THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
345 E. 47th St., New York, N.Y. 10017

96-TA-39

NEURAL NETWORKS FOR THE DIAGNOSTICS OF GAS TURBINE ENGINES

G. TORELLA+ G. LOMBARDO*

+Italian Air Force Academy 80078 Pozzuoli Napoli Italia
*Università di Palermo Palermo Italia

Abstract

The paper describes the activities carried out for developing and testing Back Propagation Neural Networks (BPNN) for the gas turbine engine diagnostics.

One of the aims of this study was to analyze the problems encountered during training using large number of patterns.

Each pattern contains information about the engine thermodynamic behaviour when there is a fault in progress.

Moreover the research studied different architectures of BPNN for testing their capability to recognize patterns even when information is noised.

The results showed that it is possible to set-up and optimize suitable and robust Neural Networks useful for gas turbine diagnostics. The methods of Gas Path Analysis furnish the necessary data and information about engine behaviour.

The best architecture, among the ones studied, is formed by 13, 26 and 47 neurons in the input, hidden and output layer respectively. The investigated Nets have shown that the best encoding of faults is the one using a unitary diagonal matrix.

Moreover the calculation have identified suitable laws of learning rate factor (LRF) for improving the learning rate.

Finally the authors used two different computers. The first one has a classical architecture (sequential, vectorial and parallel). The second one is the Neural Computer, SYNAPSE-1, developed by Siemens.

1.-INTRODUCTION

The diagnostic is one powerful tool for the maintenance of gas turbine engines. During engine operating life, the activities based on effective Engine Condition Monitoring allow early intervention assuring the security margins. Moreover, by permitting the maintenance only when it is necessary, diagnostics avoid unnecessary and expensive engine stops and save both human and money resources.

Obviously information about the engine behavior makes possible and easy the engine diagnostics.

Unfortunately, sometimes the causes of engine malfunction and the right interventions must be detected by poor information.

Moreover the values used for diagnostics may be either approximate or, at the worst, incorrect. Nevertheless the diagnostics must proceed and the maintenance must start.

Neural Networks are one of the Artificial Intelligence fields and they are particularly useful for many tasks [1-3].

They work satisfactorily also with poor and inaccurate information therefore they might give an important help to diagnostics and fault detection.

The specialized literature and different authors have shown the possibility of using Neural

Presented at the 1996 ASME Turbo Asia Conference
November 5–7, 1996 — Jakarta, Indonesia

Networks for fault isolation and for diagnostics in different fields. [1-11]

This paper deals with the activities carried out for developing, testing and using Neural Networks for engine diagnostics. The methods, the techniques and the computer programs for Gas Path Analysis give the necessary data about the engine behaviour.

The main aims of this research were:

- the analysis of the possibility of setting-up reliable, flexible and robust Neural Networks having high ability to detect the right fault even if information about engine behaviour is poor and/or affected by noise;
- the study of training problems linked to the use of a large number of patterns;
- the analysis of some architectures of Neural Networks;
- the utilization of two different type of computers: a 'classical' computer and a Neural-Computer. The scope was the comparison of benefits offered by each type.

2.-THE SELECTION OF NEURAL NETWORK AND THE RELATED PROBLEMS

Neural Nets are information processing systems that have certain performance characteristics in common with biological Neural Systems. Elemental units, called artificial neurons, form the nets. The different arrangements of neurons and the connections among them define a particular Neural Network.

The analysis of different types of Neural Networks, the experience heaped-up during past studies by authors and specific literature [9] showed that different types of networks have capability for fault diagnosis.

Each Neural Net has advantages and disadvantages. For instance BPNNs must be trained again for learning new patterns. The Adaptive Resonance Theory Neural Network has not this problem.

On the other side BPNNs seem to be quite robust and has capability to work satisfactorily even with poor data.

This paper consider the use of the Back Propagation Neural Networks (BPNN) for the diagnostics of gas turbine engines [12-14].

BPNNs are multilayered networks formed by units (the neurons) arranged in different layers. There are one input layer, one or more hidden layers and one output layer.

