Detection of holes in a plate using global optimization and parameter identification techniques


UNIFEI – Federal University of Itajubá, IEM – Mechanical Engineering Institute, Itajubá, MG, Brazil
patricia.silva.lopes@gmail.com, ariosto.b.jorge@unifei.edu.br, ss.cunha@unifei.edu.br

1. Abstract
Several types of static and dynamic loads and the structural deterioration process can cause different types of structural damages. The distribution of stresses in a plate with damage can be modeled through boundary elements methods for elasticity. The damage can be characterized by changes on the structure, such as the presence of holes and cracks. In this work, the boundary element method is used as the direct problem, and two different techniques are used for the inverse problem, in order to localize and to identify damages on a structure. The first technique adopted is a global optimization technique using genetic algorithms. Using a global optimization technique, the global optimum of the system is larger chances of being obtained. The second approach is a parameter identification technique using artificial neural networks. The artificial neural networks are computational techniques that present a mathematical model to represent the human brain and to try to simulate the learning process of this brain. In this work, the presence and localization of holes in the structure is studied and a comparison is done between the results obtained using these two techniques for the inverse problem. Genetic algorithms and artificial neural networks are independent techniques to obtain the damage location, thus providing a means to verify the results.

2. Keywords: damage detection, boundary element method, optimization, genetic algorithm, artificial neural network

3. Introduction
The life time of any structure can be predicted through the correct determination of the damage. To determine the damage, the numerical modeling consists in a direct problem and an inverse problem. For the direct problem, a model is required to obtain information on the distribution of the quantity of interest throughout the structure, given the boundary conditions and the presence of the damage. For the inverse problem, a model is required for the procedure of locating the damage in the structure given some (partial) information on the quantity of interest at some particular locations (for example, where some sensors are placed). The boundary element method (BEM) is used in the numerical modeling of the direct problem. The model investigated in this work includes the elastostatics formulation, in which the quantities of interest are the interior point displacements and stresses. The damage detection problem can be considered as a problem of system identification or an inverse problem. In this work, the inverse problem of identifying the presence, location and size of damages, such as cracks and holes, in a plate structure is modeled using optimization and parameter identification techniques. Among the optimization techniques, the genetic algorithms (GA’s) are used. GA’s belong to the category of global optimization where the global optimum of the system has larger chances of being obtained. In the parameter identification techniques, the artificial neural networks (ANN’s) are very used. GA’s and ANN’s are independent techniques to obtain the damage location, thus providing a means to verify the results.

In the works presented in [1], [2] and [3], the BEM is used to model the direct mechanical problem numerically. The sequential quadratic programming (SQP), to determine the local optimum of the error (difference between the measured and the computed value), and the genetic algorithm (GA), to determine the global optimum of the same function are used in [1]. A backpropagation neural network (BPN) for the on-line identification of holes or cracks in composite structures is applied by [2]. In [3], evolutionary algorithms at the identification of cracks are used and the problem is formulated as the minimization of the difference between the measured and computed values of displacements or stresses at selected boundary nodes. The work presented in [4] proposes a method of inverse analysis using a BPN and the computational mechanics, matching the of the finite element method with the boundary integral equation. The BPN uses a backpropagation learning rule where the adjustment of weights from input to hidden layers is made by back-propagating the errors of the neurons of the output layer to the hidden layers.

In this work, a BEM model was built for the plate with an internal hole, considering some boundary conditions (traction on the external surface of the plate, for example). The stresses at internal holes in the plate are assumed null. The software MATLAB® was used for the development of the damage detection program.

4. Direct Problem: Boundary Element Methods
The boundary element method (BEM) is a numerical procedure well adapted for the modeling of a structure with damage. In this method, the distribution of the quantities of interest in the domain is obtained from the information of the distribution of certain quantities in the boundary. Thus, the problem is described based on what happens in its boundaries, reducing the dimension of the problem and simplifying numerically the treatment. The use of a simple direct boundary element problem for the distribution of a potential field in a domain was already considered in [5]. In this work, the model investigated includes the elastostatics formulation (see references [6] and [7] for elastostatics). When modeling the damage detection problem by means of an analysis of the elastic response of the structure under excitation, perturbations in the expected response imply in the presence of damage. Thus, the damage in the structure will characterize its behavior, static or dynamic.

