International Journal of Fracture (2006) 137:151–172
DOI 10.1007/s10704-006-6582-7

Health assessment of concrete dams by overall inverse analyses and neural networks

R. FEDELE[1], G. MAIER[1,*] and B. MILLER[2]
[1]*Department of Structural Engineering, Technical University (Politecnico) of Milan, Piazza Leonardo da Vinci 32, 20133, Milan, Italy*
[2]*Department of Structural Mechanics, Rzeszów University of Technology, ul. W.Pola 2, 35–959 Rzeszów, Poland*
[*]*Author for correspondence. (E-mail: giulio.maier@polimi.it)*

Received 3 January 2006; accepted in revised form 6 January 2006

Abstract. In several existing dams alcali–silica reaction (ASR) during several decades of service life, or diffused micro-cracking (due to concrete ageing and/or past extreme loads, such as earthquakes) give rise to deterioration of concrete stiffness and to correlated reduction of its strength. An inverse methodology is presented herein apt to identify damage in concrete dams on the basis of hydrostatic loading, measurements by traditional monitoring instruments, such as pendulums and collimators, and artificial neural networks trained by means of finite-element simulations. The arch-gravity dam referred to in this study is sub-divided into homogeneous zones, to which a constant Young modulus is attributed as unknown parameter which quantifies possible damage. These elastic moduli are estimated on the basis of pseudo-experimental data and identification procedures. After a suitable 'training' process, artificial neural networks (ANNs) are employed for numerical solutions of the inverse problem, and their potentialities and limitations are examined to the present purposes. In particular, they turn out to be robust and practically useful in the presence of information which are scarce quantitatively (few available measurements) and/or qualitatively (large noise-to-signal ratio).

Key words: Artificial neural networks, concrete dams, damage diagnosis, inverse analysis, parameter identification, statical tests.

1. Introduction

Integrity assessment of large dams is an important engineering task, since many existing concrete dams are deteriorated by physical–chemical processes and/or past extreme loadings, and their failure might have catastrophic consequences (see e.g., Darbre and Kobelt, 2000). Most of the large dams all over the world, built several decades ago, are generally equipped with instruments which are supposed to provide, continuously in time, information useful to the structural assessment of the dam conditions, on the basis of its response to hydro-statical load consisting of a significant change of the water level in the reservoir.

These instruments are usually of two kinds: (i) *collimators*, which measure the absolute displacements of some selected points of the dam crest, orthogonal to the straight line between two fixed positions in its neighborhood; (ii) *pendulums*, positioned inside the dam in *ad hoc* vertical tunnels, the number and location of which have been selected in the design process. In the majority of cases, measurements are

available also of air, water and structure temperatures in order to account for thermal effects (not dealt with in this study). These effects however have negligible consequences on diagnostic analysis if the relevant hydro-static loading occurs in a few days, as it is assumed in this study.

Innovative monitoring techniques by radar, not to be considered herein, have been recently proposed for dams and are likely to be frequently adopted in the future (see e.g., Schwesinger and Wittmann, 2000; Rhim, 2001; Maier et al., 2004; Ardito et al., 2005).

The number of monitoring instruments in existing dams is, at present, usually quite limited. Purpose of this study is to investigate the informative content provided by the traditional monitoring techniques mentioned above, and their combination with a non-traditional parameter identification methodology in order to estimate both location and intensity of possible damages.

The main simplifying hypothesis adopted herein concerns damage modelling. Damage is assumed to be equivalent to a stiffness deterioration, i.e. to a reduction of the Young modulus in one or more regions inside the dam. The dam concrete is supposed to behave as an isotropically elastic material, whereas Poisson ratio is regarded as known *a priori* and unaltered by the deterioration process. Reduction of elastic modulus and tensile strength may be due to past extreme loadings such as earthquakes; and/or it can occur slowly in time in the presence of alcali–silica reaction (ASR), also denominated alcali-aggregate reaction (AAR). Along decades of service life, such physico–chemical process determines a correlated deterioration of both strength and stiffness in dam concrete (see e.g., Swamy and Al-Asali, 1988; Ahmed et al., 2003), particularly in the presence of diffused cracking at the micro-scale, due to concrete ageing.

In order to identify damage locations and intensities within the volume of the dam considered, in this paper a fairly recent technique, artificial neural networks (ANNs), is employed and critically assessed as for potentiality and limitations. This methodology (see e.g., Haykin, 1999; Waszczyszyn and Ziemiański, 2001; Mróz and Stavroulakis, 2005) differs from traditional identification techniques so far employed in dam engineering both in the inverse problem formulation and in the solution algorithm.

All global health monitoring methods (both statical and dynamical) of large dams exhibit limitations as for identifiability of damages especially in zones close to foundation and abutments. These limitations can be circumvented with recourse to in situ tests by dilatometric measurements and/or flat-jacks. Also these local diagnostic techniques can at present be combined with inverse analysis and ANN (see Fedele et al., 2005).

The paper is organized as follows. Section 2 describes the reference dam and the monitoring instruments currently available. The adopted finite element model of the dam is presented in Section 3. The inverse problem is formulated in Section 4 and diverse approaches to its solution are briefly discussed. Section 5 corroborates by numerical exercises the adopted parameter identification procedure centered on the use of artificial neural networks. Pros and cons of the proposed method are pointed out in Section 6.

Notation. Matrix symbology and formalism are adopted. Boldface letters denote vectors and matrices. $(\cdot)^T$ means transpose. Inequalities hold componentwise. Words in the jargon of the ANN literature are written in *italics*.

2. The selected representative case

The dam considered herein (see Figure 1) had been chosen by the engineering community as a case study for a benchmark (ICOLD, 2001), since it was subjected in the last three decades to significant ASR deterioration. It is a double curvature arch-gravity dam, with maximum height 80 m and crest length 515 m (at 1919 m a.s.l.). Other geometrical properties are as follows: dam volume 372×10^3 m^3; reservoir total capacity about 23.50×10^6 m^3; total catchment area 31.9 km^2. The construction was completed in 1956 and the power production started in 1958.

The dam is equipped with the following instruments for its monitoring: (i) pendulums (see Figure 2a) with plumb-lines on four vertical sections (Figure 3), provided in the original design in order to measure in-plane relative displacements between two points of the dam placed on the same vertical line, or between a point belonging to the dam and another located in the foundation; (ii) three optical collimators

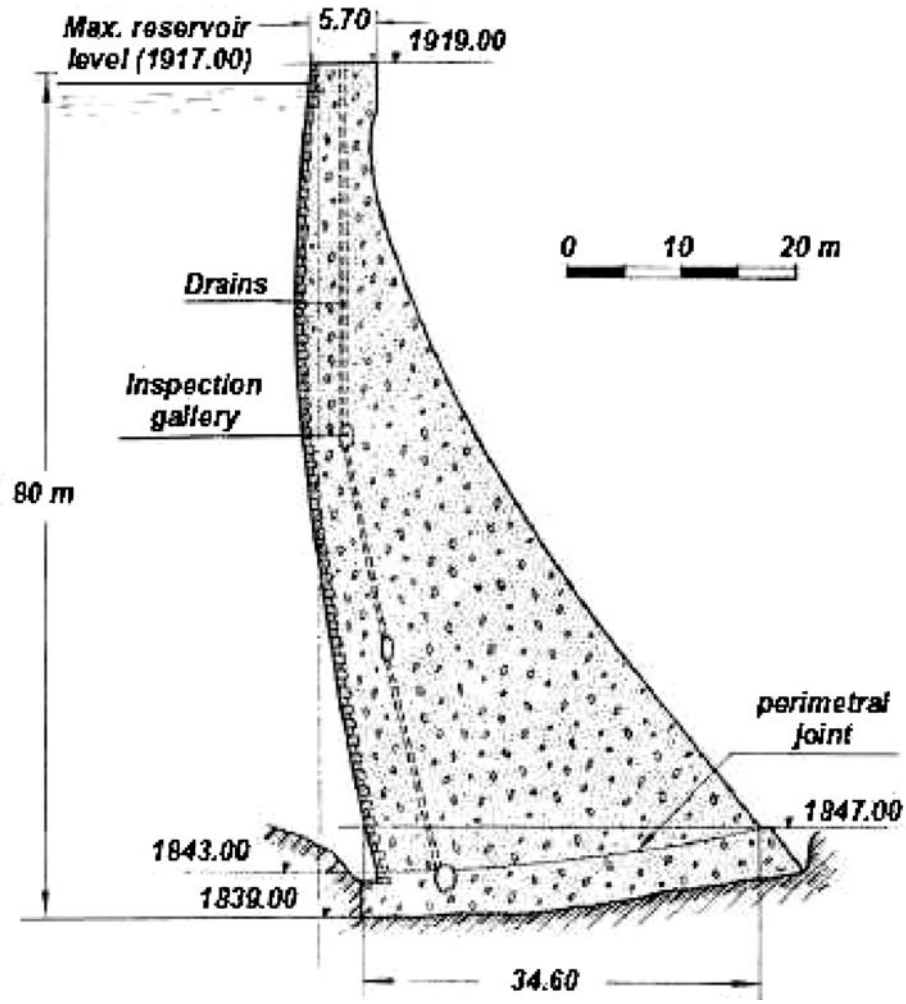

Figure 1. Central vertical section of the considered dam (ICOLD, 2001).