The number of units of input layer is strictly linked to the number of available information about engine health. The studied Nets had 13 units. This means that all nets use 13 parameters describing the thermodynamic behaviour of engine.

This paper considered networks having only one hidden layer. This choice was suggested by necessity to reduce the time for training Nets.

There are not fixed and sure criteria for selecting the number of elements of hidden layer. It should be useful ιo use few units for reducing the time of training. Nevertheless convergence problems might arise [11]. The result of previous studies suggested to use 26 or 52 neurons in the hidden layer.

Owing to the very long time for training, a larger number of hidden neurons was not considered.

Finally the number of units of output layer is variable and it is strictly linked to the criterion used for encoding the faults.

Therefore the different architecture are characterized by the encoding of faults and by the number of element of hidden units.

One of the aim of this study was the selection of the best BPNNs among the ones studied.

As previously stated one of the scope of this paper was to increase the number of patterns in order to study the problems linked to training.

The literature contains results about the application of Neural Networks to fault diagnosis using 140 patterns [9].

During this research was used a number of patters ranging from 47 to 4324.

Each patter contains the values of parameters describing the thermodynamic behaviour of the engine when there is a fault in progress.

During this study two different sets of patterns were used. The first one describes the effects of different faults of engine behaviour. The second one, the largest one, contains disturbed patterns derived from all patterns used in the first set.

The following section discusses the criteria used for developing and evaluating the used patterns.

2.1-The training of BPNNs and the criteria for selecting the patterns for training

Learning or training is the most important and critical activity carried out during Network development because it strongly influences the final qualities of Nets.

From a computational point of view, the training allows to calculate the values of the strength of connections among the neurons. The BPNNs derive their name from the procedure used for training. It is depicted in Fig. 1.

