STRUCTURAL CONTROL AND HEALTH MONITORING
Struct. Control Health Monit. 2012; **19**:3–21
Published online 5 November 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/stc.420

Bayesian neural networks for bridge integrity assessment

S. Arangio[1,*,†] and J. L. Beck[2]

[1]*Department of Civil and Geotechnical Engineering, University of Rome 'La Sapienza' Via Eudossiana 18, 00184–Rome, Italy*
[2]*Division of Engineering and Applied Science, California Institute of Technology, Mail Code 104-44, Pasadena, CA 91125, USA*

SUMMARY

In recent years, neural network models have been widely used in the Civil Engineering field. Interesting enhancements may be obtained by re-examining this model from the Bayesian probability logic viewpoint. Using this approach, it will be shown that the conventional regularized learning approach can be derived as a particular approximation of the Bayesian framework. Network training is only a first level where Bayesian inference can be applied to neural networks. It can also be utilized in another three levels in a hierarchical fashion: for the optimization of the regularization terms, for data-based model selection, and to evaluate the relative importance of different inputs. In this paper, after a historical overview of the probability logic approach and its application in the field of neural network models, the existing literature is revisited and reorganized according to the enunciated four levels. Then, this framework is applied to develop a two-step strategy for the assessment of the integrity of a long-suspension bridge under ambient vibrations. In the first step of the proposed strategy, the occurrence of damage is detected and the damaged portion of the bridge is identified. In the second step, the specific damaged element is recognized and the intensity of damage is evaluated. The Bayesian framework is applied in both steps and the improvements in the results are discussed.

Received 22 May 2009; Revised 28 July 2010; Accepted 7 August 2010

KEY WORDS: Bayesian neural networks; probability logic approach; integrity assessment; long-span bridges; damage identification; model selection; automatic relevance determination

1. INTRODUCTION

Complex structural systems, such as long bridges, tall buildings or offshore platforms, are exposed to various external loads during their service life and they may deteriorate and degrade with time; this may lead to structural failures and possibly loss of human lives. To ensure safety and to minimize financial loss, early detection of damage becomes important so that preventive and remedial works can be carried out as early as possible [1,2]. Consequently, an effective structural integrity monitoring system can play a key role in the maintenance scheduling of the structures [3–5]. A great deal of research in the past 30 years has been aimed at establishing effective local and global methods for integrity monitoring in civil structures. An extensive survey of global methods is presented in Doebling *et al.* [6] and in an update by Sohn *et al.* [7].

Large structures, like long-span bridges for example, are often affected by environmental and traffic loads that cannot easily be controlled or measured. In such cases, it is possible to assess

*Correspondence to: S. Arangio, Department of Structural and Geotechnical Engineering, University of Rome 'La Sapienza', Rome, Italy.
†E-mail: stefania.arangio@uniroma1.it

the structural integrity and performance by measuring and analyzing the structural responses. As such responses are usually small in amplitude and often corrupted by noise, for monitoring efficiency, it is important to develop practical techniques for extracting useful information from the measured data.

Conventional methods may encounter difficulties for dealing with these data, whereas adaptive models, like the neural networks applied in this work, represent a useful modeling tool. Such models are robust and fault tolerant and can effectively deal with qualitative, uncertain and incomplete information, making them highly promising for smart monitoring of civil structures [8–14]. For example, Wu *et al.* [15] discussed the use of neural network models for detecting damage in a multistory frame with rigid floor. Elkordy *et al.* [16] proposed a neural network method by using displacements and strain mode shapes; Pandey and Barai [17] described the use of neural networks for damage detection of steel-truss bridge structures; Masri *et al.* [18] proposed a neural network approach to detect changes in the structural parameters. Kim *et al.* [19] described a two-stage procedure where, in the first stage, traditional sensitivity analysis is used to rank and select critical members, and in the second stage, the results of the sensitivity analysis and a neural network are used to identify the optimal number and locations of monitoring sensors. This method is applied to two-dimensional trusses and multistory frames. Ni *et al.* [20] discussed the relevance of the right construction of the input vector. Ko *et al.* [21] proposed a multi-stage method for the monitoring of the cable-stayed bridge Kap Shui Mun Bridge in Hong Kong. Xu and Humar [22] proposed a two-step algorithm that uses the modal energy-based index to locate the damage and a neural network model to determine the extent.

To improve the performance of neural network models, it is interesting to consider their framework from the probability logic viewpoint [23]: in fact, for the task of inferring a best estimate of the model parameters, the so-called learning phase, the traditional and the Bayesian approach can be made mathematically equivalent, if Laplace asymptotic approximation is used [24], but the potential enhancements that can be obtained by applying Bayesian inference at further levels in a hierarchical fashion are often not appreciated. These various levels can be summarized as follows:

- *Level 1*—Model fitting: task of inferring appropriate values for the model parameters, given the model and the data.
- Level 2—Optimization of the regularization terms that make Level 1 a better conditioned inverse problem.
- Level 3—Model class selection: the Bayesian approach allows a data-based comparison between models using alternative network architectures.
- *Level 4*—Automatic relevance determination (ARD): the relative importance of different inputs can be determined by using separate regularization coefficients.

In this work, after a historical overview of the probability logic approach and its application in the field of neural network models, the existing literature is revisited and reorganized according to the enunciated four levels. Then, this framework is applied in a multi-step strategy for the identification and quantification of damage in a long-suspension bridge. In the first step of the proposed strategy, the occurrence of damage is detected and the damaged portion of the bridge is identified. In the second step, using a pattern recognition approach, the specific damaged element is recognized and the intensity of damage is evaluated. The hierarchical Bayesian framework is applied in both steps and the improvements in the results are discussed.

2. TRADITIONAL NETWORK TRAINING AND PROBABILITY LOGIC INTERPRETATION

The neural network concept has its origins in attempts to find mathematical representations of information processing in biological systems. A neural network can also be viewed as a way of constructing a powerful statistical model for non-linear regression. It can be described by a

DOI: 10.1002/stc

series of functional transformations working in different correlated layers [25]:

$$y_k(\mathbf{x}, \mathbf{w}) = h\left(\sum_{j=1}^{H} w_{kj}^{(2)} g\left(\sum_{i=1}^{D} w_{ji}^{(1)} x_i + w_{j0}^{(1)}\right) + w_{k0}^{(2)}\right) \tag{1}$$

where y_k is the kth neural network output; $\boldsymbol{x}$ is the vector of the D variables in the input layer; $\boldsymbol{w}$ consists of the adaptive weight parameters, $w_{ji}^{(1)}$ and $w_{kj}^{(2)}$, and the biases, $w_{j0}^{(1)}$ and $w_{k0}^{(2)}$; H is the number of units in the hidden layer; and the quantities in the brackets are known as activations: each of them is transformed using a non-linear activation function (h and g) [26]. Input–output data pairs from a system are used to train the network by 'learning' or 'estimating' the weight parameters and biases.

In the traditional training approach, the values of the components of $\boldsymbol{w}$ are estimated from the training data by minimizing a proper error function, e.g. the sum of squared errors with weight decay regularization [27]:

$$E = E_D + E_W = \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{N_o}\{y_k(\boldsymbol{x}^n; \mathbf{w}) - t_k^n\}^2 + \frac{\alpha}{2}\sum_{i=1}^{W} w_i^2 \tag{2}$$

where y_k is the kth neural network output corresponding to $\boldsymbol{x}^n$, the nth input in the training data; t_k^n is the corresponding nth target output in the training data; N is the number of input–output pairs in the target data set; N_o is the number of output variables; W is the number of parameters in $\boldsymbol{w}$; and α is a regularization parameter. The first term E_D represents the error between the data and the approximation by the neural network and the second term E_W is a decay regularization that penalizes large weight parameters.

Neural network learning can be framed as Bayesian inference, where probability is treated as a multi-valued logic that may be used to perform plausible inference [23]. Although the roots of this probability logic approach originated long ago in the work of Bayes [28], Laplace [29], Jeffreys [30] and Cox [31], the practical application of Bayesian methods was for a long time severely limited by the computational difficulties in carrying through the full Bayesian procedure. The recent development of approximate methods, especially stochastic sampling methods, along with dramatic improvements in the power of computers, has opened the door to the practical use of Bayesian techniques in an impressive range of applications across all disciplines. In recent years, in civil engineering, for example, the probability logic approach has been successfully applied to system identification problems and structural health monitoring (e.g. [32–36]) using the Bayesian probabilistic framework presented in Beck [37] and Beck and Katafygiotis [38,39]) for dynamic systems.

Starting from the early works of MacKay [24] and Buntine and Weigend [40], there has been a growing interest for the application of this Bayesian framework in the field of neural networks [41–43]. Several relevant textbooks have been published: for example, Neal [44] proposed a detailed full Bayesian approach to neural networks; Lee [45] dealt with neural networks within the context of nonparametric regression and classification; Nabney [46] developed software to complement the key reference books of Bishop [25].

