



A. García-Crespo · B. Ruiz-Mezcua · D. Fernández-Fdz
R. Zaera

Prediction of the response under impact of steel armours using a multilayer perceptron

Abstract This article puts forward the results obtained when using a neural network as an alternative to classical methods (simulation and experimental testing) in the prediction of the behaviour of steel armours against high-speed impacts. In a first phase, a number of impact cases are randomly generated, varying the values of the parameters which define the impact problem (radius, length and velocity of the projectile; thickness of the protection). After simulation of each case using a finite element code, the above-mentioned parameters and the results of the simulation (residual velocity and residual mass of the projectile) are used as input and output data to train and validate a neural network. In addition, the number of training cases needed to arrive at a given predictive error is studied. The results are satisfactory, this alternative providing a highly recommended option for armour design tasks, due to its simplicity of handling, low computational cost and efficiency.

Keywords Neural network · Numerical simulation · Steel armour · Ballistic impact

1 Introduction

Over the last 50 years, scientists and engineers have devoted more and more effort for solving problems relating to the impact of solids. Until a couple of decades

ago, the analysis of these problems turned almost exclusively to collisions of solids, of simple geometry and low impact energy, with limited plastic deformation, given the mathematical complexity of the equations involved in these phenomena and the limited information available on the mechanical behaviour of materials at high-speed deformation [1]. The evolution of powerful computers and instrumental techniques has provided new test data which have helped to solve many problems of impact. Models are now available to assist engineers in the study of car crashes, impacts on turbine blades, ballistic protection and impacts on structures or vehicles.

Now focusing on the ballistic range, there are numerous systems which, during their service, can suffer the impact of objects moving at high speed (over 500 m/s). The most characteristic examples are found in the military field although over the last decades, this type of problem has become of interest in civil applications, in which structural elements are required to absorb the projectile energy so that it does not damage critical parts of the global system. *Weight* is a key factor in the design of protections against impact—whether on vehicles, civil or military aircraft, or personnel of the security and defence corps. Hence the interest in developing high-performance ballistic materials able to save weight [2, 3], which, however, means higher costs that are prohibitive in certain applications. When weight is not the primary design concern and cost benefits are welcome, *steel* would be one of the most adequate materials, due to its high hardness, capacity of energy absorption by plastic deformation and its low cost.

The design of structures subjected to impact was traditionally carried out empirically, relying on real impact tests, each using the given projectile/target. The mathematical complexity of solving the equations that rule the impact phenomenon and the relative ignorance of the mechanical behaviour of the materials at high strain rates discouraged any simulation of the problem.

Therefore, the experimental results, with a statistic treatment, gave curves of probability of perforation

A. García-Crespo · B. Ruiz-Mezcua
Computer Science Department, University Carlos III of Madrid,
Avda. de la Universidad 30, 28911 Leganés, Madrid, Spain

D. Fernández-Fdz · R. Zaera
Department of Continuum Mechanics and Structural Analysis,
University Carlos III of Madrid, Avda. de la Universidad 30,
28911 Leganés, Madrid, Spain

A. García-Crespo · B. Ruiz-Mezcua · R. Zaera (✉)
Research Institute “Pedro Juan de Lastanosa”,
University Carlos III of Madrid, Avda. de la Universidad 30,
28911 Leganés, Madrid, Spain
E-mail: rzaera@ing.uc3m.es

depending on the velocity of impact (Fig. 1). These curves defined different values of *ballistic limit velocities*, V_P , the critical value above which a probability of perforation P exists.

This method is reliable but the results are valid only for the conformation tested, and they are not readily extrapolated; any variation of the impact velocity or the characteristics of the projectile or the target invalidates the test data. It is also a costly procedure; testing in these conditions requires sophisticated installations and equipment. It becomes even more costly if the aim of the tests goes beyond the discovery of the strength quality of a target or the characteristics of the deformation of a projectile capable of perforating an armour or destroying a structure. To obtain information on the deformation of the target or on the position of the projectile during penetration, ultra-rapid photography—either optical or X-ray—is required, and this again increases the cost of the tests. With a view to extend the validity of real-fire tests to other projectile-target conformations, some so-called *semi-empirical* techniques were developed; these proposed algebraic equations, with a certain theoretical base, calibrated from a large number of experimental data and allowed an interpolation or even extrapolation of results.