5. Inverse Problem: Optimization and Parameter Identification Techniques
The inverse problem might be modeled by means of optimization and parameter identification techniques. The damage is simulated by the presence of small holes in the domain, and the goal is to obtain size and location of the damage. The direct method (BEM)
provides some (partial) information for any desired point in the domain. Without the hole, the distribution of the displacement and stresses is known a priori. If a small hole is included, this information is unknown and must be obtained numerically from the BEM solution. The BEM for the elasticity problem can be used. In this case, boundary conditions for the displacement and traction will be provided. The BEM for elasticity (in a 2D problem) provides two pieces of information at a single interior point – one normal stress and one shear stress. But this information cannot be used directly in the optimization problem, as it depends on the system of coordinates being used, or on the normal direction of the cutting plane that passes through the point of interest. Therefore, a choice is made to adopt the stress invariants of the stress tensor at the point of interest – in 2D, the mean stress and the octahedral stress – as the vector field to be analyzed and used in the optimization problem.

5.1. Optimization using Genetic Algorithms

The genetic algorithm (GA) is a search method based on the processes of natural evolution. This method works with a set of possible solutions for a given problem, composing the initial population. In other words, GA uses multiple points to search for the solution rather than a single point in the traditional gradient based optimization method [8]. In this algorithm the problem variables are represented as genes in a chromosome (each chromosome is also denominated an individual of the population). Starting from an initial population, the individuals with better adapted genetic characteristics have higher chances of surviving and reproducing. According to [3], the GA’s are methods that do not depend on the choice of the initial point, increasing the chances of obtaining the optimum global of the system. So that the population is diversified and maintain certain acquired adaptation characteristics by the previous generations, the genetic operators (selection, crossover and mutation) can be used. These operators transform the population through successive generations, extending the search until arriving to a satisfactory result.

In this work, GA is used to find the optimal solution by using a functional defined as the difference between the measured (simulated) values of the local difference in the mean stress (between the undamaged plate and the plate with the damage) and the values of the same differences in mean stress calculated at the same points by the code (assuming several different locations and sizes for the “numerical” damage). This functional corresponds to the fitness function of the GA. The minimization of this fitness function allows the damage detection program to find the unknown parameters of the damage. The functional formulation is shown at Eq. (1)

\[
J_j = \frac{1}{2} \sum_{i,s} [\text{measured}_i - \text{calculated}_i]^2
\]

being \( n \) the number of internal points \( i \) (“sensors” placed in the plate) where the differences are evaluated; \( \text{measured}_i \), the vector of simulated values for the differences obtained using BEM, for a given damage; and, \( \text{calculated}_i \), the vector of differences in mean stress calculated by the code for each individual \( j \).

As mentioned, GA starts with a population, representing a set of possible solutions for a given problem. To solve the damage detection problem, each chromosome (individual) of the population can be assembled according to the vector presented in Eq. (2):

\[
c = [g_1, g_2, g_3, g_4, ..., g_n]
\]

where:

- \( g_1 \) – first gene representing the \( x \)-coordinate of the hole;
- \( g_2 \) – second gene representing the \( y \)-coordinate of the hole;
- \( g_3 \) – third gene representing the hole radius;
- \( g_4 \) – fourth gene and the subsequent genes representing the measures of the mean stress difference between the undamaged plate and the plate with the damage.

As an example, Fig. 1 represents three possible configurations of chromosomes. While the location and size of the hole varies, the number and location of the sensors remains the same, for all chromosomes. The information on the quantity of interest is collected at these sensor locations for all cases.

![Figure 1. Plate with a hole: three possible configurations for the chromosomes.](image)

5.2. Parameter identification using Artificial Neural Network

The artificial neural networks (ANN’s) are computational techniques that present a mathematic model to represent the human brain and to try to simulate the learning process of this brain. These ANN’s are made by small units called neurons. A biological neuron has several entrance ramifications known as dendrites (input terminals), a cellular body where the nucleus with the whole genetic information is located, and an axon (output terminal). The communication among neurons is made in the contact area between two neurons through the transmission of nervous pulses.

A simplified model of the biological neuron is represented by an artificial neuron. According to Fig. 2, \( x_1 \) to \( x_n \) represent the \( n \) input terminals (dendrites), \( y_1 \) to \( y_m \), the \( m \) output terminals, \( w_{ij} \) to \( w_{nj} \) are the weights in the inputs, representing the synapses (communication) among the neurons, and the threshold function represents the function in the output of the neuron. Every input sign is multiplied by a weight, indicating the influence of these signs in the output of the neuron. Then, a weighted sum is made, producing an activity level. If this level exceeds a given threshold, the information is passed for other neurons. In this case, the neuron is active.
An ANN is formed by the interconnected neurons whose inputs can be obtained from the outputs of other neurons or from input nodes. Different configurations of the artificial neuron can be made to develop different network topologies [10]. The network topologies can be defined for the layer number, amount of neurons in the layers and the connection type among the neurons. Among the existent configurations, the ANN can be feedforward or feedback. At the feedforward neural networks, the neurons are interconnected in layers, but the flow of data only occurs in a direction [9]. At the feedback neural networks, there is at least a feedback cycle, in other words, a neuron receives the information of neurons of the previous layer and of a subsequent layer. The first layer at the network is the input layer, the last layer is the output layer and the layers between the input and output layer are the hidden layers. More complex problems can be implemented due to the use of the hidden layers; however the network learning becomes more difficult.

