JOURNAL OF THEORETICAL AND APPLIED MECHANICS
42, 3, pp. 629-649, Warsaw 2004

NEURAL NETWORK ASSISTED CRACK AND FLAW IDENTIFICATION IN TRANSIENT DYNAMICS

Georgios E. Stavroulakis
Department of Applied Mathematics and Mechanics, University of Ioannina, Ioannina, Greece and TU Braunschweig, Germany; e-mail: gestavr@cc.uoi.gr

Marek Engelhardt
Institute of Applied Mechanics, Carolo Wilhelmina Technical University, Braunschweig, Germany e-mail: marek.engelhardt@tu-bs.de

Aristidis Likas
Department of Computer Science, University of Ioannina, Ioannina, Greece e-mail: arly@cs.uoi.gr

Rafael Gallego
Department of Structural Mechanics, University of Granada, Granada, Spain e-mail: gallego@ugr.es

Heinz Antes
Institute of Applied Mechanics, Carolo Wilhelmina Technical University, Braunschweig, Germany e-mmail: h.antes@tu-bs.de

Crack and flaw identification problems in two-dimensional elastomechanics are numerically studied in this paper. The mechanical modelling is based on boundary element techniques, with special care of hypersingular issues for the cracks. The possibility of partially or totally closed cracks (unilateral contact effects) is taken into account by linear complementarity techniques. Backpropagation neural networks are used for the solution of the inverse problems. For dynamical problems, a suitable preprocessing of the input data enhances the effectiveness of the procedure. For the two-dimensional examples presented here, the proposed method has similar performance for classical crack and flaw identification problems. The identification of unilateral cracks is a considerably more difficult task, which nevertheless, can also be solved by the same method, provided that a suitable dynamical test loading is applied.

Key words: crack identification, neural networks, unilateral contact, wave propagation, nondestructive evaluation

1. Introduction

Crack and flaw identification problems are solved in the paper. Accurate modelling of static and dynamic problems in two-dimensional linearly elastic bodies is done by means of the boundary element method, based on Gallego and Domínguez (1996, 1997). Unilateral effects, which allow for partial or total closure of cracks, are also included, following a linear complementarity problem formulation of the unilateral problem, see Antes and Panagiotopoulos (1992).

For the numerical solution of the inverse problem, usually the problem is transformed into an optimization task, following an output error minimization formulation. It is solved subsequently by numerical optimization (see, among others, Stavroulakis, 2000; Rus and Carlborg, 2001; Rus and Gallego, 2002). Neural networks are able to directly approximate the relation between measurements and unknown parameters. They have been tested for the solution of crack identification problems in statics (Stavroulakis and Antes, 1997), and in dynamics (both steady state, harmonic elastodynamics, in Stavroulakis and Antes (1998), and time domain elastodynamics in Stavroulakis (1999), see also Stavroulakis (2000)). The mentioned optimization task is now hidden in the training phase of the neural network.

This general methodology is extended in the present paper for the solution of two-dimensional problems. From the mechanical point of view, this paper presents a comparison of the identifiability of flaws (holes) with respect to cracks and a numerical study of the effect of non-classical, due to the unilateral contact phenomenon, cracks. From the data processing point of view, the main task is the reduction of the size of data used for the training of the neural network. A partial answer to this problem is proposed and tested here. The modelling of back propagation neural networks is based on the Neural Network Toolbox of MATLAB and on home made programs (Likas *et al.*, 1998).

2. Static and dynamic identification problems

Based on the loading excitation, which is used for the testing of a structure, damage and crack identification problems can be divided into static problems, harmonic dynamic or modal analysis problems and transient dynamic problems. For solving the inverse problem, first, the unknown quantities like number and type of defects, their position and other shape constants are expressed with the help of certain variables. In the next step, a number of

mechanical tests are considered, where for each value of the unknown defect parameters, the corresponding responses of the structure are compared with the wished (measured) ones.

Several methods for the effective numerical solution to this problem have been tested (among others, numerical optimization, genetic algorithms, soft computing, see Stavroulakis (2000)). The method used here employs a neural network to solve the inverse problem, and supports on-line implementation. Analogous results were published in statics (Liang and Hwu, 2001) and in dynamics (one-dimensional problems in Ziemianski and Piatkowski (2000), two-dimensional problems in Oishi *et al.* (1995)). Moreover, the evaluation of ultrasonic data has been performed in several cases by means of neural network models, see, among others, Zgonc and Achenbach (1996) and the review articles Yagawa and Okuda (1996), Zeng (1998).

In the author's previous work, dynamical problems on relatively simple layered structures with cracks or defects, which were parallel to the layers, were considered (Stavroulakis, 1999). In the present paper, the method is extended to general two-dimensional structures with measurements of dynamical responses on several different parts of the boundaries. The size of the data (measured waveforms) increases dramatically, therefore a data reduction scheme must be used to allow for effective treatment by neural networks. Concerning the identification of unilateral cracks, except for the previous work by the authors summarized in Stavroulakis (2000), and the paper Alessandri and Mallardo (1999) which was based on the classical optimization for the solution to statical problems, no other published work is available to the best knowledge of the authors.

3. Boundary element modelling of the mechanical problem

3.1. Boundary integral equations for elastostatics

The formulation of the direct elastomechanical problem is based on the basic equations of the theory of elasticity, i.e., on Navier's equation

$$\mu u_{i,jj}(\boldsymbol{x}) + (\mu + \lambda) u_{j,ij}(\boldsymbol{x}) + b_i(\boldsymbol{x}) = 0 \tag{3.1}$$

where the *Lamé* constants μ and λ are used. Further, one formulates the *weighted residuum*

$$\int_{\Omega} [u_{i,jj}(\boldsymbol{x}) + (\mu + \lambda) u_{j,ij}(\boldsymbol{x}) + b_i(\boldsymbol{x})] u^*_{ik}(\boldsymbol{x}, \boldsymbol{\xi})\, d\Omega = 0$$

where $u^*_{ik}(\boldsymbol{x}, \boldsymbol{\xi})$ is the so called *fundamental solution*, i.e., the displacement field at the point $\boldsymbol{x}$ (observation point), due to a unit force applied at the point $\boldsymbol{\xi}$ (collocation or source point), in an infinite domain.