To pose the neural network model within the Bayesian framework, the learning process needs to be interpreted probabilistically. First, it can be considered that the network output represents the mean output conditional on the input [27]. Second, as a neural network gives an empirical model, it is not expected to make perfect predictions (in fact, if it could match the data exactly, one would suspect over-fitting due to an excessive number of parameters), so the prediction error $\boldsymbol{\varepsilon} = \boldsymbol{t} - \boldsymbol{y}(\boldsymbol{x}; \boldsymbol{w})$ between the target vector $\boldsymbol{t}$ and the network output vector $\boldsymbol{y}$ needs to be interpreted probabilistically using an appropriate prediction-error probability model, such as independent and identically distributed (i.i.d.) Gaussian PDFs (probability density functions) with mean zero and variance $\sigma_D^2 = 1/\beta$ whose choice is supported by the principle of maximum differential entropy [23]. Thus, the likelihood function for a training data

DOI: 10.1002/stc

set $D = x^n, t^n: n = 1,\ldots,N$ is given by

$$p(D|\boldsymbol{w},\beta,M) = \left(\frac{\beta}{2\pi}\right)^{N\cdot N_0} \exp\left[-\frac{\beta}{2}\sum_{n=1}^{N}\sum_{k=1}^{N_o}\{y_k(x^n;\boldsymbol{w}) - t_k^n\}^2\right] \tag{3}$$

where M denotes the Bayesian model class that specifies the form of the likelihood function here and also the prior probability distribution discussed next.

Although the likelihood function does take into account the uncertain prediction error, it does not quantify the uncertainty in the values of the parameters $\boldsymbol{w}$. In the Bayesian framework, this can be represented by a prior PDF $p(\boldsymbol{w}|M)$ which expresses the relative plausibility of each value of the parameter vector $\boldsymbol{w}$ [23]. Since generally there is a little idea of what the values should be, it is usual to select the prior as a rather broad distribution expressing a large amount of uncertainty. Under the constraints of specified zero mean and the same variance $\sigma_W^2 = 1/\alpha$ for each component of vector $\boldsymbol{w}$, the principle of maximum differential entropy leads to a Gaussian prior PDF of the form:

$$p(\boldsymbol{w}|\alpha,M) = \left(\frac{\alpha}{2\pi}\right)^{W/2} \exp\left(-\frac{\alpha}{2}\|\boldsymbol{w}\|^2\right) \tag{4}$$

where $\|\boldsymbol{w}\|^2$ is the Euclidean norm as in the last term of Equation (2). Other choices for the prior $p(w|\alpha,M)$ for neural networks have been investigated [45,47].

Using available data, Bayes' Theorem updates the prior PDF $p(w|\alpha,M)$ to give the posterior PDF:

$$p(\boldsymbol{w}|D,\alpha,\beta,M) = \frac{p(D|\boldsymbol{w},\beta,M)\,p(\boldsymbol{w}|\alpha,M)}{p(D|\alpha,\beta,M)} \tag{5}$$

This posterior distribution is always more compact than the prior distribution if the data informs the model, as indicated schematically in Figure 1, expressing the fact that something has been learned; it gives a measure of the relative plausibility that the model's parameters should

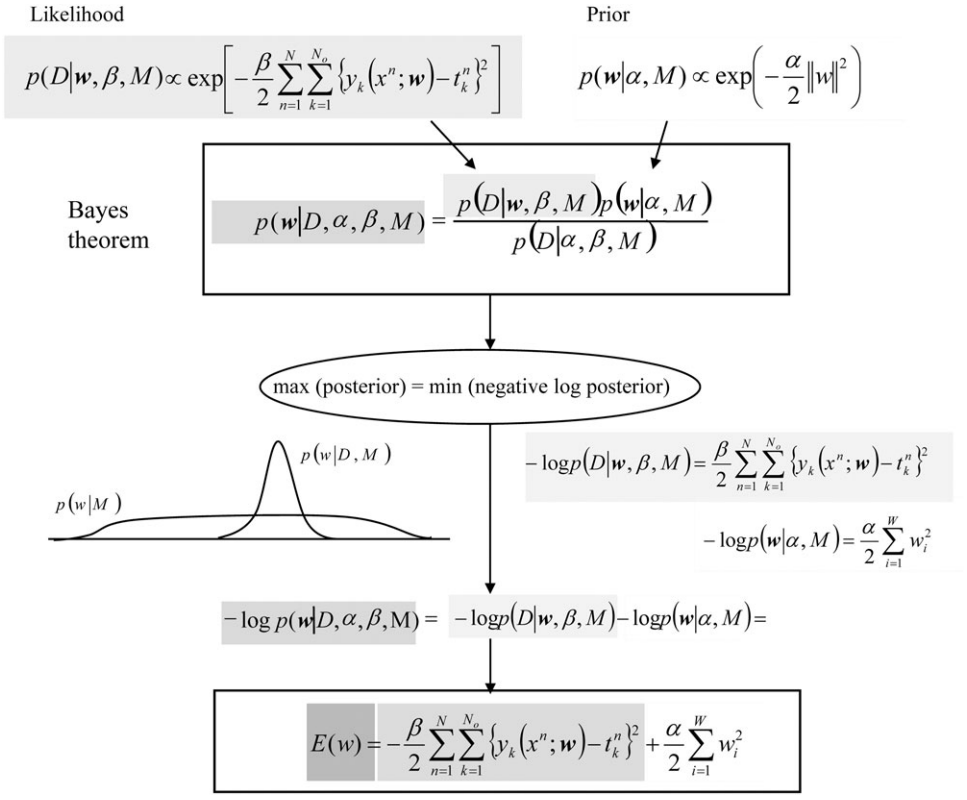

Figure 1. Learning as inference.

DOI: 10.1002/stc

have a specified value $\boldsymbol{w}$ based on the data and the prior information. Therefore, by maximizing the posterior, the most plausible (maximum *a posteriori*—MAP) values $\boldsymbol{w}_{\mathrm{MAP}}$ of the parameters can be found.

Instead of finding a maximum of the posterior probability in Equation (5), one can seek instead a minimum of its negative logarithm. As shown in Figure 1, for the chosen prior distribution and likelihood function, the negative log posterior probability is just the usual regularized sum of squares function in Equation (2). Therefore, the conventional learning approach can be derived as a particular approximation of the Bayesian framework where only the MAP parameter values are utilized and so the post-data uncertainty in $\boldsymbol{w}$ is ignored.

3. BAYESIAN ENHANCEMENTS FOR NEURAL NETWORKS

3.1. First level of inference: model fitting

In Figure 2, the steps for the practical application of this first level of inference are schematized. First, network input and output are selected and a data set is collected or created. Usually, the data need to be pre-processed to reduce their complexity. Then, a suitable network architecture is selected and the probability model is defined. At this point, network training starts: it consists

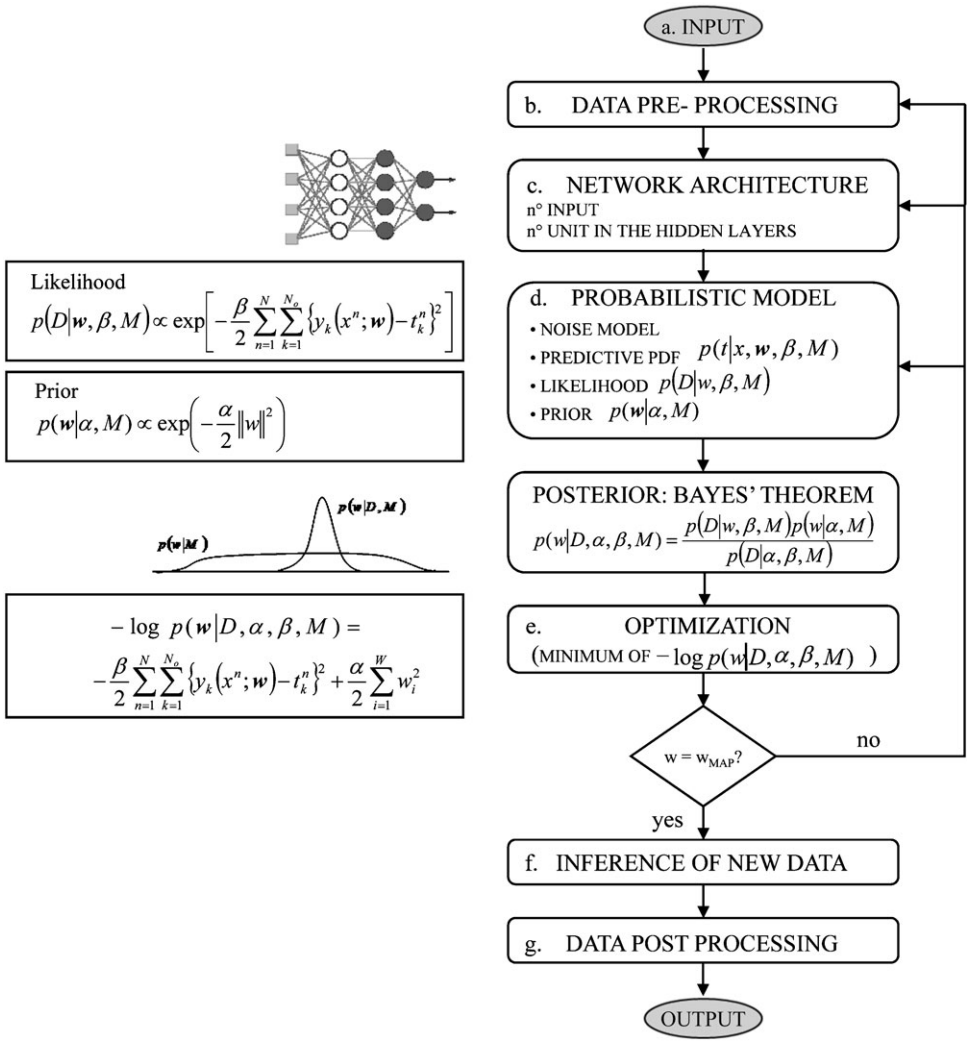

Figure 2. Level 1 inference: flow chart of model fitting using neural networks.

