Proceedings of the XII International Symposium on Dynamic Problems of Mechanics (DINAME 2007),
P.S.Varoto and M.A.Trindade (editors), ABCM, Ilhabela, SP, Brazil, February 26 - March 2, 2007

Investigations in the hybrid FEM/Neural Network crash test simulation of vehicle structures

Adam Machnik, Emmanuel Pagnacco, and J. Eduardo Souza de Cursi

LMR, EA 2838, INSA-Rouen Avenue de l'Universitée BP 8 76801 Saint-Etienne du Rouvray CEDEX

Abstract: The Finite Element Method has itself asserted as a standard tool for the crash test numerical assessment of complex vehicle structures. However, despites the development of computer power, the numerical computation remains still very costly and time consuming activity. The problem thereby is the creation of simplified (reduced complexity) models destined for the analysis. This work considers the use of coupled FEM and Artificial Neural Network (ANN) models: ANN based models are used to replace some parts of the whole complex FEM model of the vehicle, providing a significant simplification of the initial model. Two possible uses of ANN are considered herein in order to replace a part of the whole structure. The first one consists in an ANN used for the identification of model behaviour. The second ANN produce an evaluation of the accelerations field at the interface between the two substructures. The approaches are studied and validated on simple but significant nonlinear unidimensional situation. Then, the efficiency of the hybrid model is demonstrated on a real situation when used in conjunction with an explicit crash test code. Initial results show that using the hybrid FEM/ANN models may represent an interesting alternative for crash test simulation of complex vehicle structures.

Keywords: Neural Network, Identification, Hybrid Models, Crash simulation

INTRODUCTION

Crash is a strongly non linear dynamic phenomenon which takes a central place in vehicle design. Namely, it is significant when analyzing the dynamic behavior of automobile structures. The numerical simulation of crash test may considerably speed up the design procedure and has been considered in many studies. In this framework, the computational cost is a severe limitation: the pioneer works have considered simple one-dimensional mass/spring simplified models(see for example Mentzer, 1982, Mentzer, 1992, Cheva et al. 1996). These models are computationally cheap, but they do not represent the vehicle CAD geometry and their practical use is not straightforward. Nowadays, the evolution of computational facilities makes that crashe simulations may be more conveniently handled with commercial Finite Element (FE) explicit sofwares (Hallquist and Tsay, 1999; PAM-CRASH, 2000; RADIOSS, 2000,...). Nevertheless, an important limitation of the current models remains the mesh size (which quickly becomes very large): in fact, the CAD models used involve a detailed geometry, since a single model must be used for the analysis of all types of crash (frontal, side, offset or even "roll-over").

More recently, another approach has been proposed for the reduction of FE Models (FEM) destined to crash simulations has been proposed, using simplified or complete analysis of isolated parts of the vehicle, which are subsequently assembled (Drazetic et al., 1993; Cornette et al., 1998). This work considers an analogous approach, but our main goal is to construct a hybrid complete model where some parts are simplified parts while others remain detailed. More precisely, we focus on the particular problem introduced by the use of Artificial Neural Network (ANN) in the simulation of substructures of the global vehicle: ANN must be able to reproduce a nonlinear mechanical behavior in a very efficient calculation (much faster than FEM).

Crash tests consist generally in the projection of the vehicle against a rigid wall or a deformable barrier with an initial speed V_0. In practice, differents speeds going from 10km/h to 65km/h are used. Each speed needs a complete crash simulation. Thus, it appears that a significant improvement may be obtained by a reduction of the number of speeds for which a complete calculation is requested. This naturally suggests the use of a hybrid FEM/ANN model generated as follows:

1. the complete vehicle FEM simulation of the crash is carried out for some selected initial crash velocities. This generates a database containing information about the dynamic behavior of the vehicle.

2. the database is used for training the ANN model.

3. the hybrid FEM/ANN model is used to simulate the crash test for the velocities not included in the database.

So, the improvement introduced by the hybrid model grows with the number of analysis in the third stage upper defined since it has only a partial cost of the complete FE analysis (FEA), carried out in the first stage.

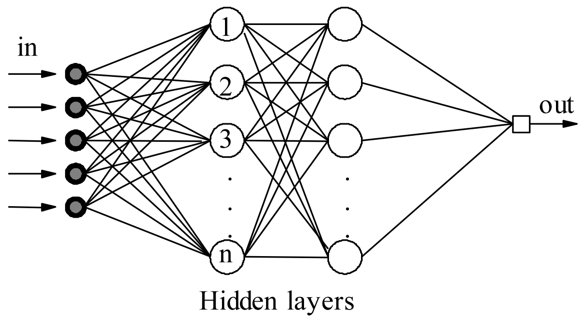


Figure 1 – “Multi-layered” ANN with 5 input neurons, 2 hidden layers of *n* neurons and 1 output neuron.

In the sequel, we briefly present the guiding principles of the operation of ANN networks. In order to generate an ANN able to reproduce a typical signal function $\mathbf{a}_s(t)$, we consider a simple but significant situation involving a one Degree Of Freedom (DOF) system. The analysis of this simple situation leads to the definition of the general principles for the generation of an efficient ANN model able to produce the temporal sequence prediction of $\mathbf{a}_s(t)$ based on its past and initials conditions. Then, we analyze the behaviour of the proposed ANN when additional ecxcitation is considered, in order to take into account interface accelerations which intervenes when couplingo to other substructures must be performed. Finally, an efficient application of the hybrid model on a real situation is performed, involving coupling with an explicit crash test code.