After defining the structure of the ANN, an iterative process of weight adjustment of this network is made. This process is known as training process. Following the training, the ANN learns how to proceed for other input data in the problem domain. An ANN learns when a generalized solution for a class of problems is reached, in other words, when a given input leads to a target output. The training or learning algorithms differ in the manner how the weights are modified. When an external agent is used to indicate to the network an acceptable solution of the problem, the learning is said to be supervised. In this kind of training, the input and output vectors are known in the problem. The lack of the external agent leads to an unsupervised learning.

In this work a backpropagation neural network (BPN) is used, through a feedforward configuration and the backpropagation learning algorithm. The backpropagation algorithm carries out a supervised learning where the desired outputs are given as part of the training vector. In the training stage, this algorithm operates in a sequence of two steps. First, a sign is presented to the input layer of the network and this sign is propagated through the network until an answer is produced by the output layer. In the second step, the adaptation stage of the network is initiated. In this stage the obtained output is compared to the desired output for the input sign, producing an error. Finally, the error is passed back through the network for the weight adjustment among the layers to produce the correct output [11].

6. Numerical Results and Discussion

For the elasticity problem, a BEM model was built for the plate with a hole with the boundary conditions illustrated in Fig. 3(a). Two discretizations were implemented for the external contour, a coarse mesh with 12 constant elements and a fine mesh with 48 constant elements. Fig. 3(b) shows the discretization for the case of 48 elements in the outer boundary and 12 elements in the hole, as well as the position of the nine sensors. At the present work, the sensors were uniformly distributed on the plate and no positioning study of the sensors was performed. The plate was simulated with shear modulus equal to 94,500 MPa and a Poisson’s ratio for plane strain equal to 0.1.

![Figure 3. Elasticity plate model: (a) dimensions, loading, and boundary conditions. Insert shows a stress-free hole; (b) boundary discretization (fine mesh) and sensor locations. Insert shows hole discretization.](image)

Fig. 4 shows the influence of the numerical errors due to the BEM discretization in the optimization results. A comparison is made for the optimization results (using 10 runs of a GA algorithm) and for two meshes (a coarse mesh and a fine mesh) using the elastostatics formulation for the plate shown in Fig. 3(a). In Fig. 4 is an illustrative result for the mean values of the error in the
location \((x\,\text{and}\,y\text{ coordinates})\) and size \((\text{radius } r)\) of a central hole.

![Graph showing percent error in location and size](image)

**Figure 4.** Mean values for the error in location and size of a center hole.

The initial population of GA and the ANN input/output data were configured as follow: holes with radius equal to 0.05 cm, 0.10 cm and 0.15 cm was considered to assemble the data. For each radius, the \(x\) and \(y\)-coordinate of the center of the hole was varied of 0.5 cm to 5.5 cm with a step size of 0.5 cm. Then, 121 different positions for each radius in the plate were simulated and the respective values of the difference in the mean stress at the 9 internal points (sensor location; see, Fig. 3(b)) were founded by means of BEM and these values were stored for \textit{a-posteriori} processing.

6.1. Analysis of the results obtained from the genetic algorithm

For the initial population of GA, before using the data directly, the values of the difference in the mean stress were normalized, taking in consideration the maximum value of this difference. The values of \(x\) and \(y\)-coordinate of the center of the hole and its radius were also normalized, considering the respective maximum value. After that, the initial population with 363 individuals can be formed.

The plots of the location and size of the holes obtained from 10 different runs of the GA is presented in Fig. 5. The GA, due to its own randomness, generates a different optimal solution every time it is run; nevertheless the GA results present a tendency to be concentrated near the “real” hole. Fig. 5(a) shows the results for a central hole; Fig. 5(b) shows a hole located at \(x = 2\) and \(y = 2\); and Fig. 5(c), a hole located at \(x = 5\) and \(y = 2\). The radius of each plot was considered equal to 0.12 cm. The real position of the hole is represented in continuous line and the results found by the GA in non continuous lines.

![Graph showing plots of hole locations](image)

**Figure 5.** Plot of holes located using mean stress: (a) for a central hole; (b) for a hole at \((2,2)\); (c) for a hole at \((5,3)\).

