PREDICT CONSUMPTION IN INTERNAL COMBUSTION ENGINE USING NEURAL NETWORKS FROM NON-HOMOGENEOUS DATA FOR VEHICLE SIMULATION

Adriano da Mota Santos, adrmsantos@gmail.com
Opencadd Advanced Technology. São Paulo, SP, Brazil.
João Bosco Diniz, joaobdiniz@eaton.com
Eaton Transmissões. Valinhos, SP, Brazil.

Antônio Carlos Canale, canale@sc.usp.br
University of São Paulo. Trabalhador Sãocarlense Avenue, 400, Centro.
São Carlos, SP, Brazil.

Abstract. *Artificial Neural Networks (ANNs) are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the connections between elements largely determine the network function. One can train a neural network to perform a particular function by adjusting the values of the connections (weights) between elements. They are widely used in all engineering, banking, financial and entertainment sectors.This study proposes the application of ANNs in an internal combustion engine model to obtain the specific consumption as a performance feature. This model was applied in entire vehicles simulation, using MatLAB®/Simulink® platform. The engine model in ANNs is based on the engine map and on its main characteristics. Furthermore, it is designed the best net architecture, transfer functions and training styles.To validate the study was mapped a stretch of road, in which an instrumented vehicle provided, besides other variables, the final consumption. These givens were compared within the vehicle simulation using Look-Up Table and ANNs, showing best performance for the latter.*

***Keywords**: neural network, consumption, vehicle simulation, internal combustion engine, look-up table*

1. INTRODUCTION

For years, the automotive industry and several research institutes have developed simulation models to evaluate vehicle performance, evaluating the vehicle as entire, or studying individually each one of its subsystems and components. This need appears during all product development phases: conception, design and validation; providing a cost and time saving and avoiding prototype elaboration.

As sayd by Sayin *et al* (2007), manufacturers and application engineers usually want to know the performance of an engine, or any other element, for the entire range of operating conditions. This requirement can be met either by conducting a comprehensive testing study or modelling the engine operation. Testing the engine under the all possible operating conditions is time consuming and expensive. On the other hand, developing an accurate model for the operation of an engine is too difficult due to the complex processes involved. As an alternative, the performance and exhaust emissions of an engine can be modelled using ANNs.

The structure and function of the ANN attempt to mimic that of the biological neural network (Schalkoff, 1997), as shown in Fig. 1. The most popular fully interconnected ANN consists of a large number of processing units known as nodes or artificial neurons, organized in layers. There are, in general, three groups of node layers, namely, the input layer, one or more hidden layers, and an output layer, each of which is occupied by a number of nodes. All the nodes of each hidden layer are connected to all nodes of the previous and following layers by means of internode synaptic connectors or simply connectors. Each of the connectors, which mimic the biological neural synapsis, is characterized by a synaptic weight.

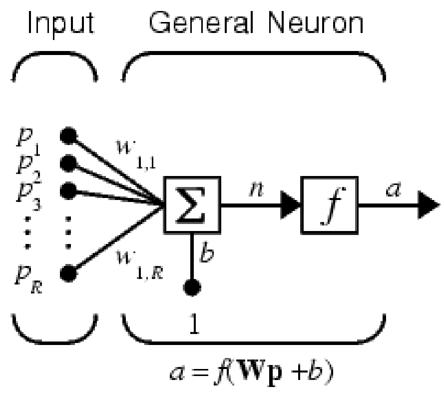

Figure 1. Description of a general neuron, where R is the number of inputs in input vector (Mathworks, 2008b).

Another possibility is the utilization of the well-known tool Lookup Table, Matlab®/Simulink®. The initial motivation of this work is the huge difficulty of use this tool in engine mapping. According, Mathworks (2008a), the Lookup Table (2-D) block computes an approximation to some function z = f(x,y) given x, y, z data points. The Lookup table utilization needs a table data which for each row-variable and column-variable, there is only filled cells. In other words, data-variables as rotation and pressure must be fixed; what doesn't always occur for this case, where sometimes there are different values for the pressure variable between a rotation range and another. A consumption data pre-linearization must be done fixing pressure and rotation values, already causing process errors. In a nut shell, for the Lookup Table utilization, interpolations in the data treatment and other interpolations during simulation to get the desired consumption value are performed, and errors are imminent in this process. There is a strong belief ANNs are most precise than Lookup Table.

The aim of this work is perform a comparative study of average consumption results got from the engine models based on neural network and also on Lookup Table with the real data obtained from the vehicle monitoring in coursed stretch. The study was split in the following steps: travel and road mapping; engine and car data selection; architecture, training and simulation of ANNs; vehicular simulation with look-up table and two best ANNs; comparison and analysis of simulated and real situations.

2. METHODOLOGY

2.1. Vehicular Simulation

2.1.1. Steerability strategy

The orientations given to the motorist, as steerability strategy, were:

- to keep the maximum velocity according to the road signalization, which varies between 80 and 110 km/h;
- to maintain always the same rotation to the gear changes, achieving the plane torque region, that comprehend the range of 1300 to 2300 rpm.;
- as much as possible, not perform stops in road, except for highway tolls.