The need for design tools to simulate this process triggered the development in recent years of a large number of models of different types; all of them belong to two families: those of *analytical modelling* and of *numerical modelling*. Thus, the use of expensive experimental tests has been relegated to the final stage of the design. All the previous stages can be covered by the use of simulation tools.

In *analytical modelling*, a series of simplifying hypotheses is assumed in the physical mechanisms that define the behaviour of the solids, to produce by means of simple equations a model that enables each problem to be solved in a few minutes or even seconds. These

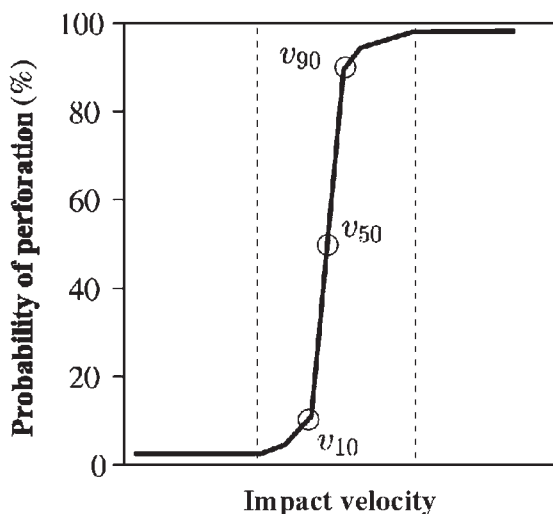


Fig. 1 Probability of perforation versus impact velocity for a given projectile/target set

methods are not so precise as the numerical methods, but they are so rapid that a great number of calculations can be made in a short time (see the classic references [4, 5]). Also known as *engineering models*, these tools are useful in the design of armour plating; they are not only quick and easy to use but also give detailed information about the process of penetration. The representation of the materials in these models is expressed in simple equations using experimental parameters such as the elastic modulus, stiffness or failure deformation. The equations used in continuum mechanics are greatly simplified by the introduction of hypotheses obtained from the analysis of the global behaviour of the phenomenon. The approximation obtained with this type of model depends largely on its complexity. The introduction of a larger number of variables into the model gives a more precise representation of the physical phenomena involved in the impact although this complicates the solution of the differential equations and delays the operation. So a compromise has to be made between the capacity of the model to predict the experimental results and the need to avoid complexity in the calculations.

Using *numerical simulation* leads to a result of an impact problem through the complete solution of the differential equations of the mechanics of continuous media, much more reliable than that obtained with the analytical model. A number of commercial programmes of finite elements or finite differences are available, and these are very successful in simulating impact problems. However, their capacity of close approximation to reality depends more on the constitutive equations used for the materials than on the errors inherent in the numerical method itself [6]. The impact produces temperature increase due to irreversible processes—plastic flow, shock waves—and deformations at very high speed that can be simulated correctly only by equations of state and constitutive equations whose expression is often unknown. The main drawback is the amount of time required to solve each impact problem, usually several hours or even days of modelling. Its high cost and long execution time required for the simulation of a single case of impact hamper a wide number of calculations during the process of design, which are needed for the optimisation of the protection.

Taking account of these difficulties, a *neural network* has been designed for the prediction of the response of steel armours. The neural networks have become a centre of intense activity in engineering during the past years [7] due to their power of predicting results and the easy handling at user's level. In these networks, the biological model of learning and computation is emulated, permitting the network to modify its own structure (weight matrix) and adapt it for achieving an algorithm of execution. This algorithm is built up from the characteristics extracted from the cases whose solutions are known. Training is done with these algorithms. In the neural network it is necessary to select the significant variables, in both the input network and the output. The expected result must be related to the

output variable and the input related to the variables that are of value in the physical problem. In this way, the network then predicts the response of a system to a group of input data.