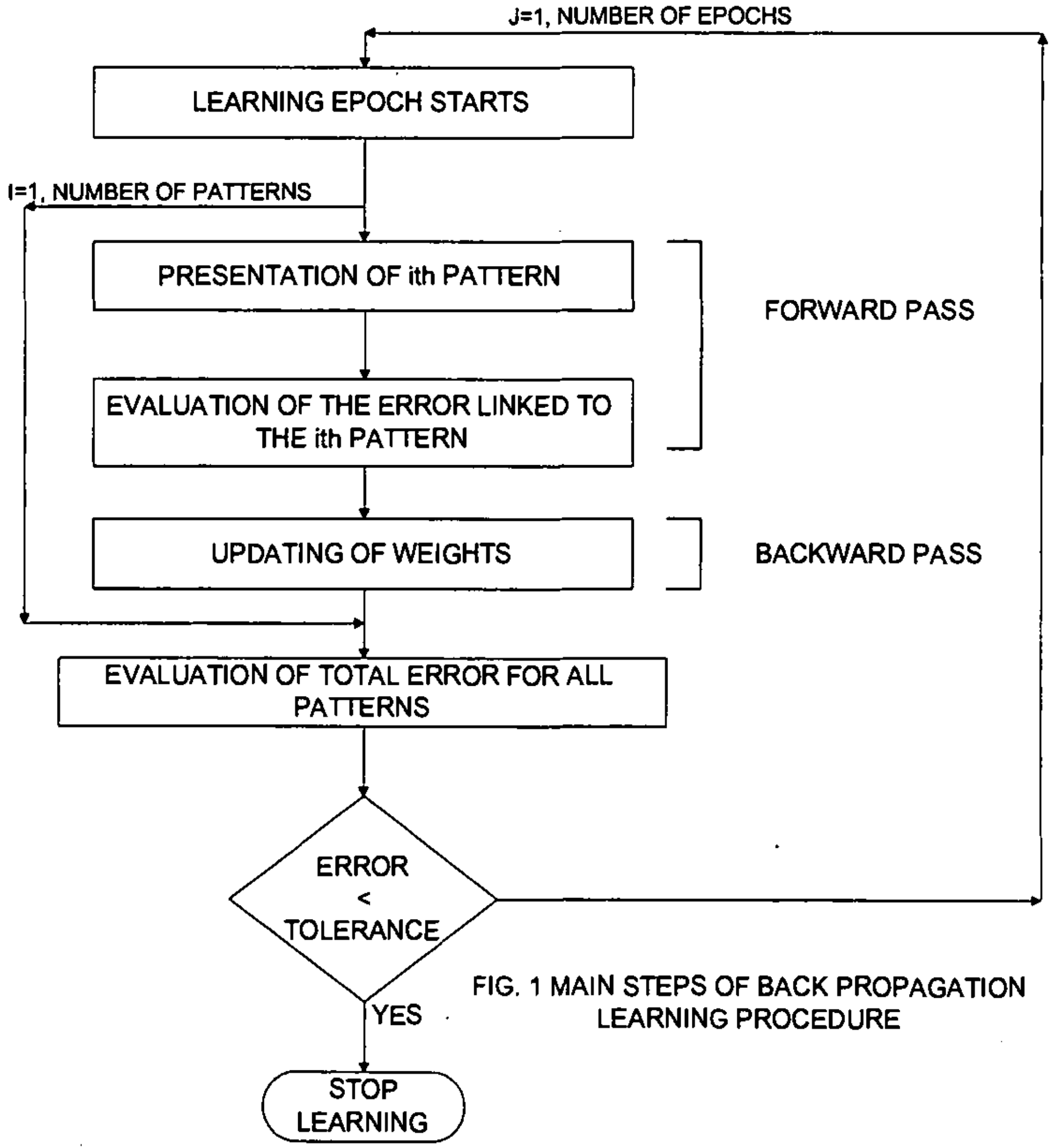

FIG. 1 MAIN STEPS OF BACK PROPAGATION LEARNING PROCEDURE

Neural Networks for diagnostics use patterns containing the variations of engine thermodynamic and performance parameters, with respect to the same values of a 'healthy' engine, when known faults are active. Therefore the training procedure may start when patterns, describing known situations and faults, are available.

One source furnishing the necessary patterns is the engine simulation. It permits the calculation of many patterns containing information about the engine behaviour when there are one or more faults in progress [15-17]. For this aim the authors used the codes for simulating the engine working developed in the past. These programs are based on a set of non-linear equations describing the thermodynamic cycle of engine, on the state vector technique and on the component matching criterion.

The results of calculations are usually collected in files called *'matrices of influence'* [18-19].

These files are the main data of codes for engine diagnostics and they derive from the criteria of Gas Path Analysis used in Engine Condition Monitoring [18-19].

Each row of a matrix is the replay of engine to the decay of one or more performance of engine component. As an example: a row shows the variation of engine performance when the fault causes the decrease of compressor efficiency, another row the contemporary decay of efficiency of compressor and turbine, etc. Therefore each row is a pattern useful for the training of Neural Networks for engine diagnostics.

Obviously each column represents the variation of each parameter of engine (thrust, rotational speed, exhaust gas temperature, compressor delivery pressure, etc.) due to the decay of component performance.

The engine studied was a 600 SHP single spool turboshaft engine with a power turbine.

For this study the codes for engine simulation gave 47 basic patterns for the training of BPNNs; some of the simulated faults are shown in Table 1.

The basic patterns describe the engine behaviour when the faults cause the variation of one, two or three performance of engine components.

TABLE 1
FAULT EFFECTS (component performance variation)
(+) Increase; (-) Decrease

1	COMPRESSOR EFFICIENCY (-)
2	COMPRESSOR PRESSURE RATIO (-)
3	COMPRESSOR CAPACITY (-)
4	GAS GENERATOR TURBINE EFFICIENCY (-)
5	GAS GENERATOR TURBINE CAPACITY (+)
6	POWER TURBINE EFFICIENCY (-)
7	POWER TURBINE CAPACITY (+)
8	GAS GENERATOR TURBINE EFFICIENCY (-) and CAPACITY (+)
9	POWER TURBINE EFFICIENCY (-) and CAPACITY (+)

The data whose variations are stored in each element of a row of matrix of influence are the most important performance and thermodynamic parameters of engine. Table 2 shows the parameters considered here.