Figure 2. Map showing (in yellow) the defined route for study.

2.1.2. Travel and Road Mapping

The defined stretch of nearly 290km was mapped in terms of distance, velocity, geographic position and altitude, using a GPS, brand Garmin, model 76CSx, with accuracy less than 10m for distance, 0.05m/s for stead state velocity and from 3 to 5m for altitude. It was taken 23 route laps showed in Fig. 2 and its profile is showed in Fig. 3. The route and travel's mapping information was obtained from Bosco (2009) work and employed as input data in this simulation.

It was weighed the tank before and after the travel to measure the fuel volume spent. The ration between the coursed distance and this volume is the mean consumption performed by the car. Table 1 and Fig. 4 show the results for mean fuel consumption and per sample, respectively.

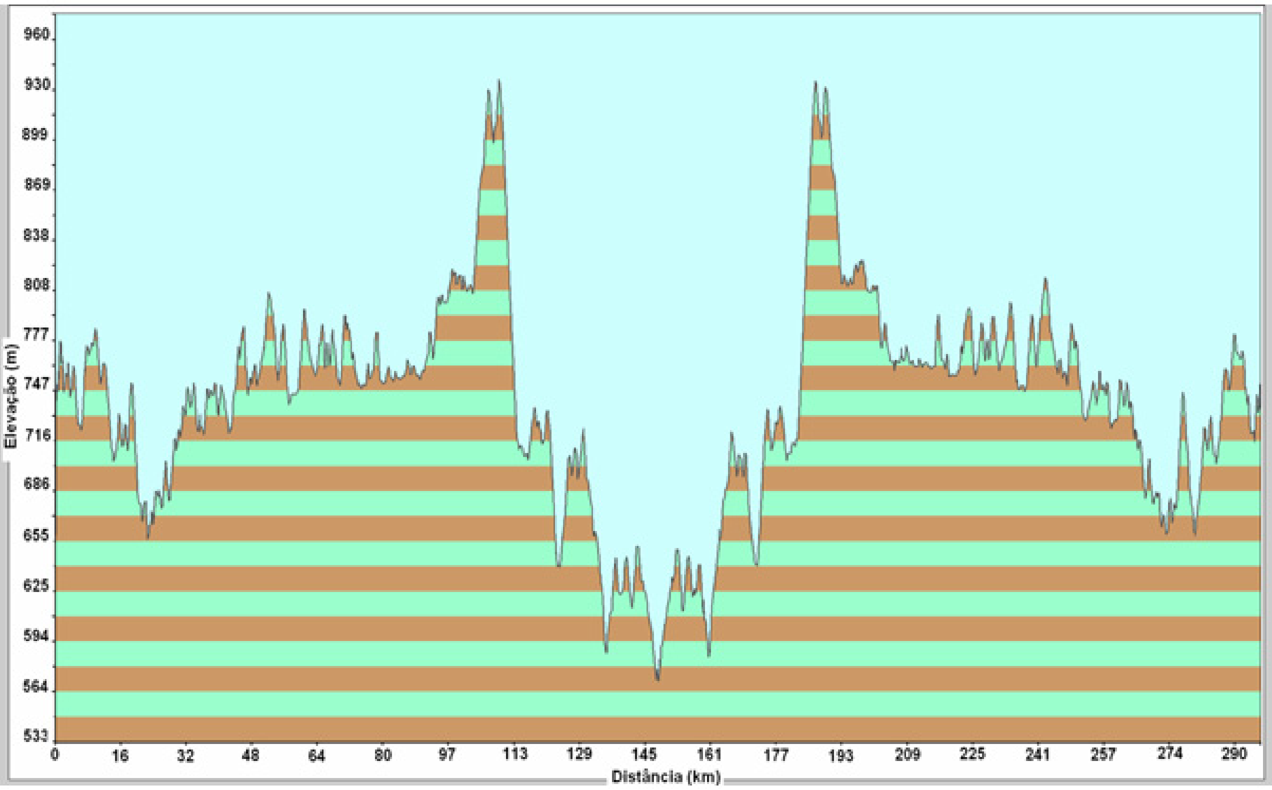

Figure 3. Mapped stretch profile (distance x altitude).