DOI: 10.1002/stc

of utilization of input–output data by maximizing the posterior probability of the model specified by $\boldsymbol{w}$. At the end of training, if the level of learning and generalization is considered sufficient, the iterative network optimization process is stopped and predictions can be made using the trained network. If the result is not satisfactory, one or more of the previous steps is repeated.

3.2. *Second level of inference*: *evaluating the hyperparameters*

In the first level of inference, it has been assumed that the hyperparameters α and β are known. To select the appropriate values of the hyperparameters, Bayes' Theorem can be applied at a second level [41] as follows:

$$p(\alpha, \beta|D, M) = \frac{p(D|\alpha, \beta, M)\, p(\alpha, \beta|M)}{p(D|M)} \tag{6}$$

The correct Bayesian treatment to evaluate these parameters will be, in general, computationally expensive. Different computational methods can be used [44]. In this work, an iterative method based on Laplace's asymptotic approximation, which leads to Gaussian approximations of the posterior PDF, has been used [24,39,44]. This method, known as the *evidence procedure* in the neural network field [24], is not a full Bayesian approach because it searches for optimal hyperparameters instead of integrating them out using, for example, stochastic simulation methods. Nevertheless, it has given good results for many applications (e.g. [48]) and is considerably less computationally demanding than the full Bayesian procedure. Applying this method, the hyperparameters are estimated at the same time as the parameters $\boldsymbol{w}$.

As shown in Figure 3, initial values for α and β are chosen and the related values of $\boldsymbol{w}$ are obtained by maximizing their posterior probability. Then, the hyperparameters are re-estimated using the following relations [27] which are their MAP values, based on a uniform prior for α

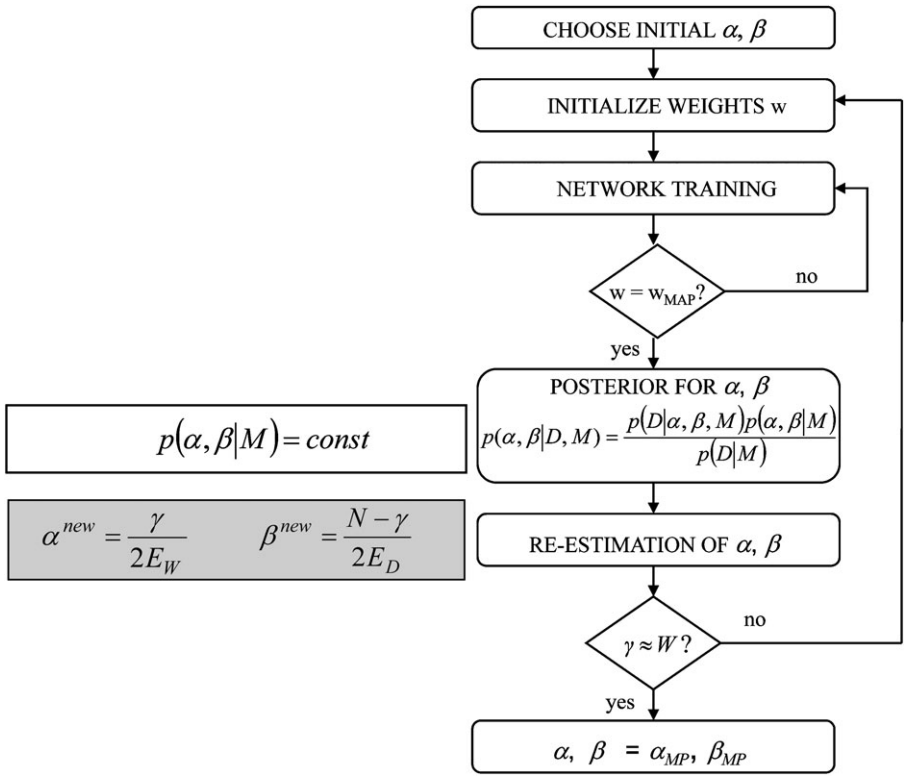

Figure 3. Level 2 inference: implementation of the evidence procedure.

DOI: 10.1002/stc

and β, that is, these values maximize the evidence $p(D|\alpha,\beta,M)$ in Equation (6):

$$\alpha^{\text{new}} = \frac{\gamma}{2E_W} \tag{7}$$

$$\beta^{\text{new}} = \frac{N-\gamma}{2E_D} \tag{8}$$

where E_W and E_D, already discussed in the previous section, indicate the part of the error function related to the prior and likelihood for the parameters, respectively, and N is the number of available sample input-output pairs in the data set. The parameter γ measures the effective number of parameters whose values are controlled by the data rather than by the prior, i.e. the number of well-determined parameters. It is computed using the following equation, obtained by differentiating the expression of the evidence based on Laplace's asymptotic approximation and setting the derivative to zero [24]:

$$\gamma = W - \alpha \text{Trace} \boldsymbol{A}^{-1} = W - \sum_{\alpha=1}^{W} \frac{\alpha}{\lambda_\alpha + \alpha} = \sum_{\alpha=1}^{W} \frac{\lambda_\alpha}{\lambda_\alpha + \alpha} \tag{9}$$

where W is the total number of parameters in $\boldsymbol{w}$. The relation has been written in terms of the eigenvalues of the matrix $\mathbf{A}$, which is the Hessian of the total error function (given by the sum of E_D and E_W). Each eigenvalue measures how strongly one parameter is determined by the data, while α measures how strongly the parameters are determined by the prior. Each term $\gamma_a = \lambda_\alpha/(\lambda_\alpha + \alpha)$ is a number between 0 and 1, which measures the strengths of the data relative to the prior in direction a. A direction in parameter space for which λ_a is small compared with α does not contribute to the number of good parameter measurements.

The iterative estimation of the hyperparameters is repeated until convergence is achieved. According to Bishop [27], it is possible to consider that the algorithm converges when γ is close to the total number of parameters ($\gamma \cong W$).

3.3. Third level of inference: model class selection

Although it is a crucial aspect in neural network applications, very few publications deal with the choice of the optimal model complexity: for example, how many hidden nodes should be included in a layer? It is not correct to choose simply the model that fits the data better: more complex models will always fit the data better but they may be over-parameterized and so generalize poorly by making poor predictions for new cases. Therefore, in model class selection, it is necessary to penalize models that are more complicated.

In the fourteenth century, William of Ockham proposed: 'Frustra fit per plura quod fieri potest per pauciora', now known as Ockham's razor, which can be translated as 'It is vain to do with more what can be done with fewer' [49]. In the present context, this philosophy implies a principle of model parsimony that simpler models should be preferred over unnecessarily complicated ones that offer only slight improvements in the fit to the data. This point was recognized by Jeffreys [30], who pointed out the need for a simplicity postulate that would give a quantitative expression of Ockham's razor. Akaike [50], for example, introduced the AIC, which combines a measure of the data fit with a term penalizing a larger number of uncertain parameters but the form of the penalty term was determined in a somewhat *ad-hoc* manner. In recent years, there has been a growing interest in the application of the Bayesian approach to model class selection, which has been shown to automatically enforce a quantitative expression of Ockham's razor without the introduction of *ad-hoc* penalty terms [34,41,49].

The selection of the appropriate model class is the third inference task where Bayesian techniques can be successfully applied to neural network models. It can lead to significant improvements in the neural network model performance; in fact, many heuristic methods are suggested in the literature to determine the optimal network architecture (see, for example, [51,52]) but most of them are lacking a rigorous basis or, like the widely used early stopping technique, need the data set to be portioned into a training set and a validation set. On the

DOI: 10.1002/stc

contrary, the Bayesian approach permits an objective comparison between different models and allows all of the available data to be used for training [53].

In Bayesian model selection, the most plausible model class among a set $\mathscr{M}$ of N_M candidate ones is obtained by applying Bayes' Theorem as follows:

$$p(M_j|D,\mathscr{M}) \propto p(D|M_j)\,p(M_j|\mathscr{M}) \tag{10}$$

The factor $p(D|M_j)$ is known as the evidence for the model class M_j provided by the data D. The user's judgment about the initial plausibility of each neural network model class is expressed by the prior probability $p(M_j|\mathscr{M})$ over the set of model classes M_j for $j = 1,\ldots,N_M$ where:

$$\sum_{j=1}^{N_M} p(M_j|\mathscr{M}) = 1 \tag{11}$$

Equation (10) shows that the most plausible model class is the one that maximizes $p(D|M_j)$ $p(M_j|\mathscr{M})$ with respect to j. If there is no particular reason *a priori* to prefer one model over another, they can be treated as equally plausible a priori and a non-informative prior, i.e. $p(M_j|\mathscr{M}) = 1/N_M$, can be assigned, then different models can be compared just by evaluating their evidence [24].

3.4. Fourth level of inference: automatic relevance determination of inputs

Once the optimal architecture has been determined, the last issue that should be considered is the relative importance of each input variable. Using data from real systems, it could be difficult to separate the relevant variables from the redundant ones. In the Bayesian framework, this problem can be addressed by the Automatic Relevance Determination (ARD) method, proposed by Mackay [41] and Neal [44].