ARTIFICIAL NEURAL NETWORKS

Artificial Neural Network are considered with a growing interest in the field of computational mechanics. The basic principles of ANN have been stated in the 40-s starting by analogy with a human nervous system. A large number of different ANN models concerning complex processes may be found in the literature. There is a strong analogy between identification problems and ANN training, which is exploited in this work: if an arbitrary nonlinear function in time domain is given, we may generate ANN in order to identify a simplified substructure having a dynamical behavior which corresponds to an approximation of these function. The main advantage of ANN in such a case is the use of a small number of parameters when compared to other methods.

We do not introduce here the various existing ANN models or algorithms and we focus on the generation of a “black box” which replaces part of the complete FEM. ANN consist in several basic unities - usually called neurons - which exchange information by weighted connections. ANN are mainly characterized by the type of the units used and the topology of the connections - which is called the network architecture. The ANN performance is close connected to this architecture, which generally must be adapted to the task under consideration. In this work, we use a “multi-layered” architecture, illustrated in figure 1, which is one of the most popular network architectures (Cichocki and Unbehauen, 1994).

The information flux between neurons involves weights weights w_{ij} which characterize the importance of the connections. Training an ANN consists in the determining these weights, by using optimization algorithms - this process is also called ”learning”. We use in this work a supervised training which requests the knowledge of all inputs and their corresponding desired outputs. When the input data is introduced in the network, ANN perform a calculation and produce an output. The difference between this output and the desired output gives an measurement of the error to be reduced by a modification of the weights, performed by the classical “ gradient back-propagation” algorithm (Rumelhart, 1986). This algorithm is based on the standard gradient descent methods: the derivatives of the error with respect to the weights generate a gradient and the opposite direction leads to a diminution of the error.

ANN BASED PREDICTION OF AN ACCELERATION TEMPORAL SIGNAL

From the theoretical stand point, ANN are able to “predict the future” of an arbitrary signal from known information about its past (see, for instance, Box and Jenkins, 1976 and Wong, 1991). For the prediction of the temporal acceleration signal $\mathbf{a}_s(t)$ of a mechanical system, an ANN defines a nonlinear function $y(k) = \Psi(X(k))$ where $y(k) = \{a_s(t),\ t = (k+1)\Delta t, k = N, N-1, ..., n\}$ is the desired output, while $X(k) = [\, y(k-1),\, y(k-2), ..., \, y(k-n)\,]$ is the history of the accelerations. We underline that this approach implicitly supposes that the present value of the temporal sequence is connected to the N preceding values. The architecture diagram for training the ANN is illustrated in figure 2 (the operator z^{-1} stands for accessing the preceding values: $z^{-1}y(k) = y(k-1)$).

In order to evaluate the ability of the ANN for the prediction of the acceleration signal, let us consider a simple (1D) but non linear dynamic system (figure 3), where the rigidity is a function of displacement x: $K(x) = 1000.e^{-10x^2} + 100$

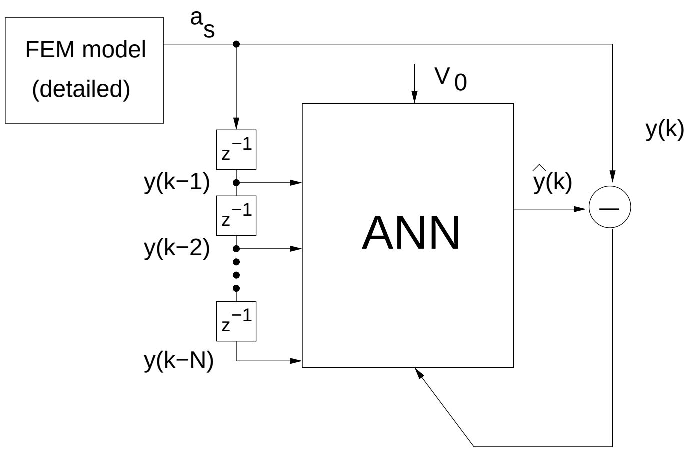


Figure 2 – Diagram of the ANN training.

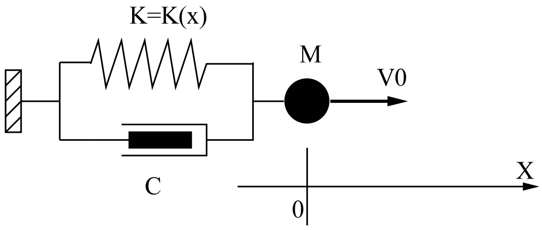


Figure 3 – 1 D mass/spring system that describes crash behaviour.

N/m. With $C = 5$ N/m/s and $M = 1$ kg, the acceleration response for two initial velocities $V_0 = 8$ and $V_0 = 14$ m/s is illustrated in figure 4.

