

Predicting the fuel flow rate of commercial aircraft via multilayer perceptron, radial basis function and ANFIS artificial neural networks

T. Baklacioglu 

tbaklacioglu@eskisehir.edu.tr

Eskisehir Technical University

Eskisehir 26555

Turkey

ABSTRACT

A first attempt is made to use recently developed, non-conventional Artificial Neural Network (ANN) models with Multilayer Perceptron (MLP), Radial Basis Function (RBF) and Adaptive Neuro-Fuzzy Interference System (ANFIS) architectures to predict the fuel flow rate of a commercial aircraft using real data obtained from Flight Data Records (FDRs) of the cruise, climb and descent phases. The training of the architectures with a single hidden layer is performed by utilising the Delta-Bar-Delta (DBD), Conjugate Gradient (CG) and Quickprop (QP) algorithms. The optimum network topologies are sought by varying the number of processing elements in the hidden layer of the networks using a trial-and-error method. An evaluation of the approximate fuel intake values against the ideal fuel intake data from the FDRs indicates a good fit for all three ANN models. Thus, more accurate fuel intake estimations can be obtained by applying the RBF-ANN model during the climb and descent flight stages, whereas the MLP-ANN model is more effective for the cruise phase. The best accuracy obtained in terms of the linear correlation coefficient is 0.99988, 0.91946 and 0.95252 for the climb, cruise and descent phase, respectively.

Keywords: aircraft; fuel flow rate; artificial neural networks; radial basis function; adaptive neuro-fuzzy interference system; flight data records

NOMENCLATURE

Abbreviations

ANFIS	adaptive neuro-fuzzy interference system
ANN	artificial neural network
BP	back-propagation

CG	conjugate gradient
CV	cross-validation
FFNN	feed-forward neural network
FIS	fuzzy inference system
HL	hidden layer
nMSE	normalised mean squared error
MAE	mean absolute error
MAPE	mean absolute percentage error
MLP	multilayer perceptron
MSE	mean squared error
PE	processing elements (neurons)
QP	Quickprop
RBF	radial basis function
T	training
TSK	Takagi–Sugeno–Kang fuzzy model

Symbols

a_i, b_i, c_i	consequent parameters in ANFIS
A_i, B_i	linguistic labels in ANFIS
c_j	centre (vector) of j^{th} hidden neuron
f	transfer function
h_j	output of j^{th} hidden neuron
m	number of neurons in hidden layer
n	number of data
O_i^j	output of i^{th} node in layer j
R	linear correlation coefficient
w_i	firing strength of a rule in ANFIS
w_{k0}	bias
w_{kj}	synaptic weight connecting j^{th} hidden neuron and k^{th} output neuron
x	n -dimensional input
y_k	k^{th} output of ANN
Y_{actual}	actual value of ANN model output
Y_{mean}	mean value of ANN model output
$Y_{\text{predicted}}$	predicted ANN model output
μ	membership function in ANFIS
ϕ	radial basis function

1.0 INTRODUCTION

Recently, the climatic and economic impacts of increasing economic growth and output have led to a challenging trade-off. Regardless of any enhancements in the efficiency of industrial systems, any percentage increase in terms of economic performance escalates the

impact on the climate as a result of the irregular progress towards greater economic growth. Moreover, in addition to legislation aimed at environmental protection, industry now has more flexible choices, such as the consideration of substitute fuels, ground-based activities, and comparatively minimal safety and back-up requirements, with aviation having high specific limitations which ultimately complicate the achievement of the trade-off necessary to alleviate the climatic effect. Furthermore, given the significance of the aviation sector in steering the economy, it is highly resistant to ecological considerations.

Currently, there are two key issues associated with air transportation: environmental impact and efficiency. Efficiency includes aspects such as fuel intake, time and the most favourable use of resources within the airport and terminal zone, such as slots, continuous descent operations and direct routing. On the other hand, noise and emissions are linked to the environmental impact. Regardless of the considerable progress in engine and airframe design, inefficient air traffic management, resulting from capacity concerns, has an adverse effect on flight efficiency, such as fuel or time cost, thus escalating emissions.

Several effects influence the fuel consumption during a particular flight, including the weather conditions, flight speed and altitude, mass and air traffic management. Equally, the intricate relationships between these variables make it quite difficult to measure the size of each variable. Nevertheless, note that several studies have been conducted on the estimation of the sole impact of a given variable on fuel intake. Additionally, each flight phase offers diverse possibilities for fuel savings, which must be weighed against aspects such as time, cost, capacity, noise and contrail formation. When considering the engine power, it is observed that the fuel flow rate is considerably high in the climb phase, which has a limited duration. Therefore, limitations implemented by air traffic management may lead to inefficient operations with the aim of attaining decreased noise levels^(1,2).

To alleviate the harmful environmental effects associated with fuel consumption, it is imperative to preserve fuel energy sources, curtail flight costs, obtain highly precise estimations of aircraft trajectories and achieve faultless and efficient air traffic management. In the present state of commercial aviation worldwide, this requires a precise fuel flow rate model for turbofan engines⁽¹⁾.

The primary literature approaching such predictions of fuel intake comprises the model in Ref. (3), which was considered in the SIMMOD (Airport and Airspace Simulation Model) of the FAA. Later, an Artificial Neural Network (ANN) model was applied in the prediction of the fuel intake by Trani et al.⁽⁴⁾ based on flight data for a Fokker F-100 aircraft. Consistent with Senzig et al.⁽⁵⁾, given that this model requires comprehensive aerodynamic information or an extensive database of aircraft operations, along with data on the associated state of the aircraft, it has met with limited acceptance. Regardless of the accurate results of the ANN model proposed by Trani et al.⁽⁴⁾, it is necessary to improve this model, including the form of its input and the optimisation of the modelling architecture. In their study, Trani et al.⁽⁴⁾ established the impact of variation of the Mach number as an input parameter, including the only primary and final altitudes but excluding the impact of altitude variation in the model's input parameters. Furthermore, the additional input parameters defined included the initial temperature and weight of the aircraft. It is critical to note that a definite air range was provided as an output parameter, but not the fuel flow rate, while modelling the cruise flight phase. Eight candidate topologies were tested, using a sensitivity analysis, with a three-layer model with eight neurons in the initial two layers plus a single neuron in the third (output) layer finally being chosen. The number of layers and neurons therein were selected using a trial-and-error approach rather than optimisation of the ANN model architecture. In their study, Bartel and Young⁽⁶⁾ developed a thrust-specific fuel intake model for the cruise phase.