To use this technique, a separate hyperparameter α_i is associated with each input variable, which represents the inverse variance of the prior distribution of that input parameter. In this way, every hyperparameter explicitly represents the relevance of one input: a small value means that a large weight parameter value is allowed and the corresponding input is important; on the contrary, a large value of α_i means that the corresponding weight parameter is constrained near zero, and hence the corresponding input is less important. Therefore, the ARD allows a fourth level of inference to be added in which the hyperparameters α_i are chosen by maximizing the evidence for the model class defined by these hyperparameters. Once the architecture of the model is defined, the importance of every input is evaluated: if some hyperparameter is very large, the related input will be dropped from the model and the optimal architecture for the new model will be re-estimated.

The four levels are summarized in the flow chart in Figure 4. Starting from the simple process in Figure 2, further steps have been added to include the other three levels of inference: evaluation of the hyperparameters, model class selection, and ARD. For the model class selection, it has been assumed that the evidence will have a single peak, so the optimal model is selected in an iterative way considering models of increasing complexity: in Figure 4, the index i, which represents the number of units in the internal layers, gets larger at every loop and the process continues until the increasing of the the complexity corresponds to a decreasing of the evidence $p(D|M_j)$.

4. STRATEGY FOR INTEGRITY ASSESSMENT OF A LONG-SUSPENSION BRIDGE

The theoretical framework developed in the previous sections is applied in a multi-step strategy for the assessment of the integrity of the bridge in Figure 5 (Arangio, [54]). It is a long-span suspension bridge whose construction has been recently re-proposed for crossing the Strait of Messina in Italy. A finite element model based on a preliminary design scheme has been considered for numerical calculations. It has a main span of 3300 m; the towers are 383 m high and the bridge suspension system relies on two pairs of steel cables, each with a diameter of 1.24 m. The secondary suspension system consists of 121 pairs of rope hangers. The cross-section of the deck is composed of three box elements supported every 30 m by transversal

DOI: 10.1002/stc

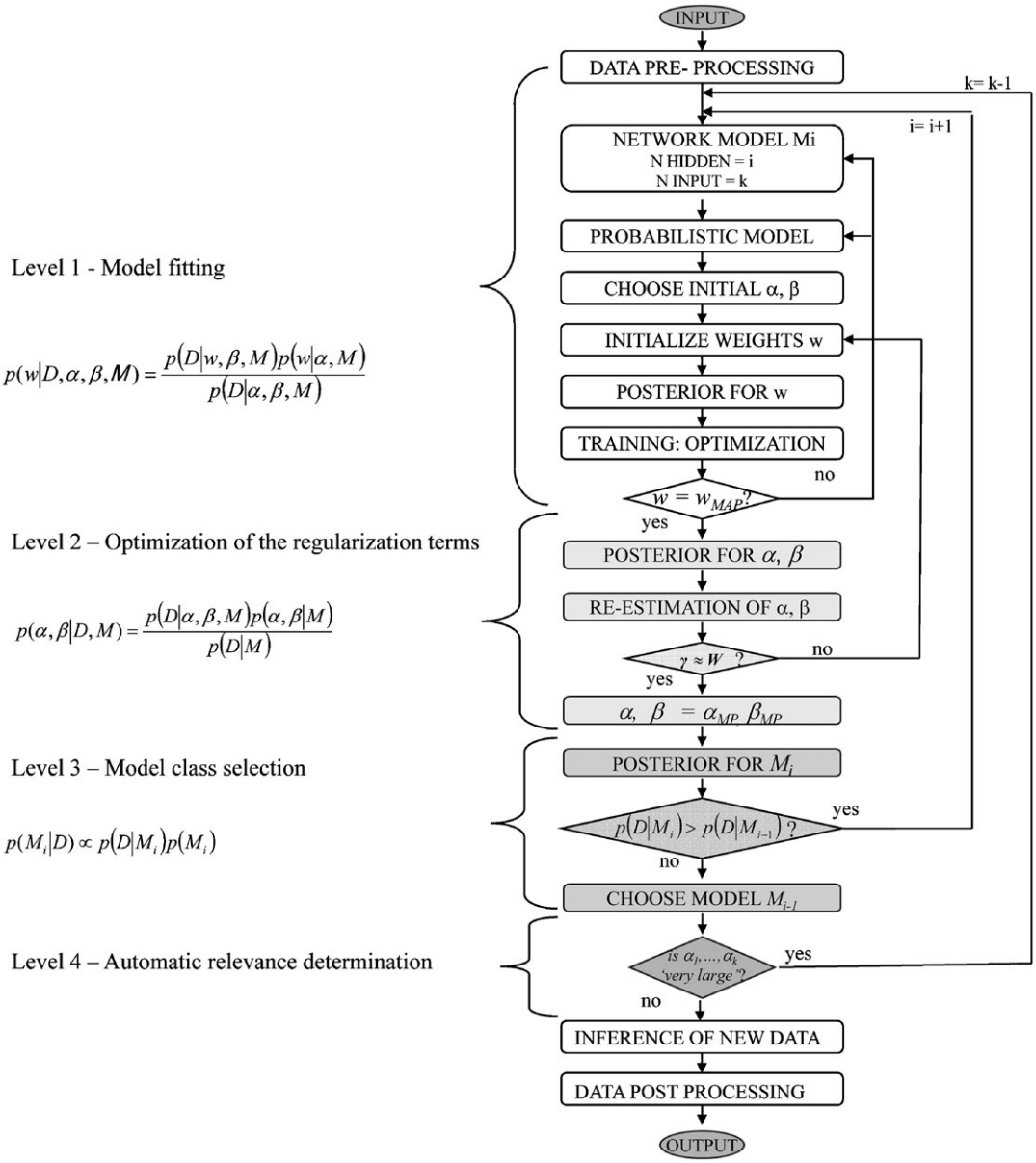

Figure 4. Hierarchical Bayesian framework for neural networks.

beams; the deck carries six road lanes in the external box girder and two railway tracks in the central one. Bontempi [55] gives more detailed information about the bridge design.

A multi-step damage identification strategy has been followed because it has been shown that it is more effective to consider independently the tasks of damage detection, location and quantification [21,56]. In the first step of the strategy, the occurrence of anomalies or damage in the bridge is detected, and the damaged portion of the structure is identified. If some damage is detected, the second step of the procedure is initiated using a pattern recognition approach, where the location of the specific damaged member within the whole area is identified, and the extent of damage is evaluated. The two steps are illustrated in more detail in the following.

4.1. Step 1: *damage detection*

In the first step of the proposed strategy, the response of the structure is monitored at various measurement points located in a group of three (A, B, C in Figure 5) every 30 m along the bridge

DOI: 10.1002/stc

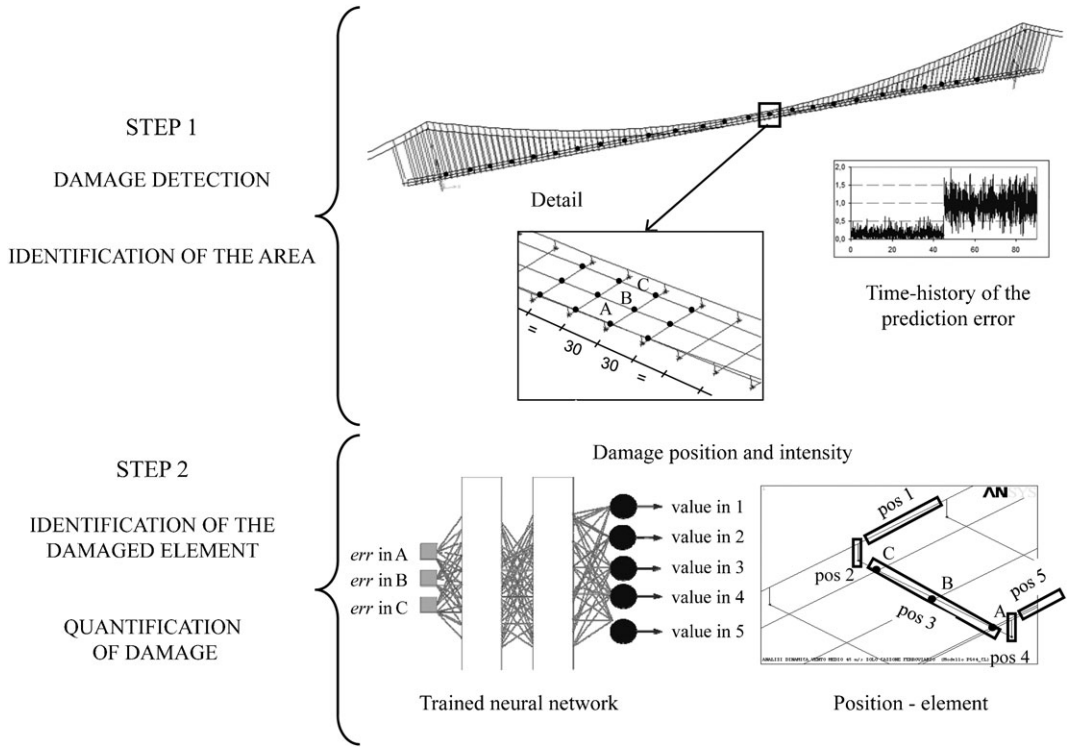

Figure 5. Scheme of the damage identification strategy.

