ENHANCEMENT OF CONTROL ORIENTED ENGINE MODELS USING NEURAL NETWORK

I. ARSIE, C. PIANESE, G. RIZZO

Dipartimento di Ingegneria Meccanica - Università di Salerno, 84084 Fisciano (Sa), Italy
E-mail: pianese@unisa.it

In the last years a hierarchical model structure has been developed by the authors for the optimal design of engine control strategies. This structure is composed of a group of models based on different approaches ranging from fast black-box models to phenomenological ones. The black-box models together with grey-box mean value dynamical models are linked with the computer code O.D.E.C.S. which is used in industrial environment for the optimization of engine control strategies.
In the framework of black-box models development, some approaches have been considered in order to enhance model precision and to maximize the level of information derivable from experimental tests. The former set of models is based on classical regression techniques to compute steady state engine performance (i.e. fuel consumption and HC, CO, NO_x emission levels) together with interpolating techniques in order to evaluate performances in the domain not investigated during experimental tests.
To overcome some limitations coming out from the use of regressions and interpolating techniques, a Neural Network model structure has been developed. The Neural Networks, well suited for non linear phenomena modelization, are able to deal with high uncertainty input level (independent data variables) or noised data as well as are able to operate outside their range of training experience. For the purpose of the present application a Multi Layer Perceptron (MLP) Neural Network structure has been selected with a Backpropagation training procedure.
The results of simulations obtained by using the Neural Network model developed are compared with the previously used regression technique and the advantages emerging from the new approach are discussed.

1. Introduction

The reduction of both engine emissions and fuel consumption together with vehicle driveability enhancement during steady-state and transient operations are the main tasks of automotive industry. In the last decade many efforts have been addressed to the design of new engines and to develop innovative control systems in order to achieve the targets imposed by government emission limits and customer demand. On the side of engine control, many approaches have been followed toward the assessment of reliable methodologies for the achievement of the optimal engine control strategies. These methodologies are based on the development of easy to handle fast engine computer models able to synthetize the information derived from experimental data. Nevertheless, the trade-off between engine model predictivity, computer resources and number of experiments required is still one of the major drawback in the building of comprehensive control oriented engine models.
The interest of the authors on new engine models is of relevant importance for their activity in the field of optimal engine control strategies definition[1] and related modelization problems which are treated in a hierarchical way. This models structure, described elsewhere[2], takes advantages of different model approaches from black-box to phenomenological ones combining several mathematical models at variable levels of details. The models are "embedded" in a more general engine control strategy rapid prototyping procedure which makes use of experimental design techniques [3]. In this hierarchical model structure a recent developed thermodynamical model for the simulation of engine pressure cycle is also included and an identification procedure based on decomposition approach has been developed for the experimental validation of the HC, CO, NO_x emission sub-models [4,5]. A view of the hierarchical model structure is reported in the Figure 1.

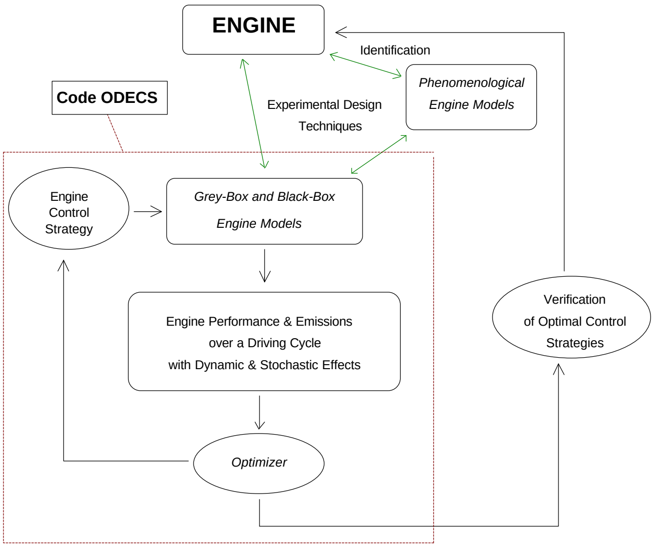

Figure 1: A hierarchical model structure for engine control design.