Some of these data are directly obtained from experimental facilities: test beds and airborne data acquisition systems. From the analysis of these data, carried out by suitable computer codes, it is possible to obtain the values of parameters not directly measurable.

The dimensions of starting matrix of influence, containing all patterns used for training, are 47x13. The matrix was evaluated for fixed values of both fuel flow and power turbine rotational speed.

Other matrices were developed during the setting-up and the tests BPNN for improving and testing the robustness and ability of Nets.

The training of BPNNs required the coding of each fault. This study used different criteria.

The first one was based on a 6 bit binary code of the fault and Table 3 shows part of the coding for some faults.

TABLE 2
ENGINE PERFORMANCE and THERMODYNAMIC PARAMETERS

1	POWER
2	PRESSURE RATIO
3	EXHAUST GAS TEMPERATURE
4	SPECIFIC FUEL CONSUMPTION
5	INLET MASS FLOW RATE
6	COMPRESSOR EXHAUST PRESSURE
7	COMPRESSOR EXHAUST TEMPERATURE
8	GAS GENERATOR ENTHALPY DROP
9	TURBINE INLET TEMPERATURE
10	GAS GENERATOR TURBINE EXHAUST TEMPERATURE
11	GAS GENERATOR TURBINE EXHAUST PRESSURE
12	GAS GENERATOR ROTATIONAL SPEED
13	POWER TURBINE PRESSURE DROP

TABLE 3
6 BITS CODING OF FAULTS

000001	decay of compressor efficiency
000010	decay of compressor pressure ratio
000011	decay of compressor capacity
....	
101110	increase of burner and interduct pressure losses
101111	increase of intake, burner and interduct pressure losses

The second criterion was based on the construction of a matrix whose element are all set to zero except one whose value is 1. The position of this non-zero digit in the row indicates the fault. The final matrix was an unitary diagonal matrix and Table 4 shows a part of this matrix with the related faults.

TABLE 4
UNITARY DIAGONAL MATRIX FOR FAULT CODING

10000000...00000	decay of compressor efficiency
01000000...00000	decay of compressor pressure ratio
00100000...00000	decay of compressor capacity
....	
00000000...00010	increase of burner and interduct pressure losses
00000000...00001	increase of intake, burner and interduct pressure losses

Finally a third coding criterion stated that each fault were described by a 13 elements vector. The first 4 elements contained the tens and the successive 9 elements contained the units.

For instance the codes of fault 3 and fault 33 are:

0 0 0 0 0 0 1 0 0 0 0 0 0 fault 3
0 1 0 0 0 0 1 0 0 0 0 0 0 fault 33

3.-THE DEVELOPMENT OF NEURAL NETWORKS

The study followed two different approaches. The first one dealt with the development, testing and use of BPNN by using computers ranging from Personal to high speed computers with advanced architecture.

The second approach considered the use of a Neural Computer: the Siemens SYNAPSE 1.

The aim of this double choice was to study and to observe the difference in developing the BPNNs and in testing their ability and robustness in fault recognition. This way two different philosophies were compared and enough data, for selecting the more effective way for developing BPNNs, were collected.

3.1-The '*classical*' computer approach

The study began with the selection of patterns for the BPNN training. The 47x13 matrix of influence furnished the first set of patterns.

For improving the robustness of Neural Network, the training used two more sets of patterns derived from the original matrix (47x13 matrix). This way .it was possible to study the behavior of BPNN when it must works with either uncompleted patterns or with patterns containing wrong data.

Therefore the second set contained 658 patterns; 611 of them were obtained form the basic 47 patterns by setting, one by one, each element of each row equal to zero.

The third set contained 4324 patterns; 4277 of them were obtained considering, from time to time, two elements of each row of basic 47 patterns equal to zero. Finally both sets contained the basic 47 patterns

Finally, for controlling the quality of BPNN learning, the matrix shown in Table 4 was used as the set of output patterns.