Integrating by parts and taking into account that the fundamental solution fulfills governing equations (3.1), the previous equation leads to

$$u_k(\boldsymbol{\xi}) = \int_\Gamma [p_i(\boldsymbol{x}) u^*_{ik}(\boldsymbol{x};\boldsymbol{\xi}) - u_i(\boldsymbol{x}) p^*_{ik}(\boldsymbol{x};\boldsymbol{\xi})]\, d\Gamma_x + \int_\Omega b_i(\boldsymbol{x}) u^*_{ik}(\boldsymbol{x};\boldsymbol{\xi})\, d\Omega_x \quad (3.2)$$

where $p_i(\boldsymbol{x})$ and $p^*_{ik}(\boldsymbol{x};\boldsymbol{\xi})$ are the traction vectors at the boundary Γ corresponding to the displacement fields u_i and u^*_{ik}, respectively. In equation (3.2), the collocation point belongs to the interior of the domain, i.e., $\boldsymbol{\xi} \in \Omega$ but $\boldsymbol{\xi} \notin \Gamma$. To obtain the displacement Boundary Integral Equation a limiting process must be carried out, which finally leads to

$$c_{ki}(\boldsymbol{\xi}) u_i(\boldsymbol{\xi}) = \fint_\Gamma [p_i(\boldsymbol{x}) u^*_{ik}(\boldsymbol{x};\boldsymbol{\xi}) - u_i(\boldsymbol{x}) p^*_{ik}(\boldsymbol{x};\boldsymbol{\xi})]\, d\Gamma_x \quad (3.3)$$

where the *free term* c_{ik} depends on the position of the collocation point: $c_{ik} = \delta_{ik}$ if $\boldsymbol{\xi}$ belongs to the interior of Ω; $c_{ik} = 0$ if $\boldsymbol{\xi} \notin \Omega$; if $\boldsymbol{\xi} \in \Gamma$, the value of c_{ik} depends on the local geometry of the boundary at $\boldsymbol{\xi}$; if the normal to the boundary is continuous at $\boldsymbol{\xi} \in \Gamma$ then $c_{ik} = \delta_{ik}/2$. $\fint$-stands for the Cauchy Principal Value (CPV) of the integral.

3.2. Displacement boundary integral representation for transient elastodynamics

Consider an elastodynamic problem in the domain Ω subject to transient loads. The displacement field u_i satisfies Navier's differential equations in the domain

$$(c_1^2 - c_2^2) u_{i,ij}(\boldsymbol{x}, t) + c_2^2 u_{j,ii}(\boldsymbol{x}, t) - \ddot{u}_j(\boldsymbol{x}, t) = -\frac{b_j(\boldsymbol{x}, t)}{\rho} \quad (3.4)$$

where c_1 and c_2 are the dilatational and shear wave velocities, ρ is the density and b_j are the components of the body force per unit volume. The initial and boundary conditions have to be satisfied in the domain at $t = 0$, and on the boundary Γ, respectively. The stresses at a point in the domain can be obtained from the displacement field using Hooke's law

$$\sigma_{im} = \rho[\delta_{im}(c_1^2 - 2c_2^2) u_{j,j} + c_2^2 (u_{i,m} + u_{m,i})] \quad (3.5)$$

and the traction at a point on Γ, whose outward normal has components n_k, can be computed from the stresses

$$p_i = \sigma_{ik} n_k = \rho[n_i(c_1^2 - 2c_2^2)u_{j,j} + n_k c_2^2(u_{i,k} + u_{k,i})] \tag{3.6}$$

where p_i are the components of the traction vector.

The displacement at a point $\boldsymbol{\xi}$ and time t can be represented in terms of the displacements and tractions on the boundary by

$$\begin{aligned} c_{ij}(\boldsymbol{\xi})u_j(\boldsymbol{\xi},t) + \int\!\!\!\!\!\!-_{\Gamma} \int_0^{t^+} \bar{p}^*_{ij}(\boldsymbol{x}, t-\tau; \boldsymbol{\xi}) u_j(\boldsymbol{x},\tau)\, d\tau\, d\Gamma = \\ = \int_{\Gamma} \int_0^{t^+} (u^*_{ij}(\boldsymbol{x}, t-\tau; \boldsymbol{\xi}) p_j(\boldsymbol{x},\tau)\, d\tau\, d\Gamma \end{aligned} \tag{3.7}$$

where $u^*_{ij}(\boldsymbol{x}, t-\tau; \boldsymbol{\xi})$ is the displacement field in an infinite medium due to a unit impulse in the direction x_i located at the point $\boldsymbol{\xi}$ and acting at the time τ; $\bar{p}^*_{ij}(\boldsymbol{x}, t-\tau; \boldsymbol{\xi})$ is the corresponding traction field obtained from the displacements by Hooke's law (3.6); c_{ij} is equal to the free term in equation (3.3); $\int_0^{t^+} = \lim_{\varepsilon \to 0} \int_0^{t+\varepsilon}$. Zero initial conditions and zero body forces b_j have been assumed.

The traction tensor $\bar{p}^*_{ij}$ contains Dirac's delta functions which preclude a direct numerical integration of eqaution (3.7) (see Mansur (1983), Domínguez (1993) for technical details). To eliminate Dirac's delta terms, the spatial derivative of the Heaviside function is related to its time derivative, and integration by parts is performed. This leads to the integral equation

$$c_{ij}u_j + \int_{\Gamma} \int_0^{t^+} (p^*_{ij}u_j + p^{v*}_{ij}\dot{u}_j)\, d\tau\, d\Gamma = \int_{\Gamma} \int_0^{t^+} u^*_{ij} p_j\, d\tau\, d\Gamma \tag{3.8}$$

where $\dot{u}_j$ is the velocity field, and p^*_{ij} and p^{v*}_{ij} are new kernels where Dirac's delta terms have been removed.

3.3. Traction boundary integral representation for transient elastodynamics

From the integral representation of the displacements at an interior point (3.8), the corresponding integral representation for the stress tensor at this

point can be obtained by Hooke's law

$$\sigma_{im}(\boldsymbol{\xi},t)+\int_{\Gamma}\int_{0}^{t^+}\bar{d}^*_{imj}p_j\;d\tau\,d\Gamma=\int_{\Gamma}\int_{0}^{t^+}(\bar{s}^*_{ij}u_j+\bar{s}^{*v}_{ij}\dot{u}_j)\;d\tau\,d\Gamma \tag{3.9}$$

where

$$\bar{d}^*_{imj}=\rho[\delta_{im}(c_1^2-2c_2^2)p^*_{kj,k}+c_2^2(p^*_{ij,m}+p^*_{mj,i})] \tag{3.10}$$

and similar equations for the rest of the kernels.