In the following the problem related with the definition of spark ignition engine performance models used in the code O.D.E.C.S. will be focuses (see the link between Engine and Grey/Black-box models in the Figure 1).

2. Engine model

2.1 Mean Value Engine Model

The whole dynamic engine model implemented in the code O.D.E.C.S. is basically a Mean Value Engine Model[6,7,8] and is build up of various submodels each related with the relevant engine phenomena concurring to the performance evaluation during transient operations. The engine dynamic model structure, for the calculation of both air mass flow and two phase fuel flow into the manifold and for the mean combustion wall temperature (grey-box models), is linked with a set of black-box regression models for the evaluation of fuel consumption and emission levels for HC, CO and NO_x. This latter set of models are derived from steady-state experimental tests as function of both actual engine state (Torque, rpm) and control variables (Spark Advance, Air-Fuel Ratio). The experimental data are measured over an equally spaced grid in the Torque-rpm plane (see Figure 2), in each grid point the fuel consumption and the emissions are measured as function of Spark Advance and Air-Fuel ratio. Major details on the grey-box models used and on the global model structure can be found in the references[6,7,9].

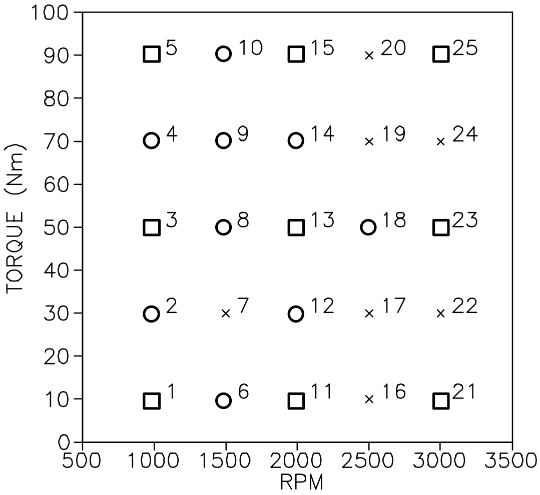

Figure 2: Grid of experimental points on the engine operating plane.

2.2 Regression model

In each point of the Torque-engine speed plane a set of polynomial regressions as function of control variables are derived for fuel consumption and for the specific mass flow rate of the three main emissions (HC, CO, NO_x):

$$\dot{m}_f^*\Big|_i = f_i(\mathrm{a},\mathrm{q}) \quad i = 1, N_g \tag{1.}$$

$$\dot{e}_j^*\Big|_i = g_i(\mathrm{a},\mathrm{q}) \quad i = 1, N_g \quad j = HC, CO, NO_x \tag{2.}$$

The generic *i-th* polynomial formula in each point is written as follow:

$$G(\mathrm{a},\mathrm{q}) = a_0 + b_1\mathrm{a} + ... + b_p\mathrm{a}^p + c_1\mathrm{q} + ... + c_q\mathrm{q}^q + d_1\mathrm{aq} + ... + d_{r+s}\mathrm{a}^r\mathrm{q}^s \tag{3.}$$

where the coefficients a_0, $b_{1...p}$, $c_{1..q}$, $d_{1...r+s}$ are found with a least square technique.

The equations (1) and (2) represent the steady-state models for each of the N_g Torque-speed working conditions investigated during the experimental analysis, therefore the resulting engine global model assumes a discrete form. Hence, to evaluate the performance in a generic (T,rpm) point a bilinear interpolation of the four nearest surrounding values derived from equations (1) and (2) is made. Due to the non linear dependencies with respect to the independent variables (T,rpm,a,q) an acceptable level of precision can be reached only making use of a quite dense experimental grid with equidistant points, while coarse grids could generate models with poor accuracy.

Since the use of a fine grid with equidistant points requires a considerable set of experimental data, some new techniques are undertesting with the objective of reducing the experimental effort needed. The first is based on phenomenological engine models to generate off-line the data required for the definition of the regression models (1-2), in this case a limited set of experimental data are used only for the identification phase of physical models[4,5] (see Figure 1). The other approach consists in the use of black-box Neural Network models for which an equidistant regular grid data is not required, hence resulting in a reduced number of experimental data needed compared with regression based model system. The Neural Network approach will be discussed in the next part of the paper.