The successive step of BPNN development dealt with the definition of its architecture. The considered BPNN has one input layer with 14 neurons (13 for each engine parameter + 1 bias neuron); one hidden layer with 52 neurons (51+1 bias) and one output layer with 47 units. When the sets of input and output patterns were constructed the training procedure started.

In the past, several Networks were developed by using personal computers and the studies considered comparatively few patterns (8-17).

The present research considered a larger number of patterns so the training time become very high (from 10 hours to several days) so a faster computer, the CONVEX 3880 a vectorial-parallel supercomputer with 1024 Megabyte RAM and 8 processors, was used.

The faster computer performed training only because the test on robustness and ability of BPNNs were carried out by Personal Computers with 486 66 Mhz microprocessors.

During the training, the following problems appeared and required attention and solution.

The first problem was to improve the speed of calculation.

One important parameter of BPNN learning iterative procedure is the learning rate factor (LRF) controlling the updating of neuron weights between two successive epochs.

When the Network used few patterns the LRF had a fixed values. The training with 47 patterns used satisfactorily a value equal to 0.05. The training required 28606 epochs in about 10 hours (it is the total time shared with other applications).

When the number of patterns increased to 658 and 4324 it was necessary to use variable LRF. In fact the study showed that it is convenient to use high value of learning factor at the beginning of learning phase and lower values in the final part of the learning phase. This way, at the beginning, the variation of weights between two successive steps is large and the between the calculated and the actual patterns decreases quickly. When the learning proceeds there is the necessity to decrease the variation of weight so the LRF must be lower.

Moreover variable LRF avoid the fluctuation of global value of error. The general law used was:

$$LRF = \frac{A * IPASS + B}{IPASS}$$

. IPASS indicates the number of epochs and different laws, deriving from different values of A and B, were used, table 5.

TABLE 5 Different laws of LRF		
LAW 1	A=0.001	B=0.099
LAW 2	A=0.01	B=0.98
LAW 3	A=0.05	B=0.94

The Law 1 avoids the fluctuation of global error but the learning speed is too low owing to the high number of patterns.

The Law 2 decreases the initial error very quickly but, when the number of learning epochs increases, the learning speed decreases very much.

The Law 3 allows fluctuation of global error higher than previous law but the learning speed is noticeably increased. Moreover the high value of learning rate during the first epochs lowers the global error quickly.

Law 3 was used for the BPNN training with 658 and 4324 patterns. The total epochs required were about 36667 and 21651 respectively.

3.2-The Neural Computer approach

The second approach studied the use of the neural computer SYNAPSE-1 by Siemens for BPNN development.

SYNAPSE-1 is a neural computer with processors arranged in a matrix and it allows parallel computation. Its architecture is quite different from usual computers therefore it is more suitable for Artificial Intelligence applications.

The training used again three sets of patterns. The first was the basic 47x13 matrix. Moreover, for studying the robustness requirements, two more sets of patterns were used. One contained the already used 4324 patterns. The third set was built by altering each pattern of matrix 47x13. The alteration was performed by adding to each element a random quantity proportional to its original value. The range of random values was between -0.08 and +0.08. Since each pattern was changed 17 times, the third set contains 799 patterns.

As regard to the fault coding, BPNNs may use either the 6 bits binary coding (Table 3) or tbe unitary diagonal matrix (Table 4) or the decimal coding already described.

The BPNN development used ECANSE (Environment for Computer Aided Neural Software Engineering). It is based on the concepts of object oriented programming and permits to develop Neural Networks by using an user-friendly environment containing many objects each carrying out a particular activity [20].

Owing to its structure ECANSE allows the development of Neural Networks even if the user is not expert in a particular programming language.

The aim of SYNAPSE-1 utilization was the selection of BPNN showing the highest ability and robustness. Moreover particular attention was paid to the time necessary for training. The developed BPNNs are shown in Table 6.

The Net 1 and 2 studied the influence of number of elements of hidden layer on the reliability and robustness of networks. They use a 6 bits fault coding.