Again, Dirac's delta function terms appear in the kernels of equation (3.9), due to the differentation of the Heaviside functions in the kernels of (3.8). Integration by parts leads to the equation

$$\begin{aligned}\sigma_{im}+\int_{\Gamma}\int_{0}^{t^+}(d^*_{imj}p_j+d^{v*}_{imj}\dot{p}_j)\;d\tau\,d\Gamma=\\ =\int_{\Gamma}\int_{0}^{t^+}(s^*_{imj}u_j+s^{v*}_{imj}\dot{u}_j+s^{a*}_{imj}\ddot{u}_j)\;d\tau\,d\Gamma\end{aligned} \tag{3.11}$$

where $\ddot{u}_j$ is the acceleration field; $\dot{p}_i$ is the time derivative of the traction vector; the kernels, d^*_{imj}, d^{v*}_{imj}, s^*_{imj}, s^{v*}_{imj} and s^{a*}_{imj} do not contain any strongly singular term (Dirac's delta functions).

Before carrying the former equation to the boundary it is necessary to asses the order and location of the singularities involved in the time and space integrations. It can be shown that the kernels have the same singularities as the corresponding ones in the dynamic antiplane formulation (Gallego and Domínguez, 1995, 1996), and therefore a similar regularization approach can be performed to find the traction boundary integral representation (Guiggiani, 1992). The final boundary integral representation for the traction at a smooth boundary, whose outward normal is n_m, can be written as

$$c_{ij}p_j+\oint\!\!\!\!\!\!=_{\Gamma}\int_{0}^{t^+}(d^*_{ij}p_j+d^{v*}_{ij}\dot{p}_j)\;d\tau\,d\Gamma=\oint\!\!\!\!\!\!=_{\Gamma}\int_{0}^{t^+}(s^*_{ij}u_j+s^{v*}_{ij}\dot{u}_j+s^{a*}_{ij}\ddot{u}_j)\;d\tau\,d\Gamma \tag{3.12}$$

where $c_{ij}=\delta_{ij}/2$; $d^*_{ij}=d^*_{imj}n_m$, $d^{v*}_{ij}=d^{v*}_{imj}n_m$, $s^*_{ij}=s^*_{imj}n_m$, $s^{v*}_{ij}=s^{v*}_{imj}n_m$ and $s^{a*}_{ij}=s^{a*}_{imj}n_m$; $\oint\!\!\!\!\!\!=$ stands for the Hadamard Finite Part of the integral. Equation (3.12) can be written at interior points and points outside Ω with $c_{ij}=1$ and $c_{ij}=0$, respectively.

3.4. Application of the traction integral equation to crack problems

A mixed Boundary Element formulation has been proposed to solve fracture mechanics problems in static (Portela *et al.*, 1992; Sáez *et al.*, 1995) and dynamic loadings (Fedelinski *et al.*, 1994). A more refined approach, taken from Gallego and Domínguez (1997), which is essential for problems with a nonzero traction on the crack faces, is used here.

Let us call Γ^+ and Γ^- the upper and lower face of the crack, respectively, and Γ_c the rest of the boundary. The superscripts '+' and '−' denote any function evaluated on the upper and lower face of the crack, respectively. It could be checked that $d_{ij}^{*+} = d_{ij}^{*-}$, $d_{ij}^{v*+} = d_{ij}^{v*-}$, $s_{ij}^{*+} = -s_{ij}^{*-}$, $s_{ij}^{v*+} = -s_{ij}^{v*-}$ and $s_{ij}^{a*+} = -s_{ij}^{a*-}$. Therefore, assuming that $\Delta p_j = p_j^+ + p_j^- = 0$, equation (3.12) for an interior point can be written as

$$p_i + \int_{\Gamma_c} \int_0^{t^+} (d_{ij}^* p_j + d_{ij}^{v*} \dot{p}_j)\, d\tau\, d\Gamma = \int_{\Gamma_c} \int_0^{t^+} (s_{ij}^* u_j + s_{ij}^{v*} \dot{u}_j + s_{ij}^{a*} \ddot{u}_j)\, d\tau\, d\Gamma + \\ + \int_{\Gamma^+} \int_0^{t^+} (s_{ij}^* \Delta u_j + s_{ij}^{v*} \Delta \dot{u}_j + s_{ij}^{a*} \Delta \ddot{u}_j)\, d\tau\, d\Gamma \tag{3.13}$$

where $\Delta u_j = u_j^+ - u_j^-$ is the crack displacement jump. If this equation is carried to a point on the boundary Γ^+, the following expression is obtained

$$p_i^+(\boldsymbol{\xi}, t) + \int_{\Gamma_c} \int_0^{t^+} (d_{ij}^* p_j + d_{ij}^{v*} \dot{p}_j)\, d\tau\, d\Gamma = \int_{\Gamma_c} \int_0^{t^+} (s_{ij}^* u_j + s_{ij}^{v*} \dot{u}_j + s_{ij}^{a*} \ddot{u}_j)\, d\tau\, d\Gamma + \\ + \text{=}\!\!\!\!\!\!\int_{\Gamma^+} \int_0^{t^+} (s_{ij}^* \Delta u_j + s_{ij}^{v*} \Delta \dot{u}_j + s_{ij}^{a*} \Delta \ddot{u}_j)\, d\tau\, d\Gamma \tag{3.14}$$

The difference between this equation and the general traction representation, Eq (3.12), is the free term, which is 1 for the crack point instead of 1/2.

Hypersingular boundary integral equation (3.14) in the crack face Γ^+ and the standard displacement boundary integral representation on the rest of the boundary Γ_c provide a complete set of equations to compute the tractions and/or displacement or displacement jump on the boundary and the crack faces.

3.5. Boundary element discretization

The existence of the hypersingular boundary integral equation demands C^1 continuity on the displacement at the collocation point. On the other hand, the collocation point should be placed at a smooth boundary point, since otherwise the value of the traction is not unique. The boundary element discretization must fulfill both conditions, as it has been discussed in Gallego and Domínguez (1997). Taking this into account and using standard Boundary Element Techniques otherwise for spatial and time discretization, the resulting set of algebraic equations is written in a compact matricial notation as

$$\mathbf{G}^{nn}\boldsymbol{p}^n = \mathbf{H}^{nn}\boldsymbol{u}^n + \sum_{m=1}^{n-1}(\mathbf{H}^{nm}\boldsymbol{u}^m - \mathbf{G}^{nm}\boldsymbol{p}^m) \tag{3.15}$$

where the arrays $\boldsymbol{u}^m$ and $\boldsymbol{p}^m$ contain the displacements and tractions of the nodes on Γ_c and the displacement jumps and tractions on Γ^+, for the time step m.