This model included the variation of the thrust-specific fuel intake with the Mach number, but excluded the impact of the temperature ratio. The temperature term was replaced by using a factor considered to be the ratio between the thrust in a given cruise condition versus a reference condition. Equally, Senzig et al.⁽⁵⁾ examined a fuel intake model based on the terminal area. Their model based on the thrust-specific fuel intake was proposed by Hill and Peterson and modified by Yolder. For the terminal area, the model further included the Mach number dependence, the temperature ratio effect and the net thrust correlated with the pressure ratio. A novel fuel intake model based on the Genetic Algorithm (GA) was proposed by Turgut and Rosen⁽⁷⁾, where the descent flight phase considered the fuel flow rate dependence, with only altitude as an input parameter. Except for these models, considering the latest version (3.13) of Eurocontrol's Base of Aircraft Data (BADA) and based on its user manual, the nominal fuel flow was computed by using the thrust and thrust-specific fuel intake, defined as a function of the practical airspeed for all flight phases, excluding the idle descent and cruise conditions. The difference between the expression for the cruise and nominal fuel flows was observed by an extra correction coefficient of cruise fuel flow, with the fuel flow expression in the idle descent condition being considered as a function of the pressure altitude and the fuel flow coefficients during descent⁽⁸⁾. A fuel flow rate estimation model based on the GA was developed by Baklacioglu⁽⁹⁾ for the climb phase, and another model⁽¹⁰⁾, based on GA-optimised Back-Propagation (BP) and Levenberg–Marquart (LM) Multilayer Perceptron (MLP) ANNs was developed for the cruise, climb and descent phases by using flight data records from a transport aircraft. Recently, a statistical model predicting the fuel flow rate of a piston engine aircraft was derived by Huang et al.⁽¹¹⁾ using general aviation flight operational data, including the altitude, ground speed and vertical speed of the aircraft.

The present study is the first attempt in literature to estimate the fuel flow rate of a medium-weight transport aircraft using ANNs with three architectures (MLP, RBF and ANFIS), trained using three different learning algorithms (CG, QP and DBD). The newly designed models in this study take the variation of both the true airspeed and altitude as input parameters and provide the fuel flow rate as an output parameter for the cruise, climb and descent flight stages. A unique aspect of this research is that the projected models are built using real raw flight data records from a Boeing 737-800 operated by a local airline in Turkey. Flight altitude and real airspeed variations were included as inputs to the modelling approach to avoid the need for comprehensive information on aircraft data and operation. This facilitates precise fuel flow rate modelling using a simplified method.

2.0 ANN MODELLING ARCHITECTURES

2.1 MLP-ANN

The multilayer perceptron (MLP) topology comprises an output layer, an input layer and one or more layers of hidden neurons⁽¹²⁾. Accepting input signals from the outside world, the input layer distributes these signals to all the neurons in the hidden layer(s), which identifies patterns in the input and symbolises them through their weights. The output pattern from the network is then proposed by the output layer^(12–16).

The overall error of the network is computed and the network weights thus determined by employing three diverse learning algorithms, viz. CG, QP and DBD, in this study. As an example, the general structure of the learning procedure of a feed-forward ANN is illustrated in Fig. 1. Based on the forward propagation process, the data are distributed from the input

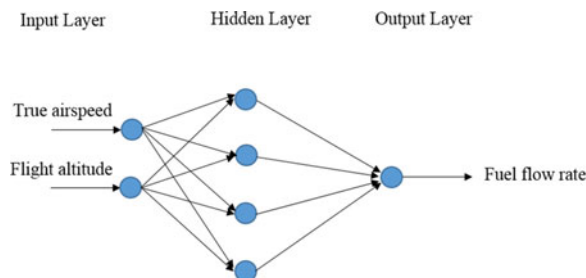


Figure 1. General structure of the feed-forward ANN model for fuel flow rate prediction.

layer and measured by the hidden layer to be passed onto the output layer, where the result corresponding to the input of the network is obtained.

If an error is identified between the actual and preferred output values, it is propagated layerwise through the network weights and thresholds from the output layer via the hidden layer to the input layer, with the objective of reducing the error. Based on this backward error propagation, the network weights and thresholds are constantly modified, thereby improving the accuracy of the network in reaction to the input data. The primary objective of this learning procedure is to amend and update the link weights and thresholds to achieve suitable values based on constant training. However, excessive or inadequate training may prevent an optimum outcome and even lead to phenomena such as overfitting or decreased generalisability of the network⁽¹⁵⁾.

2.2 RBF-ANN

The RBF network is a type of FFNN comprising a single hidden layer, an input layer and an output layer^(12, 17). The neurons in the input layer are openly linked to the neurons in the hidden layer. Employing non-linear radial basis functions, the hidden layer transforms the data received from the input space to the hidden space. Within an RBFN, a radially symmetrical activation function is used in the hidden layer. The Gaussian function, which is the natural choice for this activation process, is applied in this study. Considering an n -dimensional input $x \in R^n$ to an RBFN, the output of the j^{th} hidden layer can be expressed as

$$h_j(x) = \phi_j (\|x - c_j\|), \quad j = 1, 2, \dots, m \quad \dots (1)$$

Here, c_j denotes the centre (vector) of the j^{th} hidden neuron, m symbolises the number of neurons in the hidden layer and $\phi(\cdot)$ signifies the radial basis function. A linear activation function is employed in the output layer. A weighted sum of the output of each neuron in the hidden layer is linked to a specific neuron in the output layer, thus the k^{th} output of the neural network is

$$\hat{y}_k(x) = \sum_{j=1}^m w_{kj}h_j(x) + w_{k0} \quad \dots (2)$$

where w_{kj} represents the weight linking the j^{th} hidden neuron to the k^{th} output neuron, w_{k0} stands for the partiality and m is the number of neurons in the hidden layer. Throughout the training procedure of the RBFN, the parameters describing the (centre and width) of the Gaussian functions and the values of the weights between the hidden and output layers are

amended. The weights are optimized by using the least mean square algorithm, while the centres might be allocated randomly or measured by the application of various clustering algorithms.

2.3 ANFIS

The adaptive network-based fuzzy inference system (ANFIS) developed by Jang is one of the main, widely applied, fuzzy inference systems. The architecture of this system is based on the introduction of a fuzzy inference system (FIS) into the structure of an adaptive neural network. The first-order Takagi–Sugeno model is implemented for the ANFIS topology in this research^(12, 18). The general rule set, as well as the base fuzzy if–then rules, are given as follows:

$$\begin{aligned} \text{If } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } B_1 \text{ then } f_1 &= a_1x_1 + b_1x_2 + c_1 \\ \text{If } x_1 \text{ is } A_2 \text{ and } x_2 \text{ is } B_2 \text{ then } f_2 &= a_2x_1 + b_2x_2 + c_2 \quad \dots (3) \end{aligned}$$

The ANFIS topology comprises five layers, each consisting of a number of nodes depending on their function. Let O_i^j denote the output of the i^{th} node within layer j .

First Layer: Each node within the initial layer is an adaptive node. The outputs of the first layer are the fuzzy membership degrees of the inputs, as shown below:

$$O_i^1 = \mu_{A_i}(x_1) \quad i = 1, 2 \quad \dots (4)$$

$$O_i^1 = \mu_{B_{i-2}}(x_2) \quad i = 3, 4 \quad \dots (5)$$

Here A_i and B_i signify the linguistic labels and μ_{A_i} and μ_{B_i} denote the corresponding membership functions. Constant and piecewise differentiable functions such as trapezoidal, Gaussian, triangular, and generalised Bell membership functions are frequently employed for the nodes in this layer. The outputs of this layer define the membership values of the principle part and the active parameters within the membership functions of the fuzzy sets, which are known as premise parameters.