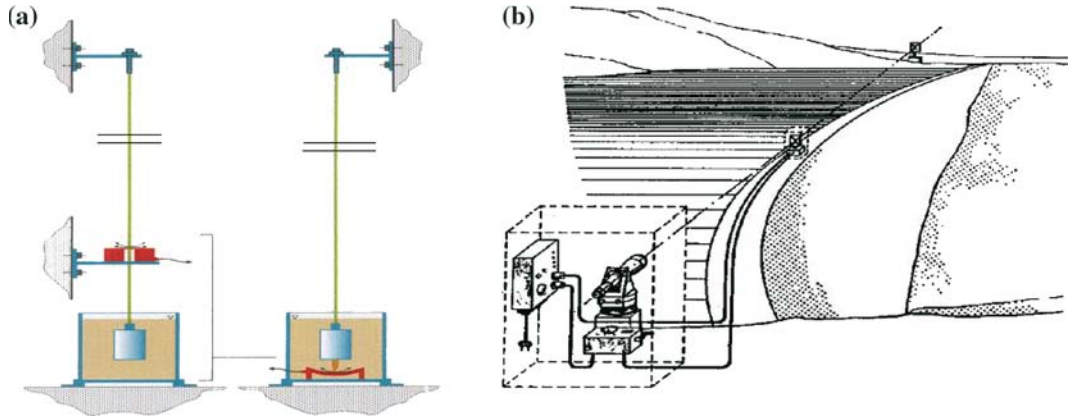

Figure 2. Schematic representation of a pendulum (*a*) and a collimator (*b*).

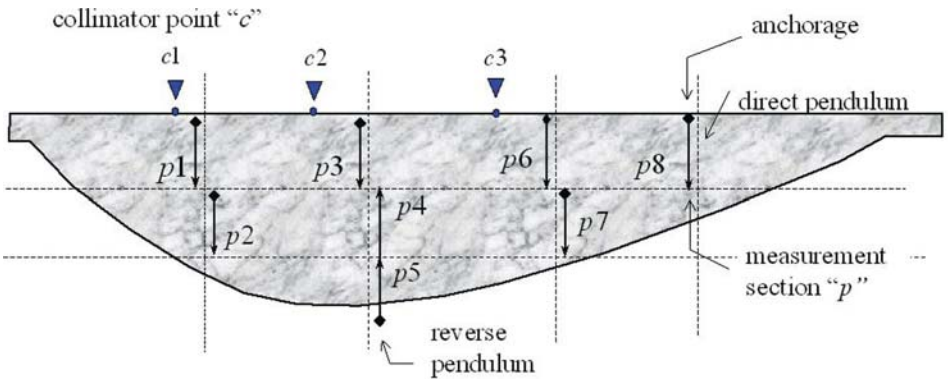

Figure 3. Location of the monitoring instruments in the considered dam.

(see Figure 2b), activated rather recently, measuring absolute displacements of points along the dam crest, as indicated in Figure 3.

A 'direct' pendulum (Figure 2a) is composed of a steel wire anchored in the upper part of the structure, ballasted at the bottom by a proper weight that can move inside a tank filled with a damping fluid. The relative horizontal displacements are measured between the anchoring point and the detecting point of the wire, e.g., along upstream–downstream and right–left directions. Reverse pendulum works according to the same concept, but the wire anchorage is on the bottom part, generally inside a bore-hole in a non-accessible location, e.g., in the dam foundation.

As shown in Figure 2b, each collimator (ii) measures the distance between a line created by two fixed reference points on the rocky slopes near the dam abutments, and another point on the crest of the dam. For each monitoring point on the dam crest, the horizontal displacement component can be measured perpendicularly to the dam profile.

During many years of service life, the above mentioned instruments have been recording relative displacements and their variations in time. These data are periodically examined by experts, who may suggest possible visual inspections. Generally, mathematical modelling and inverse diagnostic analyses are not routinely employed in order to exploit the informative content of the available data. Statistical techniques,

primarily based on comparisons with accumulated past experiences, are frequently used in order to formulate integrity assessments and predictions of the future structural behavior, see e.g., Darbre and Kobelt (2000).

3. Computer simulations

The mathematical model of the dam in point is based on a finite element (FE) discretization (Figure 4), consisting of 468 brick elements (each one with 20 nodes and quadratic interpolations) and 48 wedge elements (with 15 nodes and quadratic interpolations). Self weight is considered by assuming a concrete density $\rho = 24.5\,\mathrm{kN\,m^{-3}}$.

The lowest part of this dam, called *pulvino*, represents the connection between the structure and the rock foundation. *Foundation* means here the volume of the geological formation which is supposed to be stress-perturbed by the presence of the dam and by its hydrostatic load. The pulvino and the foundation have been modelled by means of 112 and 1,320 brick elements, respectively. The total number of degrees-of-freedom in the FE model is 10,836.

In order to reduce computing time in repeated direct analyses, the foundation is statically condensed into a sub-structure, the stiffness of which provides elastic constraints to the nodes belonging to the pulvino-foundation interface. This provision is based on the assumptions that, under the loading to be considered, the foundation will behave elastically (Young modulus $E = 28\,\mathrm{GPa}$, Poisson ratio $\nu = 0.2$) and that its elastic properties are known (i.e. they are not affected by the degradation to be investigated).

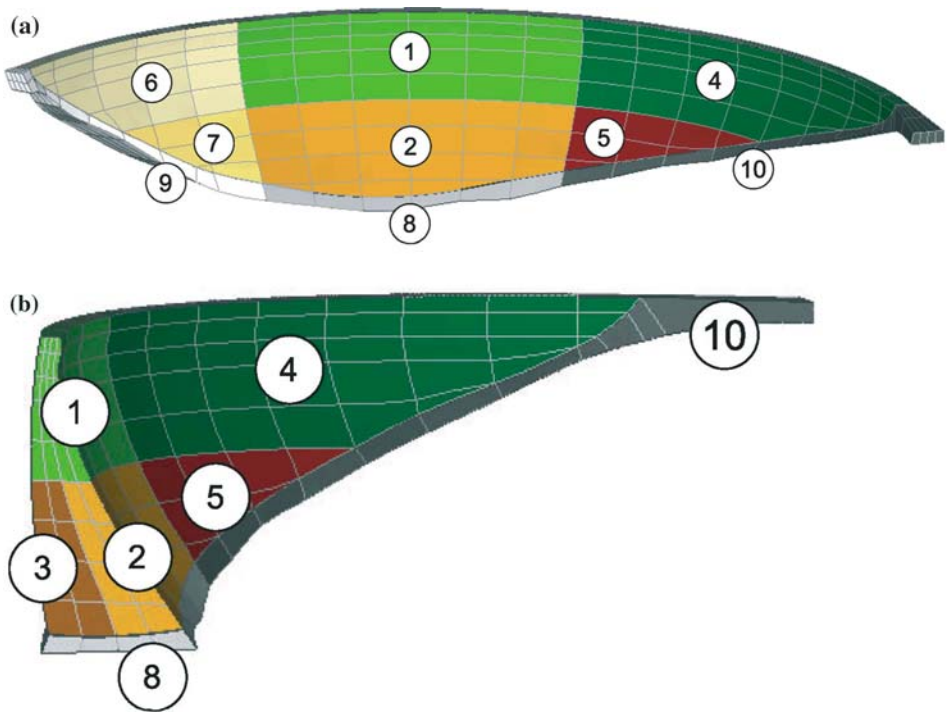

Figure 4. The FE model of the considered dam, sub-divided into ten homogeneous zones.

The behavior of the bulk concrete is assumed to be isotropically elastic. The dam is sub-divided into individually homogeneous zones (in the present case ten, see Figure 4), to each one of which an unknown Young modulus is attributed as representative of the damage to be estimated. Number and location of these zones should be chosen on the basis of mechanical considerations, taking into account the actual concrete mixture and the sequence of casting blocks. Such information are not available for the dam referred to herein: the present choice of the ten zones shown in Figure 4 pursues merely illustrative and methodological purposes.

The homogenous zones adopted herein are as follows: (1) Dam Center High, (2) Dam Center Low Down, (3) Dam Center Low Up, (4) Dam Left High, (5) Dam Left Low, (6) Dam Right High, (7) Dam Right Low, (8) Pulvino Center, (9) Pulvino Left, (10) Pulvino Right. The location of the measurement tools (collimators and pendulums) was specified in Figure 3.

Let vector $\boldsymbol{x} = \{E_1, \ldots, E_{10}\}^T$ gather the $n_x = 10$ Young moduli of the ten zones, and let $\mathbf{y} = \{\Delta u_1, \ldots, \Delta u_{11}\}^T$, with $\Delta u_i = u_i^{\max} - u_i^{\min} (i = 1, \ldots, n_y = 11)$, be the vector of measurable quantities, computed when the reservoir water level decreases from the maximum (1917 m a.s.l.) to the minimum (1878 m a.s.l.). The assumed change of reservoir water level lasts about one week (with seasonal transitions thermal effects, not considered here, would be essential).

The measurable quantities adopted herein are as follows: (i) three data provided by the collimators, i.e. the in-plane displacements of three points on the dam crest, projected along the corresponding normal directions with respect to the dam profile; (ii) eight data provided by the pendulums, specifically the Cartesian norms of the in-plane relative displacements between two points of the dam, or between a point belonging to the dam and another on the foundation. The norm is employed, instead of the two-component vector itself measured by each pendulum, in order to reduce the dimensionality of the vector $\mathbf{y}^{\text{comp}}$ to be input into the artificial neural networks. However, under the assigned loadings, the upstream-downstream component of the displacement vector clearly is bound to prevail on the component orthogonal to it.