At this point, the boundary conditions of the elastomechanical problem at the time m are taken into account, known and unknown quantities are separated and a system of linear equations is formulated and solved for all unknown boundary displacements with respect to tractions $\boldsymbol{y}$

$$\mathbf{G}^{nn}\boldsymbol{p}^n = \mathbf{H}^{nn}\boldsymbol{u}^n + \boldsymbol{R}^n \;\;\Rightarrow\;\; \mathbf{A}^{nn}\boldsymbol{y}^n = \boldsymbol{f}^n \tag{3.16}$$

This system of equations is solved step by step as in the standard displacement formulation in the time domain.

3.6. Unilateral contact at cracks (partial closure): a linear complementarity problem

Unilateral contact problems can be considered by using a LCP-BEM (linear complementarity – boundary element) method, analogous to the one developed in the case of the two-region BEM in Antes and Panagiotopoulos (1992), Stavroulakis (1997, 2000). Without going into details, let us mention that, with a suitable sign assumption, a unilateral contact mechanism implying no-penetration and no-tension between the boundary displacements u and boundary tractions t is described by the set of inequalities and the complementarity condition

$$u \leqslant 0 \qquad t \leqslant 0 \qquad u^\top t = 0 \tag{3.17}$$

The LCP is composed of the latter set of relations for all unilateral mechanisms (here, on all crack nodes relating crack opening and contact traction) together with the underdetermined system of equations shown in the first part of (3.16). For other approaches see, in addition, Guz and Zozulya (2002).

4. Neural networks for crack and flaw identification

4.1. The direct-inverse neural modelling technique

Let a given structure which contains an unknown crack be considered. The crack is characterized by a set of parameters $\boldsymbol{z} = [z_1, \ldots, z_m]^\top$. For example, in the crack identification, the coordinates of the crack center and the length of the crack are the parameters of interest. Let, moreover, the response of the structural system for a given loading $\boldsymbol{b}^l$, $l = 1, \ldots, l_1$ and for a given crack $\boldsymbol{z}$ be given by the vector $\widetilde{\boldsymbol{x}}(\boldsymbol{z}, \boldsymbol{b}^l)$ obtained as the solution to the corresponding static or dynamic mechanical problem. Here l_1 is the total number of different loading cases. Obviously, the response of the considered mechanical system is parametrized by the unknown crack parameters $\boldsymbol{z}$. Let, moreover, the response of the examined structure with the known crack subjected to the same loading $\boldsymbol{b}^l$ be denoted by $\widetilde{\boldsymbol{x}}_0(\boldsymbol{z}, \boldsymbol{b}^l)$.

Here, a feedforward neural network is used to learn the inverse mapping

$$\widetilde{\boldsymbol{x}}(\boldsymbol{z}, \boldsymbol{b}^l) \to \boldsymbol{z} \tag{4.1}$$

for a given value of the loading vector $\boldsymbol{b}^l$ using an appropriate dataset of the example cases. The network takes the vector $\widetilde{\boldsymbol{x}}$ as the input and provides the corresponding vector $\boldsymbol{z}$ of crack parameters as the output.

The data pairs composed of the vectors $\widetilde{\boldsymbol{x}}(\boldsymbol{z}, \boldsymbol{b}^l)$ and the corresponding parameter vectors $\boldsymbol{z}$ are used as training examples. In the prediction mode, the nonlinear network reproduces the mapping $\boldsymbol{x} \to \boldsymbol{z}$, i.e., for a given vector of measurements $\widetilde{\boldsymbol{x}}$ (different from the ones used for training) it gives a prediction for the variables characterising the internal crack.

The previously outlined method for the direct inverse modelling can be extended to treat problems in elastodynamics by enlarging the input vector such that it takes into account the whole time series. This method was used in our previous investigations (see Stavroulakis, 2000). The problems which arise with the above *simple* treatment of the inverse problems in elastodynamics is that the dimension of the input vectors is dramatically increased and that a lot of this huge information is actually redundant. The redundancy is easily explained from the fact that, for example, all measurements at the external boundary before the appearance of the first wave reflected from the unknown defect do not convey any information about this defect and therefore do not help at all for the solution to the inverse problem. In this work, we have chosen a few suitably selected time instances along each waveform which seemed characteristic for the problems under study. These time instances correspond to the local maxima and minima, as well as turning points of the waveforms.

This way, only 4-5 time steps are kept for every measurement point, thus the increase in the number of input dimension is not large compared to the static case.

4.2. The neural network model and training algorithm

The neural network model that we used to implement the inverse mapping is the well-known Multilayer Perceptron (MLP). It is the most widely used neural network model for function approximation with numerous successful applications in almost every scientific and engineering domain. The most attractive feature of the MLP is that it exhibits excellent interpolation capabilities (even when trained with sparse datasets) which make it an ideal solution for data-driven inverse modelling problems.

The MLP model (also known as backpropagation neural network) is a feedforward neural network with one or more hidden layer containing units (called hidden units) with a nonlinear activation function (usually of sigmoid type). In our experiments, to implement a mapping from a d-dimensional input space to an m-dimensional output space, we have used the MLP with d inputs, m outputs and one hidden layer with H hidden units with the hyperbolic tangent sigmoid activation function

$$\tanh x = \frac{\mathrm{e}^{x} - \mathrm{e}^{-x}}{\mathrm{e}^{x} + \mathrm{e}^{-x}}$$

More specifically, if $[W] = [w_{ij}]$ denotes the weight matrix from the input units to the hidden units, $\mathbf{V} = [v_{ij}]$ denotes the weight matrix from the hidden units to the output units and b_i denotes the bias of the hidden unit i then for a given input vector $\boldsymbol{x} = [x_1, \ldots, x_d]$, the corresponding output vector $\boldsymbol{y} = [y_1, \ldots, y_m]$ is computed as follows:

- First the outputs of the hidden units are computed

$$z_i = \tanh\Big(\sum_{j=1}^{d} w_{ij} x_j + b_i\Big) \qquad i = 1, \ldots, H$$

- Next, the network outputs y_k are computed using the z_i values

$$y_k = \sum_{j=1}^{H} v_{kj} z_j \qquad k = 1, \ldots, m$$

The weights (w_{ij}, v_{ij}) and biases (b_i) constitute the neural network parameters to be adjusted during training in order to learn the network to implement the desired mapping.

