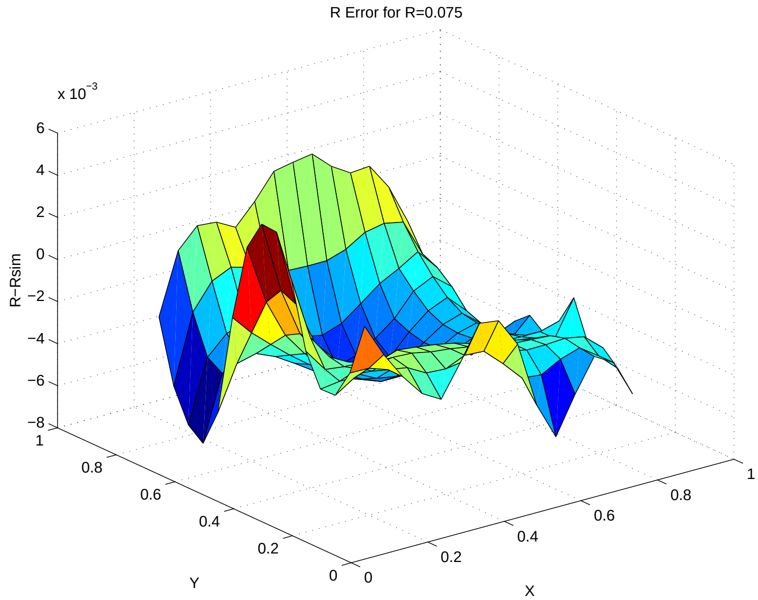

Figure 5:

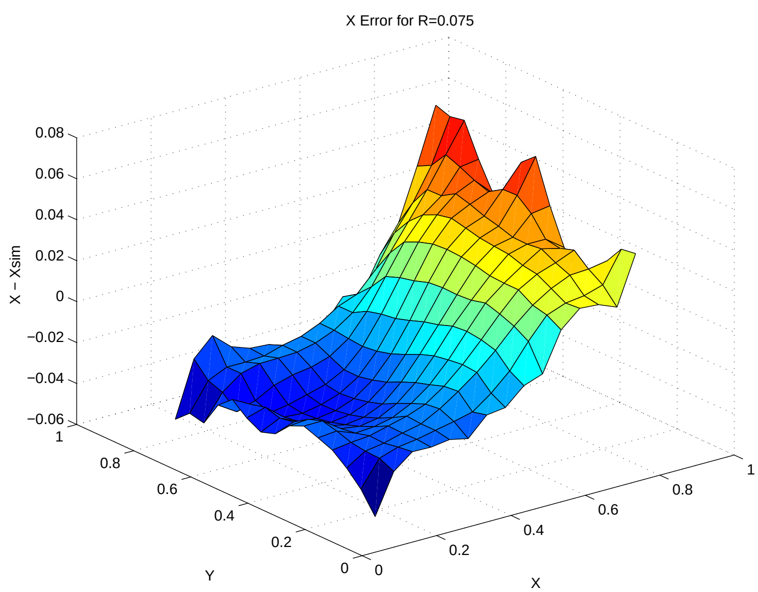

Figure 6:

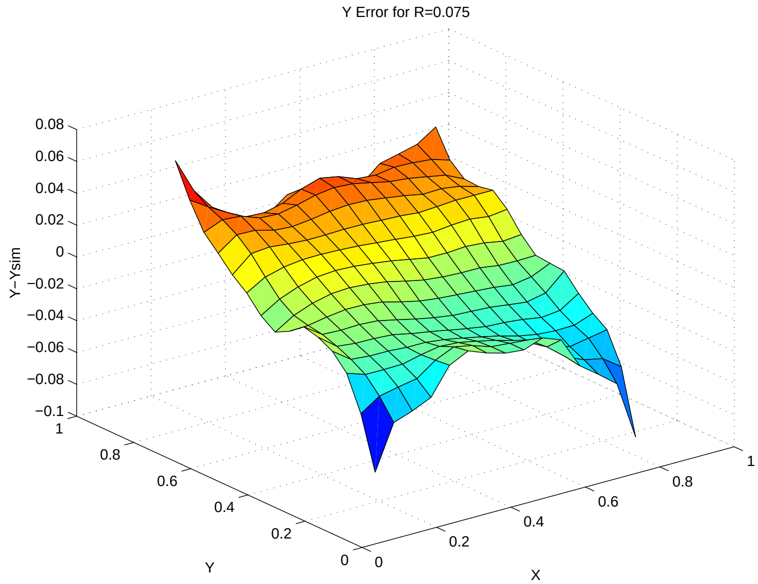

Figure 7:

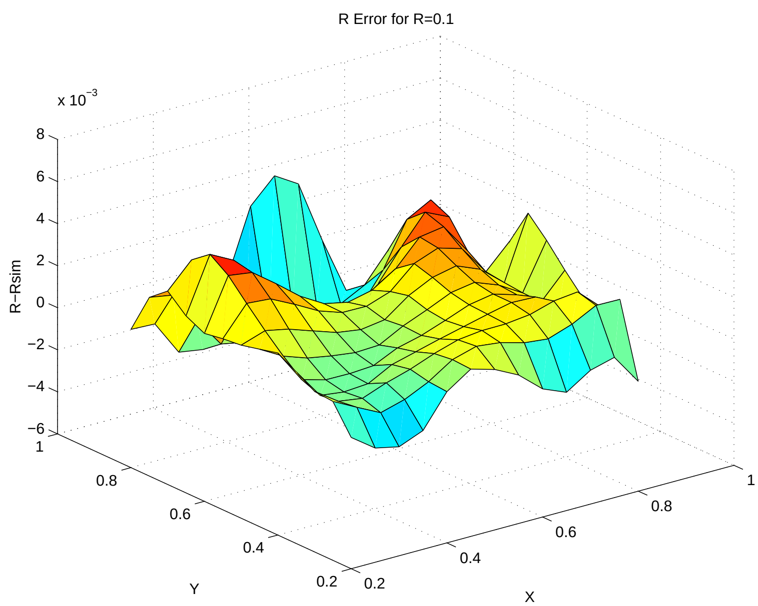

Figure 8:

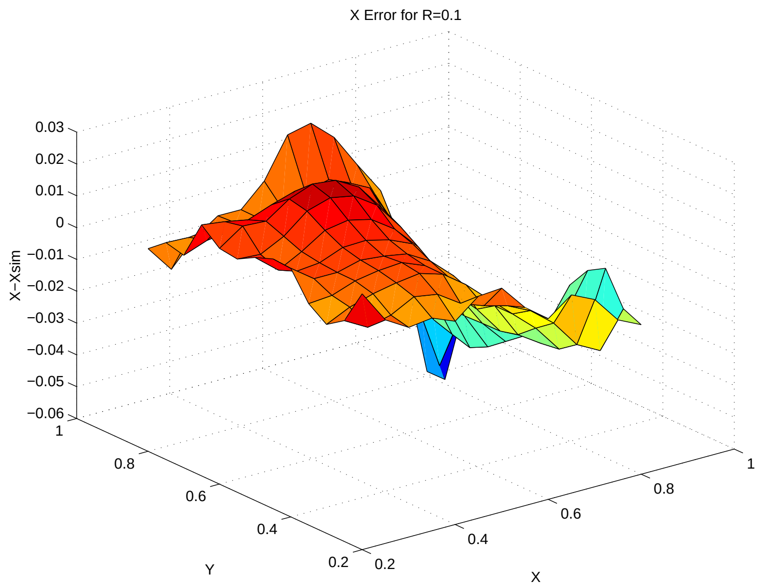

Figure 9:

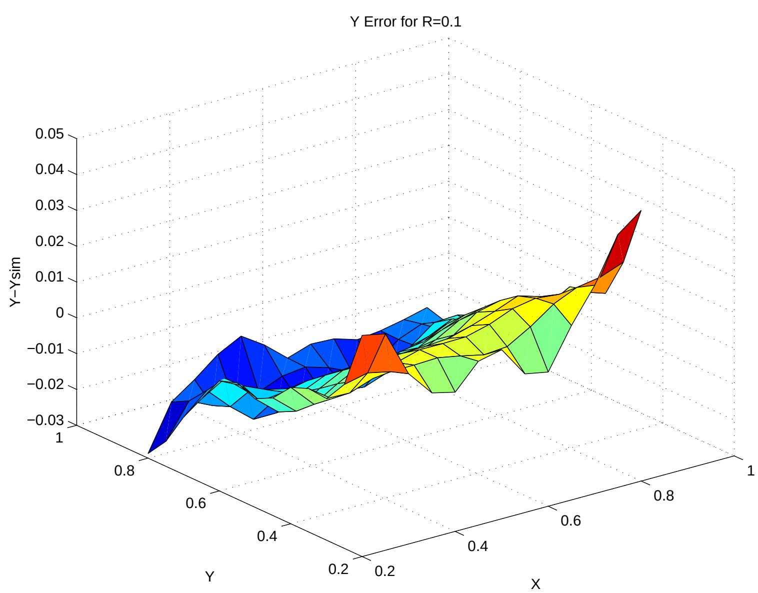

Figure 10:

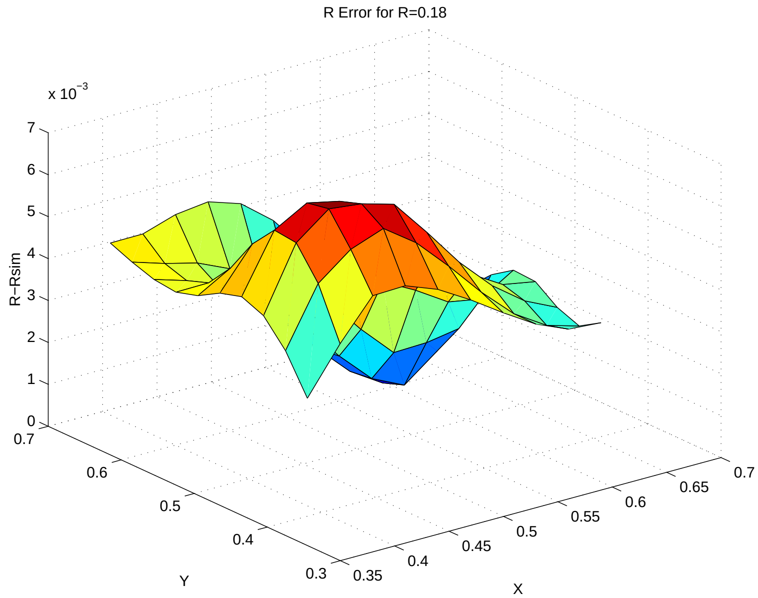

Figure 11:

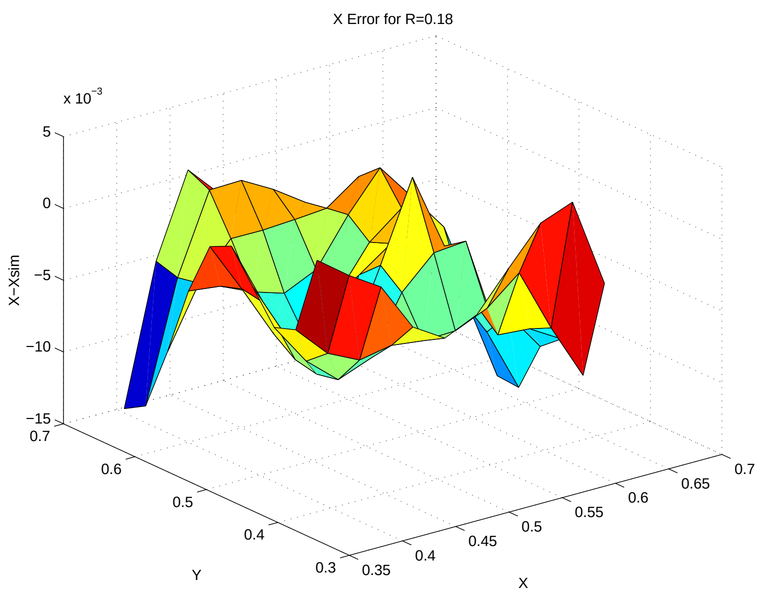

Figure 12:

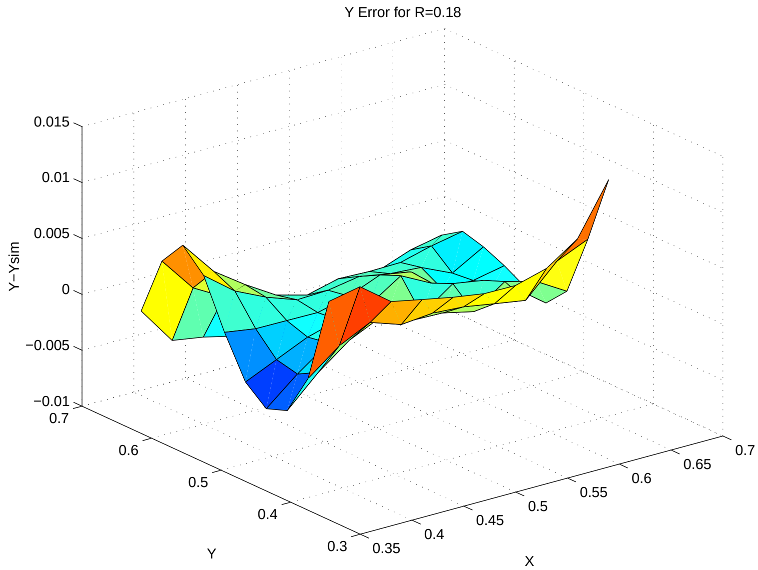

Figure 13:

[3] HAYKIN, S. *Neural Networks : a comprehensive foundation - 2nd ed.* Prentice-Hall , USA, 1999.

[4] HECHT-NIELSEN, *Neurocomputing*, Addison-Wesley Publishing Company, 1990.

[5] A. HERWIG, *Elliptische randvertprobleme zweiter Ordnung in Gebieten mit einer Fehlstelle*, Zeitschrift für Analysis und ihre Anvendungen No.8(2)(1989), pp. 153–161.

[6] K.HORNIK, M.STINCHCOMBE, H.WHITE, *Multilayer Feedforward Networks are Universal Approximators*, Neural Networks, Vol.3, pp. 551-560, 1990.

[7] A. M. IL'IN, *Matching of Asymptotic Expansions of Solutions of Boundary Value Problems*, Translations of Mathematical Monographs, Vol. 102, AMS 1992.

[8] T. LEWINSKI, J. SOKOLOWSKI, *Optimal Shells Formed on a Sphere. The Topological Derivative Method.* Rapport de Recherche No. 3495, 1998, INRIA-Lorraine

[9] W.G. MAZJA, S.A. NAZAROV, B.A. PLAMENEVSKI, *Asymptotische Theorie elliptischer Randwertaufgaben, I,* Akademie-Verlag, Berlin, 1991.

[10] J.R. ROCHE, J. SOKOŁOWSKI *Numerical methods for shape identification problems* Special issue of Control and Cybernetics: Shape Optimization and Scientific Computations, 5(1996), pp. 867-894.

[11] M. SCHIFFER, G. SZEGÖ, *Virtual mass and polarization*, Transactions of the American Mathematical Society, **67**(1949), pp. 130-205.

[12] A. SHUMACHER, *Topologieoptimierung von Bauteilstrukturen unter Verwendung von Lochpositionierungkriterien*, Ph.D. Thesis, Universität–Gesamthochschule–Siegen, Siegen, 1995.

[13] J. SOKOŁOWSKI, J-P. ZOLESIO, *Introduction to Shape Optimization. Shape Sensitivity Analysis*, Springer Verlag, 1992.

[14] SOKOŁOWSKI, J., ŻOCHOWSKI, A. *On topological derivative in shape optimization*, SIAM Journal on Control and Optimization. Volume 37, Number 4, 1999, pp. 1251-1272.

[15] Sokołowski, J., Żochowski, A. *Topological derivative for optimal control problems*, Proceedings of Fifth International Symposium on Methods and Models in Automation and Robotics, Międzyzdroje, Poland, August, 1998, pp. 111-116; paper to appear in Control and Cybernetics.

[16] Sokołowski, J., Żochowski, A. *Topological derivatives for elliptic problems*, Inverse Problems, Volume 15, Number 1, 1999, pp. 123-134.

Unit´e de recherche INRIA Lorraine, Technop^ole de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rh^one-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399