The training is supervised, since the fixed input $y(k-1)$, $y(k-2)$, $y(k-3)$ is imposed and $y(k) = \mathbf{a}_s(t)$ is sought at the ANN output. The training strategy consists of repeating randomly the presentation of the 2 simulated functions until they are conveniently approximated by the output of the ANN. This prevents the ANN from an eventual dependence on the order of introduction of the data. Different tests have been performed, involving the learning of these two functions by ANN with 2 and 3 layers of neurons. The target was a prediction error threshold of 5% . The best results obtained correspond to a success rate of only 3/20 and concern a 2 layers network having 10 neurons.

This situation is often found when using ANN and methods of prevention may be found in the literature. For instance, supplementary variables may be introduced in the ANN input or "recurrent" networks may be used. This last strategy consists in presenting the history of the signal which has been already carried out by the network, in its current state of training, instead of the target signal history at the network input (Narenda and Parthasarathy, 1990).

Recurrent learning is considered as more adapted to the simulation of dynamic systems: for instance, it should be

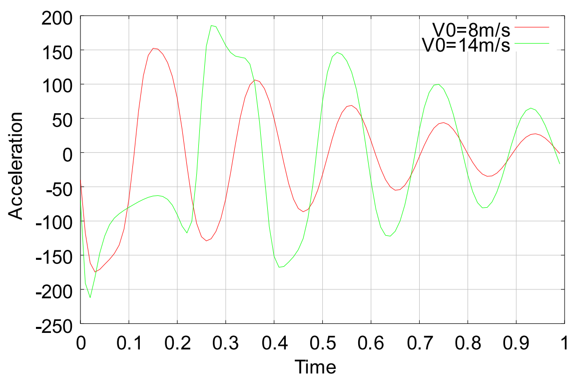


Figure 4 – Accelerations functions of the 1D system for 2 initial velocities.

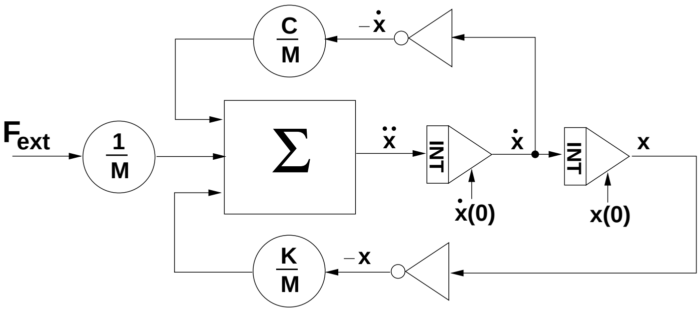


Figure 5 – Block diagramm representing the equilibrium of 1D nonlinear system.

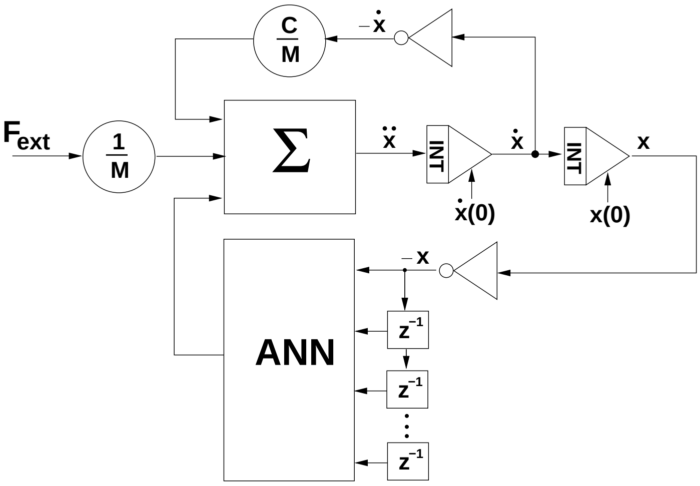


Figure 6 – Optimal ANN architecture.

noted that our simple system can be described by 3 generalized inputs $x(t)$, $\dot{x}(t)$ and, in general $F_{ext}(t)$, for one output $\ddot{x}(t)$. For a unit mass, the motion equation of this system can be written:

$$\ddot{x}(t) = F_{ext} - C\dot{x}(t) - Kx(t) \tag{1}$$

and it is thus possible to represent it by a block diagram, shown in figure 5.

One of the main characteristics of this diagram is the presence of the two closed loops representing two forces - their sum is referred as the “recall force”. Thus, the identification of this system may be carried out by using convenient approximations of these forces. In fact, the structural parameters (C and K) are taken into account in the “recall force”. However, the ANN used to simulate the system must perform all the mathematical operators present in the diagram 5, including the double integration of the acceleration signal and some inversions. These two last operators do not introduce significant difficulties, since only the stiffness of the system is nonlinear. So, it appears that the architecture may be simplified in order to let to the ANN the representation of the internal recall forces, while the simple mathematical operators may be let to the circuit. Even if the proposed diagram in figure 5 represents the state of the system at the instant t, it is clear that the system parameters are also history dependent. In order to fully represent the dynamic behavior of the part, the history of the loads must be also furnished to the ANN. Moreover, during the simulation of the crash, recall forces depend on the position of the structure. Therefore, the history of displacements should also appear at the input furnished to the ANN. An ANN architecture corresponding to these ideas is proposed in figure 6 where the external forces are kept to allow a generalization thereafter (although those are not present in the system now). It should be noted that the “closed loop” in this diagram corresponds to the recursion introduced in recurrent network, what connects the mathematical procedure to a physical characteristic of the system.