Among the different artificial neural networks, the *multilayer perceptron* (MLP) is by far the most commonly applied in mechanics; even the *radial basis function network* (RBFN) has been used successfully in this field. Both feed-forward networks are efficient tools for classification and prediction in non-linear problems. The main difference between both of them lies in the activation function of the hidden neurons. In MLPs, the log-sigmoid or hyperbolic-tangent-sigmoid function is used; this type of function divides the input variability space into a lower half with low output and an upper half with high output. In RBFNs, hidden neurons have a Gaussian function, selective to a small range of the variability space. Therefore, RBFNs effectively model a large output gradient over a small region, and MLPs capture the overall trend over the whole problem domain. There is also a consensus on the higher training speed of RBFNs over MLPs.

In the problem dealt with, the input data include the geometrical parameters of the solids involved (radius and length of the projectile, thickness of the steel protection) and the impact velocity, while the response of the system is the velocity and the residual mass of the projectile after perforation. The computational cost required to train the network in the problem herein considered is small in any case. Moreover, based on previous experience acquired through the analysis of this type of problems, the output is expected to be a smooth and monotonic function of each input data, without oscillations or large local variations. Thus an MLP was chosen as predictive tool. Since the tool is quick and easy to use, it allows a simulation of a large number of impact problems in the early stages of design and a great saving of the time and money that would be spent if the earlier methods were used.

2 Numerical simulation

Several impact problems were generated from which some of them were used for the training and others for the validation of the results obtained by the network after learning. Modelling and simulation were done with the commercial software ABAQUS/Explicit v6.4.1. [8], a finite element explicit code widely used in highly non-linear dynamic simulation.

A dynamic analysis was carried out with integration times varying from 120 to 180 μs , enough to guarantee, depending on the impact velocity and the geometry of the target and projectile, the complete arrest of the projectile or perforation of the steel plate.

The impact velocity V was generated randomly between 500 and 1,200 m/s, covering the usual range of values for this kind of problem. The projectile was modelled as a cylinder of radius R and height L and the

target as a square plate of thickness H and edge Y . The values of these geometric parameters were also generated randomly with a uniform probability density function keeping the following restrictions to maintain reasonable geometrical proportions:

$$3 \text{ mm} \leq R \leq 8 \text{ mm},$$

$$4R \leq L \leq 8R,$$

$$Y = 2L,$$

$$L/6 \leq H \leq L/3.$$

The generation of the mesh is parametric, taking as parameters the geometrical input data of the network (L, R, H) in order to achieve a rapid generation of all the cases, due to the vast number that have to be analysed. Eight-node trilinear hexahedral elements with reduced integration to a single Gauss point and hourglass control (C3D8R in ABAQUS notation) were used for the regular mesh with a characteristic element length $L_C = 1.2 \text{ mm}$.

The boundary conditions of the target do not influence the problem, given the short duration of the process, so a clamping condition was adopted on the target sides to reduce the degree of freedom of the problem.

A pure master–slave contact algorithm (Fig. 2) was used to model the contact between the projectile and target during the impact. The master surface is formed by the lower sides of the projectile elements while the slave surface is formed by all the nodes of the target. A friction coefficient of $\mu = 0.2$ was fixed between the two surfaces.

A 1006 SAE steel was chosen for both the projectile and the target. For the numerical modelling of this material, a thermoviscoplastic isotropic hardening equation was considered, dependent on the plastic equivalent strain $\bar{\epsilon}_{pl}$, plastic equivalent strain rate $\dot{\bar{\epsilon}}_{pl}$ and temperature T . These two last variables must be considered in this type of phenomena in which strain rates are very high (10^4 s^{-1}), and the plastic work generates a notable adiabatic increase of the temperature in the metallic material. A Johnson–Cook equation [9] was used:

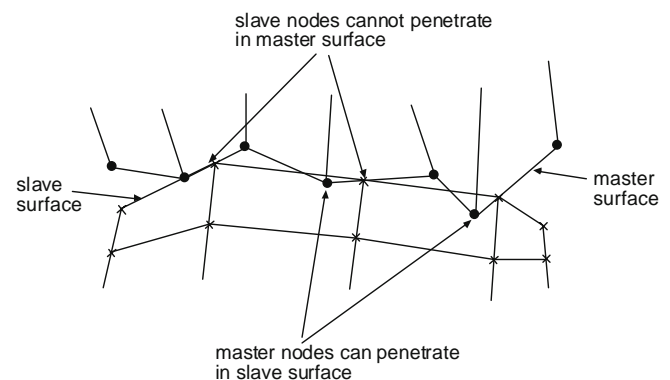


Fig. 2 Contact algorithm: master and slave surfaces

$$\bar{\sigma}(\bar{\varepsilon}_{pl}, \dot{\bar{\varepsilon}}_{pl}, T) = (A + B\bar{\varepsilon}_{pl}^N)(1 + C \text{Ln} \frac{\dot{\bar{\varepsilon}}_{pl}}{\varepsilon_0})(1 - \bar{\theta}^M), \quad (1)$$

$\bar{\theta}$ being the homologous temperature defined as:

$$\bar{\theta} = \frac{T - T_0}{T_M - T_0} \quad (2)$$

where T_M and T_0 are the melting temperature of the steel and a reference temperature (commonly taken as the room temperature), respectively. A , B , N , C , ε_0 and M are material constants determined experimentally. The first and second factors introduce the material hardening by plastic strain and plastic strain rate, respectively, while the third factor introduces the material softening by adiabatic heating. The values of the parameters of the thermo-elastoplastic constitutive equation for the steel, as well as the density, are given in Table 1.

In impact problems, in which strains are very high, the treatment of the excessive distortion of the elements in Lagrangian meshes—the one adopted in this simulation—is critical. Therefore an erosion criterion was considered to delete each element when the equivalent plastic strain at the integration point exceeds a threshold value $\bar{\varepsilon}_{failure}$. This value was fixed at 1.5 for the projectile and 2.5 for the target to obtain results consistent with experimental observations. When elements in contact between the target and projectile are deleted, new contact surfaces are introduced in the contact algorithm to guarantee the continuity between them (Fig. 3). This avoids numerical problems resulting from this distortion, and the mass loss in the projectile, a typical phenomenon in impact problems, is properly modelled.

Figure 4 shows a projectile before and after the impact as one of the 60 cases simulated. When perforation of the target took place (Fig. 5), the projectile mesh was post-processed to obtain the residual mass and velocity. An application was developed to calculate these values from the nodal coordinates and velocities of the projectile nodes at the end of the simulation.

Table 1 SAE 1006 steel properties

Property	Value
Density	7,850 kg/m ³
Young modulus	207 GPa
Poisson ratio	0.3
Specific heat	4.5 kJ/kg K
Thermal expansion coefficient	22.9×10 ⁻⁶ K ⁻¹
Quiney–Taylor coefficient	0.7
A	350 MPa
B	227 MPa
N	0.36
C	0.022
ε_0	1 s ⁻¹
M	1
T_0	30°C
T_M	1,400°C