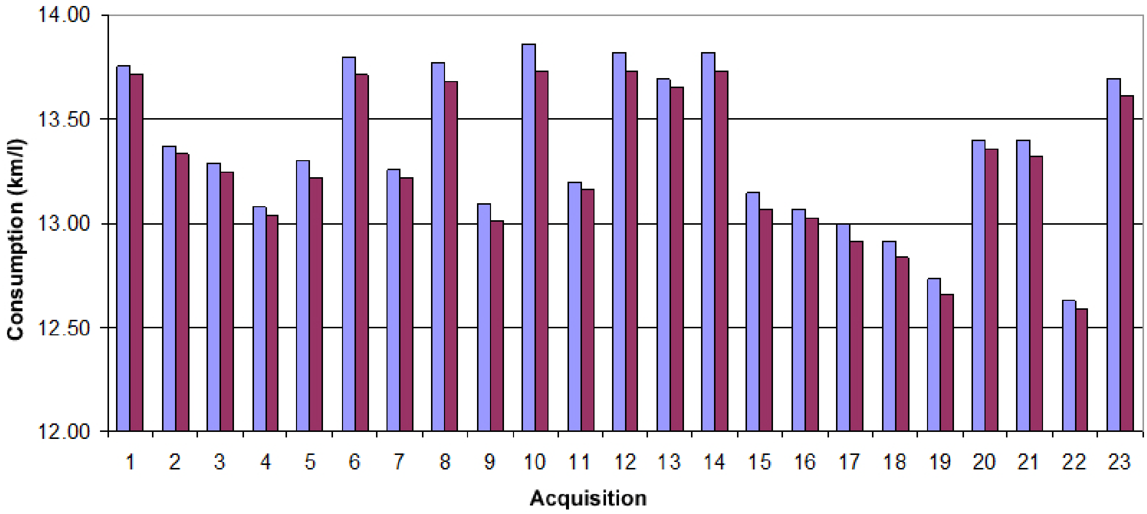

Figure 4. Consumption: odometer (blue) x GPS (burgundy)

Table 1. Comparative data between the consumption obtained by odometer and GPS.

	Odometer	GPS	Error (%)
Mean Consumption (km/l)	13,35	13,28	0,5
Standard Deviation	0,369	0,361	

The difference between the mean consumption calculated by odometer against GPS is owing to the tire slipping, indicating a greater mean distance coursed than GPS. The value used as comparative to simulated results was the 13,28 km/l, relative to GPS.

2.1.3. Vehicular Simulation Software

The validation of the best neural models was made by comparison of the simulated data with the data obtained from the road tests, using the vehicular simulation commercial software AutoSIM® (Opencadd, 2008). The software has rigid axis suspension, springs and dampers both nonlinear, suspension geometry by roll center and pitch center, nonlinear tire model, suspended mass and non-suspended mass, brake system model and automatic transmission, mapped engine (Lookup Table/ANN). It has also available a controlled system (PID) of Brake Mean Effective Pressure (B.M.E.P.) by engine and brake actuators, which reproduce the GPS-mapped stretch in terms of velocity, position and elevation.

AutoSIM® was already applied in the present vehicle, being validated for fuel consumption by Bosco (2009). Figure 5 shows the software velocity control to follow the mapped velocity information.

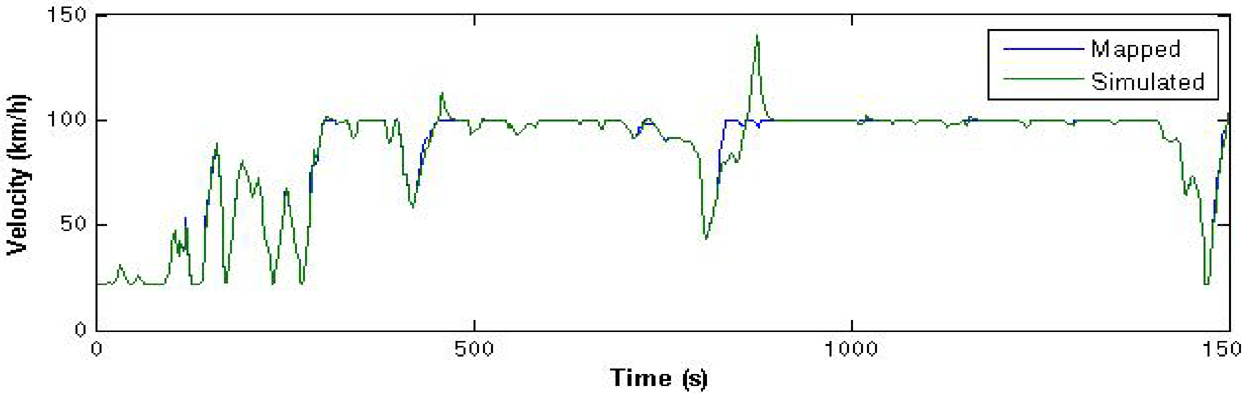

Figure 5. Mapped Velocity x Simulated Velocity.

2.2. Engine and Car Data Selection

In this step, was selected the necessary van parameters to vehicular simulation input data. The vehicle used for study was a mid-size van, double cabin, 4x2-rear traction, curb weight and total mass of 1860 kg (1101 kg on front axle and 743 kg on rear axle). Table 2 shows the engine information used in the modeling of this work.