Net 3 and 6 considered the influence of LRF and of fault coding. They used 6 bits and the unitary diagonal matrix coding respectively. Both nets used the same laws of LRF, table 7.

ECANSE allowed to train nets 3 and 6 contemporary.

The aim of nets 4 and 5 was similar to the one of net 1 and 2 but they used unitary diagonal matrix for fault coding.

Nets 7 and 8 considered only 47 faults for training. This choice was performed for evaluating the influence of both fault coding and of a 'low' number of patterns on network robustness and for optimizing the learning time. Both nets have 26 elements in the hidden layers but net 7 used 6 bits binary coding while net 8 used the unitary diagonal matrix.

Moreover both nets used variable LRF. The present study showed that the best law for optimizing the learning velocity is the one in Table 8.

TABLE 6
Summary of developed Neural Networks

network	elements of layers	number of patterns	fault coding	learning steps $x10^6$
NET 1	13-52-6	799	6 bits	20 (23 h)
NET 2	13-26-6	799	6 bits	31 (30 h)
NET 3	13-52-6	799	6 bits	48
NET 4	13-52-47	799	table 2 mat.	48
NET 5	13-26-47	799	table 2 mat.	50
NET 6	13-52-47	799	table 2 mat.	117
NET 7	13-26-6	47	6 bits	35
NET 8	13-26-47	47	table 2 mat.	35
NET 9	13-52-47	4324	table 2 mat.	52
NET 10	13-26-47	4324	table 2 mat.	52
NET 11	13-26-13	47	decimal	20
NET 12	13-52-13	47	decimal	20

TABLE 7
Variation of LRF used by nets 3 and 6

$LRF = 0.1 - 2x10^{-6}\,IPASS$	$0 < IPASS < 10000$
$LRF = LRF_{starting} - 2x10^{-9}\,IPASS$	$0000 < IPASS < 10 millions$
$LRF = 0.000$	$IPASS > 10 millions$

TABLE 8
LRF law used by nets 7 and 8

$LRF = 0.1$	$IPASS < 10000$
$LRF = 0.05$	$0000 < IPASS < 100000$
$LRF = 0.01$	$00000 < IPASS < 300000$
$LRF = 0.005$	$300000 < IPASS < 900000$
$LRF = 0.001$	$IPASS > 900000$

Nets 9 and 10 considered a very large number of patterns (4324) and consider different numbers of elements in the hidden layers (52 and 26).

Finally nets 11 and 12 studied the influence of both hidden layer elements and of decimal fault coding.

4.-THE RESULTS

This part of paper discusses the results obtained with the BPNNs developed by the two approaches.

4.1-The BPNNs developed by '*classical*' computers

The weights evaluated by CONVEX 3880 were used by Personal computers for evaluating the quality of BPNNs.

The first tests dealt with the BPNN robustness. The effectiveness of a neural network is founded in its capability to isolate the faults even when the input information about the engine behaviour are wrong and/or incomplete.

The evaluation of robustness required two series of tests. The first one used the same patterns employed for training. The second one required a new set of patterns whose elements were suitably disturbed and altered. Table 9 shows the results of first series of tests.

Table 9

Percentage of faults correctly recognized by network

	test with 47 patterns	test with 658 patterns	test with 4324 patterns
network trained with 47 patterns	100%	52%	15%
network trained with 658 patterns	100%	96%	78%
network trained with 4324 patterns	100%	100%	98%

The robustness of network is fully demonstrated by the first row. The networks recognizes all patterns it was trained by and the 52% of the set containing 658 patterns and 15% of set composed by 4324 patterns. This means that even if the starting patterns are relatively few (47) the networks is able to correctly recognize 342 of 658 (295 patterns beyond the original 47) and 648 of 4324 (641 patterns beyond the original 47). In other words the net is able to detect the right fault even if the input contains one or two information wrong or absent.