3. Neural Networks

In the last decade the application of Neural Networks has seen a large widespread in different fields. To analyze only automotive related modelization problems, the use of Neural Networks is well suited for many applications such as control problems[12,13], diagnostics[14], mapping and modelization problems[15,16], optimization, sensor data fusion and pattern recognition[11]. In this section the features and the basic theory of Artificial Neural Networks are reviewed with particular attention to the experimental data mapping problem. For more details the reader is addressed to the basic textbooks on this topic[10,11].

3.1 Artificial Neural Networks basics

An Artificial Neural Network is composed of several elementary interconnected processing elements working in a parallel way, the perceptrons. From the analogy with human brain behaviour, Artificial Neural Networks are able to reproduce a process from training examples (neurocomputing approach[10]) rather than from a coded algorithm which simulate the process on the basis of a mathematical model (programmed computing approach[10]). Due to the use of experience knowledge, Neural Networks have relevant capabilities in term of generalization from limited training data sets making them able to work outside the training domain (i.e. extrapolation). Moreover, robust performances in presence of noisy input data are guaranteed because the stored knowledge is spread over the entire perceptron structure instead of being concentrated in few units[11].

Each neuron constituent the most elementary Artificial Neural Network (i.e. the perceptron) can be viewed as a single output black-box computing element with multiple inputs. The output is obtained processing the weighted sum of the inputs with a transfer function called activation function, which in general is a non linear monotonic function. In the present work the activation functions for the perceptrons belonging to the hidden layers is a sigmoid function while at the output layer a linear activation function is considered. The former one is a bipolar continuos activation function:

$$h(net) = \frac{2}{1+\exp(-2net+b)} - 1 \qquad [-1,1] \qquad (4.)$$

the linear transfer function has the following form:

$$h(net) = net + b \qquad]-\infty,\infty[\qquad (5.)$$

where *net* is the weighted sum of the inputs to the single perceptron in the layer and *b* is a bias term.

A single perceptron has limited memorizing capacity and many neurons should be arranged in group of layers in order to improve the learning capabilities. The resulting Multi Layer Perceptron Artificial Neural Network has a typical structure composed of one input, one or more hidden and one output layers as shown in the Figure 3. The vectors X and Y refer to the input of the Neural Network (independent variables) and to the output (dependent variables) respectively. The particular structure in the Figure 3 represents a Multi Layer Feed Forward Neural Network since the external input signals propagate through the hidden layers to the output layer; in the rest of the paper such structure will be considered and the full name will be replaced by the usual acronym MLP.

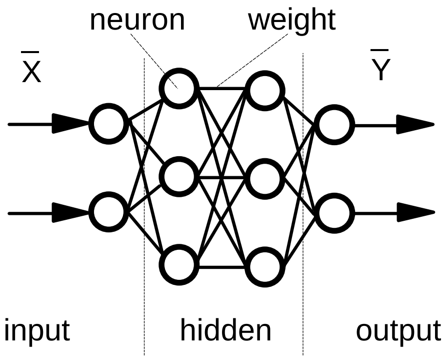

Figure 3: Multi Layer Neural Network structure.

3.2 Neural Network capabilities

From the theory of Neural Networks, the mapping capabilities of a MLP structure are based on the Kolmogorov’s Mapping Neural Network Existence Theorem who stated that a three layer feedforward structure is able to reproduce any generic continuos function[10,11]. This theorem allows the decomposition of any continuos map in a finite number of one-dimensional monotonic functions. Though the Kolmogorov’s theorem does not give any practical information on the way of deriving these functions, it has been interpreted[10,11] as a starting point to derive the basic topology of the Neural Network structure.