Second layer: The nodes in layer 2 are defined in contrast to layer 1. The output O_i^2 from node i is given as

$$O_i^2 = w_i = \mu_{A_i}(x_1) \cdot \mu_{B_i}(x_2) \quad i = 1, 2 \quad \dots (6)$$

where w_i denotes the firing strength of a function.

Third layer: The normalisation procedure is accomplished in layer 3, where the nodes are fixed. For each node, the ratio of the firing strength of the i^{th} rule to the firing strength of the sum of all rules is calculated. The outputs from this layer are thus called standardised firing strengths, expressed as

$$O_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2} \quad i = 1, 2 \quad \dots (7)$$

Fourth layer: The next section of the fuzzy rule is addressed in layer 4. Each node i within this layer is an adaptive node that computes the contribution of the i^{th} rule to the output function of the model. According to the first-order Takagi–Sugeno technique, the output O_i^4 of node i is specified as

$$O_i^4 = \bar{w}_i f_i = \bar{w}_i (a_i x_1 + b_i x_2 + c_i) \quad i = 1, 2 \quad \dots (8)$$

where $\{a_i, b_i, c_i\}$ are fixed parameters. The parameters for this layer are known as the resultant parameters.

Fifth layer: Layer 5 comprises a single set node. This is the final layer, where all the incoming signals are combined and the output is generated as

$$O_i^5 = y = \sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad \dots (9)$$

The final output can be formulated as a linear combination of the resultant parameters that five the values of the premise parameters in the derived ANFIS topology. Therefore, the output y can be expressed as

$$y = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 = \bar{w}_1 f_1 + \bar{w}_2 f_2 = (\bar{w}_1 x_1) a_1 + (\bar{w}_1 x_2) b_1 + \bar{w}_1 c_1 + (\bar{w}_2 x_1) a_2 + (\bar{w}_2 x_2) b_2 + \bar{w}_2 c_2 \quad \dots (10)$$

To optimise the resultant parameters with fixed principle parameters, the least square method (forward pass) is applied during the training phase. The backward pass starts as soon as the optimal values for the resultant parameters are obtained. The gradient descent method (backward pass) is applied to amend the premise parameters in the optimal manner with respect to the fuzzy sets in the input domain. Using the resultant parameters obtained in the forward pass, the ANFIS output is then obtained. According to the learning algorithm, the output error is then applied to amend the premise parameters⁽¹²⁾.

3.0 ANN METHODOLOGY AND APPLICATION

ANN models for the fuel intake were developed using randomly chosen flight data records for a Boeing 737-800 aircraft operating by Pegasus Airlines, a domestic airline in Turkey. The evaluated raw data comprises engine speeds, Mach number, flight altitude, fuel flow rate and flight time, which includes the inputs for the considered ANNs, viz. the true airspeed and flight altitude. When these data were applied as input parameters, the MLP, RBF and ANFIS architectures output the fuel flow rate value corresponding to the same instant of the input flight data records. Table 1 presents the corresponding values of the input and output factors for the FDRs containing 347, 404, and 483 data points in the climb, cruise and descent phase, respectively. Before training the neural networks, all the input variables and the objective function of each of the three models were normalised to lie in the interval [0–1]. The implementation of the CG, DBD and QP learning algorithms involved data pre-processing, randomisation and splitting into training (60%), cross-validation (15%) and testing (25%) groups. Modelling of non-linear problems using ANNs requires the execution of various phases, which are affected by the characteristics of the preceding and subsequent phases⁽¹⁹⁾. To choose the correct configuration of the ANN for estimating the model output parameters, a statistical analysis was employed. The ANN design requires the description of the inputs, network types, topology, training standard and transfer functions. Specifically, the modelling process can essentially be categorised into three phases: The first phase includes the design of the network topology by considering the type of ANN, the input parameters, and the numbers of hidden layers and

Table 1
Number of data and ranges of model parameters

Parameters and data	Climb	Cruise	Descent
Number of data	347	404	483
Flight altitude (ft)	10,056–29,974	31,998–32,036	7,028–32,004
True airspeed (m/s)	150.795–219.36	216.95–225.382	142.08–224.212
Fuel flow rate (kg/h)	4,113.158–6,621.183	1,465.025–2,883.552	146.436–1,465.025

neurons. This phase also entails the choice of the transfer function, the training and validation samples, as well as the training algorithm. The next phase involves a training step, where data are applied to the ANN models to regulate their weights and biases as a function of a predefined condition. The final phase is the testing step, in which the ANN models are tested using a novel dataset and their accuracy evaluated based on statistical parameters.

ANNs are described as intelligent systems with the potential to learn, memorise and form relationships among data, becoming a non-linear tool for time-series modelling⁽²⁰⁾. The importance of each input variable is related to the outputs of interest. Therefore, the optimisation procedure used in this work to evaluate the optimum network configuration considered the exogenous and endogenous variables selected for each model. The parameters considered in the recombination procedure thus included the type of transfer function, the number of hidden layers and the number of hidden neurons.

3.1 MLP-ANN results

The MLP model comprised a three-layer network, formed from a single input layer, a single hidden layer and a single output layer. The FDRs obtained from the aircraft were used to provide the training, cross-validation and test datasets. Linear (pure-lin) and hyperbolic tangent (tansig) functions were used for the output and hidden layer, respectively. The developed MLPs were trained using the CG, DBD and QP algorithms. Considering the MLP models, the number of neurons in the hidden layer was defined by using a trial-and-error method.

Given that the number of hidden layers and the number of neurons therein are features of the ANN that affect its accuracy, different ANN configurations with three different learning algorithms were examined in five runs over 1,000 epochs. Throughout this trial-and-error process, the network weights and biases were initialised randomly at the start of the learning stage. The termination criterion during the model generalisation was also defined based on the mean square error (MSE), typically the average squared error between the ANN outputs and the target on the cross-validation dataset⁽²¹⁾.

The accuracy of the model was evaluated based on the results for 19 topological configurations of the network. The Mean Squared Error (MSE), Normalised Mean Squared Error (nMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and linear correlation coefficient (R) were considered as statistical indicators in this comparison process, as expressed below:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_{\text{predicted}} - Y_{\text{actual}})^2 \quad \dots (11)$$

$$\text{nMSE} = \frac{1}{n} \sum_{i=1}^n \frac{(Y_{\text{predicted}} - Y_{\text{actual}})^2}{\bar{Y}_{\text{predicted}} \bar{Y}_{\text{actual}}} \quad \dots (12a)$$

$$\bar{Y}_{\text{predicted}} = \frac{1}{n} \sum_{i=1}^n Y_{\text{predicted}} \quad \dots (12b)$$

$$\bar{Y}_{\text{actual}} = \frac{1}{n} \sum_{i=1}^n Y_{\text{actual}} \quad \dots (12c)$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |Y_{\text{predicted}} - Y_{\text{actual}}| \quad \dots (13)$$

$$\text{MinAbsError} = \text{Min} \sum_{i=1}^n |Y_{\text{predicted}} - Y_{\text{actual}}| \quad \dots (14)$$

$$\text{MaxAbsError} = \text{Max} \sum_{i=1}^n |Y_{\text{predicted}} - Y_{\text{actual}}| \quad \dots (15)$$

$$R = \frac{n \sum_{i=1}^n Y_{\text{predicted}} Y_{\text{actual}} - \sum_{i=1}^n Y_{\text{predicted}} \sum_{i=1}^n Y_{\text{actual}}}{\sqrt{n \sum_{i=1}^n Y_{\text{predicted}}^2 - \left(\sum_{i=1}^n Y_{\text{predicted}} \right)^2} \sqrt{n \sum_{i=1}^n Y_{\text{actual}}^2 - \left(\sum_{i=1}^n Y_{\text{actual}} \right)^2}} \quad \dots (16)$$

where n , $Y_{\text{predicted}}$, Y_{actual} and Y_{mean} are the number of data, and the predicted, actual and mean value of each output parameter, respectively⁽²²⁾. Additionally, MaxAbsError and MinAbsError are the maximum and minimum absolute errors, respectively. nMSE estimates the overall variation between the measured and predicted values using the MAE measure, since the predictions lie within the range of observed values. The direction and strength of the linear relation between any two variables is measured by R , which is a significant measure as it enables an evaluation of the overall accuracy of a regression model.