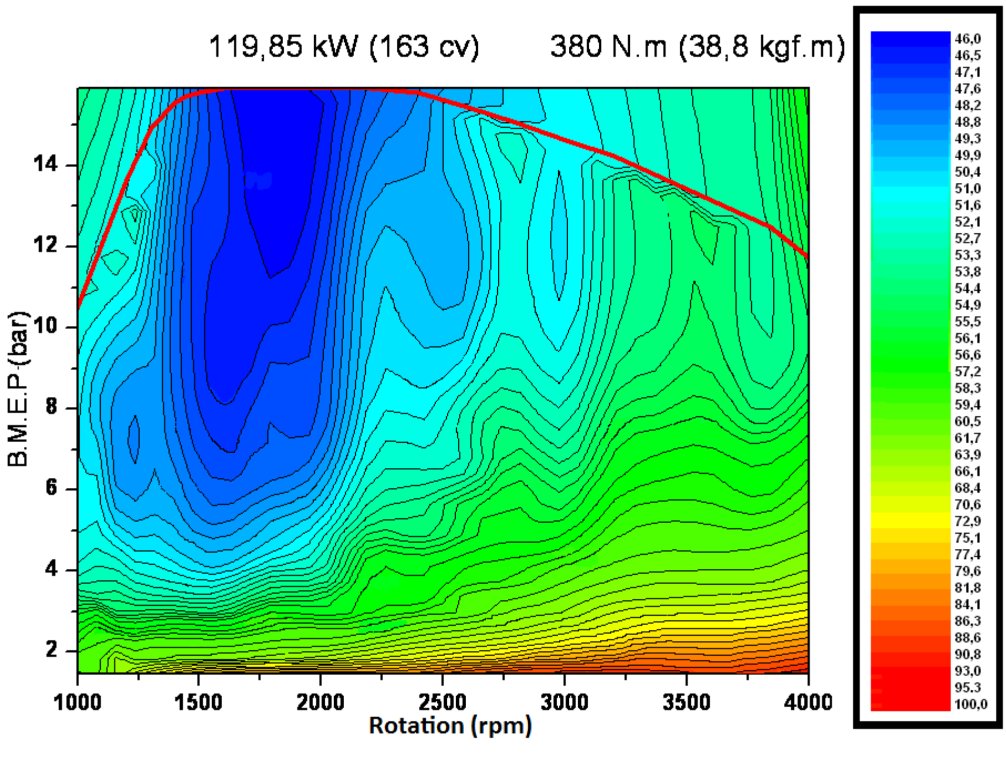

Figure 6. Graphical illustration of break specific fuel consumption, in % of maximum consumption.

The engine map data, B.M.E.P., engine speed and Break Specific Fuel Consumption (B.S.F.C), was experimental gotten in dynamometer tests, showed in Fig. 6. It was fixed 16 rotation ranges, between 1000 and 400 rpm, and measured the torque.

Table 2 - Specifications of the test engine.

Number of Cylinders	4 (in line)
Cycle	Diesel, 4 strokes
Total Cylinder Volume	3,0 l liters
Maximum Free Rotation	4640 rpm
Idle Gear Rotation	800 rpm
Power (NBR5484)	163cv@3800 rpm
Torque (NBR5484)	380 Nm@1600-2200 rpm

2.3. Lookup Table (2-D)

The same artificial neural network input data were employed to feed a Lookup Table block, needing just to do a pre-linearization in the data to eliminate the empty table cells.

2.4. ANNs: Architecture, training and simulation

This is a central step for this design because it corresponds to the data standardization and to the study of the several architectures, training techniques and transfer functions, based on the studies of Arcaklioğlu and Çelikten (2005), Sayin *et al* (2007), Cruz and Jiménez (2007). Regarding the architecture, networks were tested with one (5, 10, 15, 20 neurons) and two hidden layers (5-10, 5-15, 10-5, 20-5 neurons). For the hidden layers, a hyperbolic tangent sigmoid transfer function (tansig) was used, contradictory to the output layer, which used a saturating linear transfer function (satlin). Regarding the training, several styles were tested: Levenberg-Maquardt Backpropagation (trainlm), Gradient Descent with Momentum & Adaptive Learning Backpropagation (traingdx) and the Scaled Conjugate Gradient Algorithm (trainscg). To compare the performance was fixed the number of epochs for training in 100.

The combination among the different architectures and training styles results in a great variety of networks. In order to avoid the training and evaluation of so many networks, all the combinations among architecture and training for networks with one hidden layer were initially analysed, resulting in 12 different networks. The training style that had better performance was then used for the 2 layers, resulting in 4 more networks, giving a total of 16 network test sets. The ANN is normally used with random and well-distributed data, reaching the maximum number of possible behaviours for the phenomenon that one wishes to model or simulate. Considering that the purpose of this study is the utilization of a usual engine map, a problem shows up regarding the homogeneous data, since the rotation is fixed in the dynamometer, when measuring the torque. Therefore, the input 'Rotation' provides data for each 100 rpm and not in a random ideal form, causing some loss for the intermediary output data. However, comparing with the LookUp Table, for instance, the ANN would still have a better performance. Thus, the attempt to choose the architecture and training has the purpose of minimizing these intermediary errors.