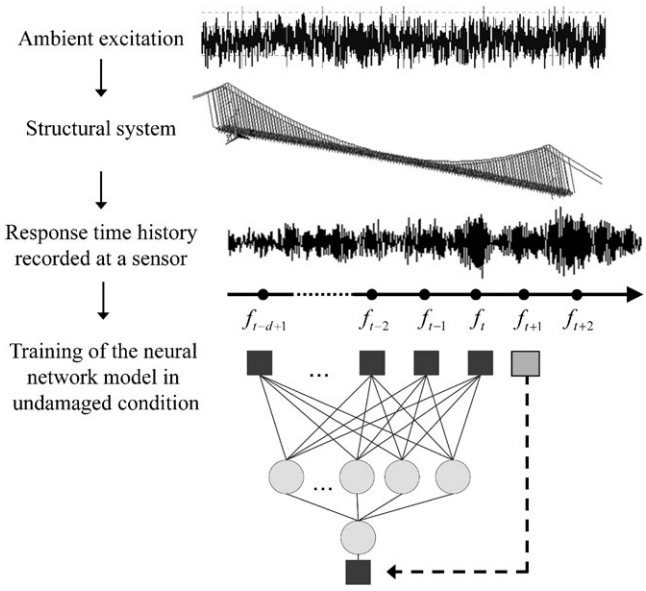

Figure 6. Procedure for network training.

deck; correspondingly, different neural networks are trained, one for each intermediate point B. The neural network models are built and trained using the time-histories of the response of the structure subjected to wind actions and traffic loads (due to the passage of a train) in the undamaged situation.

The procedure for network training is showed in Figure 6. The time-history of a response parameter f is sampled at regular intervals, generating series of discrete values f_t. A set d of such values: $f_{t-d+1}, \ldots, f_t$ is used as input of the network model, while the next value f_{t+1} is used as the target output. By stepping along the time axis, a training data set consisting of many sets of

DOI: 10.1002/stc

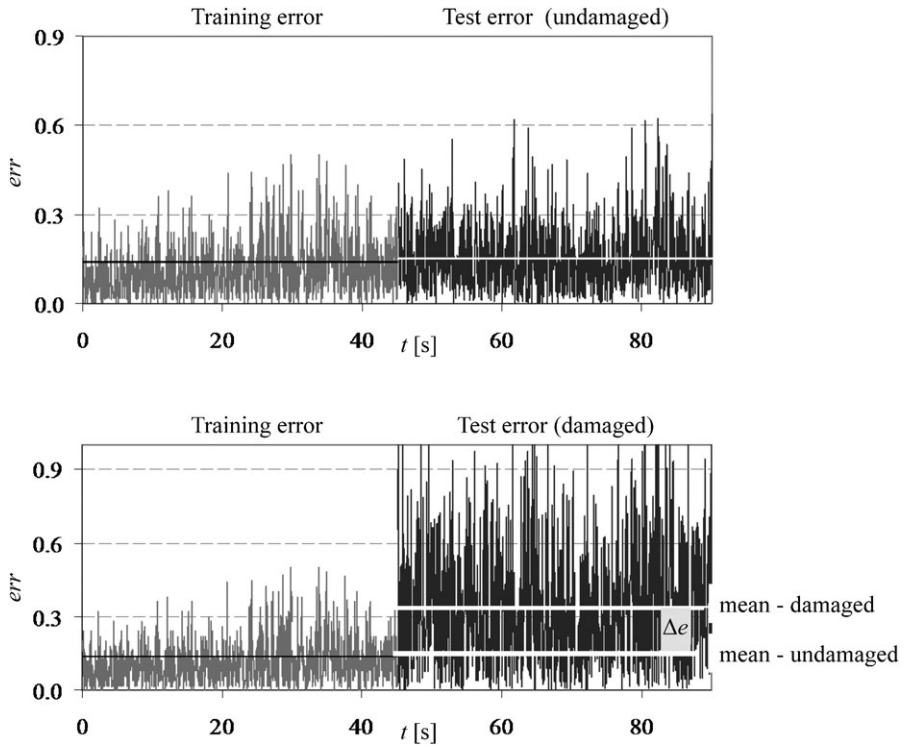

Figure 7. Difference *err* between output *y* and target *t* values for training and testing in undamaged and damaged conditions in a case example (considered damage: 5% reduction of stiffness in one cable).

input vectors with the corresponding output values is built, and the network models are trained. The trained models are then tested with a set of observed values $f_{t+n-d}, \ldots, f_{t+n}$, to predict the value of f_{t+n+1}, according to the procedure of one step ahead prediction [27].

Then, new input patterns, corresponding to both undamaged and damaged situations, are tested on the trained models. For each pattern, the next value is predicted and compared with the target output. If the error in the prediction is negligible, the structure is considered as undamaged; if the error is large, the presence of an anomaly is detected.

The results of the training and test phases are elaborated as shown in Figure 7. The two plots show the difference *err* between the network output value *y* and the target value *t* at several time steps for both training and testing, in undamaged and damaged conditions. It is possible to note that the mean values of *err* (indicated by the straight lines) obtained both in training and test are comparable if the structure remains undamaged. On the contrary, in case of anomalies that may correspond to damage, there is a difference Δe between the mean values of *err* in testing and training.

It has to be noted that the detected anomaly may correspond to a damage state or simply to a change of the characteristics of the excitation. To distinguish the actual cause of the anomaly, the intensity of Δe is checked in different test sections, according to the procedure represented in the flow chart in Figure 8, where all the aspects related to the first steps are schematized. In the left side the start-up is shown: given a data set, the optimal neural network model is selected and trained according to the Bayesian approach. Then, as shown in the right side of Figure 8, the trained model is tested with new input data sets. At this point, there are three possibilities: if the difference Δe between the mean errors in training and test is close to zero, the structure is considered undamaged; if Δe is different from zero in several test sections, it can be concluded that the external actions are probably changed. In this case, the trained neural network models are unable to represent the time-histories of the response parameters, and they have to be updated and re-trained according to the modified characteristics of the action. If Δe is different from zero only in one or few test sections (Figure 9) and generally decreases with the distance from the selected section, it can be concluded that the considered section of the structure is

DOI: 10.1002/stc

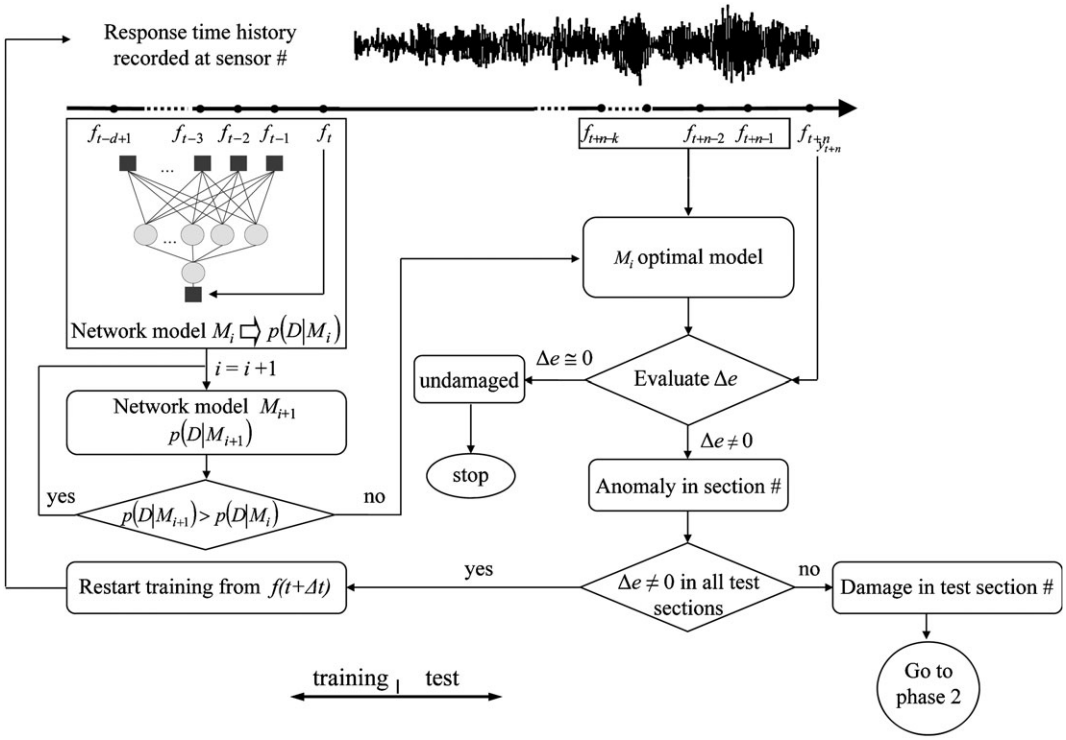

Figure 8. Flow chart of the procedure for damage detection.