Using such a strategy, the convergence is easily achieved for different layer architecture. For instance, a 2 layers network having 10 neurons each have now a full success rate (20/20) within a 1% threshold error prediction. The following series of experiments gives us an evaluation of the convergence of the training and the capacities of generalization of the considered ANN. To show the ANN performance, prediction for the 1D system accelerations over the time 0-1 second are superimpose to the finite difference simulation on figure 7. On this figure, 9 graphs are presented for 9 different initial velocities: V_0 =2, 6, 8, 10, 12, 14, 16, 18 and 20 m/s respectively. The straight line concern the ANN prediction while

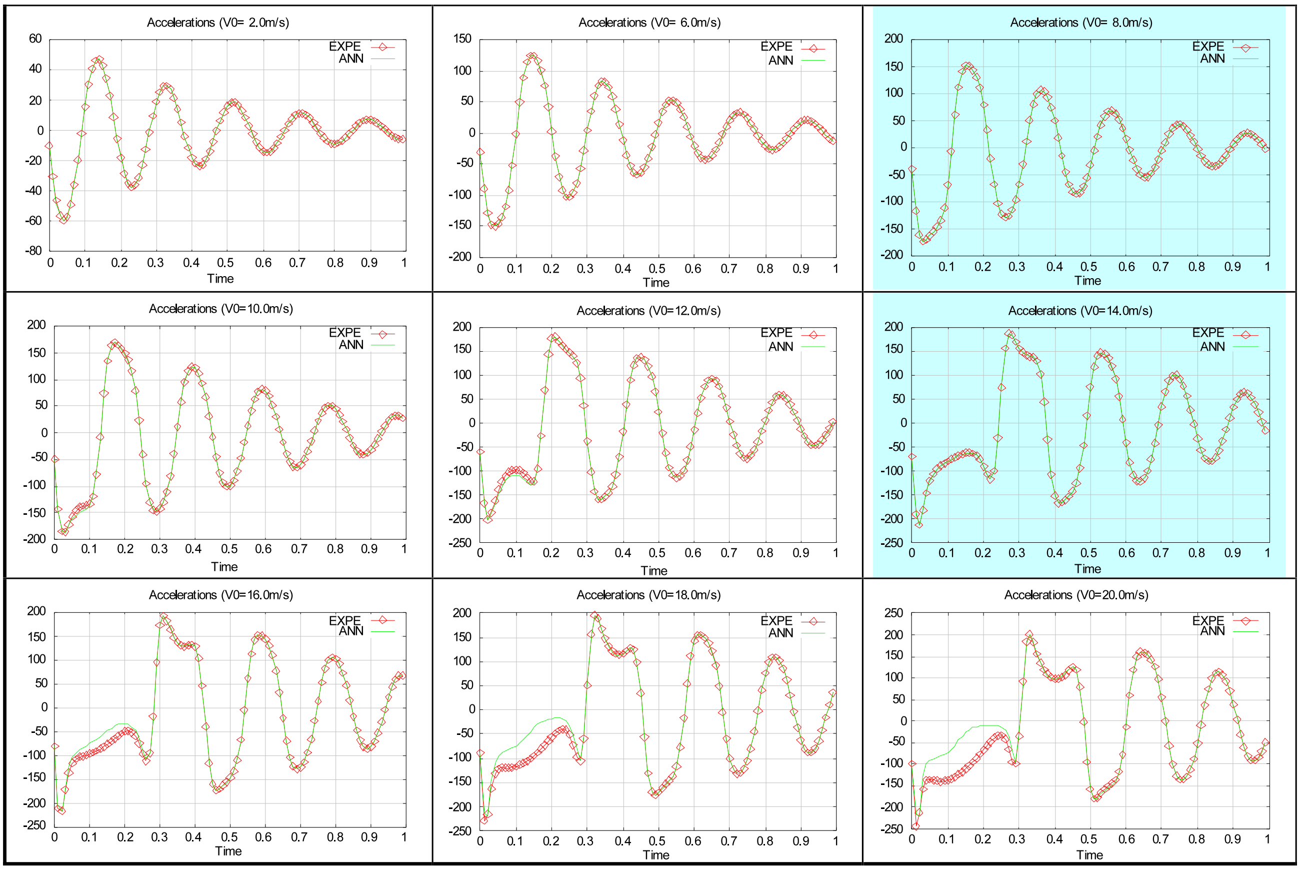


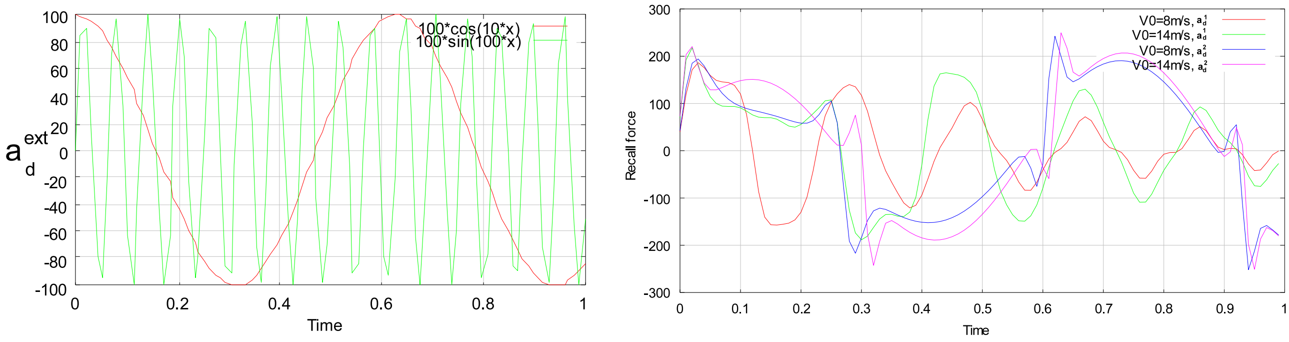
Figure 7 – Discrepancies between ANN responses and exact solutions when varying initial velocities for the free 1D system.

the line with circle marker concern the finite difference simulation. Since the ANN is train with the finite difference simulation for the 8 and 14 m/s initial velocities, good agreement is achieve in this case. We notice that the ANN behaves perfectly in the case of interpolation *i.e.* for initial velocities V_0 =10, 12 m/s. It performs well also the extrapolation for the low velocities: V_0 =2, 4, 6 m/s. However, in the case of extrapolation with higher velocities (V_0 =16, 18, 20 m/s) the ANN provides erroneous answers, but as soon as the velocity goes down up to the learned level, the ANN provides a correct response. Another experiments show that the results are much better when we use the high velocity curves V_0 =14 m/s and V_0 =18 m/s as a training base of the ANN. It also comes out from this series of experimentations that the training time of the ANN is much longer (8 times) for high velocity curves. This is certainly explained by the fact that the ANN must learn how to simulate the behavior of the nonlinear system in a wider domain of amplitudes. In order to decrease the learning time, we may increase the number of neurons: the first results obtained show that learning time is much longer for the networks whose size is not large enough (2x6, 2x7), but it decreases quickly when the number of the neurons becomes higher.