2.5. Performance evaluation of ANNs

Correlation coefficient, mean relative error and root mean square error were applied for measuring the network performance for each rotation range and subsequently calculated a mean value (Bechtler *et al*, 2001, Lucas *et al*, 2001, Yuanwang, *et al*, 2002).

In order to calculate the correlation coefficient, a mean value of the 16 rotation ranges was calculated, as well as for the 4 rotation ranges, which have the greatest error level due to the reduced amount of points.The correlation coefficient assesses the strength of the relationship between the predicted and experimental results, and it is defined (Looney, 1997) as Eq. (1).

$$R(a,p) = \frac{cov\,(a,p)}{\sqrt{cov\,(a,p)cov\,(p,p)}} \qquad (1)$$

Where cov(a,p) is covariance between a and p sets that refer to the actual output and predicted output sets, respectively. The correlation coefficient ranges between -1 and +1. R values closer to +1 indicate a stronger positive linear relationship, while R values closer to -1 indicate a stronger negative relationship.

Equation 2 calculates the mean linear correlation coefficient among the rotation ranges.

$$R_{nf} = \frac{1}{NF}\sum_{i=1}^{NF} R_i \qquad (2)$$

Where NF is the number of rotation ranges and R_i is the correlation coefficient for the rotation range i. As stated previously, R_{nf} was calculated for two values of NF, 4 and 16.

The mean relative error shows the mean ratio between the error and the experimental values. Its average value for all the rotation ranges is determined from Eq. (3).

$$MRE_{nf}(\%) = \frac{1}{NF}\sum_{j=1}^{NF}\left[\frac{1}{N_j}\sum_{i=1}^{N_j}\left|100\frac{(a_{ij}-p_{ij})}{a_{ij}}\right|\right] \quad (3)$$

Where Nj is the number of the points in the data set for the range j. Finally, the root mean square error, calculated in function of all the rotation, is obtained from Eq. (4).

$$RMSE_{nf} = \frac{1}{NF}\sum_{j=1}^{NF}\sqrt{\frac{1}{N_j}\sum_{i=1}^{N_j}(a_{ij}-p_{ij})^2} \quad (4)$$

3. RESULTS AND DISCUSSION

When analysing the output data of the 16 types of neural networks tested and after comparing them with the real data from the dynamometer, performance results were obtained. The Fig. 7 illustrates the training curves for one of the simulated networks and the projection of the data predicted for this network with the real data for a certain rotation range.

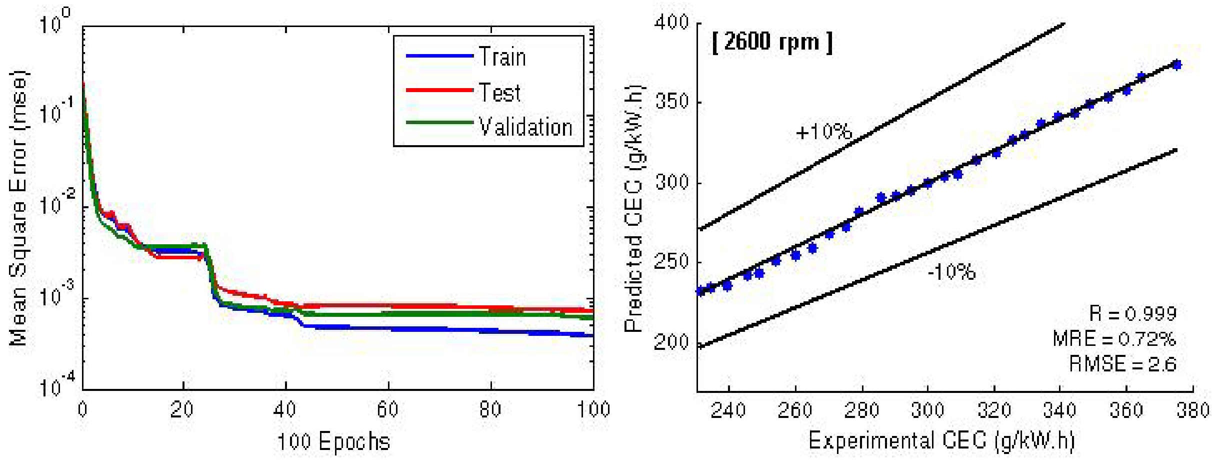

Figure 7. Left: Behavior of error during net training. Right: Example of net predicted data compared with experimental data for 2600 rpm.

Projections like that were made for each one of the 16 rotation ranges and the statistical results were consolidated, as shown in Tab. 3.

Table 3. Best results from all 16 ANNs training process.