The transfer function, the number of hidden neurons and the number of hidden layers were varied, taking into account several plausible values. Considering the reviewed literature, various pairs of transfer functions were selected for the output and hidden layers, with the linear function often being selected for the output layer. Therefore, the transfer function for the hidden layer must essentially be non-linear, although the hyperbolic tangent transfer function was found to be preferable due to its excellent results.

The training was considered to have converged when the MSE stabilised over a particular number of iterations⁽²³⁾. Figure 2 illustrates the least MSEs attained when training using the CG, DBD and QP algorithms and the cross-validation for the single-hidden-layer MLP architectures for the climb, cruise and descent flight stages. During the implementation of the QP training, a momentum factor of 0.85 was chosen, due to its ability to provide excellent results based on the minimum MSE values. The best candidate, being considered to be the most suitable MLP model, is the architecture achieving the lowest MSE in the cross-validation stage.

To estimate the fuel flow rate for the turbofan engines in the climb, cruise and descent flight phases using the single-hidden-layer ANN designs, the best topologies seemed to be the 2–16–1 CG, 2–9–1 CG and 2–5–1 DBD structures, respectively, where the three values indicate the total number of neurons in the input, output and hidden layers. The lowest cross-validation MSE values with the MLP-ANN architecture were 3.663×10^{-5} , 1.141×10^{-2} and 6.923×10^{-3} , for each flight phase respectively.

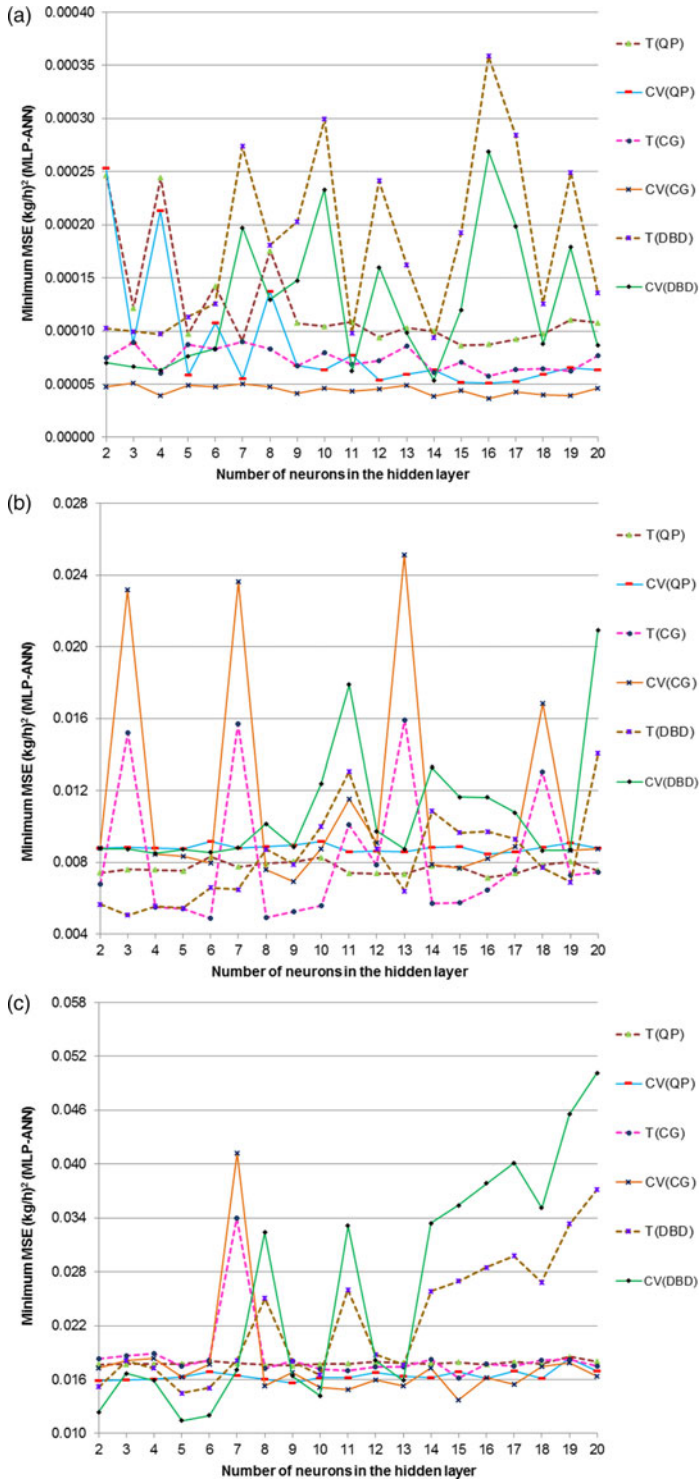


Figure 2. Minimum MSEs for the training and cross-validation of the MLP-ANNs trained with CG, DBD and QP in the (a) climb, (b) cruise and (c) descent phase.

Table 2
Testing phase error values in fuel flow rate for the optimal MLP architecture

Performance	Climb	Cruise	Descent
MSE	1.82854E-08	5.98921E-07	1.55805E-06
nMSE	0.000270896	0.164655555	0.185742985
MAE	0.000111361	0.000453283	0.000943351
Min Abs Error	5.71922E-07	1.0199E-06	1.9042E-05
Max Abs Error	0.00036113	0.003451375	0.005096066
<i>R</i>	0.999864633	0.919455151	0.903879521

Climb: CG-MLP, cruise: CG-MLP, descent: DBD-MLP.

During the testing phase of the MLPs, the highest linear correlation coefficients between the model-predicted and actual fuel flow rate, indicating the accuracy of each ANN model, were 0.999865, 0.919455 and 0.903880 for the climb, cruise and descent stage, respectively. The accuracy obtained using the best ANN architecture in terms of the preset error types MSE, nMSE and MAE and the linear correlation coefficient based on the MLP model testing dataset are presented in Table 2. Additionally, it is important to mention that the maximum and minimum absolute errors are also illustrated in this table to clarify the error level attained.