ANN SUBJECT TO EXCITATIONS

The strategy for obtaining a simplified FE crash model is based on the coupling of the two substructures, where the less loaded is replaced by ANN. Thus, it is necessary to evaluate the ANN behaviour not only for different initial conditions, but also for different loads, since this is essential in order to coupling substructures. Moreover, such an approch unifies the couplings realised by load or acceleration transfer. In this work, substructure FEM model to be simplified is considered as a system having an input $F_{ext}(t)$ or $a_d(t)$ and an output $a_s(t)$. According to the previous strategy, the external force $F_{ext}(t)$ is furnished by measurements, while displacements and velocities are obtained by numerical integration of the accelerations. Then, it is possible to get an approximation of the recall force function $g(x(t), \dot{x}(t))$ by the ANN.

To evaluate the capacity of the previously proposed ANN architecture, the previous 1D system (using the same parameters) is loaded by 2 external arbitrary acceleration $a_d^1 = 100\sin(100 \cdot t)$ and $a_d^2 = 100\cos(10 \cdot t)$ (see figure 8a) . Then, 4 functions from each 2 initial velocities $V_0 = 8$ and $V_0 = 14$ m/s and the 2 external accelerations constitute the data base of training (figure 8b). Here, the ANN has 2x20 neurons and the training converges to an acceptable level of error (10^{-5}) after 81100 presentations.



a) excitation functions b) recall force functions

Figure 8 – External excitation and recall force functions for the 1D system.

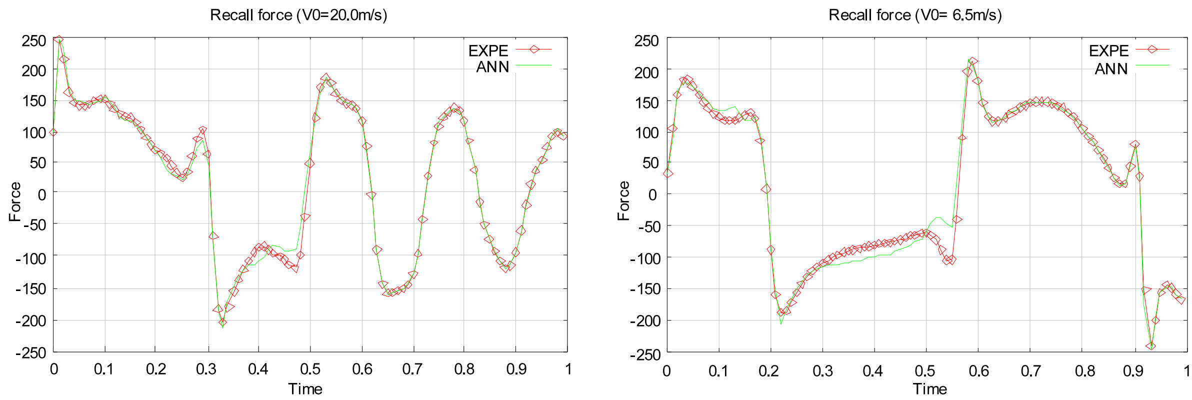


Figure 9 – Discrepancies between ANN responses and reference signal for $V_0 = 20$ m/s and a_d^1(left), and $V_0 = 6.5$ m/s and a_d^2 (right) for the 1D system.