Code	Architecture (Number of Neurons)	Training	R_4	R_{16}	MRE_{16}	$RMSE_{16}$
ANN 1	20	trainlm	0,911	0,981	1,185	4,222
ANN 2	5-15	trainlm	0,978	0,987	2,582	8,377
ANN 3	10-5	trainlm	0,955	0,990	1,246	4,142
ANN 4	15	trainscg	0,963	0,983	3,055	9,940

It was selected, from Tab. 3, the networks 1 and 2 to be inserted into the vehicle simulations because they had the lowest MRE_{16} and $RMSE_{16}$ and the greatest R_4 and R_{16}, respectively. Figure 8 and Fig. 9 show the statistical studies comparing the givens obtained from the networks with the real givens obtained from the dynamometer.

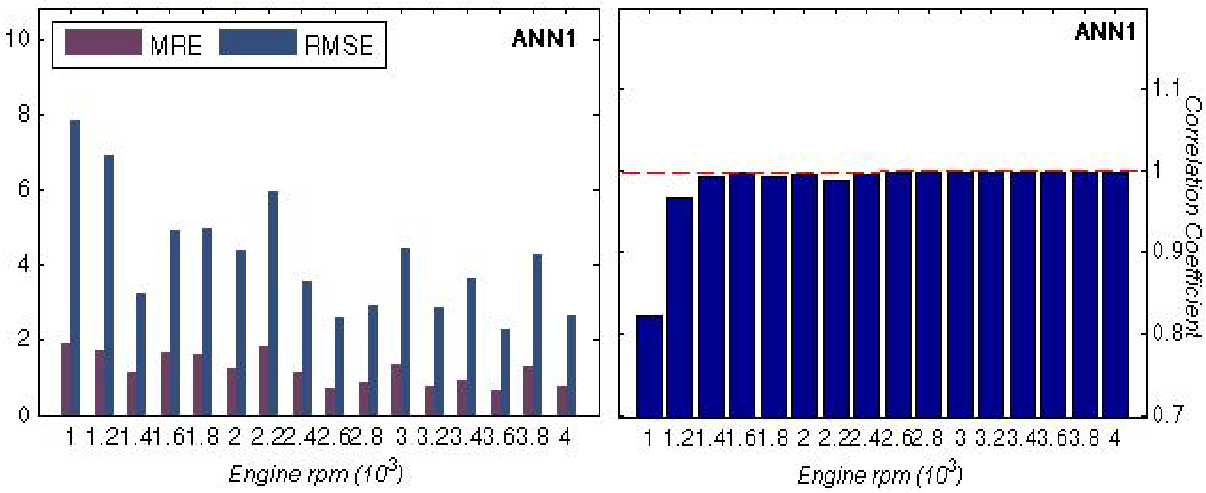

Figure 8. Statistical results for net architecture ANN1.

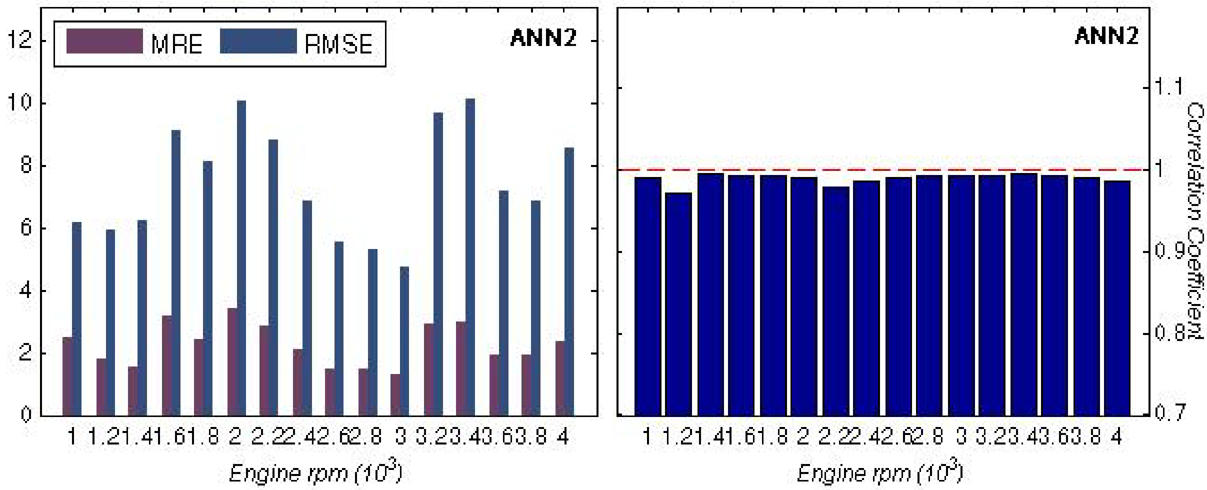

Figure 9. Statistical results for net architecture ANN2.

In order to validate the use of artificial neural network to predict the average fuel consumption of vehicles, the blocks of the two ANNs were generated and inserted into de vehicle simulator, in the sub-module ENGINE. Three scenarios were then simulated: considering each one of the ANN and utilizing the Lookup Table with the data obtained from the dynamometer. The Tab. 4 compares the results of the simulations with the real data from the road mapping.