3.2 RBF-ANN results

This study also evaluated the RBF network comprising three layers: one hidden layer, one input layer and one output layer. While seeking the optimal network using the trial-and-error method, the number of cluster centres was assumed to be 15, selected based on the competitive rule, and a Euclidean metric was considered in the design of the RBFN architectures. As for the MLP-ANNs, 19 configurations were tested, varying the number of hidden layers from 2 to 20 in the trial-and-error method for the RBFNs. Such RBF networks are based on a static Gaussian function, with nonlinearity in the Processing Element (PE) layer, while a linear transfer function is selected for the output processing elements.

The response of the Gaussian function is limited to a region of the input range. To ensure that these networks are successfully implemented, it is imperative to establish appropriate centres for these Gaussian functions. In this study, this was achieved via supervised learning, although the unsupervised method was found to provide better output values. As a result, a hybrid supervised–unsupervised topology was applied in this work. The first phase of the simulation involved the training of the unsupervised layer. This step was performed to obtain the Gaussian centres and widths using the input data. These centres were encoded using the weights of the unsupervised layer by competitive learning. While conducting the unsupervised learning, the Gaussian widths were calculated according to their neighbours. The output of this layer was obtained from the input data weighted by a Gaussian mixture.

After the training of the unsupervised layer had finished, the supervised part subsequently set the centres of the Gaussian functions, depending on the weights of the unsupervised layer, determining the width (standard deviation) of each Gaussian based on the centres of its neighbours. A hidden layer was included to make the supervised section into a multilayer perceptron (MLP), rather than a plain linear perceptron, in this study. The supervised part employed the weighted inputs, rather than the input data interpreted from the data records.

Table 3
Testing phase error values in fuel flow rate for the optimum RBF network architecture

Performance	Climb	Cruise	Descent
MSE	1.62088E-08	8.18435E-07	7.77966E-07
nMSE	0.000240131	0.225004329	0.092745227
MAE	0.00010333	0.000459805	0.000623736
Min Abs Error	1.39918E-06	5.59812E-06	2.7715E-06
Max Abs Error	0.00032498	0.005910281	0.003434451
R	0.999881531	0.897203513	0.952523067

Climb: CG-RBF, cruise: DBD-RBFN, descent: DBD-RBF.

It is not feasible to propose a suitable number of Gaussians for each case, as this depends on the problem in hand. The number of patterns within the training set influences the number of centres, as extra patterns mean extra Gaussians. However, this can be addressed via the distribution of clusters. If the information is well clustered, then few Gaussians are required. Conversely, if the data are spread widely, many more Gaussians are needed to achieve excellent performance. In this regard, the number of cluster centres in this study was assumed to be 15.

Competitive learning has an inherent metric, which was used in this study. This is known as the Euclidean metric, which determines the dissimilarity between two vectors based on the distances within the input space. Moreover, competitive learning maintains inherent probability sharing of the input data. However, it has the disadvantage that some PEs may never fire, while others may win the contest every time. To prevent these extremes, a 'conscience' mechanism, which maintains a count of how frequently a PE wins the competition and imposes a steady winning frequency across the PEs, was incorporated into the RBF models at this point.

As for the MLP-ANNs, the DBD, CG and QP algorithms were implemented to train the RBFNs in order to identify the best network model. The minimum MSE values for the training and cross-validation of the DBD, CG, and QP algorithms (with the latter having a momentum rate of 0.85) are shown in Fig. 3(a), (b) and (c) for the climb, cruise and descent phases. The best RBF network architecture candidate was found to be the 2–12–1 CG, 2–3–1 DBD and 2–3–1 DBD structure, while the least cross-validation MSE value was found to be 5.045×10^{-5} , 7.097×10^{-3} and 8.471×10^{-3} for the mentioned flight stages.

For the structures of the RBF networks with the best testing performance, the linear uppermost correlation coefficient was found to be 0.999882, 0.897204 and 0.952523 for the climb, cruise and descent flight phase, respectively. The values of the predefined types of error for the case of the RBF networks are presented in Table 3.

3.3 ANFIS results

Applying fuzzy guidelines as a pre-processor to an ANN permits the integration of human awareness to carry out decision-making and inference. The ANFIS model optimises the fuzzy laws (membership role parameters) with back-proliferation, thus human knowledge is not required. To start the development of the ANFIS prediction model, a fuzzy model first has to be obtained. To define this model, it is necessary to determine the number of inputs and linguistic variables and, thereby, the number of laws for the final fuzzy model. As in the initial

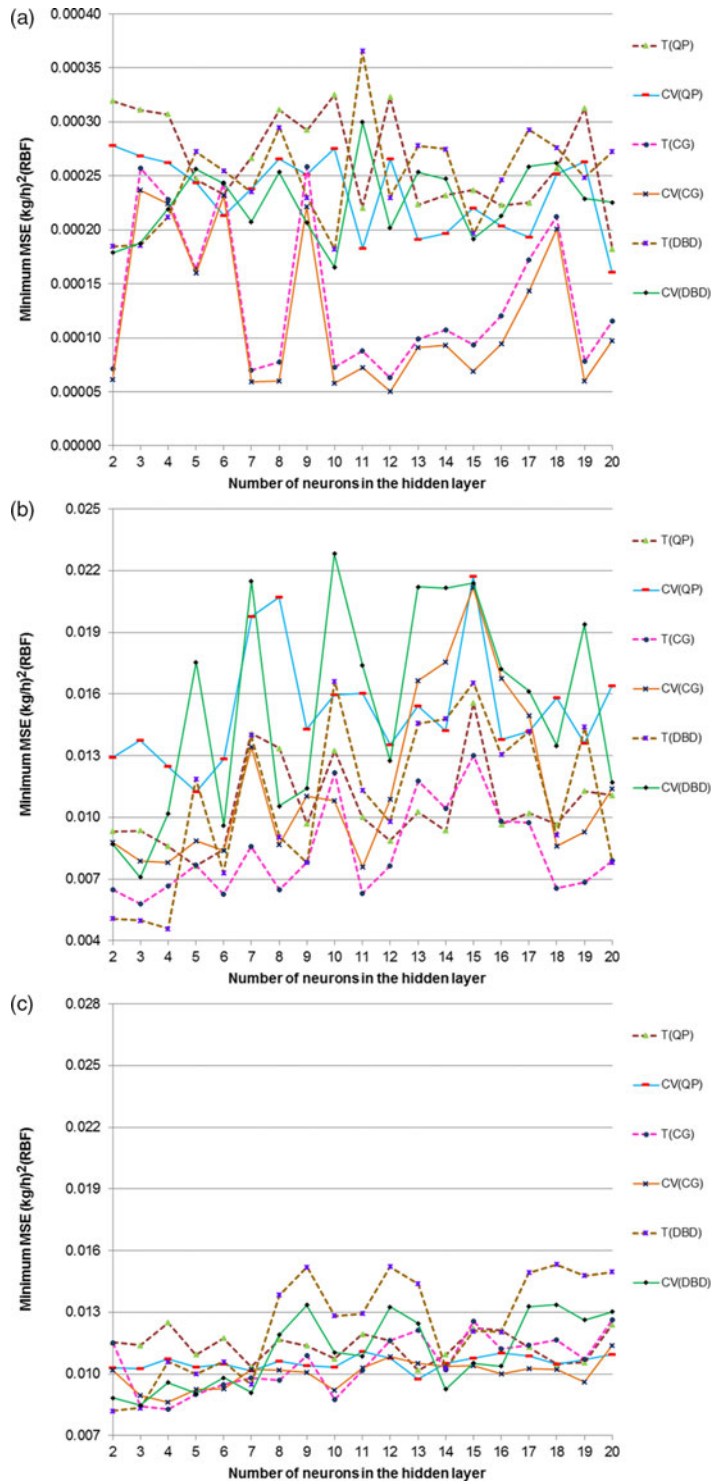


Figure 3. Minimum MSEs for the training and cross-validation of the RBFNs trained with CG, DBD and QP in the (a) climb, (b) cruise and (c) descent phase.