The dependence of the $n_y = 11$ measurable computed quantities (displacements) in vector $\mathbf{y}^{\text{comp}}$ on the $n_x = 10$ zone-wise homogeneous Young moduli, gathered in vector $\boldsymbol{x}$, will be referred to as *forward operator* in what follows and denoted by the symbol:

$$\mathbf{y}^{\text{comp}} = \mathcal{H}(\boldsymbol{x}) \tag{1}$$

The adopted symbology does not explicit the obvious linear dependence of displacements on the external actions which is here, for the dam in Figure 1, the lowering of the reservoir level from 1917 above the sea level to 1878 m.

4. Inverse problem formulation and solution strategies

As anticipated in Section 3, the parameters sought are the possibly deteriorated elastic properties, such as one Young modulus E in each zone under the isotropy hypothesis (or two moduli E_h and E_v in the horizontal and vertical directions, respectively, for dam concrete interpreted as orthotropic transversally-isotropic, a situation not considered herein). Poisson ratios, which generally play a minor role in overall analyses of dams, are assumed as unaltered by the deterioration process and known a priori.

It is assumed in this Section that the pseudo-experimental information gathered in vector $\mathbf{y}^{\exp}$ are the $n_y = 11$ measurable displacements, and the parameters to identify are the Young moduli E_k, $k = 1, \ldots, n_x$, gathered in vector $\boldsymbol{x}$, in the $n_x = 10$ zones visualized in Figure 4.

4.1. Least-square approach

The parameter identification problem to solve is formulated first by the simplest, deterministic batch approach (least-squares), as follows:

$$\boldsymbol{x}^\star = \{E_1^\star, \ldots, E_{10}^\star\}^T = \arg \min_{E_1, \ldots, E_{10}} \{\omega_y = \| \mathbf{y}^{\exp} - \mathcal{H}(E_1, \ldots, E_{10}) \|^2\} \tag{2a}$$

$$\text{subject to:} \quad E_k^{'} \leqslant E_k \leqslant E_k^{''} \quad (k = 1, \ldots, 10) \tag{2b}$$

where: $\| \cdot \|$ denotes a properly chosen norm of the difference between experimental and computed quantities (herein the Euclidean norm); stars mark the estimates resulting from the constrained minimization of the discrepancy function ω_y, with respect to the sought parameters $\boldsymbol{x}$; $E_k^{'}$ and $E_k^{''}(k = 1, \ldots, 10)$ specify bounded intervals of practical interest for the Young moduli, with positive lower-bounds ($E_k^{'} > 0$). The 'box-constraints', Equation (2b), ensure that the stiffness matrix of the material (and, hence, of the assembled FE model) is positive definite. Bounds $E_k^{'}$ and $E_k^{''}$ can be conceived as conjectured by an 'expert' in the dam engineering practice.

In a statistical framework, the discrepancy norm in Equation (2a) is often assumed in a non-dimensional form as follows: $\omega_y = \mathbf{r}^T \mathbf{W}^{-1} \mathbf{r}$, where $\mathbf{r}$ denotes the vector of 'residuals', and $\mathbf{W}$ the covariance matrix of the experimental data (diagonal if they are uncorrelated).

The objective function in Equation (2a), say $\omega_y(\boldsymbol{x})$, which quantifies the discrepancy between experimental data $\mathbf{y}^{\exp}$ and computed measurable quantities $\mathbf{y}^{\text{comp}}(\boldsymbol{x})$, is regarded as a function of the unknown parameters in the material model to be calibrated. The generally non-convex minimization in Equation (2a) can be carried out by employing algorithms of diverse kinds: primarily, those which imply repeated computations of the gradient ('first-order' procedures, like Levenberg–Marquardt or Trust Region Method) or others based on function evaluations only ('zero-order' procedures, like direct search or genetic algorithms), see e.g., Sherali and Shetty (1983).

By using the above approach and computational tools, every time an engineering situation is considered and a set of experimental data becomes available, it would be necessary to run the minimization procedure, which requires a suitable computer and finite element code for repeated, usually large-scale test simulations.

In the presence of scarce information, when the number of measurable quantities available is lower than the number of unknown parameters ($n_y < n_x$), the problem is 'under-determined' and the solution vector $\boldsymbol{x}^\star$ Equation (2a) cannot be expected to be unique. In this case, nonlinear least-square problem can be given a different, 'regularized' formulation, such as the following one:

$$\boldsymbol{x}^\star = \arg \min_{\boldsymbol{x}} \{\omega_y = \| \mathbf{y}^{\exp} - \mathcal{H}(\boldsymbol{x}) \|^2 + \varrho \, \| \boldsymbol{x}^\star - \boldsymbol{x}^\sharp \|^2 \mid \mathbf{E}^{'} \leqslant \boldsymbol{x} \leqslant \mathbf{E}^{''}\} \tag{3}$$

where $\boldsymbol{x}^\sharp$ represents the parameter vector indicated by a hypothetical expert as a (possibly subjective and uncertain) Bayesian conjecture, and constant ρ is a penalization

factor chosen by the user. The second term in Equation (3) expresses a minimum norm condition, which enforces the uniqueness of the solution vector $\boldsymbol{x}^{\star}$.

An approach to inverse problems, related to but conceptually different from those mentioned above is allowed by ANNs, as outlined and motivated in what follows (in *italic* the words typical of ANN jargon).

4.2. Artificial neural networks as parameter identification tools

Artificial neural networks (ANNs) can basically be interpreted as a mathematical construct consisting of a sequence of elementary operations apt to approximate a generally complex (possibly non-analytically describable) relationship between two variable vectors, see e.g., Haykin (1999) and Waszczyszyn (1999). Some fundamentals of ('feed-forward') ANN methodology employed herein are summarized below with reference to the present inverse problems in dam engineering.

(a) *Generation of patterns*. Suppose that the direct problem Equation (1) is solved for a certain set of N parameter vectors, say $\boldsymbol{x}^{j}$, $j=1,\ldots,N$, and for fixed boundary conditions (given hydro-static loading); namely, with the previously defined symbols:

$$\mathbf{y}^{\mathrm{comp}\,j}=\mathcal{H}(\boldsymbol{x}^{j})\,,\quad j=1,\ldots,N \tag{4}$$

'Noisy pseudo-experimental' data, say $\mathbf{y}^{\mathrm{exp}}$, can simply be generated by corrupting the model output for given $\boldsymbol{x}^{j}$, Equation (4), through an additive random perturbation $\mathbf{e}^{j}$, namely (N being the above number of simulations):

$$\mathbf{y}^{\mathrm{exp}\,j}=\mathcal{H}(\boldsymbol{x}^{j})+\mathbf{e}^{j}=\mathbf{y}^{\mathrm{comp}\,j}+\mathbf{e}^{j}=\mathcal{H}_{e}(\boldsymbol{x}^{j})\quad j=1,\ldots,N \tag{5}$$

The hypotheses underlying Equation (5) are as follows: a negligible modelling error (i.e. the absence of a deterministic bias); additivity of measurement noise; null mean values of the perturbation. Clearly, operator $\mathcal{H}_{e}$ in Equation (5) coincides with the forward operator $\mathcal{H}$ for vanishing error.

The inverse analysis problem for given data $\mathbf{y}^{\mathrm{exp}\,j}$ ($j=1,\ldots,N$) can concisely be represented as:

$$\boldsymbol{x}^{j}=\mathcal{H}_{e}^{-1}(\mathbf{y}^{\mathrm{exp}\,j}) \tag{6}$$

An artificial neural network plays the role of a 'representation' of the perturbed operator $\mathcal{H}_{e}^{-1}$, Equation (6), which leads to the output $\boldsymbol{x}^{j}$ corresponding to an assigned input vector $\mathbf{y}^{\mathrm{exp}\,j}$ (see e.g., Ziemiański and Miller, 2000). In other words, ANNs are intended to reconstruct a continuous locus in the $\boldsymbol{x}$-space on the basis of an assigned set of points $\boldsymbol{x}^{j}$ which correspond through Equation (6) to points $\mathbf{y}^{\mathrm{exp}\,j}$ selected in the space of measurable quantities.

Pairs of vectors $(\mathbf{y}^{\mathrm{exp}\,j},\boldsymbol{x}^{j})$, $j=1,\ldots,N$, related to each other through Equations (5) and (6), are called *patterns* (or *examples*). Generally, the use of 'pseudo-experimental' patterns corrupted by random noise for ANN '*training*' (as seen later) makes the neural network more robust and apt to deal with disturbed ('noisy') input data.

The so-called *universal approximation* theorem, see e.g., Haykin (1999), states that every continuous function of several variables, defined over a compact set, can be approximated by a neural network, properly designed and calibrated. This remarkable

mathematical result (an extension to a class of nonlinear functions of the classical Weierstrass–Stone theorem on polynomial approximation) is not a constructive theorem. The choice of the optimal network is almost exclusively left to the researcher's ability, in order to find a compromise between the conflicting requirements of estimation accuracy and simplicity of the ANN 'architecture'.