The MLP model can be trained to implement the desired inverse mapping from a d-dimensional to an m-dimensional domain by using a training set that contains N examples of the mapping, ie. pairs of the form $(\boldsymbol{x}_i, \boldsymbol{t}_i)$ where $\boldsymbol{x}_i = [x_{i1}, \ldots, x_{id}]$ is an input vector and $\boldsymbol{t}_i = [t_{i1}, \ldots, t_{im}]$ the desired output for the input $\boldsymbol{x}_i$. Once the training set is available, the training process is actually an optimization procedure that adjusts the network parameters (weights and biases) to minimize the error function

$$E = \frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{m} (y_k(\boldsymbol{x}_i) - t_{ik})^2$$

To achieve the error minimization, any numerical optimization method can be applied, from simple gradient-descent (also called backpropagation training algorithm) to more sophisticated quasi-Newton methods or even global optimization methods. In this work, the Levenberg-Marquadt method has been used for the minimization of the error function that is available in the Matlab Neural Network toolbox. This training method has been found to be the most effective among several tested local optimization techniques and achieved to provide near zero minima of the error function even in the case of networks with a small number H of hidden units.

An important issue for the construction of an effective neural network model is the specification of the number of hidden units H. It is well-known that for large values of H the network tends to overfit the training set. This means that although the network learns the training set very accurately (the training error becomes very small), the prediction performance of the network on new examples (not used for training) is poor. On the other hand, if the number of hidden units H is very small the network does not manage to learn the training set with acceptable accuracy. Therefore, a procedure called the *complexity control* is needed to find a reasonable value for the number of hidden units H. The objective of the complexity control is to identify the *smallest* neural architecture that is able to learn the training set with acceptable accuracy. In this work, we applied the complexity control by starting with a small network having $H = 2$ hidden units and gradually increasing the value of H by one, until a sufficiently trained network (with a low error value) is obtained. Since the training algorithm (Levemberg-Marquadt) is local and depends on the initial values of the weights, for each value of H we applied the training

algorithm at most 20 times, starting each time from random weight initial values.

Finally, for every problem examined, after the completion of training, the prediction accuracy of the constructed network was assessed by using a separate test set of cases (different from the training set).

5. Mechanical modelling

Consider an academic two-dimensional problem that is concerned with defect idenfication within a rectangular elastic plate from measurements along its external boundaries, see Fig. 1. It should be pointed out that we are considering relatively small defects, with length or diameter equal to 1, in a plate with the external side equal to $l = 100$.

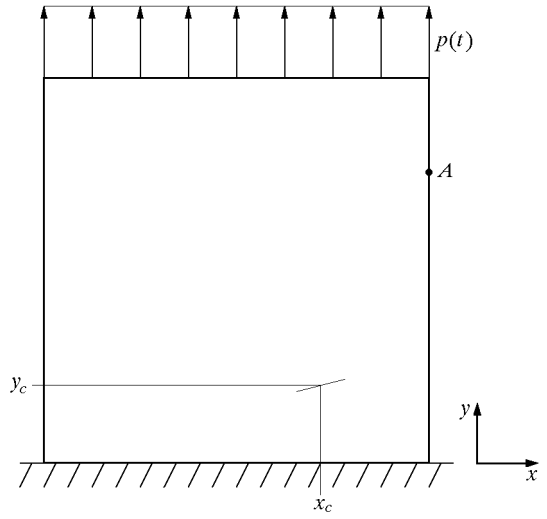

Fig. 1. Geometry and loading of the considered plate with a crack

For the assumed defect, the coordinates of its center are denoted by x_c and y_c. The lower boundary of the plate is fixed and the loading is applied at the upper boundary. For the elastic material an elasticity modulus $E = 1000000$ and Poisson's ratio $\nu = 0,3$ are used. For the discretization of the whole external boundary of the plate 10 quadratic elements are used. For each hole, 20 quadratic elements in statics and 10 in dynamics, and for every crack 7 quadratic crack elements are used.

We have considered three cases of defects. First, a circular hole is used for the modelling of the defect. Further, a horizontal rectilinear crack is used. Finally, unilateral contact phenomena are considered along the crack sides. The dynamic loading has either a form of the Heaviside function or of a dynamical pulse of one wave duration. The pulse loading has a form of a sinusoidal excitation or an impact-like one. The displacement records at specific points of the boundary are used as the input for the neural networks. For the model problem presented in Fig. 1, this information is shown, for one point at the boundary of the plate, in Fig. 2. The various lines correspond to various positions of the defect. Clearly, one can use these lines in order to identify the defect.

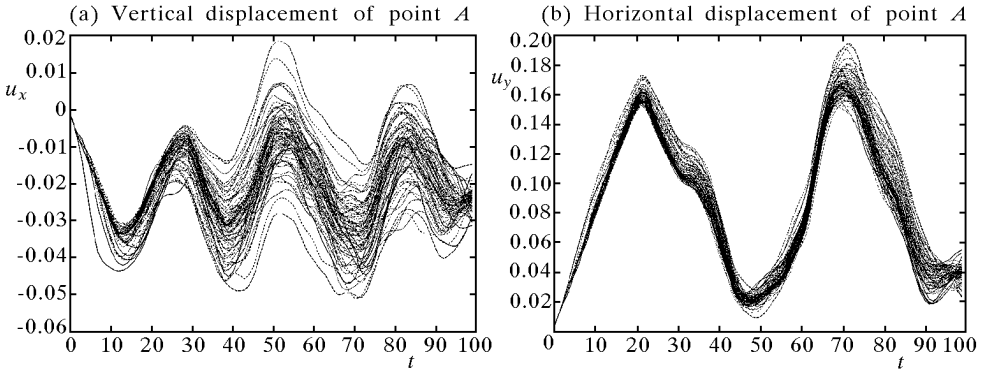

Fig. 2. Horizontal (x) and vertical (y) displacements at boundary point A: Dependence on the crack position