Further characteristics of a Neural Network are related with its learning and generalization capabilities: the learnability is the ability of a learning algorithm to find the set of weights which gives the desired accuracy of the mapping problem, the generalization is only possible if the training set contains enough information[11]. These features are strictly dependent from both the Neural Network structure and the training set of “examples” chosen to instruct the Neural Network. Thus the following problems arise: 1) define the Network structure, 2) define the training set which contains the necessary knowledge to be generalized, 3) choose the learning procedure. From a practical point of view the first problem is related with the choice of the minimum number of layers and perceptrons and the kind of activation functions, the second with the minimum number of experiments representing the mapping problem while the third addresses the problem of how to find the weights of the connections between the neurons. Usually a loss of generalization occurs when an excessive number of neurons are used (i.e. overparametrization) or when a reduced training set is used for many times (i.e. overtraining). Unfortunately for real mapping problems theoretical methods are not available and some heuristic rules should be followed to avoid overparametrization and overtraining.

3.3 Training procedure

The objective of the training process is to find the biases and the weights of each connection between the neurons of the MLP Neural Network. The method considered in the present work is based on the Backpropagation error algorithm which is an iterative supervised learning technique. A set of training examples is considered and for each training the desired output of the MLP is known (i.e. experimental data). For each iteration the error between the experimental data and the corresponding estimated value is propagated backward from the output layer to the input one through the hidden layers. During the backpropagation the weights and biases are adjusted as function of the error gradient through the learning rate, the weights updating is also controlled by a momentum constant term. The learning rate $\eta \in [0,1]$ influences the rate of convergence, generally a small value leads to a slow convergence, while faster convergence can be achieved with an high value of the learning rate which in turn could generate high instabilities. The momentum term acts as an inertia and is proportional to the previous weight update through a constant $\varphi \in [0,1]$: it has the effect of smoothing the convergence path in presence of high irregular or wavy error surfaces. The recursive weight update formula for the *j-th* connection is[10,11]:

$$\Delta w_j^{t+1} = \eta \frac{\partial E}{\partial w_j^t} + \varphi \, \Delta w_j^t \qquad (6.)$$

where E is the error function (i.e. the sum of the squared differences between predicted and training data). To improve the convergence the learning rate should be decreased as the error gradient reduces while as the error tends to reach a stable value the inertia term should be increased to skip out the presence of local minima[10,11].

4. Neural Networks application

In the following a preliminary application of Multi Layer Feedforward Neural Networks with Backpropagation for the mapping of engine fuel consumption and emission levels will be presented. Two tests have been carried out, the first concerns with the use of a simple one hidden layer MLP while in the second test a more complex structure with two hidden layers has been considered. For both structures the input layer has four neurons, one for each independent variable (Torque, rpm, Air-Fuel ratio, Spark Advance) and the output layer has one neuron only. The hidden layer of the first MLP structure is composed of five neurons for the calculation of the fuel consumption and nine neurons for the emissions, while in the second MLP structure tested five neurons have been considered for both hidden layers. The activation functions used are the ones shown in the equation (4) for the hidden layers and in equation (5) for the output.

The available experimental data have been measured on a commercial FIAT two liters, four cylinder engine, equipped with multi-point injection system with speed-density electronic control system. The whole set is composed of more than 500 experimental data for the 18 torque-speed conditions corresponding to the points with circle and square symbols in the Figure 2.

For the training of the first test case 328 experimental data corresponding to the nine points of the (*T,rpm*) plane with square symbols in the Figure 2 are used (Set A). To analyze the predictivity level of the Neural Networks trained a second set of 187 experimental data, belonging to the nine points of the (*T,rpm*) plane with circle symbols in the Figure 2, is considered (Set B). These results are then compared with the data of the regression models, presented in the section *2.2*; in this case the nine regressions models are derived from the same data used for the training of the Neural Network (Set A). Furthermore, the predictivity level of the Neural Network has been also compared with the results of a bilinear interpolation analysis conducted on the Set B. In order to have a single parameter to analyze the precision level reached in each test, the following error index is considered:

$$\varepsilon = \frac{\sqrt{\sum_{i=1}^{N_t} \left(y_i^* - y_i\right)^2 \Big/ N_t}}{\bar{y}^*} \qquad (7.)$$

where y^* is the measured value y the predicted one, $\bar{y}^*$ is the mean of the measured data and N_t is the number of data used.