An *artificial neuron* is composed of a linear and a nonlinear operator. The former performs an affine transformation on the input vector, i.e. all components of this vector are multiplied by coefficients called *synaptic weights* and added to a constant called *bias*. The scalar variable thus obtained, called *potential*, is transformed by the latter operator, namely it becomes argument of a nonlinear function f (*activation function*, generally, like in what follows, a monotonically-increasing bounded sigmoidal function).

An artificial network (see Figure 5) is structured in *layers* (therefore it is often referred to as a *multi-layer perceptron*): the first layer is called *input layer*, the last one *output layer* and the others are called *hidden*. The input layer consists of *source nodes* (squares in Figure 5), one for each component of the input n_y-vector. The k-th *source node* transmits unaltered the input scalar y_k $(k = 1, \ldots, n_y)$ to all neurons of the first hidden layer. The hidden layers and the output layer consist of artificial neurons which transform, as specified above, the relevant signals received from neurons of the preceding layer.

In Figure 5, neurons of the hidden and output layers are *computational nodes* (circles): namely, each one of them operates on its input vector the above specified affine transformation and nonlinear transformation. The *synapses*, i.e. links outgoing from each *computational node*, transmit unaltered the scalar signal obtained from that node to all the neurons of the subsequent layer connected with it. Neurons of the same layer do not interact with each other, and each one of them is connected to every neuron (*source nodes*) of the previous layer and to all neurons of the next layer. The hidden layers play the role of *features detectors*, i.e. they capture and memorize,

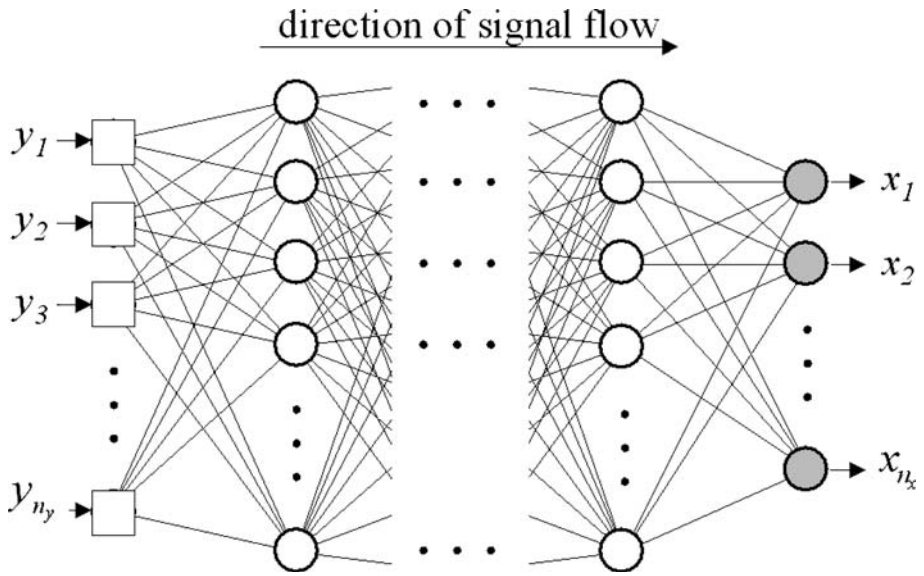

Figure 5. Scheme of an artificial neural network.

through the *training* process (see later), salient features that characterize the model and the relevant operator, like $\mathcal{H}$ in Equation (4), on which the inverse problem in point is centered (see e.g., Haykin, 1999).

The *architecture* of a network is fully specified by the sequence of numbers of components in its layers, see Figure 5: the sequence *I-H- ... H-O* indicates a network with *I source nodes* in the input layer, *O* neurons in the output layer, *H* neurons in each one of the hidden layers. The dimensions of the input and output vectors, here n_y and n_x, coincide with I and O, respectively. For the network *architecture I-H-O* adopted in the numerical exercises of Section 4, the overall number of internal weights and biases (which may exceed one hundred in complex ANNs) reads: $n_y \times H + H \times n_x + H + n_x$.

According to the features of the sigmoidal *activation function* (the values of which are bounded to the interval [0, 1]), the output parameters x_k are scaled to the same interval [0, 1] by referring them to the maximum component of the target vector $\boldsymbol{x}$; hence, the mean-square error ω_x is non-dimensional.

The output response $\boldsymbol{x}$ of an ANN under assigned input $\mathbf{y}$ depends on its *weights* and *biases*, which are calibrated in the *training* phase (also called *learning*) on the basis of available *patterns*, i.e. of vector pairs $(\mathbf{y}^{\exp\, j}, \boldsymbol{x}^j)$, $j = 1, \ldots, N_{\text{train}}$. The set of N patterns generated by means of the mathematical model, i.e. through the forward operator $\mathcal{H}$, is sub-divided into a subset of N_{train} patterns used for the network *training*, and a subset of N_{test} employed for the *testing* ('cross-validation', see later).

Let all the (non-dimensional) internal *weights* and *biases* in the ANN in point be gathered in vector $\mathbf{w}$. For their computation consider the following mathematical programming problem:

$$\mathbf{w}^\star = \arg\min_{\mathbf{w}} \left\{ \omega_x(\mathbf{w}) = \frac{1}{2\, N_{\text{train}}} \sum_{j=1}^{N_{train}} \sum_{k=1}^{n_x} (\widehat{x}_k^j(\mathbf{w}) - x_k^j)^2 \right\} \tag{7}$$

The objective function $\omega_x(\mathbf{w})$ is the output mean-square error (or *network energy*), namely a norm of the discrepancies between the components x_k^j, $k = 1, \ldots, n_x$, of the n_x *output vector* in each previously generated pattern j ($j = 1, \ldots, N_{\text{train}}$), and the corresponding parameters $\widehat{x}_k^j$ estimated by the ANN as functions of the unknown *weights and biases* gathered in vector $\mathbf{w}$; these are now the variables to compute through least-square minimization. In pattern j, the output n_x-vector $\boldsymbol{x}^j$, which is known at the *training* stage, is often called *target vector*.

(b) *Training by back-propagation*. The nonlinear discrepancy minimization problem, Equation (7), is generally non-convex. The global optimal vector $\mathbf{w}^\star$ (not necessary unique) might be computed by means of traditional first-order minimization methods, such as Levenberg–Marquardt algorithm (see e.g., Sherali and Shetty, 1993).

In view of the frequently large number of variables in vector $\mathbf{w}$ (from some tens to more than a hundred for simple ANN), the methodology outlined below, called *back-propagation*, has been developed specifically for the ANN, Equation (7) (see Haykin, 1999), and it was adopted in the present study.

Let w_{lr} denote the l-th *synaptic weight* of the r-th neuron, namely the sought coefficient which multiplies the l-th component, say y_{lr}^j, of the vector of all signals coming to it from the preceding layer and resulting from the preceding transformations of the input vector $\mathbf{y}^j$ ($j = 1, \ldots, N_{\text{train}}$). The bias b of the r-th

neuron is included in vector **w** as a further coefficient to identify, additional to the weights.

The *back-propagation* algorithm consists of the following steps.

(i) Network weights are given a random initialization, say $w_{lr}^{(0)}$, gathered in vector $\mathbf{w}^{(0)}$.

(ii) The rth neuron receives the l-th input signal y_{lr}^{j}, and computes the derivative, say $f_r^{'\,j}$, of the *activation function* with respect to its argument. The current value of this argument is the linear combination, through $w_{lr}^{(0)}$ as coefficients, of the received signals y_{lr}^{j}, index l running over the whole set of all neurons in the preceding layer augmented by one, namely $l = 1, \ldots, H+1$, assuming $y_{H+1\,r}^{j} = 1$.

(iii) The estimate $\widehat{\boldsymbol{x}}^{(0)\,j}$ resulting at the output layer of the network is compared to the (known) *target vector* $\boldsymbol{x}^{j}$ in the j-th pattern: their difference leads to the network mean-square error $\omega_x^{(0)}$, Equation (7).

(iv) Weight $w_{lr}^{(0)}$ is updated according the following formulae:

$$w_{lr}^{(1)} = w_{lr}^{(0)} + \Delta\, w_{lr}^{(0)}\ ; \qquad \Delta\, w_{lr}^{(0)} = -\eta\ \frac{\partial\, \omega_x}{\partial\, w_{lr}}|_{\mathbf{w}^{(0)}} \tag{8}$$

where η is a constant to be chosen by the user, with a default value η=1/2 (*learning-rate*, which governs the convergence rate of the back-propagation algorithm, Equation (8)).

(v) Steps (ii)–(iv) are iterated: at iteration i vector $w_{lr}^{(i-1)}$ plays the same initialization role as $w_{lr}^{(0)}$ at the first iteration.

Convergence is assumed to be attained when change per iteration (i.e. per *epoch*) of the mean-square error ω_x, Equation (7), becomes lesser than a pre-assigned tolerance; alternatively, the algorithm is stopped when the maximum iteration number chosen by the user is reached.

The above procedure is called *back-propagation* algorithm because the gradient in Equation (8) involves quantities which have to be computed recursively by passing leftward through the network (see e.g., Fedele et al., 2005).

Since the optimization of the mean-square error ω_x, Equation (7), is generally non-convex and involves a large number of variables, the ANN *training* process is usually repeated several times with different initializations (herein randomly generated), in order to empirically check possible spurious dependence of the solution vector $\mathbf{w}^{*}$ on the initial values. Such dependence may be due to local minima and/or may reflect non-identifiability of some of the sought weights and biases, as a consequence of not-correct network *architecture*, or of non-identifiability of the sought model parameters on the basis of the available data. For some details on recursive computation of the gradient in back-propagation algorithm see e.g., Haykin (1999), and, with reference specifically to local diagnostic tests on dams, see Fedele et al. (2005).