The stopping criterion of the GA was configured for the generation number equal to 100. The initial population has 363 individuals, the crossover fraction was set as 0.95, and then the mutation fraction is 0.05. The elitism considered that 10 individuals survive in the next generation. The function that performs the crossover was heuristic function, considering a value of ratio equal to 0.9 (this value represents how far the child is from the better parent). The mutation function was uniform function where each gene has a probability 0.03 of being mutated. Another GA parameter configured was migration. The migration represents how individuals move between subpopulations. In this case, the best individuals from one subpopulation replace the worst individuals in another subpopulation, only doing a copy from these individuals. The individuals are not removed from the source subpopulation. In this work, the migration fraction was set as 0.20, the direction of migration was set as "both" directions and 20 generations pass between migrations of individuals between subpopulations. Finally, the selection function was the roulette selection where a section of the roulette wheel corresponds to an individual at the population, and each section is selected with a probability equal to its area.
6.2. Analysis of the results obtained from the artificial neural network

The ANN's simulate the non-linear behavior between the values of the local difference in the mean stress (between the undamaged plate and the plate with the damage) and the hole parameters (location and size). Information regarding the difference in the mean stress is supplied in the input of the network, besides, the parameters of the hole are supplied in the output of the same network. Holes of different sizes and at different places can be part of the data supplied to the net. Having defined the input and output data, the next step is to build the network and, then, this network can be trained. Finally, the network can be tested for other data of difference in the mean stress, obtaining as answer, the location and size of the hole.

As according to the initial population of GA, the values of the difference in the mean stress and the hole parameters were also normalized, before using these values directly. After training the network with these data, this network was tested for a hole of radius 0.12 cm in each position ((3;3) cm, (1;1) cm, (1;5) cm, (5;1) cm, (5;5) cm and (3;5) cm). Fig.6 shows the results obtained, considering 9 sensors on the plate and, whose distribution is presented in Fig.3(b). The network was configured with 100 neurons in the input layer, 50 neurons in the hidden layer and 3 neurons in the output layer. The choice for other parameters of the ANN was:

- Threshold function in the input and hidden layers: tan-sigmoid transfer function;
- Threshold function in the output layer: linear transfer function;
- Training function: gradient descent with momentum and adaptive learning rate (traingdx);
- Error goal: \(1 \times 10^{-3}\);
- Number of epochs: 5000;
- Learning rate: 0.01.

In Fig.6, the results present a small area of uncertainty near the “real” hole, moreover, the size of the hole was obtained with good accuracy. These results were obtained more quickly than in the case of using GA (as a global optimization technique). For this reason, the solution of a damage detection problem through the ANN (as a parameter identification technique) is also known as an online identification. An advantage of the use of ANN in regard to the GA is that, after training the network, holes with different sizes and in different locations can be tested without running the damage detection program again.

The function traingdx is a backpropagation network training function that combines adaptive learning rate with momentum training. This function updates the weights and biases of the network only after the entire training set has been applied to the network (batch mode). An adaptive learning rate allows the performance of the steepest descent algorithm to improve, attempting to keep the learning step size as large as possible while keeping learning stable. Moreover, momentum training allows a network to respond not only to the local gradient, but also to recent trends in the error surface. Without momentum a network may get stuck in a flat local minimum.
7. Conclusions

In this work, an inverse problem of identifying damages in a plate structure was solved using both optimization and parameter identification techniques. A genetic algorithm (GA) was used as the optimization technique, and an artificial neural network (ANN) code was used as the parameter identification procedure. Also, an elastostatics formulation for the boundary element method (BEM) was used as the direct model in this inverse problem. To investigate the influence of mesh refinement, different meshes were used for this BEM model for the direct problem, and the numerical results obtained showed an improvement in the accuracy of the results for the location and size of the damage, when a fine mesh was used. Regarding the inverse problem, the GA is a technique difficult to be implemented, due to the choice of the configuration of the parameters that each problem requires. This choice depends on the realization of a great number of experiments and tests. Moreover, the GA also presents a high computational cost due to the several evaluations of the fitness function. The damage detection code using GA can only find a region for the probable occurrence of the hole, as this algorithm generates a different optimal solution every time it is run. Thus, only confidence intervals, for the different parameters being identified, can be obtained. On the other hand, the damage detection problem using parameter identification techniques can be solved more quickly than in the case of using global optimization techniques. Also, in this work, the solution of the problem through ANN presented good results for the several parameters being identified. In particular, the size of the hole was obtained with good accuracy, and the location of the hole was given by a fairly small area of uncertainty near the “real” hole, for the several cases tested. The optimization and the identification techniques adopted in this inverse problem can be used concomitantly, as independent procedures to identify the presence of a hole on the plate, thus providing a means to verify the numerical results obtained for the location and size of the damage in the structure, increasing the confidence in the damage identification results.

8. References