In the Table 1 the error levels computed with equation (7) are reported. The first column refers to the Neural Network training with set A, in the second column the error index of the prediction analysis conducted for the set B is shown and in the third column the global error level is presented. The fourth column refers to the Neural Network training error levels for the second test case performed with the two hidden layer structure. In the last two columns the errors of the regression models and the errors found making use of the bilinear interpolation procedure are shown.

For the fuel consumption the accuracy (see Table 1) of the one-layer Neural Network model is higher with respect the accuracy of the regression model, for this case the comparison between measured and computed fuel consumption is shown in the Figure 4.

Table 1: Error index e for the computational analysis performed.

	Neural Network				Regression	
	1 layer			2 layer		
	Train.	Pred.	Glob.	Glob.	Fitting	Interp.
Fuel C.	0.06	0.10	0.07	-	0.10	0.36
HC	0.32	0.53	0.38	0.33	0.27	0.36
CO	0.63	0.93	0.71	0.63	0.64	0.67
NO_x	0.26	0.45	0.34	0.28	0.21	0.63

The analysis of the error indices for the emissions shows similar levels of accuracy for the training (one-layer Neural Network) and for the fitting (regression). Nevertheless, for the Neural Network a reduction in the accuracy is found when the prediction test on the set B has been performed, while an almost stable precision is kept for HC and CO regressions when the bilinear interpolation is made on the same data set. On the other hand for NO_x the interpolation analysis exhibit a considerable increase of the error index.

The second Neural Network structure with two hidden layers has been trained on the complete data set available (set A and B) to check the possibility of improving the fitting capabilities of the Neural Network to be used in the code O.D.E.C.S. for the emission levels computation. The error levels reported in the fourth column of Table 1 show an improvement of the fitting capabilities of the Neural Network used. Moreover, the accuracy level reached is recognizable from the direct comparison between measured and fitted data in the Figures 5, 6 and 7 for HC, CO and NO_x respectively. From the analysis of HC and CO (Figures 5 and 6) emerges the presence of a limited set of high emission levels data corresponding to very rich Air-Fuel ratio working conditions. In further analysis these point will be considered as outlayers and removed from the training sets to eliminate their influence on the training process. For NO_x (Figure 7) an higher global accuracy has been obtained for MLP model with respect the corresponding calculation done with regression model.

As a first application of the Neural Network approach the objective of this work was to analyze the potentiality of the technique and at least to reach the same accuracy level of the previous method based on polynomial regressions. Furthermore, it is worth to remind that for each variable the Neural Network is a single global model while the regression model is composed of 9 independent local polynomial regressions and higher accuracy is not easy to reach for the present applications. On the other hands more appropriate and complex Neural Network structure can be considered to improve the predictivity level attained. Moreover, further enhancements toward the attainment of more accurate Neural Network models can be achieved paying more attention to the mentioned problems of overfitting and overtraining which have not been considered in a systematic way in the course of the present work.

5. Conclusions

The development of Multi Layer Perceptron Neural Network models for the simulation of fuel consumption and emissions (HC, CO, NO_x) in a spark ignition engine has been presented. The main objective of the work was to analyze the potentiality of the Neural Networks and to investigate on the possibilities of replacing the set of polynomial regression based models currently used for the simulation and the optimization of engine control strategies.

A set of more than 500 experimental data have been used to perform two preliminary test cases for different Neural Network structures and the results of the simulations have been compared with previous calculations obtained with regression models. The analysis carried out has shown that for fuel consumption the precision achieved with the Neural Network is higher with respect to the accuracy of regression models. On the other hands for HC and CO comparable error levels are found while for NO_x the Neural Network has performed more accurate results.

Further work is under course to implement the Neural Network models in the complete computational procedure for the optimal engine control strategies calculation as well as to improve their performance. For this purpose, more effort will be devoted in the next future to the investigation of different Neural Network structures and more attention will be given to the problem of overtraining and overparametrization.