Table 4
Testing phase error values in fuel flow rate for the optimum ANFIS architecture

Performance	Climb	Cruise	Descent
MSE	2.78392E-08	7.63535E-07	1.25E-06
nMSE	0.000412432	0.20991117	0.149019097
MAE	0.000133429	0.000588461	0.000808906
Min Abs Error	4.90284E-07	1.55069E-06	3.55298E-06
Max Abs Error	0.000402406	0.003107487	0.00477084
R	0.999799317	0.890046502	0.922952207

Climb: DBD-ANFIS, cruise: DBD-ANFIS, descent: QP-ANFIS.

step for defining the initial fuzzy model, the subtractive clustering technique is used on the input–output data pairs. The technique proposed by Chiu⁽²⁴⁾ is a quick, one-pass algorithm for approximating the number of clusters and cluster centres using an unsupervised method by determining the prospective data points within the feature space. After this clustering process, the number of fuzzy rules and principle fuzzy membership functions are measured. Therefore, the optimal values of these parameters, as well as the resultant and premise parameters, are obtained using the learning algorithm.

In this study, the DBD, CG and QP learning algorithms were implemented for the ANFIS architecture, as in the situation of the RBFNs and MLP-ANNs. After trying numerous network parameters in a search for the most accurate testing outcomes, the following were selected for the execution of the ANFIS networks: no hidden layer configuration, three membership functions per network input, Gaussian-shaped curve fuzzy membership function, and TSK fuzzy model (the well-recognised Sugeno fuzzy model). For ANFIS networks, the MSE values achieved for the training and cross-validation using the DBD, CG and QP algorithms (with the latter having a momentum factor of 0.85) versus the number of epochs are shown in Fig. 4(a), (b) and (c) for the climb, cruise and descent flight phase, respectively. The lowest cross-validation MSE value of 5.892×10^{-5} was obtained at epoch number 492 when using the DBD training algorithm for the climb phase, whereas a lowest value of 0.010232367 was obtained at epoch number 740 when using the QP training algorithm for the descent phase, and 0.009143089 at epoch number 367 when using the DBD training algorithm for the cruise flight phase.

Among the different ANFIS networks, the optimal testing result in terms of the linear correlation coefficient was 0.999799, 0.890047 and 0.922952 for the climb, cruise and descent flight phase, respectively. The error values for the best ANFIS configurations are reported in Table 4.

Consequently, considering these testing results for the three ANN architectures, the output parameter values estimated for the optimal ANN architectures and the data acquired from the actual FDRs are also shown, considering the associated ANN dataset, in Fig. 5(a), (b) and (c) for the climb (RBF), cruise and descent phase, respectively. Considering the results shown in these figures, it can be concluded that there is a good fit between the real values and those predicted by the model for the fuel flow rate in each flight phase.

The obtained ANN models were also employed to generate formerly inaccessible output values by feeding new input data into all the model networks. Yielding linear correlation

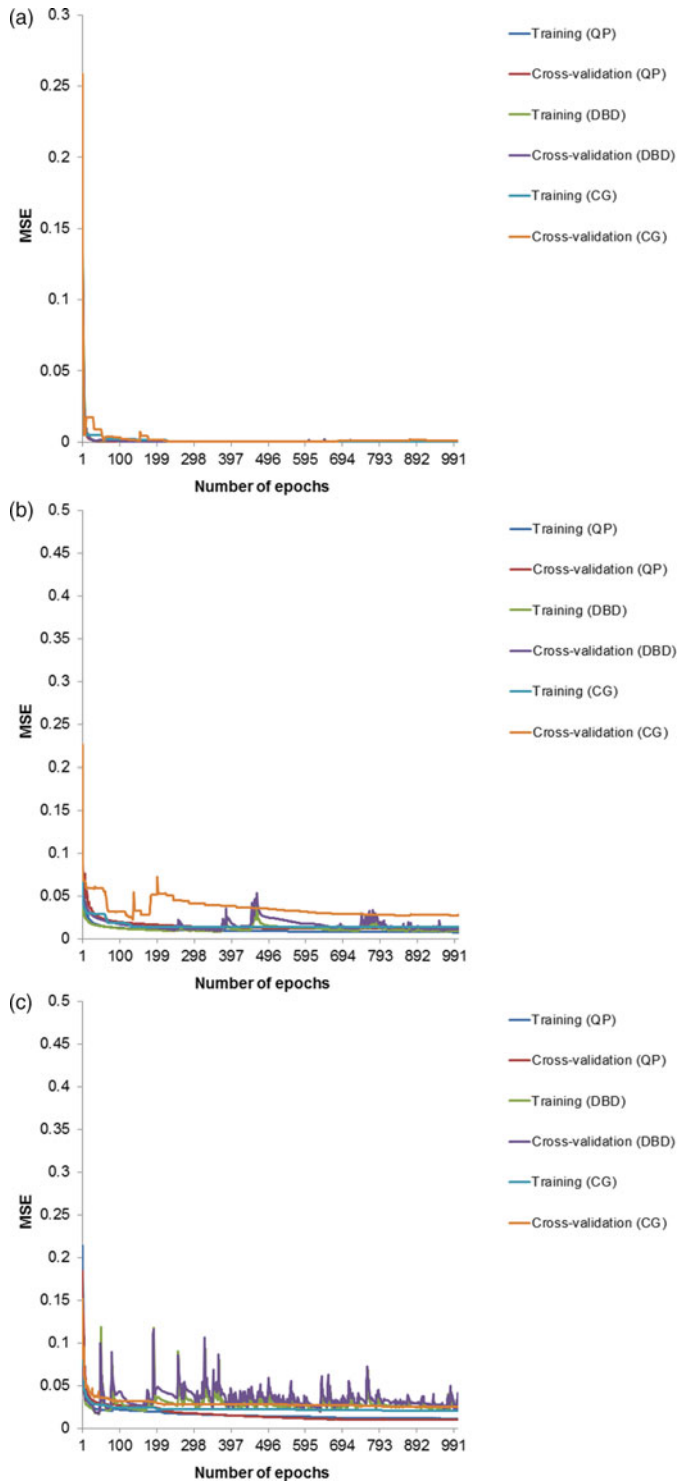


Figure 4. MSE versus number of epochs for the training and cross-validation of the ANFIS architectures trained with CG, DBD and QP in the (a) climb, (b) cruise and (c) descent phase.