(c) *Testing*. The predictive capacity (*generalization*) of the trained ANN is investigated in the *testing* phase. This phase is centered on evaluations of the mean-square error ω_x in the sense of Equation (7), for patterns $(\mathbf{y}^{\exp\,j}, \boldsymbol{x}^{j})$, $j = 1, \ldots, N_{\text{test}}$, not employed in the *training* phase. The vector **w** of *weights and biases*, employed for the computation of ω_x in the *testing* phase, is the solution $\mathbf{w}^{\star}$ of the training minimization, Equation (7).

(d) *ANN predictive improvement by early-stopping techniques.* As well known from the literature, a very small minimum $\omega_x^\star$ of the mean-square error ω_x Equation (7) attained in the *training* process, does not necessarily imply a satisfactory predictive capacity of *trained* ANN. This circumstance (called *over-training*), which motivates the *testing* phase, can be caused by training patterns not representative of the whole expected range of the parameters to identify, or by inadequate number of *patterns* compared to the number of *weights*.

In order to improve the predictive capacity, the following provisions can be adopted (*early-stopping* technique): the *mean-square error* ω_x Equation (7) over the *testing* set is computed at each *weight* $\mathbf{w}^{(i)}$ generated along the sequence of the *back-propagation* algorithm; if, during this additional process, $\omega_x(\mathbf{w}^{(i)})$ reaches a minimum $\omega_x^{\star\star}$ at a certain iteration $i^{\star\star}$, this vector of weights, say $\mathbf{w}^{\star\star}$, replaces the vector $\mathbf{w}^\star$ resulting from the *training* minimization; the ANN endowed with such *weights* and *biases* $\mathbf{w}^{\star\star}$ is finally tested over an additional set of patterns.

The above remarks and the brief outline of the main notions on ANNs, evidence the following potential advantages of ANN applications, to the present diagnostic analysis, over the traditional discrepancy minimization techniques.

(i) Patterns required for the *training* phase are computed by test simulations through the direct mathematical model (some hundreds in the present numerical exercises). These simulations may require a significant computational effort but are performed once for all: later application of a *trained* ANN demands limited processing capacity, computer storage and CPU time, and, therefore, it might be done routinely for the dam considered, and possibly in situ.

(ii) ANNs might effectively be applied when experimental data are scarce and/or when a set of patterns is available, provided only by tests on similar systems (without mathematical modelling).

5. Comparative numerical tests for different damage scenarios

The inverse methodology proposed herein has been validated by a number of numerical tests, some of which are presented below. The displacements measurable by the pendulums and collimators installed in the reference dam (Section 2) are computer-generated on the basis of a priori chosen deteriorated Young moduli, so that these parameter values can be compared to those estimated by the artificial neural network (ANN).

The 'pseudo-experimental' data have been computed by the commercial finite element code Abaqus (2003): one forward analysis requires about 17 sec of CPU time, on a PC Pentium IV, 2.6 GHz, 1 Gbyte RAM. The adopted neural network software is available in a Matlab environment (see Demuth and Beale, 1998).

In order to simulate an experiment in a deterministic framework, data vectors $\mathbf{y}^{\mathrm{comp}\,j} = \mathcal{H}(\boldsymbol{x}^j)$, $j = 1, \ldots, N$, are generated by the forward operator $\mathcal{H}$, Equation (1), namely by the FE model specified in Section 3; subsequently, they are perturbed by an additive random 'noise', Equation (5). The pseudo-experimental patterns thus obtained, namely the pair of vectors $(\mathbf{y}^{\mathrm{exp}\,j}, \boldsymbol{x}^j)$ $j = 1, \ldots, N$, are employed in part for the network calibration (*training*), and, in part, for the validation phase (*testing*).

The *network architecture* is chosen as an optimal compromise among the conflicting requirements of simplicity and estimate accuracy, according to the following

strategy: number n_y of the *source nodes* in the input layer, and that of the neurons in the output layer, denoted by n_x, are clearly assigned; one *hidden layer* is introduced, to which a small number n_H of neuron is preliminarily attributed by the user; *training* and *testing* processes on the basis of available *patterns*, are repeated for different networks with increasing n_H; the architecture of the *network* which exhibits the minimum final value of the *mean-square error* ω_x over the *testing* set, is assumed for the parameter identification.

If the available patterns are less numerus with respect to the size of vector $\mathbf{w}$, and/or large estimation errors are obtained by *training and testing* a specific network with n_y input and n_x output components, it may be useful to employ n_x separate networks, with the same input vector and with a unique output $\widehat{x}_k$, $k=1,\ldots,n_x$. These networks are trained independently on the same patterns $j=1,\ldots,N_{\text{train}}$, by assuming for each one of them a different component of the output vector, i.e. $(\mathbf{y}^j, x_k^j)$. The *training* phase is performed separately for the different n_x networks, leading to a generally different optimal weight vector $\mathbf{w}^\star$ for each one of them. In view of the reduced size of the *weight* vector in each network, patterns required for the *training* phase are significantly less numerous. This circumstance may also imply a reduction of the overall computing time required, even if error minimizations are repeated n_x times.

5.1. All zones possibly deteriorated

Within the above framework, with eleven relative displacements as input ($n_y=11$), the first, practically most frequent scenario to consider involves all ten zones (see Figure 4) as expected to be deteriorated by unknown damages. Namely, ten elastic stiffnesses are sought as output ($n_x=10$). In other terms, *patterns* to generate by the direct operator concern damage scenarios in which all the stiffnesses of the ten zones can simultaneously be deteriorated, without any restrictive assumption on the damage distribution.

In Subsection 5.3, this situation is also investigated by varying the available information, in particular by reducing the number n_y of measurable quantities employed as input of the inverse analysis, in view of situations in which only a few monitoring instruments are available.

After comparative analyses, ten ANNs with *architecture* 11-12-1 were employed, each one with a different output (the Young modulus E_k of the relevant k-th zone, $k=1,\ldots,n_x\equiv 10$). By conventional symbols:

$$\mathbf{y}^{\exp}=\{\Delta u_1,\ldots,\Delta u_{11}\}^T \Rightarrow \boxed{\mathbf{ANN}_k,\ 11\text{-}12\text{-}1} \Rightarrow x_k=E_k \quad (k=1,\ldots,10)$$

Patterns were generated by FE simulations, preliminarily noiseless, i.e. $(\mathbf{y}^{\text{comp}\,j}, \boldsymbol{x}^j)$ $(j=1,\ldots,N)$, Equation (1), and, thereafter, corrupted by additive noise, namely $(\mathbf{y}^{\exp\,j}, \boldsymbol{x}^j)$, with $\mathbf{y}^{\exp\,j}=\mathcal{H}(\boldsymbol{x}^j)+\mathbf{e}^j$, Equation (5). The components of noise vector $\mathbf{e}^j$ added to the input data were independently sampled from a uniform probability distribution over the interval $\pm 2\%$.

In order to calibrate the 144 *weights* and 13 *biases* of each one of the 10 networks in the *training* phase and to *test* them, $N=1,000$ *patterns* $(\mathbf{y}^{\text{comp}\,j}, \boldsymbol{x}^j)$, $j=1,\ldots,1,000$, have been generated as follows: values E^j randomly chosen in the

interval $10 \div 20$ GPa for each possibly damaged zone ($k = 1, \ldots, n_x$), and the corresponding measurable displacement vector $\mathbf{y}^{\text{comp}\,j}$ computed by the FE model $\mathbf{y}^{\text{comp}\,j} = \mathcal{H}(\boldsymbol{x}^j)$. Of the N patterns thus generated, 600 patterns, randomly chosen, were employed for the *training*, and the remaining 400 for the *testing* phase. The *training* process required about 1,000 iterations (*epochs*) of the *back-propagation* algorithm, to attain an increment per epoch of the *mean-square error*, say $\Delta\,\omega_x$, below the assigned tolerance ($<10^{-4}$), leading to a maximum final value $\omega_{E_9} = 1.11 \cdot 10^{-3}$.

The results concerning noiseless data are synthesized in Figure 6(a), in a way which is typical of the ANN literature. For each level of relative error (in percentage), the *ratio of success* of the network is indicated, i.e. the percentage of *patterns* in which the ANN output error does not exceed the threshold indicated in the abscissae axis. Clearly, the sooner the 100% level is obtained the better (in the ideal situation of 100% of patterns predicted by the ANN without any error, there would be a horizontal line on the level of 100%). These plots show that, by means of the ten chosen ANNs, the identification of elastic moduli on the basis of noiseless data gives very accurate results, except for the zones located in the pulvino, where the maximum error is about 10%, while for zones 1, 4, 5, 6 and 7 the maximum error does not exceed 3%.

The same networks have been calibrated and tested on the basis of pseudo-experimental data corrupted by $\pm 2\%$ random noise with uniform probability density distribution, i.e. with *patterns* $(\mathbf{y}^{\text{exp}\,j}, \boldsymbol{x}^j)$, $j = 1, \ldots, N \equiv 1,000$. The number of patterns employed for *training* and *testing* phases is the same as for noiseless data. The maximum value of non-dimensional mean-square error ω_x (over the 10 ANNs), Equation (7), at the minimum point $\mathbf{w}^\star$ is, obviously, higher with respect to that computed in the presence of noiseless data, namely: $\omega_{E_{10}} = 2.58 \times 10^{-2}$.