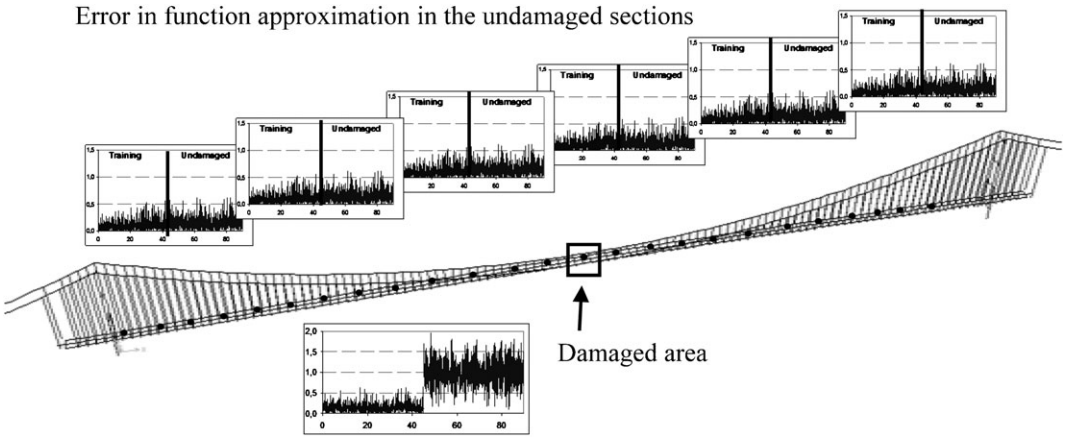

Figure 9. Identification of the damaged portion by considering the errors in the approximation.

damaged and the second step of the procedure is initiated. 'Different from zero' is a concept whose quantification is left to the experience of the operator, because a critical magnitude of the error cannot be defined in absolute terms, but it should be related to the specific structural element; to investigate this correlation and to assess the efficiency of the procedure, different scenarios of damage are considered in the numerical applications and the corresponding Δe is evaluated.

Data are simulated by analyzing the dynamic response of the finite element model of the bridge. Damage is modeled as a reduction of stiffness of a structural element. The following damage scenarios are considered:

- hangers: reduction of stiffness from 5 to 80%;
- cables: reduction of stiffness from 1 to 10%;
- transverse beam: reduction of stiffness from 5 to 30%.

DOI: 10.1002/stc

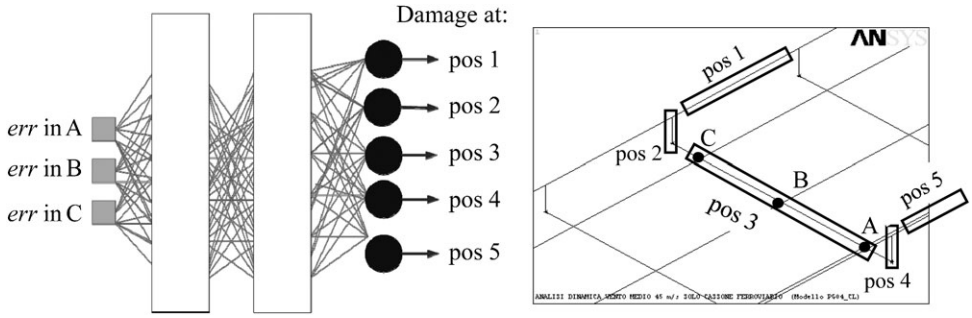

Figure 10. Neural network model for the identification of damage location and intensity.

The training data set for every network model includes 1000 time-history samples of the response parameters that were found to be more sensitive to a stiffness reduction (Arangio and Petrini, [57]): they are the rotation of the centroid of the deck cross-section about the longitudinal axis in case of wind actions, and the vertical displacement of the same point in case of traffic loads.

The number of sampled data is chosen to be 1000 after some trials. The time-histories of the responses are continuous signals so one can select as many samples as wanted, but beyond 1000 samples it was found that there was no significant improvements in the generalization capacity of the network.

4.2. *Step 2*: *identification of damage location and intensity*

If damage is detected in a portion of the structure, the second step of the procedure is initiated; it is aimed at identifying the specific damaged element (one of the two cables, the transverse beam, or one of the two hangers) in the selected area, and at evaluating the damage intensity. A pattern recognition approach is used. In this step, the mean values of the error *err* in the prediction of the response time-histories for all three measurement points (A, B, C) in the considered section are taken as inputs of the neural network model. As shown in Figure 10, each damage scenario is described by a vector of five components: each component indicates the state (presence and intensity) of damage for one of the structural elements in the test section. The data set is collected by simulating 400 damage scenarios (corresponding to different location and intensities of damage): 370 of them are used for network training and the other 30 for testing the generalization performance.

5. RESULTS OF THE BRIDGE INTEGRITY ASSESSMENT STRATEGY

5.1. *Step 1*: *Bayesian model selection for training in an undamaged condition*

Various network models were trained using the time-histories of the response of the bridge in an undamaged condition. The hierarchical Bayesian approach discussed in the previous sections was applied. Regarding the first two levels of inference, no significant differences between the traditional and the Bayesian approach were noted. On the other hand, the application of the other two levels, Bayesian model selection and Automatic Relevance Determination (ARD), led to some improvements. Only the results related to these two levels are shown here.

To select the optimal model for damage detection in Step 1, at first a class of models with four input units, corresponding to the number of previous instants needed to predict the value at the instant $t+1$, was considered. For this class, different architectures, with an increasing number of hidden units, were analyzed and compared. Treating all the models as equally plausible *a priori*, a non-informative prior $p(M_j) = 1/N_M$ over the models was assigned, and so various models were compared by just evaluating their evidence $p(D|M_j)$. In particular, the logarithm of the evidence for model class M_j was computed using the following relation based on Laplace's

DOI: 10.1002/stc

asymptotic approximation of the evidence integral [27]:

$$\ln p(D|M_j) = -\beta_j^{MP} E_D^{MP} + \frac{N}{2}\ln \beta_j^{MP} + \ln(H_j^{MP}!) + 2\ln H_j^{MP} - \alpha_j^{MP} E_W^{MP} - \frac{1}{2}\ln|\boldsymbol{A}_j^{MP}| + \frac{W_j}{2}\ln \alpha_j^{MP} + \frac{1}{2}\ln\left(\frac{2}{\gamma_j^{MP}}\right) + \frac{1}{2}\ln\left(\frac{2}{N-\gamma_j^{MP}}\right) H_j^{MP} \tag{12}$$

where α_j^{MP} and β_j^{MP} are the MAP values of the hyperparameters for model class M_j discussed in Section 3.2, γ_j^{MP} is the corresponding value of the parameter in Equations (7) and (8), N is the number of samples in the data set, H_j^{MP} is the number of hidden units of the model class M_j, the matrix $\boldsymbol{A}_j^{MP}$ is the Hessian of the total error function (given by the sum of E_D and E_W), and W_j is the number of parameters in $\boldsymbol{w}_i$.

It is important to note that in this equation it is possible to recognize two kinds of terms: those in the first row give a measure of the data fit, so they increase as the models get more complex; they are reported in the first row of the following tables. The sum of the other terms, which represents the Ockham factor, penalizes more complex models, so it decreases when the complexity increases. This sum can be viewed as the amount of information extracted from the data by model class M_j (it is the relative entropy of its posterior PDF with respect to its prior PDF) [36]. The optimal complexity is given by the model with the best compromise between these terms, which is produced by maximizing the evidence of the model and hence maximizing its posterior probability.

In the first row of Table I, some models are presented with increasing complexity (more hidden units) within the class with four input variables; one can see that the model with the highest evidence is the simplest one (model 1, one hidden unit). Once the best model within the model class is identified, the ARD is applied to determine whether this class is the optimal one or not. In the bottom part of the Model I column, the vector of the hyperparameters α_j, where $j = 1,\ldots,I$, with $I =$ number of the inputs, for the model 1, is shown. One can see that the two smallest values have the same order of magnitude, whereas the other two values are much higher, suggesting that the latter should be neglected. For this reason, another model class, with only two input variables, is analyzed.

Table I. Results of the Bayesian model selection and ARD for a model class with four input variables.

Model	1	2	4	8
Parameters W	7	13	25	49
$-\beta E_D^{MP} + \frac{N}{2}\ln \beta_{MP}$	11 081	11 081	11 081	11 081
$\ln H_j! + 2\ln H_j$	2.08	3.99	8.01	17.20
Data fit term	11 083	11 085	11 089	11 098
$-\alpha E_W^{MP} - \frac{1}{2}\ln\lvert\boldsymbol{A}\rvert + \frac{W}{2}\ln \alpha_{MP}$	−32	−44	−70	−157
$\frac{1}{2}\ln\left(\frac{2}{\gamma}\right) + \frac{1}{2}\ln\left(\frac{2}{N-\gamma}\right)$	−4.0	−4.12	−4.30	4.54
Penalizing term	−35.6	−47.7	−73.8	−152.2
Log evidence	**11 048**	11 037	11 015	10 946
	α model 1			
	121			
	84			
	9			
	6			

DOI: 10.1002/stc

In Table II, the results of this selection are shown and once again, the model with the highest evidence has only one hidden unit. Applying the ARD to this model (bottom part of the Model 1 column), one can see that, in this case, the values of the hyperparameters are comparable; thus, the optimized model has 2, 1, and 1 units in the input, hidden and output layers, respectively.

5.2. *Step 1*: *damage detection*

The results of the first step of the procedure are shown in Figure 11(a)–(c), where the differences between the mean values of the errors Δe in the damaged and undamaged conditions are shown

Table II. Results of the Bayesian model selection for a model class with two input variables.