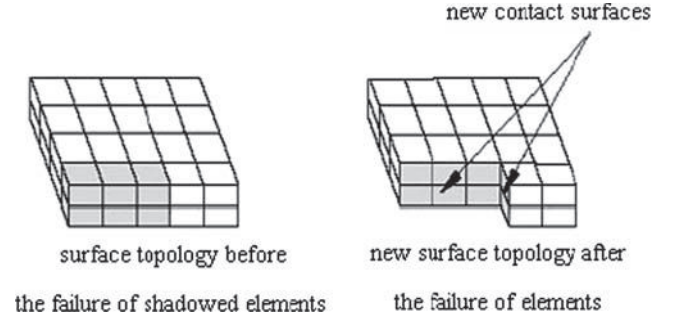


Fig. 3 Contact surfaces evolution for element failure

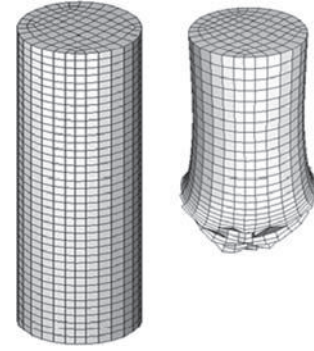


Fig. 4 Projectile before (*left*) and after (*right*) the impact

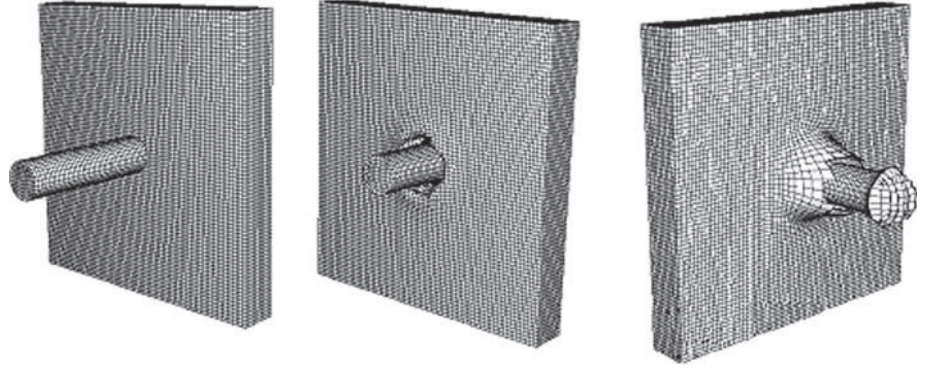
3 Neural network approach

Many different neural networks have been established over the past 20 years, with Perceptron, Hebbian, Kohonen and Hopfield networks [10–13] being some of the best known. These neural networks have been widely applied in the field of mechanics of structures and materials to predict solutions of problems such as bending in elastoplastic beams, estimation of vibrations in real buildings or damage detection in steel beams [14]. Neural networks have also been applied in fracture mechanics to estimate fracture parameters in concrete [15] and in non-destructive testing to detect cracks in brittle materials [16]. In all these applications the input data were obtained by either experimentation or numerical simulation.

3.1 Input data

Before defining the topology of the neural network, the number of inputs must be decided. In an impact problem several parameters can be selected as input data: kinetic energy, velocity, radius (R), length (L), mass or the ratio L/R —all of them related to the projectile—and the thickness of the target (H). Kinetic energy is correlated with velocity and mass. The ratio L/R and mass are correlated with R and L because the density is constant. It is important therefore to choose the appropriate

Fig. 5 Sequence of simulation in case of perforation: front views (*left* and *centre*) and final rear view (*right*)



parameters: velocity of the projectile is a key parameter in impact problems; on the other hand, the projectile cross-section area influences the perforation process of the target, so the radius will also be taken and consequently the length of the projectile. Hence, the number of inputs for the neural network will be limited to four: three geometrical parameters (R , L , H) and the impact velocity (V).

3.2 Multilayer perceptron network with backpropagation

An MLP backpropagation (BP) network with one hidden layer was proposed for this work. It is a neural network commonly used in mechanical problems such as those described in [14], a supervised, continuously valued, multi-input and multi-output feed-forward that follows a gradient descent method, altering the weight by an amount proportional to the partial derivative of the error with respect to the weight in question. The backpropagation phase of the neural network alters the weights so that the error of the network is minimised. This is achieved by taking a pair of input/output vectors and feeding the input vector into the net which generates an output vector (feed-forward process). The output vector is then compared to the supplied output vector, thus gaining an error value. The error is then passed back through the network (backpropagation process), modifying the weights. If the same set of input/output vectors was presented to the network, the error would be smaller than that previously found.

The perceptron topology is shown in Fig. 6, in which the following neurons are used:

- Four input layer neurons associated with the four input variables (R , L , H , V) with an identity transfer function
- Four hidden layer neurons with hyperbolic tangent transfer function
- Four output layer neurons, two associated with the perforation output variable (two neurons whose output is complementary were considered since this practice helps the learning task) and the other two associated with velocity and residual mass, respectively, all of them with a sigmoidal transfer function

To estimate the number of the hidden layer neurons, the guideline proposed by Tarassenko [17] has been applied

$$J \sim (IK)^{0.5}, \quad (3)$$

I , J and K being the number of neurons in the input, hidden and output layers, respectively.

The error reported to the supervised learning procedure is the squared Euclidean distance between the network's output and the desired response.