Results, shown in Figure 6(b), lead to the following remarks. The presence of 2% noise, as expected, makes the parameter estimation more difficult: Young moduli E_k concerning zones with $k = 1$, 4, 5, 6 and 7 are identified with a maximum error of about 15%. In the remaining zones, especially in those located in the pulvino, stiffness deteriorations are practically not identifiable.

5.2. Damage confined to one (unknown) of the ten zones

The dependence of ANN-estimation accuracy on the data complexity in the available *patterns*, is studied in what follows by comparing the above results to those obtainable by the same network, when it is *trained* on *patterns* relevant to a unique damage scenario, included as a particular case within those considered in the previous Subsection.

In what follows damage distributions are generated by imposing restrictive conditions to damage intensities and locations, within the adopted sub-division of the reference dam into ten zones. In other words, parameter sets $\boldsymbol{x}^j$, $j = 1, \ldots, N$, belong to a restricted subset of the original problem domain in the parameter space (bounded by box-constraints, Equation (2b)) obeying to particular geometrical conditions (which exhibits also a clear mechanical meaning).

In the situation considered now, parameters $\boldsymbol{x}^j$, $j = 1, \ldots, N$, adopted for generating *patterns* concern the scenario in which only one zone of the dam out of ten is damaged, whereas the Young modulus is unaltered in the other nine zones. If, e.g., damage is located in the zone number 4 and quantified by $E_4 = 12$ GPa, the n_x-vector of output parameters reads:

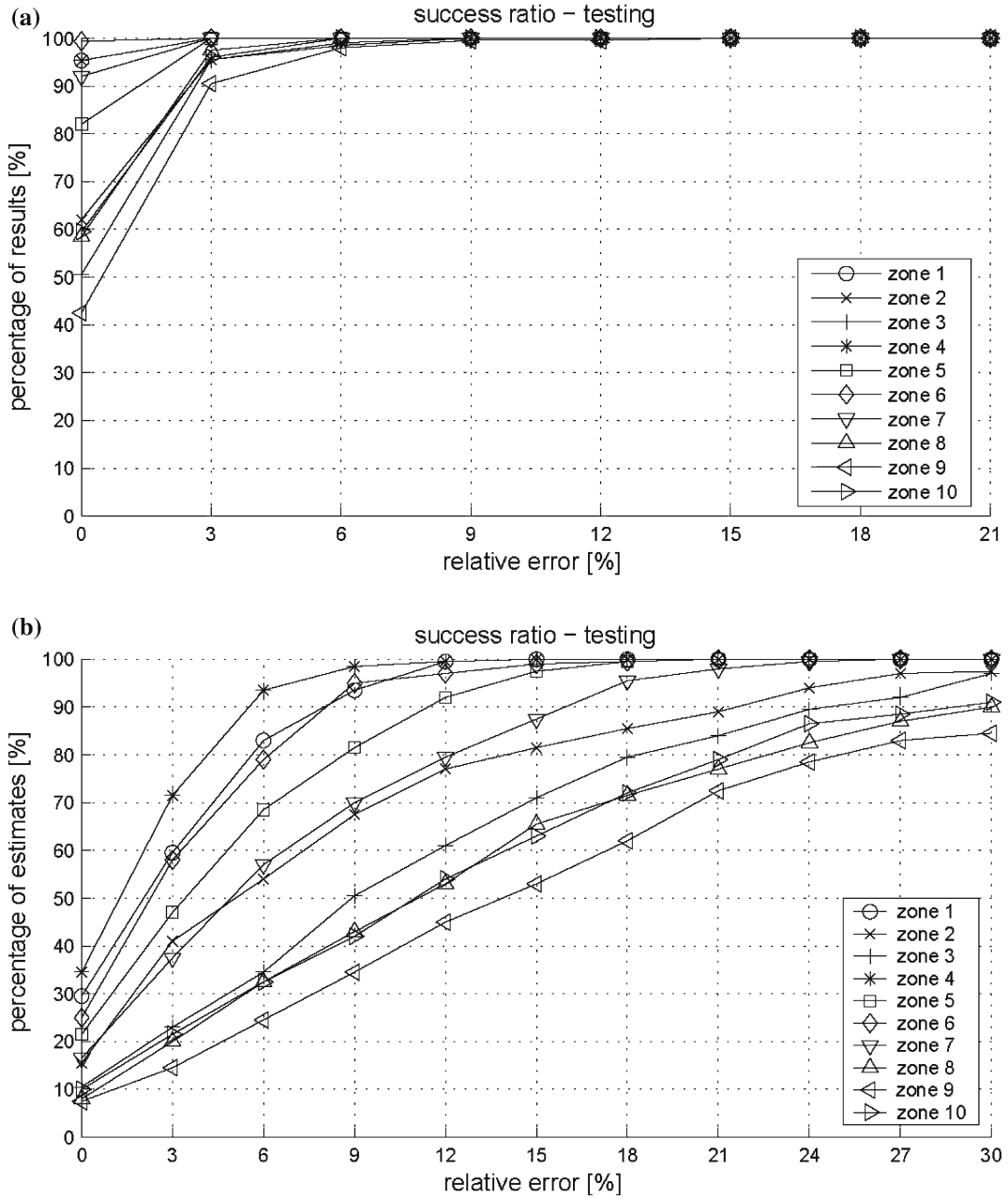

Figure 6. Ten-damaged-zone scenario: parameter identification by 11-12-1 ANNs on the basis of eleven measurements. Percentage of *testing* patterns in which the error of ANN estimates does not exceed the threshold indicated in the abscissae axis: (a) with noiseless pseudo-experimental data; (b) with data corrupted by 2% noise.

$$\boldsymbol{x} = \{E_k\}_{k=1,10} = \{20, 20, 20, 12, 20, 20, 20, 20, 20, 20\}^T \tag{9}$$

The other $N = 300$ *patterns* are generated by varying the location of the damaged zone, and its damage intensity among the reference values chosen by the user, whereas in the other zones the Young modulus remains equal to $E = 20$ GPa in all

the N *patterns*. In this exercise, parameter sets $\boldsymbol{x}^j$ in available *patterns*, $j = 1, \ldots, N$, belong to a 'star' of straight segments (indicated also as a sheaf), outgoing from the point representative of the undamaged state (defined by all Young moduli $E_k = 20\,\text{GPa}$, $k = 1, \ldots, 10$).

Ten *networks* with 11-12-1 *architecture* have been adopted, i.e. with eleven inputs, twelve hidden layer with six neurons and one output in each one of them. Figure 7 shows the success ratios of the parameter estimates obtained by ANNs, with and without noise: in the presence of 2% additive perturbation on data, errors larger than 10% can be observed for the zones numbered as 9 and 10, and, therefore, also in this simpler situation damage in the pulvino turns out to be hardly identifiable.

As a conclusion, from the above results and others not shown herein for brevity (concerning scenarios with four possibly damage zones), it can be stated that the estimation error of a neural network increases with the complexity of damage scenarios described by the *patterns* employed for the *training* process. Within the methodological study presented herein, the identification of the elastic stiffness in the ten zones without any a priori assumption on its distribution (Subsection 5.1) represents therefore a scenario of damage more difficult to identify, with respect to the last illustrative example outlined above.

5.3. Scarce experimental information

In this Subsection the inverse problem in point will be studied by employing only some selected measurements as input data. This kind of investigation may lead to determine the number and location of monitoring instruments which are optimal for the diagnostic purposes pursued. Therefore the procedure discussed below may also be regarded as supplementary or substitute of sensitivity analyses, adopted e.g., in Ardito et al. (2005); Maier et al. (2004), and not employed here.

When a reduced input vector $\mathbf{y}^{\text{exp}}$ is considered for the present parameter identification via ANNs, the number of available information for each pattern j may be smaller than the number of unknowns, namely $n_y < n_x \equiv 10$: the problem is 'under-determined' and the solution vector cannot be expected to be unique. In classical inverse analyses based on the minimization of a discrepancy function, under-determined problems, both the linear problems and the nonlinear ones, are generally solved by introducing additional regularizing conditions on the solution vector (in the spirit of Tykhonov), e.g., the condition of minimum norm as in Equation (3) (see Bui, 1994; Tarantola, 1987).

When available information is scarce, the inverse problem may admit multiple solutions, i.e. patterns exist relating different parameter sets $\boldsymbol{x}$ to the same data vector $\boldsymbol{y}$. In such situations, neural networks performs a sort of 'geometric' regularization, allowed by memory of patterns analyzed previously in the *training* phase. In fact, the *trained network* selects the solution vector $\boldsymbol{x}^\star$ which minimizes the mean-square error ω_x over the *training patterns*: in the n_x-vector parameter space, point $\boldsymbol{x}^\star$ represents as a centroid with respect to the distribution of parameter vectors $\boldsymbol{x}^j$ contained in available *patterns*, $j = 1, \ldots, N_{\text{train}}$, and corresponding through the inverse operator to the same data vector $\boldsymbol{y}$.