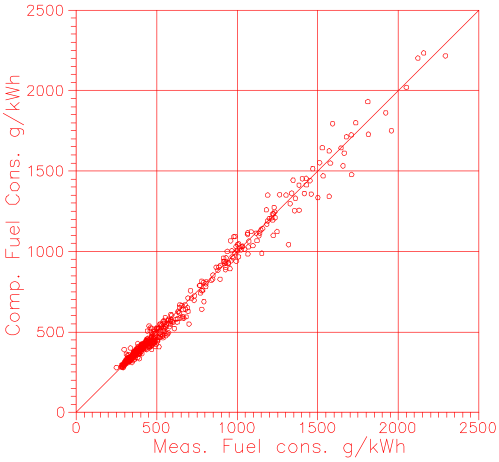

Figure 4: Comparison between computed and measured specific fuel consumption - One-layer Neural Network structure.

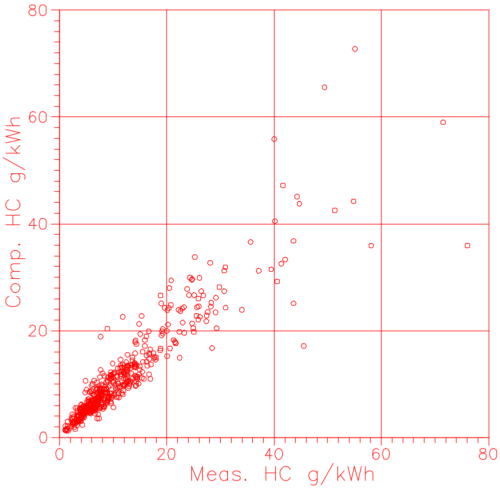

Figure 5: Comparison between computed and measured emission levels for HC - Two-layers Neural Network structure.

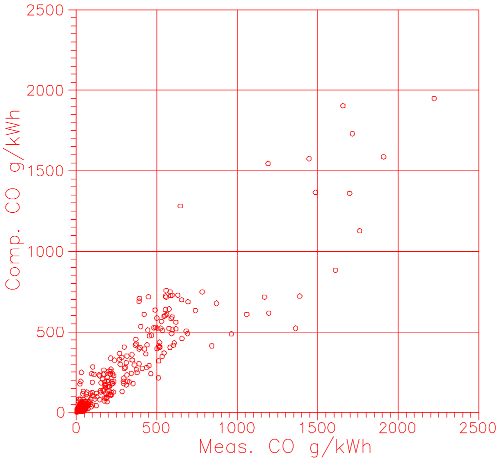

Figure 6: Comparison between computed and measured emission levels for CO - Two-layers Neural Network structure.

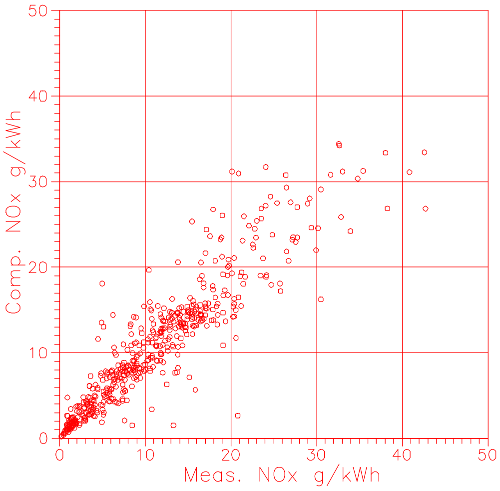

Figure 7: Comparison between computed and measured emission levels for NO_x - Two-layers Neural Network structure.

Acknowledgments

The authors would like to acknowledge the Engine Control Division of Magneti Marelli for the financial and technical support given to the present work. The PhD fellowship of Ivan Arsie is granted by European Union.