4 Results

The neural network was developed with the commercial code *Neurosolutions for Excel v4*, a software that works in an Excel environment. The number n of cases required to train the network has been estimated using the guideline proposed by Tarassenko [17]:

$$W \leq n \leq 10W, \quad (4)$$

W being the network parameters that must be adjusted in the training, which is obtained by

$$W = (I + 1)J + (J + 1)K = 40. \quad (5)$$

Due to the high computational cost of the numerical simulations, 50 randomly chosen cases were supplied to the network, in order to train it through a supervised learning process. Once the network was trained, it was validated with 10 additional cases generated for this purpose.

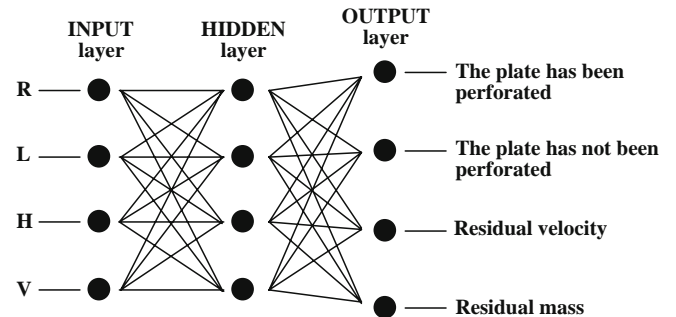


Fig. 6 Topology of the perceptron used

Table 2 depicts the input variables and the perforation results obtained by means of the network. It shows how the predicted values adjust perfectly to the expected ones, except cases 2 and 7. In case 7, a perforation value of 0.7901 was obtained, intermediate between 0 (arrest) and 1 (perforation), which does not allow the determination with accuracy of the impact result. To analyse this result, in Fig. 7 one can see the “Perforation = YES” value, keeping the projectile and target geometry of case 7 and varying impact velocity in the range 350–850 m/s. From an interpretation analogous to that used for the probability curves of perforation (Fig. 1), which are characteristic of the experimental tests, the ballistic limit velocity would be found between 500 and 670 m/s. Numerical simulations carried out afterwards confirm that at 500 m/s the target arrests the projectile, while at 670 m/s the projectile perforates the target at a residual velocity of 362 m/s. Therefore, in cases close to the ballistic limit velocity, the neural network would anticipate values of “Perforation = YES” intermediate between 0 and 1.

As for case 2, the neural network predicts perforation whilst the numerical simulation confirms the opposite. In this case, the network would provide a range between 600 and 830 m/s for the limit velocity of perforation, and the finite element tool predicts the ballistic limit velocity at around 830 m/s. The cause of this deviation would have to be sought in the singularity of the geometry not only of the projectile, but also of the target: R , L and H acquire in this case minimum values in the group of the 50 cases considered for the network training.

The results of the test for the other two output variables, velocity and residual mass, are presented in Figs. 8 and 9 for the cases resulting in perforation. The trends predicted by the network are seen to be conveniently close to the numerical simulation results.

4.1 Optimisation of the training

In a second phase, experiments were made to determine the minimum number of patterns (and in consequence, numerical simulations) to obtain a given error in the

predictions of the neural network. This serves to reduce the high computational cost of the finite element simulations performed to obtain training data.

To develop these experiments, an application was developed which chose randomly a set of training and testing patterns from the total of 60 cases from the numerical simulation. For the first experiment (A in Fig. 10), the application takes 50/10, 45/15, 40/20... training/testing patterns, trains the network and calculates the error in the prediction of the output “Perforation = YES”. This operation is repeated 100 times, each point in Fig. 10 being the average error. The experiment was performed twice with similar results, confirming that the network does not exhibit over-training.

To eliminate the influence of the number of testing cases in the pattern, a second experiment was performed (B in Fig. 10), now keeping the number of testing cases to 10, independently of the number of training cases in the pattern. Again, the results followed those obtained in experiments A , this confirming the consistency of the results. One can see a clear trend in the error obtained by the neural network for the output “Perforation = YES”, which is independent of the number of patterns used to validate the network. This trend can be fitted by the exponential curve:

$$\text{Error} = 28.31e^{-0.04X}, \quad (6)$$

X being the number of training patterns used for the learning process (correlation coefficient = 0.89).