Figure 9 shows the discrepancies between the network response and the reference signal for the worst cases ($V_0 = 20$ m/s for a_d^1 and $V_0 = 6.5$ m/s for a_d^2) showing that a good prediction confidence could be assigned to ANN responses when varying initial velocities for the a_d^1 and a_d^2 external loading. In order to check the abilities of this ANN, the system has been excited by the 3 random loads a_d^3 , a_d^4 and a_d^5 (illustrated in figure 10) and the differences between the ANN responses and exact results are presented in figure 11. These graphs show a good aggreement.

HYBRID FEM/ANN MODEL

The preceding considerations show that the ANN is able to reproduce a nonlinear crash for an one DOF system. Therefore, it is necessary to consider a more realistic configuration where the ANN and the FEM interact at once for a multiple DOF on the interface. Using one large multi-port ANN to replace the entire part of the model is not effective to calculate, since ANN apply badly to the problems of large size (in a number of inputs, outputs and neurons). This difficulty is known as “scaling effect” in the literature (Haykin, 1994). Thus, we consider the use a number of independent, small size and single input ANN for each interface DOF. Training data for the corresponding network are obtained directly at these DOF.

In order to accelerate the learning, we have introduced the training “in order”, where the training is sequentially performed for the nodes and the weights determined for node N are used as starting values for the training of concerning node N+1. This technique introduces some continuity between neighbour nodes (their weights are often very similar), while leaving the ANN free to independently reproduce the behavior for each interface DOF. The diagram illustrating this strategy is presented in figure 12.

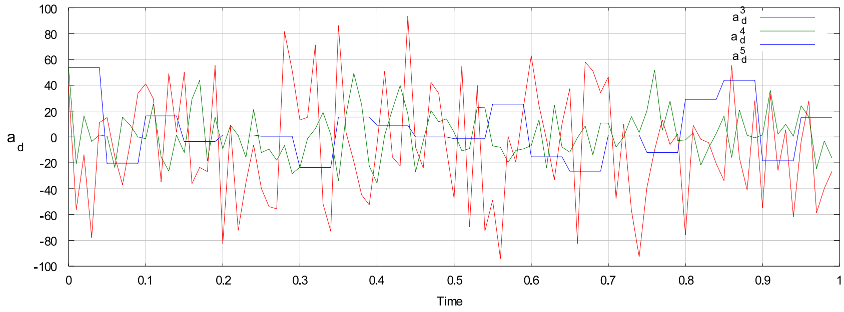


Figure 10 – External random loads applied for the ANN model of the 1D system validation.

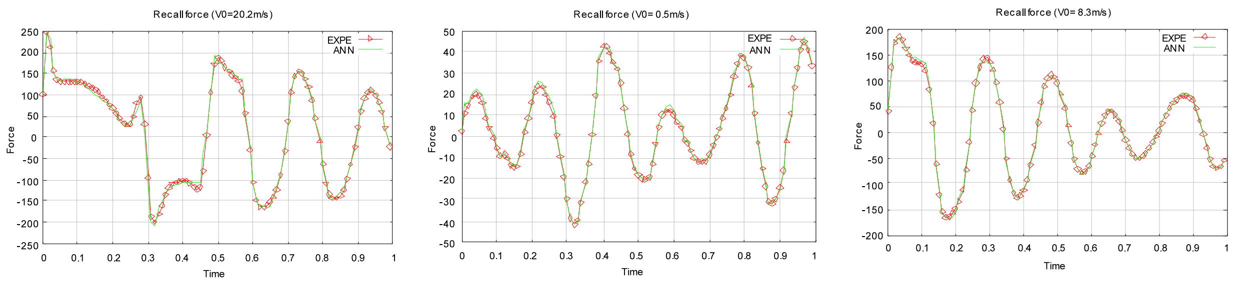


Figure 11 – Discrepancies between ANN response and exact results for the 3 random loadings a_d^3 , a_d^4 and a_d^5 of the 1D system.

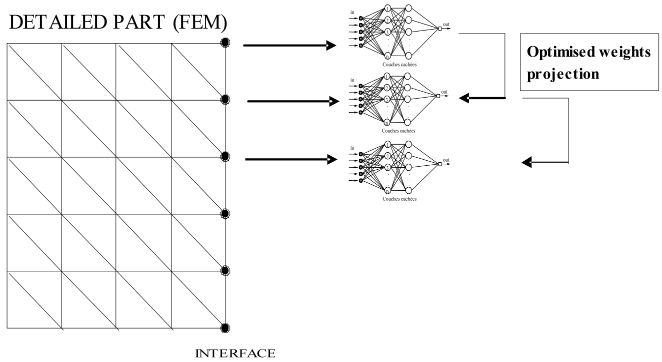


Figure 12 – Trained in order representation using one ANN per DOF.

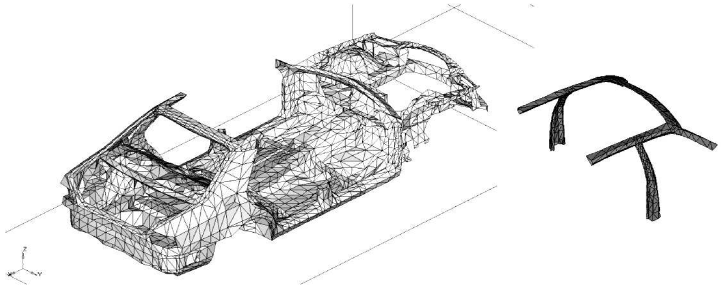


Figure 13 – Explode substructures view for the body-in-white.