References

1. Arsie I., De Franceschi F., Pianese C., Rizzo G., O.D.E.C.S. - A Computer Code for the Optimal Design of S.I. Engine Control Strategies, SAE Paper n. 960359.
2. Arsie I., Gambino M., Pianese C., Rizzo G., Development and Validation of Hierarchical Models for the Design of Engine Control Strategies, published on *Meccanica* **32** pp. 398-408, 1997Kluwer Academic Publisher.
3. Pianese C., Rizzo G., Interactive Optimization of Internal Combustion Engine Tests by Means of Sequential Experimental Design, “3rd Biennial European Joint Conference on Engineering Systems Design and Analysis - ESDA 96”, Petroleum Division of ASME, Montpellier, France, July 1-4,1996.
4. Arsie I., Pianese C., Rizzo G., Identification of Emission Models in a Spark Ignition Engine for Control Applications, “5th IEEE Mediterranean Conference on Control and Systems”, Paphos, Cyprus, July 21-23, 1997.
5. Arsie I., Pianese C., Rizzo G., Models for the Prediction of Performance and Emissions in a Spark Ignition Engine - A Sequentially Structured Approach, SAE Paper 980779 - SP-1330 pp. 59- 73.
6. Pianese C., Rizzo G., A Dynamic Model for Control Strategy Optimization in Spark Ignition Engines, Third A.S.M.E. Symposium on Transportation Systems,

Anaheim, California U.S.A., Nov., 9-13, 1992, DSC-vol 44, pp. 253-267.

7. Gambino M., Pianese C., Rizzo G., Experimental and Computational Analysis of a Dynamic Model for Control Strategy Optimization in a Spark Ignition Engine, “1994 American Control Conference”, Baltimore MD, June 29, July 1,1994, vol 2 pp. 2374 - 2378.
8. Hendricks E., Sorenson S.C., SI Engine Controls and Mean Value Engine Modelling, SAE Paper n. 910258.
9. Gambino M., Pianese C., Rizzo G., Experimental Validation of a Dynamic Model for Mixture Formation in a Multi-Point Injection SI Engine, Proc. 2nd ASME conf. on Engineering System Design and Analysis, London, UK, July, 4-7, 1994, PD, Vol. 64, 8.2 part B, pp. 489-502.
10. Hecht-Nielsen R., *Neurocomputing*, Addison-Wesley, 1987.
11. Patterson D.W., *Artificial Neural Networks - Theroy and Applications*, Prentice Hall, 1995.
12. Shayler P.J., Goodman M.S., Ma T., Transient Air/Fuel Ratio Control of an S.I. Engine Using Neural Networks, SAE Paper n. 960326.
13. Baumann B.M., Intelligent Control Strategies for Hybrid Vehicles Using Neural Networks and Fuzzy Logic, Master Thesis, The Ohio State University, 1997.
14. Ortmann S., Glesner M., Rychetsky M., Tubetti P., Morra G., Engine Knock Estimation Using Neural Networks based on a Real-World Database, SAE Paper n. 980513, SP-1357, pp. 17-31.
15. Ayeb M., Lichetenthäler D., Winsel T., Theuerkauf H.J., SI Engine Modeling Using Neural Network, SAE Paper n. 980790, - SP-1357 pp. 107-115.
16. Holzmann H., Halfmann Ch., Isermann R., Neuro-Fuzzy Modeling of Automotive SI-Engine Characteristics, in proc. of IFAC Symposium Advances in Automotive Control 98, Mohican State Park, Loudonville, OH, USA, February 26 - March 1, 1998, pp. 156-160.

Latin symbols

b	[/] Bias term.
E	[/] MLP training error.
$\dot{e}_j^*$	[g/kWh] Specific emission flow rate.
h	[/] Generic activation function.
$\dot{m}_f^*$	[g/kWh] Specific fuel consumption.
net	[/] Weighted sum of input for the generic neuron.
N_g	[/] Number of Torque-speed experimental grid points.
N_t	[/] Number of experimental data.
T	[Nm] Torque.
w,	[/] Weight of the *j-th* connection.
y_i	[g/kWh] Predicted data.
$\bar{y}^*$	[g/kWh] Mean of the measured data.
y_i^*	[g/kWh] Measured data.

Greek symbols

a	[/] Air-Fuel ratio.
e	[/]Error index.
j	[/] Momentum term constant.
h	[/] Learning rate.
q	[deg] Spark Advance.