Table 4. Final results after vehicle simulation.

Test	Final Consumption (km/l)	Relative Error (%)
Real (Not a Simulation)	13,28	-
Look-up Table	11,04	16,87
ANN 1	12,76	3,92
ANN 2	12,19	8,21

Considering as reference the real results, the artificial neural network ANN1 was the model that most had gotten near, with relative error of 3,92% against 16,87% from Lookup Table propose. If the same comparison is done by ANN1 against ANN2, the result verified is a difference of approximately 4,3%. This illustrates that a greater number of hidden layers and neurons has not shown to be more effective. Moreover, the proposed model ANN1 had a greater performance, as well as simpler architecture.

Figure 10 illustrates the consumption (km/l), the rotation (RPM) and the velocity of the vehicle (km/h) during the route. Figure 11, on the other hand, illustrates the same information, but only for the firsts 3500 meters. It is possible to observe that the discrepancy among the three models occurred after the firsts 1500 meters of the simulated route. It is also important to observe that during the first kilometre of the route the car maintained low velocity, with several stops and, consequently, several gear changes. In this steering profile the two simulations based on ANNs presented the same result, whereas when approaching the first kilometre all three models indicated the same average consumption.

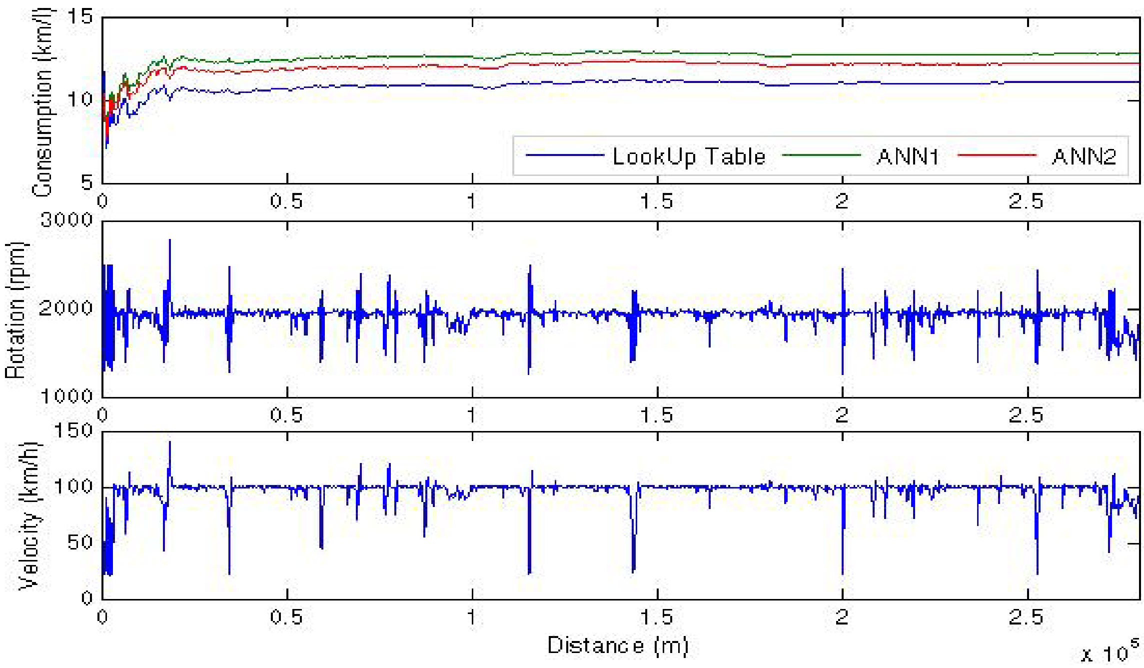

Figure 10. Consumption, rotation and velocity from vehicle simulation during all road stretch.