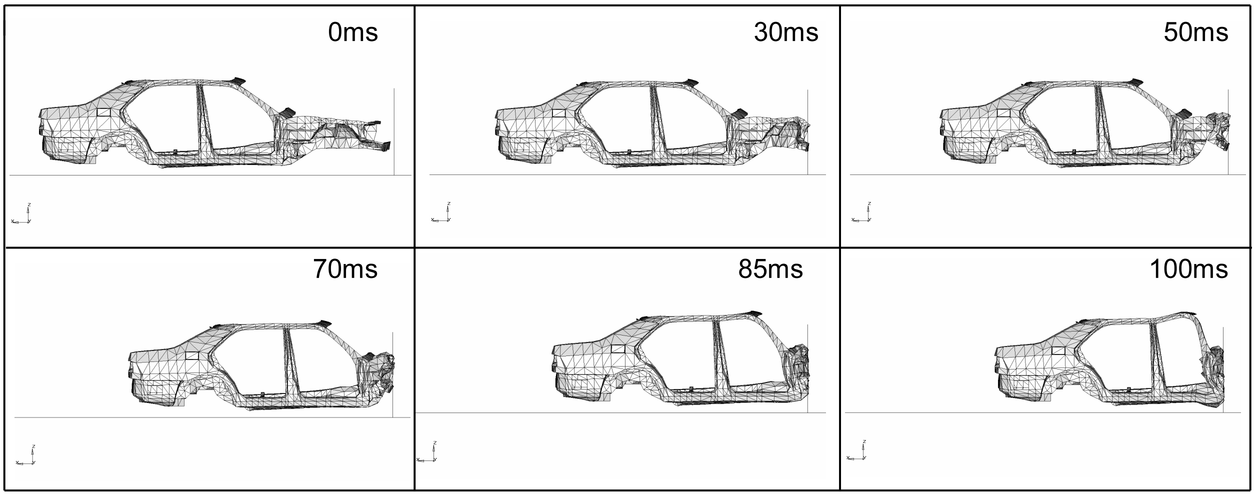


Figure 14 – Crash kinematics of the body-in-white structure for $V_0^{(4)} = 15$ m/s.

CRASH SIMULATION APPLICATION

The application concerns the crash simulation of a body-in-white structure vehicle against a rigid wall with four different initial velocities $V_0^{(1)} = 8$, $V_0^{(2)} = 12$, $V_0^{(3)} = 14$ and $V_0^{(4)} = 15$ m/s. The FEM mesh of the vehicle is composed of 8114 triangular elements. It is subdivided into two substructures at the roof interface (figure 13). This decomposition is intuitive since the roof is linked to the remainder of the structure only by four thin pillars. Nevertheless, its total suppression greatly influences the structure deformation. Model parameters are a Young modulus of $E = 210$ GPa, a Poisson ratio $\nu = 0.3$, a yield stress $\sigma_y = 220$ MPa for a density $\rho = 7900$ kg/m^3 and a thickness of 1.5 mm.

The used explicit FEA takes into account a two slopes plastic material behaviour with an isotropic hardening and a Coulomb law. The coupling is achieved with methods often used for the parallelization of the explicit simulation codes (Fahmy and Namini, 1994). By reasons of efficiency, the communication between substructures is carried out at every instant t by the exchange of accelerations $\mathbf{a}_d(t)$ and $\mathbf{a}_s(t)$ at each interface degree of freedom. Thus, a partition through the elements of the physical border is requested in this situation (they are duplicated during the decomposition process). Then, time integration of the dynamic equilibrium could be carried out independently for each sub-domain and the coupling is made at each time step after evaluating accelerations, by their transfer between the substructures.

The first step consists in a FEA for two initial velocities of $V_0^{(2)}$ and $V_0^{(4)}$ with the complete FE model. The kinematics of the crash for $V_0^{(4)}$ is illustrated in figure 14. Then, each interface node functions of roof are learned by the ANN to produce an hybrid FEM/ANN model. Training is carried out for each 102 ANN composed of 2x20 neurons (34 nodes at the interface).

In order to validate the procedure proposed, we compare the crash pulses resulting from the complete model and hybrid model FEM/ANN on the tunnel for the chosen velocities in figure 15. The results are very satisfactory. It has to be noticed that in a real experiment, the use of such hybrid FEM/ANN model allows a reduction of 15% of CPU time.