Model	1	2	4	8
Parameters W	5	9	13	17
$-\beta E_D^{MP}+\frac{N}{2}\ln\beta_{MP}$	14 554	14 555	14 555	14 554
$\ln H_j!+2\ln H_j$	2.08	3.99	5.94	8.01
Data fit term	14 556	14 559	14 561	14 562
$-\alpha E_W^{MP}-\frac{1}{2}\ln\lvert\boldsymbol{A}\rvert+\frac{W}{2}\ln\alpha_{MP}$	−32	−57	−62	−65
$\frac{1}{2}\ln\left(\frac{2}{\gamma}\right)+\frac{1}{2}\ln\left(\frac{2}{N-\gamma}\right)$	−3.9	−4.08	−4.18	−4.23
Penalizing term	−35.8	−61.3	−65.9	−69.3
Log evidence	**14 520**	14 498	14 495	14 493
	α model 1 72 18			

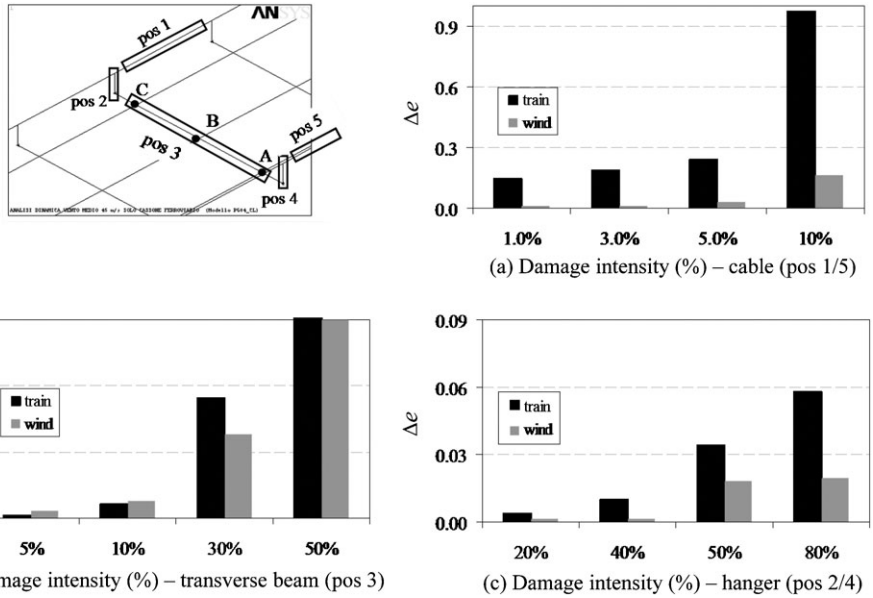

Figure 11. Differences between the mean errors Δe in training and testing for different intensities of damage in (a) one of the two cables; (b) the transverse beam; and (c) one of the two hangers.

DOI: 10.1002/stc

in the form of histograms for different intensities of damage (that is, of stiffness reduction) to the cables, the transverse beam, and the hangers, respectively. It can be noted from Figure 11 that the proposed strategy is more effective when responses to high-speed excitations (like traffic loads) are considered instead of responses to slow-speed excitations (like wind actions). Thus, in the following step, only the structural response due to the transit of a train is considered.

The possibility of detecting damage is different for the various elements, as expected: in fact, as shown in Figure 11, a small damage to the cables determines a much higher value of Δe than a strong damage to the transverse beam and the hangers. Although the sensitivity of the method to measurement noise is not fully explored here, past experience shows that the Bayesian approach is robust against noise as it is explicitly modeled probabilistically [39].

5.3. *Step 2*: *location and quantification of damage*

Once a damaged section is detected, the specific damaged element and the intensity of damage are identified using the pattern recognition approach presented in Section 4.2. The optimal network model is selected by the Bayesian approach on the base of the 370 patterns considered for training. The log evidences of the different models are compared in Table III, which shows that the optimal model (most probable model based on the data) has 11 units in the hidden layers. Thus, the selected model consists of: three input variables, that is, the errors *err* evaluated at A, B, and C; five output variables, that is, the possible locations (coincident with one of the structural element within the section) and intensities; two hidden layers with 11 units (obtained by the Bayesian selection process). After the training phase, the network is tested with the 30 patterns not included in the training set. The errors are used as inputs and the corresponding output vector representing the damage scenarios are calculated. Then, these vectors are compared with the target ones.

In order to give a global and intuitive representation of the results, two quantities are defined and evaluated for the 30 test samples: the *position*, which gives a measure of the error in locating the damage, and the *intensity*, which gives a measure of the error in estimating the damage intensity. These quantities are obtained by comparing the network output vectors $\boldsymbol{y_i}$ and the target vectors $\boldsymbol{t_i}$, where $i = 1,\ldots,30$, and are expressed as

$$\mathrm{pos}(i) = \frac{\boldsymbol{t}_i \cdot \boldsymbol{y}_i}{||\boldsymbol{t}_i||\,||\boldsymbol{y}_i||} \tag{13}$$

$$\mathrm{int}(i) = \frac{|\boldsymbol{t}_i|}{|\boldsymbol{y}_i|} \tag{14}$$

Table III. Results of the Bayesian model class selection for step 2.

Model	4	7	9	10	11	12	13	20
Parameters W	41	68	86	95	104	113	122	185
Log evidence	742	912	939	942	**962**	956	940	833

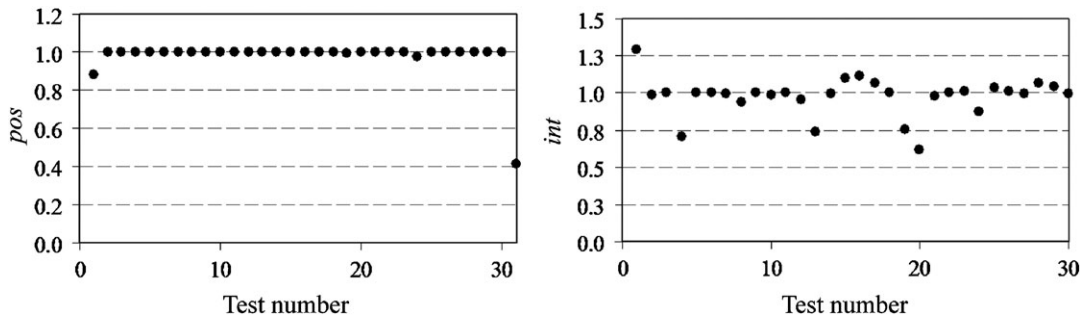

Figure 12. Location and intensity of damage using the optimal model for the 30 test samples.

DOI: 10.1002/stc

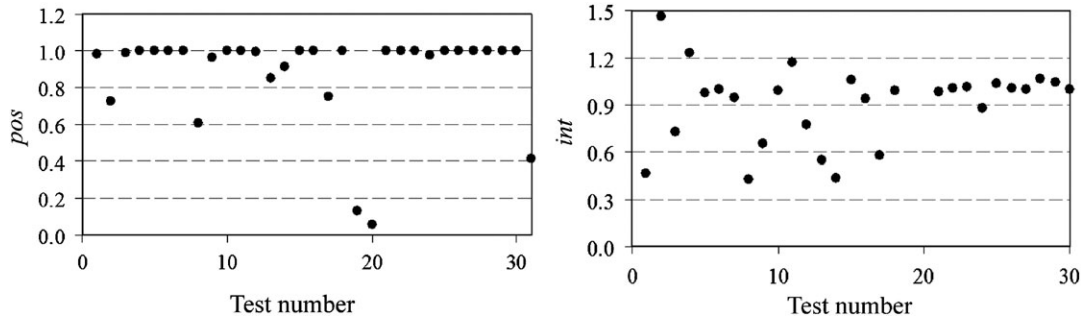

Figure 13. Location and intensity of damage using the heuristic-based 20 units model for the 30 test samples.

If these quantities are equal to one, the damage is well localized and its intensity is correctly estimated.

In Figures 12 and 13, the location and intensity results of the test carried out using the optimal model with 11 hidden units, and another heuristic-based model obtained using literature suggestions, which has 20 units, are compared. Using the optimized model, the location can be detected in 90% of the cases and the intensity in 66% of the cases, whereas for the heuristic-based model the corresponding cases give 70 and 50%, respectively.

6. CONCLUSIONS

In this paper, the probability logic approach has been applied to neural network models to develop a two-step Bayesian framework that has been utilized for identification and quantification of damage in a long-suspension bridge. In the first step of the proposed strategy, various neural network models have been trained to approximate the time-histories of the structural response in the undamaged situation. Then, new inputs, corresponding to different damage scenarios, have been presented to the trained models. Analyzing the increment of the error in the prediction, the occurrence of damage has been detected. The second step aims at the identification of the specific damaged element in the structural system and the quantification of the intensity of such damage. In this step, a pattern recognition approach has been used.

In both the damage detection and assessment steps, Bayesian techniques have been applied for four levels of inference and in both cases they allow an optimization of the model complexity and an improvement in its generalization capability. This can be noted especially in the pattern recognition step, where the choice of the optimal model instead of heuristic-based models leads to an improvement of 20% in the damage location and 16% in damage quantification.

The presented method can be useful for damage identification of large structural systems instrumented with monitoring systems that furnish measurements for both damaged and undamaged situations. Using the Bayesian neural network approach, no information about a structural model is needed.

ACKNOWLEDGEMENTS

Stefania Arangio acknowledge Prof. Fabio Casciati who brought the authors into contact by proposing the visiting period she spent at the California Institute of Technology.

REFERENCES

1. Aktan AE, Catbas FN, Grimmelsman KA, Pervizpour M. Development of a model health monitoring guide for major bridges. *Report DTFH61-01-P-00347*, Federal Highway Administration Research and Development, Drexel Intelligent Infrastructure and Transportation Safety Institute, 2002.