The main conceptual drawback in this approach is that the numerical regularization implicitly performed by *trained* ANNs is not supported by any clear engineering

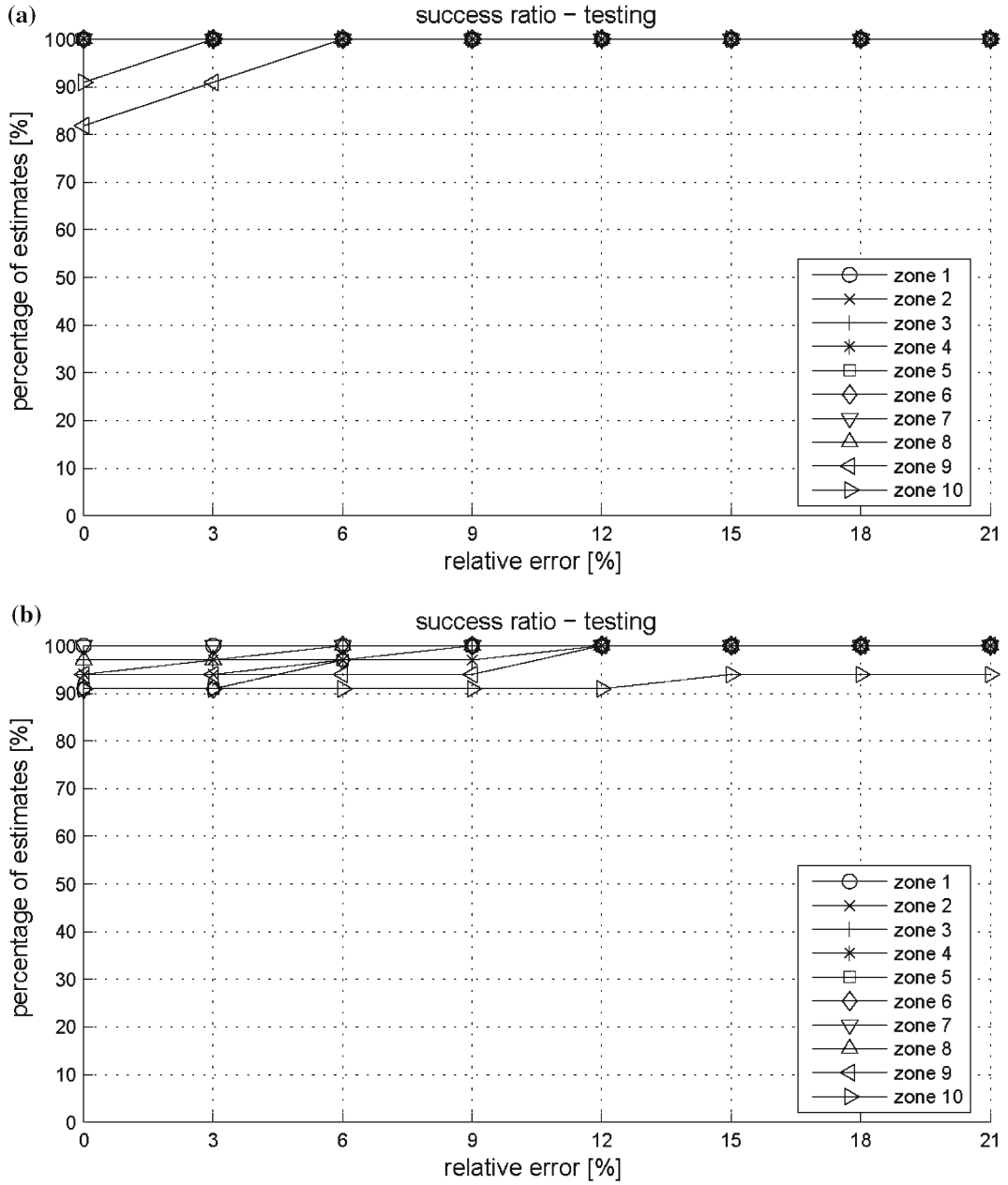

Figure 7. One-damaged-zone scenario: parameter identification by 11-12-1 ANNs on the basis of eleven measurements. Percentage of *testing* patterns in which the error of ANN estimates does not exceed the threshold indicated in the abscissae axis: (a) with noiseless pseudo-experimental data; (b) with data corrupted by 2% noise.

motivation. It is a procedure which gives straightforwardly a particular solution of the inverse problem, in this aspect not differently from the minimum norm condition typically adopted in least-square approaches.

To the present identification purposes ANN exhibits practical interest since common softwares currently available are generally not apt to manipulate problems with more unknowns than equations. It is worth noting that damages located

in specific zones of the dam may however be estimated even in the presence of under-determined measurement equations, whereas possible damages in the remaining zones do not 'excitate' the monitoring instruments currently available: from a mathematical standpoint, this circumstance implies that only a restricted number of components in the parameter vector $\boldsymbol{x}$ may turn out to be identifiable. These situations of significant practical interest can be easily investigated by the above procedure centered on ANNs.

The following situations with reduced experimental data (see Figure 3 for the locations of the monitoring instruments) have been investigated:

(1) measurements from the pendulums only are available, namely the overall number of input data is reduced to $n_y = 8$;
(2) only measurements from the collimators (labelled as c1, c2, c3 in Figure 3) are employed as input data, so that the number of input data becomes $n_y = 3$;
(3) measurements from the three collimators, and from those pendulums which are located in the central vertical cross-section of the dam (labelled as p3, p4, p5 in Figure 3); hence $n_y = 6$;
(4) measurements from the collimators and from the pendulums in the two lateral vertical sections (labelled as p1, p2, p6, p7), so that now $n_y = 7$.

Ten distinct one-output ANNs have been employed for all the situations mentioned above, with the hidden layer of 12 neurons, i.e. with the architectures n_y-12-1. By customary symbols:

$$\mathbf{y}^{\exp} = \{\Delta u_1, \ldots, \Delta u_{n_y}\}^T \Rightarrow \boxed{\mathbf{ANN}_k,\ n_y\text{-}12\text{-}1} \Rightarrow x_k = E_k \quad (k = 1, \ldots, 10)$$

It is worth noting that the 10 one-output networks employed for each situation were *trained and tested* by using always the same set of patterns: the only difference was in the components of the data vector $\mathbf{y}^{\exp\, j}$ $(j = 1, \ldots, N)$ selected as input of each specific networks. $N = 1,000$ patterns $(\mathbf{y}^{\mathrm{comp}\, j}, \boldsymbol{x}^j)$ $(j = 1, \ldots, N)$ have been generated by the mathematical model and, subsequently, have been corrupted by 2% additive noise. Of the N available patterns, 800 have been employed for *training* and 200 for *testing*.

In preliminary numerical exercises, *networks* with diverse *architectures* were *trained* on the basis of the above $N_{\mathrm{train}} = 800$ patterns: the prediction errors ω_x over the $N_{\mathrm{test}} = 200$ *testing* patterns turned out to be significant, as a consequence of the solution multiplicity implied by scarce information $\mathbf{y}^{\exp}$. Therefore, an *early stopping* technique (see Subsection 4.2, point (e)) was employed in the *back-propagation* algorithm intended to identify vector $\mathbf{w}$, gathering *weights* and *biases* $(n_y \cdot 12 + 12 \cdot 1 + 13$ neuron coefficients). This *regularization* strategy consists of the following operative steps: (i) the *training* set, constituted of $N_{\mathrm{train}} = 800$ patterns, is split into an *estimation* subset (herein 600 patterns), and a *validation* subset (200 patterns); (ii) the network is calibrated over the *estimation* set, but the *back-propagation* algorithm is stopped after a certain number of *epochs* before it achieves convergence, at a value $\mathbf{w}^{\star\star}$ which corresponds to a minimum of the mean-square error ω_x over the *validation* subset; (iii) thereafter, *testing* phase (over $N_{\mathrm{test}} = 200$ patterns) is performed.

By means of ANN inversions, in the presence of scarce information, a unique solution vector $\boldsymbol{x}^{\star}$ is obtained, which satisfies the *regularizing* condition of minimum *validation error*.

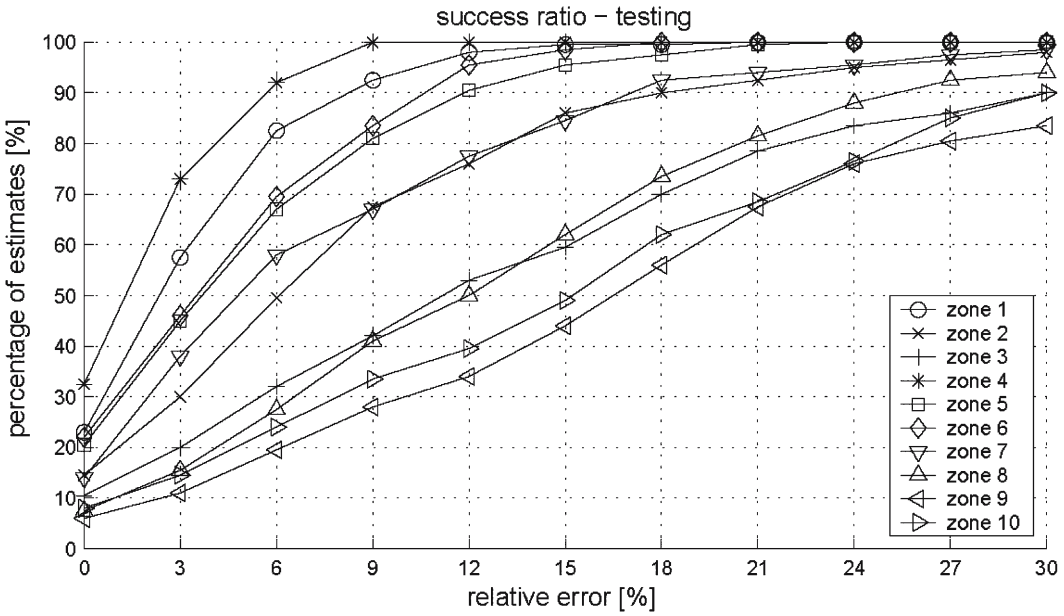

Figure 8. Ten-damaged-zone scenario: parameter identification by ANN on the basis of measurements obtained by the eight pendulums (no collimators) and corrupted by 2% noise. Percentage of testing patterns leading to ANN estimates affected by errors smaller than that in abscissae.