5 Conclusions

As shown here, the MLP could be considered an alternative tool for the design of armours against impacts. The network is reliable in predicting the projectile arrest, outdoing finite element simulation tools in the lower computational cost, once the training has been carried out. With regard to the prediction of velocity and residual mass, the achieved values are close enough, and the network trend is clearly similar to that obtained by numerical simulation.

Table 2 Results of “Perforation = YES”

Case	H (mm)	R (mm)	L (mm)	V (ms ⁻¹)	Perforation = YES (1)	
					Numerical simulation	Neural network
1	7	6	26	1,133	1	0.9990
2	4	3	14	804	0	0.9977
3	11	6	47	787	1	0.9957
4	13	6	44	1,094	1	0.9999
5	16	8	53	1,073	1	0.9998
6	10	8	53	420	0	0.0006
7	7	7	33	601	1	0.7901
8	11	7	35	613	0	0.0004
9	11	5	40	863	1	0.9918
10	5	3	21	769	1	0.9923

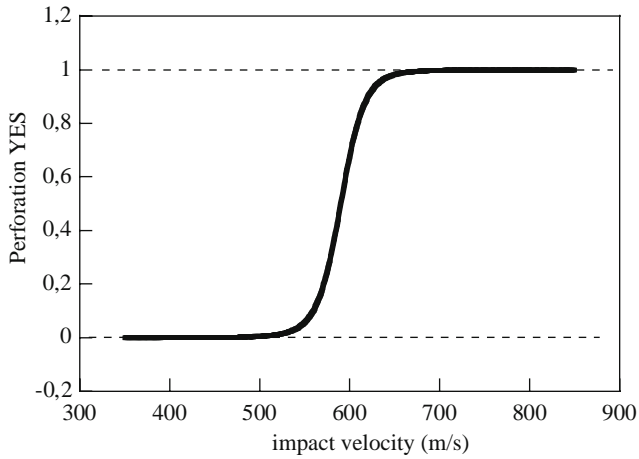


Fig. 7 Results of “Perforation=YES” for case 7. Velocity 350–850 m/s

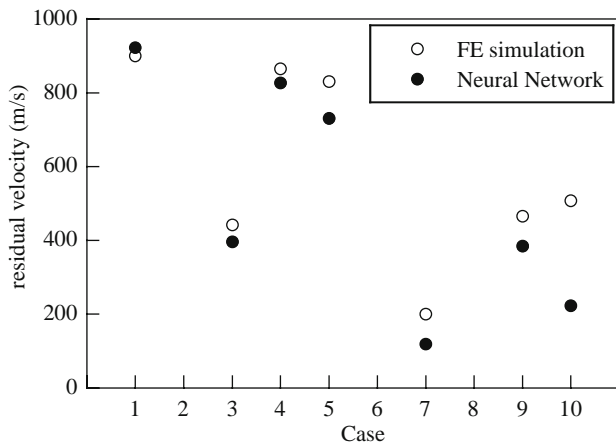


Fig. 8 Residual velocity in the perforation cases predicted by numerical simulation and neural network

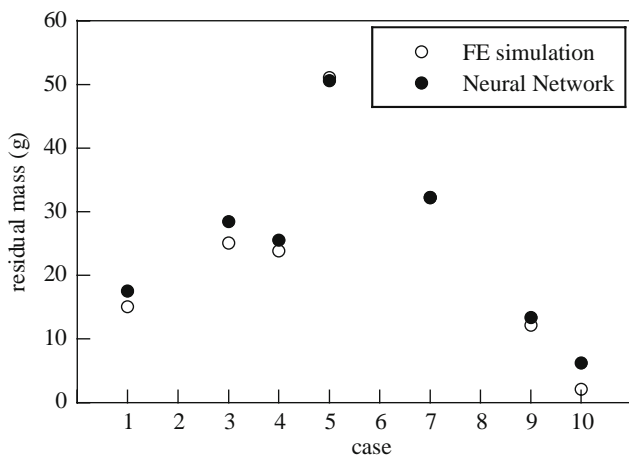


Fig. 9 Residual mass for the perforation cases predicted by numerical simulation and neural network