2. Bontempi F, Gkoumas K, Arangio S. Systemic approach for the maintenance of complex structural systems. *Structure and Infrastructure Engineering* 2008; **4**:77–94. DOI: 10.1080/1573247060115 5235.
3. Augusti G, Ciampoli M, Frangopol DM. Optimal planning of retrofitting interventions on bridges in a highway network. *Engineering Structures* 1998; **20**(11):933–939.
4. Casciati F. An overview of structural health monitoring expertise within the European Union. In *Structural Health Monitoring and Intelligent Infrastructure*, Wu ZS, Abe M (eds), Lisse: Balkema, 2003; 31–37.
5. Li H, Ou J. Structural health monitoring system for Shandong Binzhou Yellow River Highway Bridge. *Computer-Aided Civil and Infrastructure Engineering* 2006; **21**(4):306–317.
6. Doebling SW, Farrar CR, Prime MB, Shevitz DW. Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review. *Los Alamos National Laboratory Report LA-13070-MS*, 1996.
7. Sohn H, Farrar CR, Hemez FM, Shunk DD, Stinemates DW, Nadler BR, Czarnecki JJ. A review of structural health monitoring literature: 1996–2001. *Report LA-13976-MS*, Los Alamos National Laboratory, New Mexico, 2004.
8. Faravelli L, Pisano AA. A neural network approach to structure damage assessment. In *Proceedings of Intelligent Information Systems* (*IIS'97*), Adeli S (ed.). Grand Bahama Island: Bahamas, 1997.
9. Adeli H. Neural networks in civil engineering: 1989–2000. *Computer-Aided Civil and Infrastructure Engineering* 2001; **16**(2):126–142.
10. Hajela P. Application of neural networks in modeling and design of structural systems. In *Neural Networks in the Analysis and Design of Structures*, Waszczyszyn Z (ed.). Springer: Berlin, 1999.
11. Topping BHV, Khan AI, Sziveri JA, Bahreininejad A, Leite JPB, Cheng B, Ivanyi P. Neural networks in advanced computational problems. In *Neural Networks in the Analysis and Design of Structures*, Waszczyszyn Z (ed.). Springer: Berlin, 1999.
12. Kao CY, Hung SL. Detection of structural damage via free vibration responses generated by approximating artificial neural networks. *Computers and Structures* 2003; **81**:2631–2644.
13. Ko JM, Ni YQ. Technology developments in structural health monitoring of large-scale bridges. *Engineering Structures* 2005; **27**:1715–1725.
14. Waszczyszyn Z, Ziemianski L. Neural networks in the identification analysis of structural mechanics problems. In *Parameter Identification of Materials and Structures*, Mroz Z, Stavroulakis GE (eds), Springer: Berlin, 2005.
15. Wu X, Ghaboussi J, Garret Jr JH. Use of neural networks in prediction of structural damage. *Computer and Structures* 1992; **42**(4):649–659.
16. Elkordy MF, Chang KC, Lee GC. A structural damage neural network monitoring system. *Microcomputers in Civil Engineering* 1994; **11**(1):67–76.
17. Pandey PC, Barai SV. Multilayer perceptron in damage detection of bridge structures. *Computer and Structures* 1995; **54**(4):597–608.
18. Masri SF, Nakamura M, Chassiakos AG, Caughey TK. Neural network approach to detection of changes in structural parameters. *Journal of Engineering Mechanics* 1996; **122**:350–360.
19. Kim SH, Yoon C, Kim BJ. Structural monitoring system based on sensitivity analysis and a neural network. *Computer-Aided Civil and Infrastructure Engineering* 2000; **15**(4):309–318.
20. Ni YQ, Wong BS, Ko JM. Constructing input vectors to neural networks for structural damage identification. *Smart Materials and Structures* 2002; **11**:825–833.
21. Ko JM, Sun ZG, Ni YQ. Multi-stage identification scheme for detecting damage in cable-stayed Kap Shui Mun Bridge. *Engineering Structures* 2002; **24**:857–868.
22. Xu H, Humar J. Damage detection in a girder bridge by artificial neural network technique. *Computer-Aided Civil and Infrastructure Engineering* 2006; **21**:450–464.
23. Jaynes ET. *Probability Theory*: *The Logic of Science*. Cambridge University Press: Cambridge, UK, 2003.
24. MacKay DJC. A practical Bayesian framework for back-propagation networks. *Neural Computation* 1992; **4**(3):448–472.
25. Bishop CM. *Pattern Recognition and Machine Learning*. Springer: Berlin, 2006.
26. Cybenko G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control*, *Signal and Systems* 1989; **2**:304–314.
27. Bishop CM. *Neural Networks for Pattern Recognition*. Oxford University Press: Oxford, 1995.
28. Bayes T. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London* 1763; **53**:370–418.
29. Laplace PS. Théorie Analytique des Probabilités. Courcier editor, 1812.
30. Jeffreys H. *Theory of Probability* (1st edn). Clarendon Press: Oxford, 1939; 3rd edn, Oxford University Press, Oxford, 1961.
31. Cox RT. Probability, frequency and reasonable expectation. *American Journal of Physics* 1946; **14**:1–13.
32. Vanik MW, Beck JL, Au SK. A Bayesian probabilistic approach to structural health monitoring. *Journal of Engineering Mechanics* 2000; **126**:738–745.
33. Papadimitriou C, Beck JL, Katafygiotis LS. Updating robust reliability using structural test data. *Probabilistic Engineering Mechanics* 2001; **16**:103–113.
34. Beck JL, Yuen K-V. Model selection using response measurements: Bayesian probabilistic approach. *Journal of Engineering Mechanics* 2004; **130**:192–203.
35. Yuen KV, Katafygiotis LS. Bayesian modal updating using complete input and incomplete response noisy measurements. *Journal of Engineering Mechanics* 2002; **128**(3):340–350.
36. Muto M, Beck JL. Bayesian updating of hysteretic structural models using stochastic simulation. *Journal of Vibration and Control* 2008; **14**:7–34.
37. Beck JL. System identification methods applied to measured seismic response. In *Proceedings of the Eleventh World Conference on Earthquake Engineering*, Acapulco, Mexico, Disc 4, Paper No. 2004, Lee , Espinosa-Aranda (eds), 1996.

38. Beck JL, Katafygiotis LS. Updating of a model and its uncertainties utilizing dynamic test data. *Proceedings First International Conference on Computational Stochastic Mechanics*, Computational Mechanics Publications, Boston, 1991; 125–136.
39. Beck JL, Katafygiotis LS. Updating models and their uncertainties-Bayesian statistical framework. *Journal of Engineering Mechanics* 1998; **124**:455–461.
40. Buntine W, Weigend A. Bayesian back-propagation. *Complex Systems* 1991; **5**:603–643.
41. MacKay DJC. Bayesian methods for backpropagation networks. *Model of Neural Networks III*, Chapter 6. Springer: Berlin, 1994; 211–254.
42. Lampinen J, Vethari A. Bayesian approach for neural networks—review and case studies. *Neural Networks* 2001; **14**(3):257–274.
43. Barber D. Bayesian methods for supervised neural networks. *Handbook of Brain Theory and Neural Networks*. MIT Press, Cambridge, 2002.
44. Neal RM. *Bayesian Learning for Neural Networks*, *Vol. 118*. Lecture Notes in Statistics. Springer: Berlin, 1996.
45. Lee HKH. *Bayesian Nonparametrics via Neural Networks*. ASA-SIAM Series on Statistics and Applied Probability, 2004.
46. Nabney IT. *NETLAB– Algorithms for Pattern Recognition*. Springer: Berlin, 2004.
47. Neal RM. Prior for infinite networks. *Technical Report CRG-TR-94-1*, Department of Computer Science, University of Toronto, 1994. Available from: http://www.cs.toronto.edu/~radford/ftp/pin.pdf.
48. MacKay DJC. Bayesian non-linear modeling for the prediction competition. *ASHRAE American Society of Heating*, *Refrigeration*, *and Air-Conditioning Engineers Transactions* 1994; **100**:1053–1062.
49. Sivia DS. *Data Analysis*: *A Bayesian Tutorial*. Oxford Science, Oxford, 1996.
50. Akaike H. A new look at the statistical identification model. *IEEE Transactions on Automatic Control* 1974; **19**:716–723.
51. Blum A. *Neural Networks in C++*. Wiley: NY, 1992.
52. Lawrence S, Giles CL, Tsoi AC. What size neural network gives optimal generalization? Convergence properties of backpropagation. *Technical Report UMIACS-TR-96-22 and CS-TR-3617*, Institute for Advanced Computer Studies, University of Maryland, 1996.
53. Lam HF, Yuen KV, Beck JL. Structural health monitoring via measured Ritz vectors utilizing artificial neural networks. *Computer-Aided Civil and Infrastructure Engineering* 2006; **21**:232–241.
54. Arangio S. Inference model for structural systems integrity monitoring: neural networks and Bayesian enhancements. *Ph.D. Thesis* on Structural Engineering, University of Rome 'La Sapienza', 2008.
55. Bontempi F. The structural analysis of the Messina Strait Bridge. *Proceedings of the International Conference on Bridge Maintenance*, *Safety and Management*, IABMAS'08, Seoul, Korea, 2008 [CD-ROM].
56. Ceravolo R, de Stefano A, Sabia D. Hierarchical use of neural techniques in structural damage recognition. *Smart Material and Structures* 1995; **4**:270–280.
57. Arangio S, Petrini F. Application of neural networks for predicting the structural response of a long suspension bridge subjected to wind actions. *Proceedings of the Third International Conference on Structural Engineering*, *Mechanics and Computation SEMC*, Cape Town, South Africa, 2007 [on CD-ROM].

DOI: 10.1002/stc