The results obtained by the above outlined numerical exercises (and not exposed here for brevity) lead to the remarks which follow: (α) measurements obtained by the eight pendulums (situation 1) lead to a satisfactory identification results for the zones labelled as 1, 4, 6, located in the upper part of the dam: in fact, the relative output error concerning those zones does not exceed the threshold of 15%, see Figure 8; (β) measurements provided by the three collimators alone (situation 2) are not sufficient to identify any Young modulus in the present context, and, in general, their role is less significant (situation 1); (γ) in situations (3) it was possible to identify accurately damages only in zone 1; in zones 1 and 6 in situation (4).

From the above considerations it can be concluded that the monitoring instruments in the number and locations assumed in this study do not allow to infer with accuracy the presence of possible damages (in the sense specified in Section 1) in the pulvino, and, more generally, in the lower part of the dam, subdivided into zones as specified in Figure 4. Since measurements obtained by collimators seem to have less influence on the identifiability of the elastic moduli in lower zones, it would be desirable to install further pendulums in order to achieve reliable information on the lower parts of the dam. However these parts are accessible with difficulty and, clearly, tunnels for additional pendulums would imply further safety reduction additional to the material deterioration in the dam.

6. Final remarks and future prospects

Purpose of this paper is to investigate the potentialities of a parameter identification methodology not yet popular in civil engineering, namely artificial neural

networks (ANNs), in damage assessment of concrete dams possibly deteriorated by, say, alcali-silica-reactions, as many existing dams unfortunately are.

The arch-gravity concrete dam considered herein as representative, typical reference, is equipped with traditional monitoring instruments, namely pendulums and collimators. Damages are assumed to be characterized quantitatively by a reduction of local stiffness, assumed to be zone-wise constant in the structure volume. The dam was subdivided into ten zones, and the relevant Young moduli have been estimated by ANNs on the basis of the above monitoring systems in various situations.

The numerical exercises outlined and discussed in this paper show that ANNs, if properly *trained* and *tested* by means of a finite element model of the dam-foundation system, are suitable to investigate the identifiability of zone-wise homogeneous elastic moduli in the dam, even when the inverse problem is under-determined and the 'classical' techniques based on the minimization of a discrepancy norm are inadequate.

If the stiffness deterioration is sought in the upper part of the dam, the Young moduli are estimated quite precisely, almost independently from the measurement noise level. The reduction of input data, and/or high values of the noise-to-signal ratio in the available measurements, dramatically decrease the estimation accuracy when damage is located in the pulvino, and, in general, in the lower part of the dam.

Non-destructive overall diagnostic tests obviously are intended to avoid dissipative inelastic deformation in the structural responses to the loading processes. Therefore, only elastic stiffness parameters can be identified on the basis of global excitation of structures. The correlation between decrease of concrete Young modulus (average at the macroscale) and concrete damages, is fairly well understood and quantified nowadays when mechanical deterioration is caused by alcali-silica reaction, much less when it consists of diffused cracking due, e.g., to past extreme loading (and usually implying anisotropy of average behavior).

Remedies for such difficulties and limitations are represented by novel techniques of local material parameter identification by hole drilling or superficial flat-jack tests, dilatometric measurements, simulation and inverse analysis combined, see Fedele et al. (2005) and Maier et al. (2005). In particular, these new local non-destructive diagnostic procedures, if endowed with ANNs, are able to provide (*in situ*), in real time, meaningful inelastic material parameters such as fracture energy and tensile strength, see Fedele et al. (2005).

If the expected complexity of possible damage distribution in dams requires the estimation of a larger number of model parameters ($n_x > 10$), ANNs can exhibit poor performance and the application of them might become questionable: in such situations recourse is advantageous to classical identification procedures based on the minimization of a discrepancy function by 'hard computing' algorithms (e.g., Sequential Quadratic Programming or Trust Region Method, both based on gradient computations).

The methodology proposed herein, centered on ANNs, might be used as a first step of a strategy for damage assessment. On the basis of the measurements obtained from traditional monitoring techniques, the ANN is apt to assess the health state of the dam in real time and possibly in situ, since ANNs can be *trained* once for all by large-size finite element simulations and, after *training*, require relatively modest hardware, software and CPU time. If results of preliminary diagnostic investigations by ANN indicate that there is a potential reduction of structural integrity, additional

measurement tools (generally more expensive) should be used (e.g., additional collimators, or probably in the near future the radar technique, as described in Ardito et al., 2005; Maier et al., 2004), in order to collect experimental data adequate to a detailed overall diagnostic analysis.

Acknowledgements

The results presented in this article have been achieved in the frame of a research project supported by Italian Ministry of the University and Research (MIUR) on the subject: 'Concrete dam-foundation-reservoir systems', 2003–2004. Fruitful interactions with the dam engineering researchers of CESI, Segrate, are gratefully acknowledged.

References

Abaqus. (2003). Release 6.3. Theory and User's manuals. Hibbit, Karlsson and Sorensen Inc., Pawtucket, RI, USA.

Ahmed, T., Burley, E., Rigden, S. and Abu-Tair, A. (2003). The effect of alkali reactivity on the mechanical properties of concrete. *Construction and Building Materials* **17**, 123–144.

Ardito, R., Bartolotta, P., Ceriani, L. and Maier, G. (2005). Diagnostic inverse analysis of concrete dams with statical excitation. *Journal of the Mechanical Behavior of Materials* **15**(6), 381–386.

Bui, H. (1994). *Inverse Problems in the Mechanics ofMaterials: An Introduction*. CRC Press, London.

Darbre, G. and Kobelt, A. (2000). Health monitoring of dams: Objectives and predictive models. In *Proceedings of the 6th International Workshop on Material Properties and Design*. AEDIFICATIO Publishers, Freiburg, pp. 159–180.

Demuth, H. and Beale, M. (1998). *Neural Network Toolbox for use with Matlab*. The Math Works Inc., Natick, USA.

Fedele, R., Maier, G. and Miller, B. (2005). Identification of elastic stiffness and local stresses in concrete dams by in situ tests and artificial neural networks. *Structure and Infrastructure Engineering* **1**(3), 165–180.

Haykin, S. (1999). *Neural Networks. A Comprehensive Foundation*. 2nd Edition, Prentice Hall, Upper Saddle River, NJ, USA.

ICOLD. (2001). Evaluation of AAR (Alkali – Aggregate Reaction) effects on the structural behaviour of an arch dam: Interpretation of the measured behaviour and forecasting of the future trend. In: *6th International Benchmark Workshop on Numerical Analysis of Dams*, Salzburg (Austria).

Maier, G., Ardito, R. and Fedele, R. (2004). Inverse analysis problems in structural engineering of concrete dams. In: Z. Yao, M. Yuah and W. Zhong, (eds.), *Proceedings of the 6th World Conference of Computational Mechanics WCCM VI, Beijing (China)*. Invited Semiplenary Lecture. Tsinghua University Press and Springer-Verlag, Beijing, pp. 97–107.

Maier, G., Lettieri, M. and Piola, L. (2005). In situ mechanical characterisation of dam concrete and stress state by dilatometric measurements and inverse analysis. In: *Abstract Book, 11th International Conference on Fracture ICF XI (Turin)*. p. 395.

Mróz, Z. and Stavroulakis, G. (eds.) (2005). *Parameter Identification of Materials and Structures*. CISM Lecture Notes, vol. 469, Springer Verlag, Wien.

Rhim, H. (2001). Condition monitoring of deteriorating concrete dams using radar. *Cement Concrete Research* 31, 363–373.

Schwesinger, P. and Wittmann, F. (eds.) (2000). *Present and Future of Health Monitoring*. AEDIFICATIO Publishers, Freiburg.

Sherali, M.B.H. and Shetty, C. (1993). *Nonlinear Programming. Theory and Algorithms*. John Wiley & Sons, New York.

Swamy, R. and Al-Asali, M. (1988). Engineering properties of concrete affected by alkali-silica reaction. *ACI Materials Journal* **85**, 367–374.

Tarantola, A. (1987). *Inverse Problems Theory. Methods for Data Fitting and Model Parameter Estimation*. Elsevier Applied Science, Southampton.

Waszczyszyn, Z. and Ziemiański, L. (2001). Neural networks in mechanics of structures and materials – new results and prospects of applications. *Computers & Structures* **79**, 2261–2276.

Waszczyszyn, Z. (ed.) (1999). Neural networks in the analysis and design of structures. *CISM Lecture Notes*, vol. 404, Springer Verlag, Wien.

Ziemiański, L. and Miller, B. (2000). Dynamic model updating using neural networks. *Comput. Ass. Mech. Eng. Sci*. **4**, 68–86.