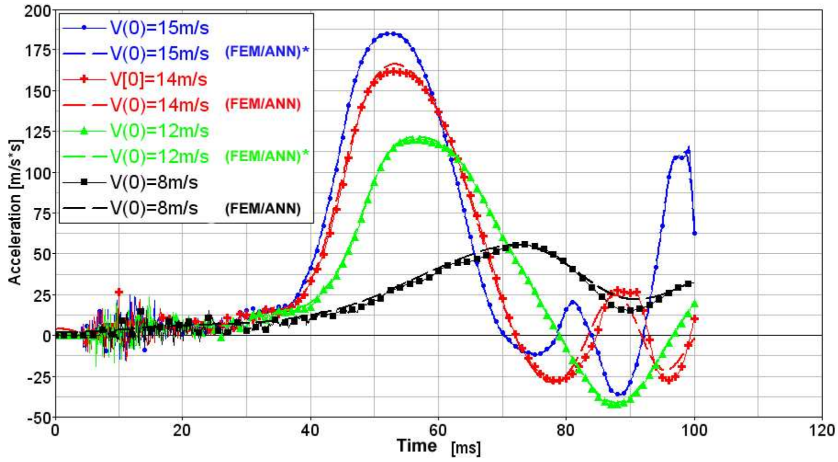


Figure 15 – Crash pulses discrepancies between original and hybrid (roof ANN) model. Four velocities are represented: $V_0^{(1)} = 8$ m/s (extrapolation) $V_0^{(2)} = 12$ m/s (learned) $V_0^{(3)} = 14$ m/s (interpolation) and $V_0^{(4)} = 15$ m/s (learned).

CONCLUDING REMARKS

We have presented a strategy for the simplification of the automobile crash simulation by a coupled FEM/ANN approach. This approach is original and interesting since it makes possible to introduce a different precision level for some model part without difficulty.

Since the ANN identifies structural parameters, its architecture is derived from physical laws (in nonlinear dynamics equations form) for a system representing the simplified part. By using a simple nonlinear system, we have analyzed the ANN ability for the reproduction of a significant nonlinear behavior and the connections between ANN design and the quality of the results. Then, the approach has been applied to a complex situation, concerning a real crash simulation.

The results show the interest of coupling the FEM and the ANN. The FEM use for a detailed substructure makes possible the preservation of the traditional design of this model part. The ANN is used to predict a temporal acceleration sequence at the interface with the FEM detailed substructure. The initial results show that the hybrid FEM/ANN models may represent an interesting alternative for crash test simulation of complex vehicle structures. Moreover, the ANN model brings a speed of calculation and flexibility, with a promising possibility of generalization.

The results show also the influence of the network architecture, which has to be connected to the physical problem under consideration. Nevertheless, in spite of the limits evoked (network size, training convergence, extrapolation simulation,...), the results emphasize that:

- The use of ANN in hybrid system FEM/ ANN has led to a significant gain in the computation time.
- The simulation experiments furnished positive results which show the interest of the approach and suggest an exploration of the connection between identification of dynamic systems and ANN.
- The identification approach has made possible the generation of an algorithm, easily translatable to the ANN framework.

REFERENCES

Box, G., Jenkins, G., 1976, “Time Series Analysis Forecasting and Control”, San Francisco: Holden day.

Chen, H.M., Qi, G.Z., Yang, J.C.S. and Amini, F., 1995, “Neural Network for Structural Model Identification”, Journ. of Engineering Mechanics, pp.1377-1381, December.

Cichicki, A., Unbehauen, R., 1994, “Neural Networks for Optimisation and Signal Processing”, Wiley NY.

Cornette, D., Thirion, J.L., Markiewicz, E., Drazetic, P., Ravalard, Y., 1998, “Localization of collapse mechanism in axial compression and bending for simplified vehicle crash simulation”, Proceedings of 2-nd International Conference on Integrated Design and Manufacturing in Mechanical Engineering, Vol. 1, pp.245-256, Compiègne - France, May.27-29.

Drazetic, P., Markiewicz, E., Ravalard, Y., 1997 “Application of kinematic models to compression and bending in simplified crash calculations”, Int. J. Mech. Sci. , Vol.35, #3/4, pp. 179-191.

Fahmy, M.W., Namini, A.H., 1994, “A survey of parallel non-linear dynamic analysis methodologies”, Computers & Structures, Vol.53, pp. 1371-1386.

Hallquist, J.O., Tsay C., 1999, “LS-DYNA User’s manual: Nonlinear dynamic analysis of structures” vs. 950, Livermore Software Techn. Corp., Livermore, California .

Haykin, S., 1994, “Neural Networks: A Comprehensive Foundation”, Macmillan College Publisher Comp.. FERENCES

Mentzer, S.G., 1982, “Analysis and Enhancement of the Fiat Methodology for Vehicle Crash Modeling and Simulation” Automated Sciences Group, Inc. Report ASG-TR-82-18.

Mentzer, S.G., 1992, “The SISAME Methodology for Extraction of Optimal Structural Crash Models” SAE Report 920358.

Narendra, K.S., Parthasarathy, K., 1990, “Identification and control of dynamical systems using neural networks” IEEE Transactions on Neural Networks, Vol. 1(1) pp. 4-5.

PAM-CRASH/PAM-SHOCK, Engineering Systems International SA, F-94588 Rungis, France.

RADIOSS, “Radioss Crash, User’s Documentation”, MECALOG, 06903 Sophia Antipolis Cedex, France.

Rumelhart, D.E. et al., 1986 “Learning representations by back-propagating errors”, Nature, Vol.323, pp. 323.

Wong, F.S. 1991, “Time series forecasting using backpropagation neural networks” Neurocomputing, Vol.2, pp. 147-159.

RESPONSIBILITY NOTICE