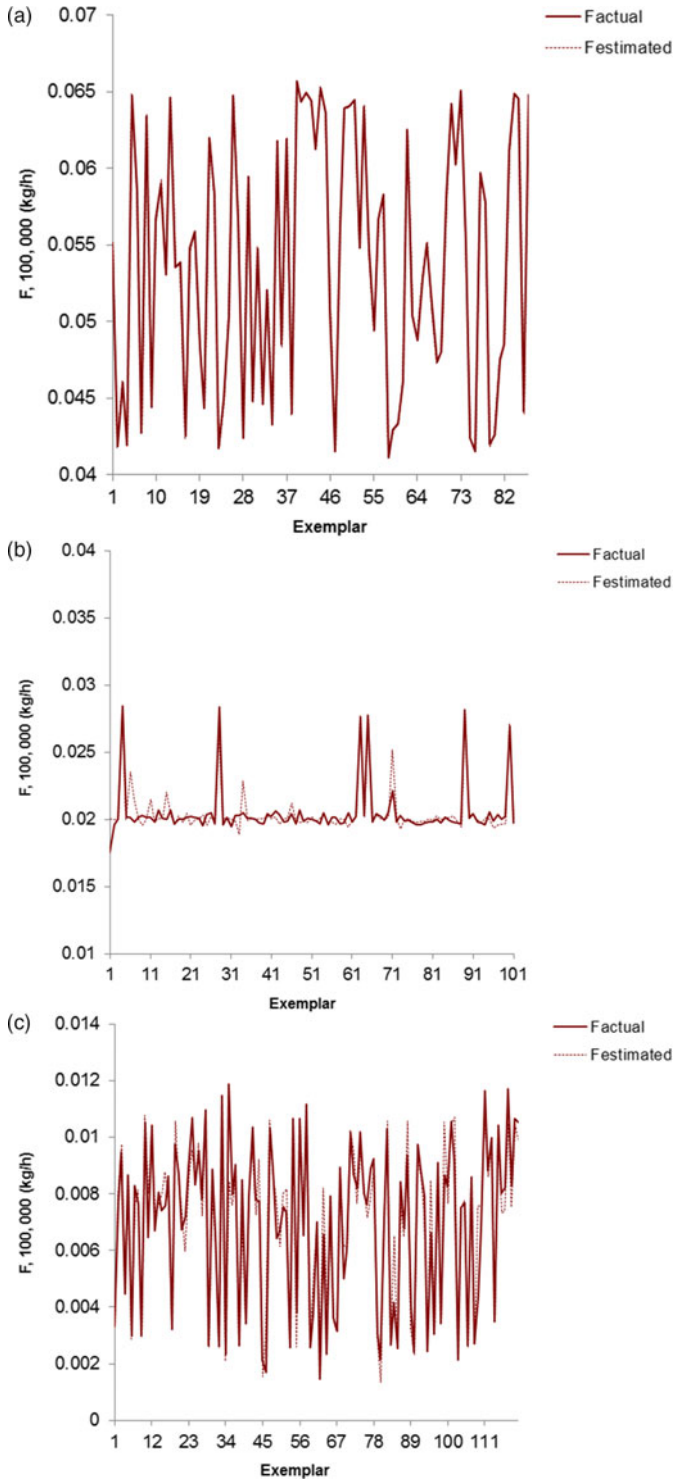


Figure 5. Test results of the best ANN architectures for the (a) climb (RBF), (b) cruise (MLP) and (c) descent (RBF) phase.

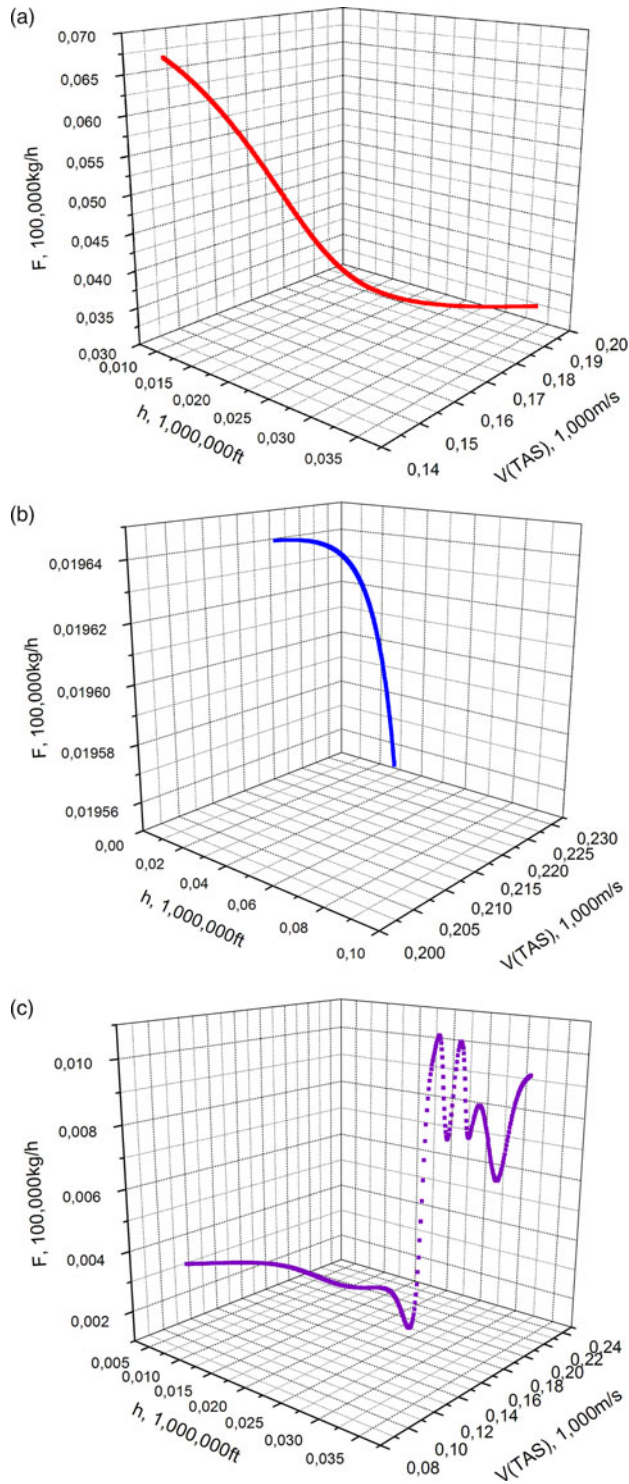


Figure 6. Fuel flow rate outputs produced by the most accurate ANNs in the (a) climb, (b) cruise and (c) descent phase.

coefficients extremely close to 1, the derived ANN models were extremely efficient for obtaining accurate new values for the fuel flow rate. The resulting outputs are shown versus the model inputs in Fig. 6(a)–(c).

4.0 CONCLUSIONS

For the first time in literature, this study implemented a new modelling method for estimation the fuel flow rate of a commercial turbofan aircraft throughout the climb, cruise and descent flight phases based on ANNs with three diverse architectures, viz. MLP, RBF and ANFIS networks, trained using three different learning algorithms, i.e. CG, QP and DBD. To device and design the models, real data derived from FDRs of a Boeing 737-800 were used, which represents an additional novelty of the projected models. Furthermore, unlike existing models in literature, the models derived in this study adopted the variation of both the true airspeed and altitude as input parameters, while offering the fuel flow rate as the output parameter. To avoid the need for comprehensive information on aircraft data and operations, flight altitude and real airspeed variations were applied as the input parameters for the developed models, thus facilitating highly accurate fuel flow rate modelling in a simplified way.