The second series of tests was carried out by developing 235 new patterns. They were obtained from the basic 47x13 matrix by adding to each element of each pattern a noise. One type of noise was equal to ±20%,±10%,±5% of original values of element. Another type is constantly equal to ±0.005,±0.001 for any element of each pattern. Table 10 shows the obtained results.

Table 10

Second set of tests for robustness network

	noise ±20%	noise ±10%	noise ±5%	noise ±0.005	noise ±0.0.01
Network trained with 658 patterns	55%	50%	89%	87%	100%
Networks trained with 4324 patterns	54%	64%	74%	81%	98%

Again BPNNs show a very high degree of robustness Infact they recognize the right faults even if the input pattern is strongly disturbed.

4.2-The BPNNs developed by Neural-Computer

Table 11 shows the most significant results about the ability of nets in recognizing the patterns and their robustness. Two tests for evaluating the net ability were carried out. The first used the basic 47 patterns. The second one used 149 of the 799 patterns already considered for training.

The robustness tests were carried out in different ways. One test used 47 patterns noised proportionally to the values of original elements. The noise was selected randomly in three different range ±0.08,±0.15,±0.8. Moreover the test used 611 patterns derived from the basic ones by setting equal to zero, from time to time, one element of each pattern.

TABLE 11
Results of calculation for testing the Net robustness

Net	149 patterns	47 patterns	noise ±0.08	noise ±0.15	noise ±0.8	611 patterns
1	86%	85%	-	83%	34%	42%
2	86%	85%	-	72%	34%	41%
3	66%	66%	-	64%	40%	43%
4	98%	98%	-	98%	66%	68%
5	98%	98%	-	98%	57%	62%
6	98%	98%	-	96%	64%	63%
7	77%	77%	77%	72%	28%	35%
8	100%	100%	100%	98%	68%	60%
9	85%	-	-	-	-	31%
10	74%	-	-	-	-	71%
11	92%	92%	92%	85%	34%	44%
12	89%	89%	92%	81%	32%	44%

The results obtained by nets 5, 6 and 2 and 3 show that the use of variable LRF helps the training and shortens the time of learning.

This result is already available in literature where there are only general indications about the value of learning rate factor. This study has shown that, for our aim, the best laws of variation are the ones shown in table 7 and 8.

There is an apparent strange results of learning time of net 3 and 2, Table 6. The former (variable learning rate) requires 48 millions of steps while the latter (fixed learning rate) requires only 20 millions of steps. The result may be explained by thinking that the net 3 and net 6 are trained together and the shown time is the total training time.

This study showed that the fault coding criterion is very important during Neural Network development.

The comparison of nets 1,2,3 and nets 4,5,6 shows the best behaviour of unitary matrix coding while the comparison of nets 4,5,6 and nets 11,12 shows that the unitary diagonal matrix coding is always better than the decimal one.

The results show that the best BPNN is net 8 that has 13 elements in input layer, 26 elements in the hidden layer and 47 elements in the output layer (the fault coding uses the diagonal unitary matrix).

5.-CONCLUSIONS

The paper considered BPNNs for the diagnostics of gas turbine engines based on thermodynamic and performance data obtained by the engine Gas Path Analysis.

The selection of BPNNs was based on the fact that these Nets seem to he more robust than others. So they might be able to detect the right fault even if the information about engine health and behaviour is poor and/or noised.

This study confirmed this behaviour.

The research considered Neural Nets with one input, one hidden and one output layer.

The number of neurons of input is related to the number of information available about engine health. This study considered an input layer with 13 neurons.

The choice of both the number of hidden layer and the elements of each layer is not controlled by well defined criteria. This study considered 26 and 52 elements in the unique hidden layer.

The number of output elements is related to the criterion for encoding the faults. The paper considered two different criteria.

The work confirmed the importance of fault coding and the unitary diagonal matrix showed to be the best coding criterion.

The previous considerations lead to some architectures and one aim of study was to select the best among them.

The best architecture, from the robustness point of view, had 13, 26 and 47 neurons in the input, hidden and output.

The study used a very large number of patterns for training.

The results confirmed that, when the number of patterns is very large, the use of variable LRF is essential for reducing learning time.