The preparation of data for the neural network postprocessing and the subsequent solution to the inverse problem has been done in the following way. A learning set, for the training of the neural network, and a test set, for the demonstration of its ability, are produced. The position of the assumed defect changes in each element of these two sets of examples (paradigms). The training set includes displacements of the upper side for positions of the hole with $x = 10$ to $x = 90$ and from $y = 10$ to $y = 90$ with all combinations produced with the step $\Delta x = \Delta y = 10$. Therefore, the training set has 81 examples (i.e. different positions of the defect, different mechanical problems), each one including 21 measurements. For the training set, the coordinates of the hole center are considered from $x_c = 15$ and $y_c = 15$ with steps $\Delta x = 10$, $\Delta y = 10$ up to the defect with coordinates at $x_c = 85$ and $y_c = 85$ (all intermediate combinations). Thus, the training set has 64 examples with different positions of the defect. We have chosen measurement points at time instants where the

displacement values comming from the considered examples, i.e. with different crack positions, differ as much as possible. These are, practically, the turning points of the waveforms shown in Fig. 2. In addition, the time instants where the wave reflections seem to play a significant role in the data are chosen (local minima and maxima of the waveform). This way one tries to have a minimum size of the input data, so that significant features of the measurements are still represented. In a future step of the investigation this point could be done automatically. For the hole identification 81 examples with various measurement points on the boundary of the plate have been used.

The crack identification follows a similar procedure. Nevertheless, a smaller area within the plate is considered for the placement of potential cracks and the calculation of training and test data. This is due to the fact that the crack-type defects with their stress singularity at the end of the crack make the dynamic boundary element method we used unstable for crack positions near the external boundaries. A finer discretization would resolve this numerical instability and make the computational effort significantly higher. Nevertheless, we did not use a finer discretization, in order to be able to directly compare the effectiveness of the procedure with the previously solved hole identification problem. The training set consisted of displacement values at the external boundary of the plate from 49 simulations. These data were produced by taking into account a crack with the center coordinates from $x_c = 20$ and $y_c = 20$ to $x_c = 80$ and $y_c = 80$ with the step $\Delta x = 10$ and $\Delta y = 10$ and all intermediate combinations. The test set had, analogously, 36 simulations.

5.1. Numerical results

5.1.1. Example 1: hole identification in elastostatics with two unknowns

A circular hole of diameter equal to 4.0 is considered to be the unknown defect. The coordinates (x, y) of its center are the unknown parameters of the inverse problem. The defect is hidden within a rectangular plate with dimensions equal to 100.0×100.0. One static loading case (pure traction on the upper side of the plate, fixed boundary on the opposite side) is considered. The boundary displacements (10 nodes per boundary) are used as measurements for the solution to the inverse problem. Thus, the dataset contains input-output pairs with 20-dimensional inputs and two-dimensional outputs with values normalized in the range $(-0.9, 0.9)$. These are the displacements of the boundary nodes, in the x and y direction with respect to the reference otrhogonal coordinate system, as they are calculated by the boundary element method. For the construction of the training set we considered all possible positions of the center. In particular, for the training set all values of x and y

coordinates in the interval $[10-90]$ with a step equal to 10 have been used. For the test set we used values in the interval $[15-85]$ with a step equal to 10. The training dataset contains 81 cases, while the test set includes 64 cases. We used a neural network model with 20 inputs, 2 outputs and $H=5$ hidden units. The training error achieved was less than 0.001, and the results for the training set are shown in Fig. 3a. The performance on the test set of cases is illustrated in Fig. 3b.

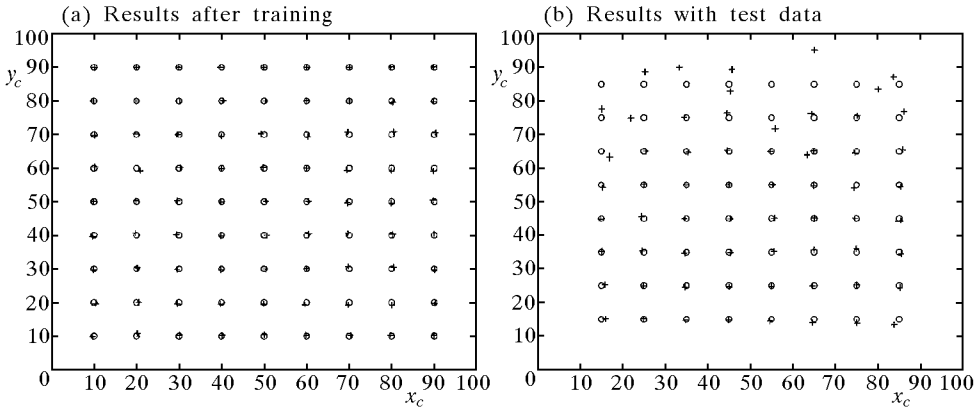

Fig. 3. Example 1. Performance of the neural network on the training and the test set; ∘ given position, + predicted position

A very interesting issue to note was that in order to achieve successful training, it was not necessary to use the complete 20-dimensional input vector. Instead, if only a small part of this vector was used (for example eight components) the same training and test performance was achieved. This fact suggests that it is possible to solve the inverse problem using sparse boundary displacements (eg. three nodes instead of ten) or a lower number of measurements during an experiment, and needs further investigation. Similar results can be obtained with different sizes of holes and with classical (bilateral) cracks.

5.1.2. Example 2: hole identification in elastostatics with three unknowns

In this case, both the center of the circular defect and its diameter are considered to be unknown. Holes with a diameter between 8 and 12 have been considered. All sites in between these values, with a step equal to 0.2 have been used for the construction of the training set. In the test set, diameters between 8.5 and 11.5 with a step equal to 0.2 have been considered.

The dataset contains input-output pairs with 20-dimensional inputs and three-dimensional outputs (two outputs for the position coordinates and one

output for the diameter) normalized in the range $(-0.9, 0.9)$. The training dataset contains 1701 cases, while the test set includes 1024 cases.

In the first experiment we tried to use the available training set to train *a single network* with three outputs that simultaneously provide both the location and diameter of the defect for a given input vector of measurements. Nevertheless, *it was impossible to successfully train such a network and obtain results of reasonable accuracy on the test set*, especially in terms of the diameter of the defect. For this reason, we followed a different (two-stage) approach that involved a cascade of *two neural networks*.

In the first stage, the neural network with two outputs is trained. It takes as the input the vector of measurements and provides as the outputs the (x, y) coordinates of the defect. It must be noted that the diameter measurements are not taken into account for the construction of this network.