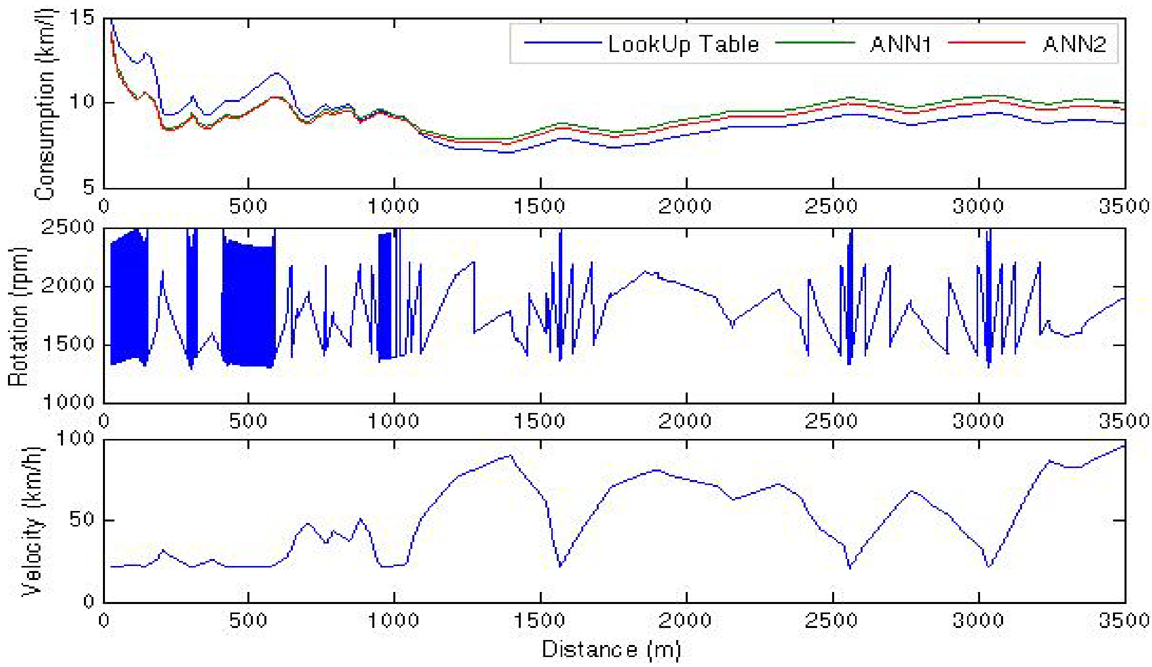

Figure 11. Consumption, rotation and velocity from vehicle simulation focusing on the firsts 3,5km.

4. CONCLUSION

The employment of tools based on artificial neural networks technique is, and must be, widely adopted in engine models for vehicle simulation, owing to the lack or inefficiency of other tools. The best ANN model obtained by this study has proved to be approximately 12,95% more precisely than the Lookup Table. The greater difference between this study and other studies, which have used the ANN for the prediction of performance/gas emissions, is the utilization of heterogeneous engine data. All the data employed was obtained from a usual engine map, acquired with the aid of a dynamometer. These maps are also given, by the manufactures, to the engineering staff of the car assembler companies, for the development of their products.

This present study has focused on predicting the consumption of fuel, as a factor of performance, from an usual engine map; however the same procedures could be used for other performance factors (torque, power) and gas emission (NO_X, CO_2 and SO_2).

5. REFERENCES

Arcaklioğlu, E. and Çelikten, İ., 2005, "A Diesel Engine's Performance and Exhaust Emissions", Applied Energy, Vol. 80, pp. 11-22.

Bechtler, H., Browne, M.W., Bansal, P.K., Kecman, V., 2001, "New approach to dynamic modelling of vapourcompression liquid chillers: artificial neural-networks", Applied Thermal Engineering, Vol. 21, pp. 941–953.

Bosco, J.D., 2009,"Desenvolvimento e Validação de um Software para Simulação de Consumo de Combustível Veicular", University of São Paulo, Brazil (in press).

Cruz, F. and Jiménez, F.J., 2007, "Design and Optimization of Neural Networks to Estimate the Chamber Pressure in Internal Combustion to Estimate the Chamber Pressure in Internal Combustion Engines by an Indirect Method", Energy & Fuels, Vol. 21, pp. 2627-2636.

Cui, Z., Zhang, W., 2008, "An Adaptive Neural Network System for Prediction of Thermal Protective Performance of Fabrics", Intelligent System and Knowledge Engineering, Vol. 1, pp. 837 – 841.

Looney, C.G., 1997, "Pattern Recognition Using Neural Networks; Theory and Algorithms for Engineers and Scientists", Oxford University Press, New York, USA.

Lucas, A., Duran, A., Carmona, M., Lapuerta, M., 2001, "Modeling diesel particulate-emissions with neural networks", Fuel, Vol. 80, pp. 539–548.

Mathworks, September 17 2008a, "Matlab Help - Lookup Table (2-D)", Software Version 7.7.0.471 (R2008b). 12 may 2009.

Mathworks, September 17 2008b, "Matlab Help - Neural Network Toolbox, Architecture", Software Version 7.7.0.471 (R2008b). 12 may 2009.

Opencadd A.T., 2008, "AutoSIM® - Vehicle Simulation Software". Brazil.

Sayn, C., Metin, H.E., Hosoz, M., Kilicaslan, I., Canakci, M., 2007, "Performance and Exhaust Emissions of a Gasoline Engine Using Artificial Neural Network", Applied Thermal Engineering, Vol. 27, pp. 46-54.

Schalkoff, R., 1997, Artificial Neural Networks, McGraw-Hill, New York.

Yuanwang, D., Meilin, Z., Dong, X., Xiaobei, C., 2002, "An analysis, for the effect of cetane number on exhaust emissions from an engine, using a neural network", Fuel, Vol. 81, pp. 1963–1970.