The research furnished suitable laws of LRF for improving the training reducing the necessary training. The laws were obtained starting from general criterion that during the training the LRF must decrease [11]. The laws seem to be effective also for a number of patterns larger than the one used.

The comparison of the use of different computer has lead to these conclusions:

The results are the same because the basic relations and calculation criteria are the same.

Even if the value of training time are not available, due to the time sharing characteristics of classical computer, this latter is faster than neural computer.

On the other side the neural computer is easy to use. Infact, thanks to the ECANSE environment and the object oriented programming, the shell allows to construct effective Nets even if the user is not an expert of computer programming.

6.-BIBLIOGRAPHY

1)-M.H. HASSOUN 'Fundamentals of Artificial Neural Networks' The MIT press, Cambridge, Massachusetts, 1995

2)-S.I. GALLANT 'Neural Network Learning' The MIT press, Cambridge, Massachusetts, 1994

3)-L.FAUSETT 'Fundamentals of Networks Architectures, Algorithms and Applications' Prentice-Hall-1994

4)-W.E. DIETZ, E.L. KIECH, M ALI 'Jet Rocket engine fault diagnosis in real time' Journal of Neural Network Computing, 1 5-18, 1989

5)-D.M. HIMMELBLAU, R.W. BARKER, W. SUEWATANAKUL 'Fault Classification with aid of artificila neural network' IFAC/IMACS Symposium SAFEPROCESS '91 Baden-Baden, Germany 1991

6)-M.A. KRAMER, J.A.LEONARD 'Diagnosis Using Back Propagation Neural Network - Analysis and Criticism' Computational Chemical Engineering, 14, 1323-1338, 1990

7)-S.R. NAIDU, E. ZEFIRIOU, T.J. McAVOY 'Use of Neural Networks for Sensor failure Detection in a Control System' IEEE Control System Magazine, 10, 49-55, 1990

8)-L.F. PAU 'Failure Diagnosis and Performance Monitoring' Marcel Dekker, New York 1981

9)-T. SORSA, H.N. KOIVO ' Application of Artificial Neural Networks in Process Fault Diagnosis' Automatica, Vol. 29, No. 4, pp843-849, 1993

10)-M.McCORD, W.T. ILLINGWORTH 'A Practical Guide to Neural Nets' Addison-Wesley Publishing Company, Inc. 1991

11)-J.A.FREEMAN, D.M.SKAPURA 'Neural Networks Algorithms, Applications and Programming Techniques' Addison-Wesley - 1991

12)-G. TORELLA, G. LOMBARDO 'Neural Networks for the Maintenance of Aeroengines' AIAA paper 95-2351 1995

13)-G. TORELLA, G. LOMBARDO 'Utilization of Neural Networks for Gas Turbine Engines' ISABE paper 95-7032, 12th International Symposium on Air Breathing Engines 10-15 September 1995, Melbourne, Australia

14)-G.TORELLA, G.LOMBARDO 'Artificial Intelligence Tools for the Maintenance of Turbofan Engines' ICAS-94 18-23 September, 1994 Anaheim, CA

15)-G.TORELLA 'Problems and Use of Scaling Factors in the Numerical Simulation of jet Engines' ASME paper 90-GT-386 1990

16)-G.TORELLA 'Study on the Behaviour and Performance of Turbofan Engines with Faults in the Components' 1992 Engineering System Design and Analysis Conference - ASME PD-Vol.47-1 pp11-21, 1992

17)-G. TORELLA, G. LOMBARDO 'The Analysis of Influence of Component Performance Decay in Gas Turbines' AIAA paper 95-3031 1995

18)-L.A. URBAN 'Gas Turbine Engine Parameters Interrelationships' 2nd edition Hamilton Standard UAC - 1969

19)-L.A.URBAN 'Gas Path Analysis Applied to Turbine Engine Condition Monitoring' AIAA paper 72-1082 December 1972

20)-SIEMENS 'Environment for Computer Aided Neural Software Engineering ECANSE' 1994