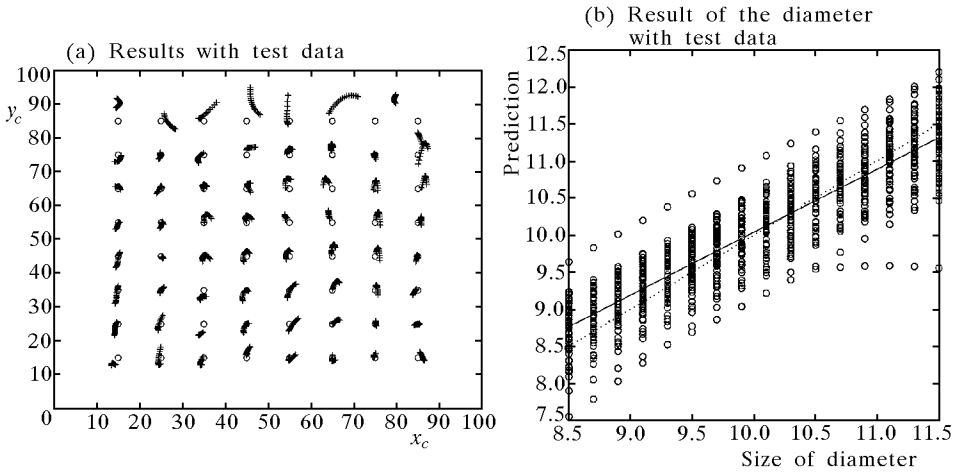

Fig. 4. Example 2. Performance of the neural network in predicting location and size for the case of the test set

Then, the computed location coordinates (outputs of the first network) along with the measurements are used as the inputs to another network (second stage) that has one output providing the diameter of the defect. Using this two-stage approach we were able to obtain sufficiently accurate results. The number of hidden units was $H = 10$ and $H = 5$ for the first and second network, respectively. The prediction accuracy of the defect location (accuracy of the first network) for the test cases is shown in Fig. 4a. In this figure, the multiple predictions '+' shown for each defect location correspond to cases (experiments) with the same defect location but different diameter. In what

concerns the accuracy of the second network, the average test set error for the estimation of the diameter was found equal to 0.25 with standard deviation 0.15, thus indicating the estimation performance of acceptable quality (see Fig. 4b).

Further investigation showed that the crack and hole identification problems using elastostatic measurements had similar performance. The detailed documentation of this investigation does not provide additional information in this paper (see Engelhardt [4]).

5.1.3. Example 3: Hole and classical crack identification in elastodynamics

The transient dynamic problem is modelled with the previously outlined theory. The calculated waveforms have been preprocessed by the authors so that only essential measurement points and time instances were used for the neural network. Only classical cracks without contact were considered in this example. The remaining of the work was similar to the static case. Predictions of the neural network for the set of experiments used in the unknown (testing) data are shown in Fig. 5.

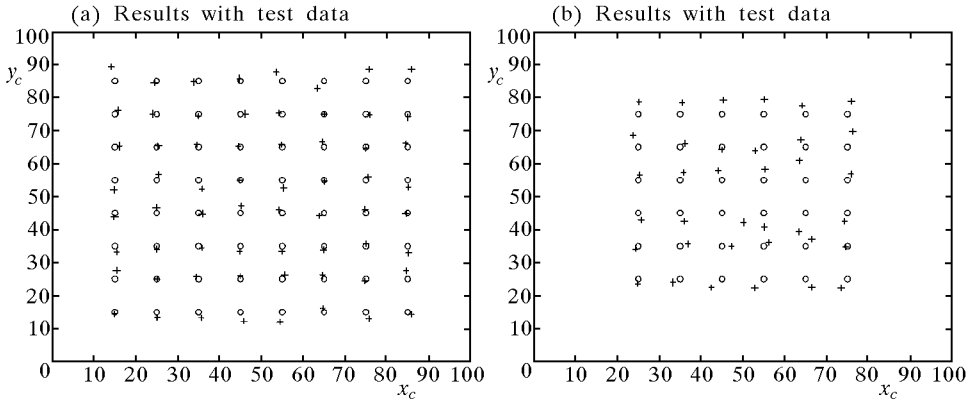

Fig. 5. Example 3. Performance of the neural network in predicting hole and classical (bilateral) location for the test set

5.1.4. Example 4: Unilateral crack identification in elastodynamics

An identification of cracks with contact using transient dynamic measurements was done in the last example. The identification task was more complicated, since one tries to use complicate waves, which include reflection and transmission from nonlinear interfaces. For the used limited time interval the

method worked quite satisfactory, as it is shown in Fig. 6. A generalization to more complicated problems should be done with care, since nonlinear dynamical phenomena may have chaotic characteristics. Further research in this direction is certainly needed.

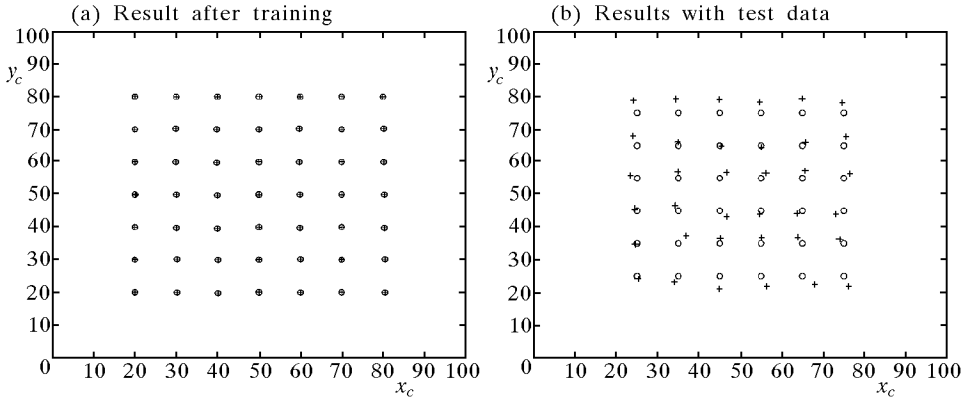

Fig. 6. Example 4. Unilateral crack identification. Performance of the neural network in predicting location for the cases of the training set

Acknowledgments

The work has been partially supported by the German Research Foundation (DFG) and by the Greek-German Scientific Cooperation Programm (IKYDA2001).