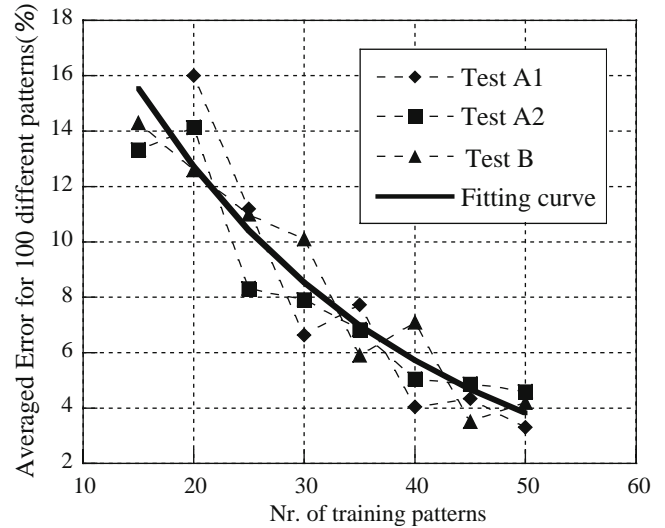


Fig. 10 Error versus number of training patterns

It is important to note the small number of training cases needed to get a low error in the predictions: with 40 training patterns (consequently 40 finite element numerical simulations), the network obtains a mean error less than 7%. In spite of this repetitive low number of training data, the network does not exhibit over-training. Since the chosen network (MLP) is one of the simplest topologies, the results show the feasibility of using neural networks in the prediction of the behaviour of armours against high-speed impact of projectiles.

Acknowledgements This research was done with the financial support of the Comunidad Autónoma de Madrid under Project GR/MAT/0507/2004.

References

1. Goldsmith W (1960) Impact. The theory and physical behaviour of colliding solids. Edward Arnold Publishers, London
2. Zaera R, Sánchez-Gálvez V. (1998) Analytical modelling of normal and oblique ballistic impact on ceramic/metal light-weight armours. *Int J Impact Eng* 21:133–48
3. Arias A, Zaera R, López-Puente J, Navarro C (2003) Numerical modeling of the impact behavior of new particulate-loaded composite materials. *Comp Struct* 61:151–59
4. Awerbuch A, Bodner SR (1974) Analysis of the mechanics of perforation of projectiles in metallic plates. *Int J Solids Struct* 10:671–684
5. Ravid M, Bodner SR (1983) Dynamic perforation of viscoplastic plates by rigid projectiles. *Int J Eng Sci* 21:577–591
6. Anderson CE (1987) An overview of the theory of hydrocodes. *Int J Impact Eng* 5:33–59
7. Minsky M, Papert S (1969) *Perceptrons*. MIT Press, Cambridge
8. ABAQUS/Explicit v6.4 User's manual (2003) ABAQUS Inc., Richmond
9. Johnson GR, Cook WH (1983) A constitutive model and data for metals subjected to large strains, high strain rates, and temperatures. In: *Proceedings of seventh international symposium ballistics*, The Hague, The Netherlands, pp 1–7
10. Kirraine DE (1990) Machine learning, Training and Development. *J* 40:24–29

11. Rumelhart DE, McClelland JL (1986) Parallel distributed processing. MIT Press, Cambridge
12. Rumelhart DE, McClelland JL (1988) Parallel distributed processing VI: foundations. Bradford Books, New York
13. Hopfield JJ (1987) Neurons with graded response have collective computational properties like those of two state neurons. Proc Natl Acad Sci 81:4947–4960
14. Waszczyszyn Z, Ziemiański L (2001) Neural networks in mechanics of structures and materials—new results and prospects of applications. Comput Struct 79(22–25):2261–2276
15. Ince R (2004) Prediction of fracture parameters of concrete by artificial neural networks. Eng Fract Mech 71(15):2143–2159
16. Liu SW, Huang JH, Sung JC, Lee CC (2002) Detection of cracks using neural networks and computational mechanics. Comput Methods Appl Mech Eng 191(25–26):2831–2845
17. Tarassenko L (1998) A guide to neural computing applications. Arnold/NCAF, London