References

1. ALESSANDRI C., MALLARDO V., 1999, Crack identification in two-dimensional unilateral contact mechanics with the boundary element method, *Computational Mechanics*, **24**, 100-109

2. ANTES H., PANAGIOTOPOULOS P.D., 1992, *The Boundary Integral Approach to Static and Dynamic Contact Problems. Equality and Inequality Methods*, Birkhäuser, Basel-Boston-Berlin

3. DOMÍNGUEZ J., 1993, *Boundary Elements in Dynamics*, Computational Mechanics Publications, Southampton and Elsevier Applied Science, London

4. ENGELHARDT M., 2004, PHD Thesis, Technical University of Braunschweig, Germany (in preparation)

5. FEDELINSKI P., ALIABADI M.H., ROOKE D.P., 1994, Dynamic stress intensity factors in mixed mode, in *Boundary Elements XVI*, Brebbia C.A. edit., Comp. Mech. Publications, Southampton, 513-520

6. GALLEGO R., DOMÍNGUEZ J., 1995, HBEM applied to Transient Dynamic Fracture, *Proc. ICES'95*, Hawaii, USA, Edit. S.N. Atluri

7. GALLEGO R., DOMÍNGUEZ J., 1996, Hypersingular BEM for transient elastodynamics, *International Journal for Numerical Methods in Engineering*, **39**, 10, 1681-1705

8. GALLEGO R., DOMÍNGUEZ J., 1997, Solving transient dynamic crack problems by the hypersingular boundary element method, *Fatigue and Fracture of Engineering Materials and Structures*, **20**, 5, 799-812

9. GUIGGIANI M., 1992, Direct evaluation of hypersingular integrals in 2D BEM, *Notes on Numerical Fluid Mechanics*, **33**, W. Hackbusch, edit., Vieweg, Braunschweig, 23-34

10. GUZ A.N., ZOZULYA V.V., 2002, Elastodynamic unilateral contact problems with friction for bodies with cracks, *International Applied Mechanics*, **38**, 8, 895-932

11. LIANG Y.C., HWU C., 2001, On-line identification of holes/cracks in composite structures, *Smart Materials and Structures*, **10**, 4, 599-609

12. LIKAS A., KARRAS D., LAGARIS I.E., 1998, Neural network training and simulation using a multidimensional optimization system, *Int. J. of Computer Mathematics*, **67**, 33-46

13. MANSUR W.J., 1983, *A time-stepping technique to solve wave propagation problems using the Boundary Element Method*, Ph.D. Thesis, Universiy of Southampton, U.K.

14. OISHI A., YAMADA K., YOSHIMURA A., YAGAWA G., 1995, Quantitative nondestructive evaluation with ultrasonic method using neural networks and computational mechanics, *Computational Mechanics*, **15**, 521-533

15. PORTELA A., ALIABADI M.H., ROOKE D.P., 1992, The dual Boundary Element Method: effective implementation for crack problems, *Int. J. Numer. Meth. Engineering*, **33**, 6, 1269-1287

16. RUS G., CARLBORG, 2001, *Numerical methods for nondestructive identification of defects*, Doctoral Thesis, Departamento de Mecanica de Estructuras, Universidad de Granada, Spain

17. RUS G., GALLEGO R., 2002, Optimization algorithms for identification of inverse problems with the boundary element method, *Engineering Analysis with Boundary Elements*, **26**, 4, 315-327

18. SÁEZ A., GALLEGO R., DOMÍNGUEZ J., 1995, Hypersingular quarter point boundary elements for crack problems, *Int. J. Numer. Methods Engineering*, **38**, 1681-1701

19. STAVROULAKIS G.E., 1999, Impact-echo from a unilateral interlayer crack. LCP-BEM modelling and neural identification, *Engineering Fracture Mechanics*, **62**, 2-3, 165-184

20. STAVROULAKIS G.E., 2000, *Inverse and Crack Identification Problems in Engineering Mechanics*, Kluwer Academic Publishers, Dordrecht, and Habilitation Thesis, Technical University of Braunschweig, Germany

21. STAVROULAKIS G.E., ANTES H., 1997, Nondestructive elastostatic identification of unilateral cracks through BEM and neural networks, *Computational Mechanics*, **20**, 5, 439-451

22. STAVROULAKIS G.E., ANTES H., 1998, Neural crack identification in steady state elastodynamics, *Computer Methods in Applied Mechanics and Engineering*, **165**, 1/4, 129-146

23. YAGAWA G., OKUDA H., 1996, Neural networks in computational mechanics, *Archives of Computational Methods in Engineering*, **3**, 4, 435-512

24. YUSA N., CHENG W., CHEN Z., MIYA K., 2002, Generalized neural network approach to eddy current inversion, *NCT and E. International*, **35**, 609-614

25. ZENG P., 1998, Neural computing in mechanics, *ASME Applied Mechanics Reviews*, **51**, 2, 173-197

26. ZGONC K., ACHENBACH J.D., 1996, A neural network for crack sizing trained by finite element calculations, *NDT and E. International*, **29**, 3, 147-155

27. ZIEMIANSKI L., PIATKOWSKI G., 2000, Use of neural networks for damage detection in structural elements using wave propagation, In: *Computational Engineering using Metaphors from Nature*, Edit. B.H.V. Topping, Civil-Comp Press, Edinburgh, U.K.

Identyfikacja pęknięć i wad strukturalnych w stanach nieustalonych za pomocą sieci neuronowych

Streszczenie

W pracy zajęto się problemem identyfikacji pęknięć i innych uszkodzeń strukturalnych w dwuwymiarowym stanie odkształceń sprężystych za pomocą metod numerycznych. Modelowanie mechaniczne oparto na metodzie elementów brzegowych ze szczególnym uwzględnieniem kwestii osobliwości pęknięć. Możliwość powstawania częściowo lub całkowicie zamkniętych pęknięć wprowadzono poprzez wykorzystanie

liniowej metody komplementarności. Dla rozwiązania zagadnień odwrotnych użyto sieci neuronowych ze wsteczną propagacją. W zagadnieniach dynamicznych efektywność zaproponowanej procedury zwiększono poprzez odpowiednią wstępną obróbkę danych wejściowych. Na przykładzie dwuwymiarowych modeli opisywanych w pracy stwierdzono podobną skuteczność metody, jak w przypadku klasycznego zagadnienia identyfikacji wad strukturalnych. Wykazano, że identyfikacja jednostronnych pęknięć, która jest znacznie trudniejszym zadaniem, jest możliwa za pomocą zaprezentowanej metody, jeśli do analizy modelu przyjąć odpowiednio dobrane obciążenie testowe.

Manuscript received December 2, 2003; accepted for print January 21, 2004