Using the most favourable ANN models, linear correlation coefficients of 0.999882, 0.919455 and 0.952523 were obtained for the climb, cruise and descent flight phase, respectively. Considering these high linear correlation coefficients, together with the low values of the predefined error types, it can be clearly deduced that the obtained ANN models are extremely accurate for the approximation of the fuel flow rate for the case study aircraft.

Among the three diverse topologies, the implementation of the ANN model with the RBF architecture and trained with the CG algorithm achieved the most accurate behaviour throughout the climb phase. Meanwhile, the CG-MLP and DBD-RBF network architectures performed better for the cruise and descent stage, respectively.

When actual flight data or manual data from an aircraft are accessible, this newly proposed ANN architecture could facilitate a further effective modelling methodology for fuel flow rate predictions in actual applications. For instance, this could decrease fuel consumption in ecological research or be used in fuel-saving policies and air traffic management, as well as aircraft trajectory prediction within air traffic control simulations and decision support systems. Moreover, to achieve accurate fuel flow rate predictions to compute the required thrust and energy of an aircraft engine, the model obtained in this research could be embedded into Full Authority Digital Engine Control (FADEC). When an accurate formulation to define the correlation between the thrust and required fuel energy is achieved, FADEC could send the accurate and minimum required fuel flow rate to the engine combustor. Such achievement of the correct fuel flow rate could minimise emissions, decrease environmental effects and wasted energy, and maximise aircraft sustainability.

The present research also opens the avenue towards future research. Indeed, the robust fuel flow rate modelling approach proposed herein could be used to construct energetic and exergetic models for the turbofan engines of contemporary commercial aircraft, to analyse their environmental effects and sustainability.

REFERENCES

1. TURGUT, E.T., CAVCAR, M., USANMAZ, O., CANARSLANLAR, A.O., DOGEROGLU, T., ARMUTLU, K. and YAY, O.D. Fuel flow analysis for the cruise phase of commercial aircraft on domestic routes, *Aerosp Sci Technol*, 2014, **37**, pp 1–9.

2. MITCHELL, D., EKSTRAND, H., PRATS, X. and GRÖNSTEDT, T. An environmental assessment of air traffic speed constraints in the departure phase of flight: A case study at Gothenburg Landvetter Airport, Sweden, *Transp Res Part D, Transp Environ*, 2012, **17**, (8), pp 610–618.
3. COLLINS, B. Estimation of aircraft fuel consumption, *J Aircr*, 1982, **19**, (11), pp 969–975. <http://dx.doi.org/10.2514/3.44799>, also AIAA Paper 81-0789R.
4. TRANI, A., WING-HO, F., SCHILLING, G., BAIK, H. and SESHADRI, A. A neural network model to estimate aircraft fuel consumption, 4th AIAA Aviation Technology, Integration and Operations (ATIO) Forum, Chicago, 20–22 September 2004, AIAA Paper 2004-6401, 2004.
5. SENZIG, D.A., FLEMING, G.G. and IOVINELLI, R.J. Modeling of terminal-area airplane fuel consumption, *J Aircr*, 2009, **46**, (4), pp 1089–1093.
6. BARTEL, M. and YOUNG, T.M. Simplified thrust and fuel consumption models for modern two-shaft turbofan engines, *J Aircr*, 2008, **45**, (4), pp 1450–1456.
7. TURGUT, E.T. and ROSEN, M.A. Relationship between fuel consumption and altitude for commercial aircraft during descent: Preliminary assessment with a genetic algorithm, *Aerosp Sci Technol*, 2012, **17**, pp 65–73.
8. NUIC, A. User Manual for the Base of Aircraft Data (BADA), Revision 3.13, EUROCONTROL Experimental Centre. EEC Technical/Scientific Report No. 15/04/02-43, 2015.
9. BAKLACIOGLU, T. Fuel flow-rate modelling of transport aircraft for the climb flight using genetic algorithms, *Aeronaut J*, 2015, **119**, (1212), pp 173–183.
10. BAKLACIOGLU, T. Modeling the fuel flow-rate of transport aircraft during flight phases using genetic algorithm-optimized neural networks, *Aerosp Sci Technol*, 2016, **49**, pp 52–62.
11. HUANG, C., XU, Y. and JOHNSON, M.E. Statistical modeling of the fuel flow rate of GA piston engine aircraft using flight operational data. *Transp Res Part D Transp Environ*, 2017, **53**, pp 50–62.
12. JOVANOVIĆ, R.Z., SRETENOVIC, A.A. and ZIVKOVIC, B.D. Ensemble of various neural networks for prediction of heating energy consumption, *Energ Buildings*, 2015, **94**, pp 189–199.
13. CHANDOK, J.S., KAR, I.N. and TULI, S. Estimation of furnace exit gas temperature (FEGT) using optimized radial basis and back-propagation neural networks, *Energ Convers Manage*, 2008, **49**, pp 1989–1998.
14. EKONOMOU, L. Greek long-term energy consumption prediction using artificial neural networks, *Energy*, 2010, **35**, pp 512–517.
15. SUN, W. and XU, Y. Financial security evaluation of the electric power industry in China based on a back propagation neural network optimized by genetic algorithm, *Energy*, 2016, **101**, pp 366–379.
16. SVORCAN, J., STUPAR, S., TRIVKOVIC, S., PETRAŠINOVIC, N. and IVANOV, T. Active boundary layer control in linear cascades using CFD and artificial neural networks, *Aerosp Sci Technol*, 2014, **39**, pp 243–249.
17. SEKHAR, P. and MOHANTY, S. An online power system static security assessment module using multi-layer perceptron and radial basis function network, *Int J Elec Power*, 2016, **76**, pp 165–173.
18. TAKAGI, T. and SUGENO, M. Fuzzy identification of systems and its applications to modeling and control, *IEEE Trans Syst Man Cybern*, 1985, **1**, pp 116–132.
19. KHATIBA, T., MOHAMEDA, A. and SOPIAN, K. A review of solar energy modeling techniques, *Renew Sustain Energy Rev*, 2012, **16**, pp 2864–2869.
20. VOYANT, C., MUSELLI, M., PAOLI, C. and NIVET, M.L. Hybrid methodology for hourly global radiation forecasting in Mediterranean area, *Renew Energy*, 2013, **53**, pp 1–11.
21. ADEWOLE, B.Z., ABIDAKUN, O.A. and ASERE, A.A. Artificial neural network prediction of exhaust emissions and flame temperature in LPG (liquefied petroleum gas) fueled low swirl burner, *Energy*, (2013) **61**, pp 606–611.
22. TAGHAVIFAR, H. and MARDANI, A. A comparative trend in forecasting ability of artificial neural networks and regressive support vector machine methodologies for energy dissipation modeling of off-road vehicles, *Energy*, 2014, **66**, pp 569–576.
23. ASADI, E., GAMEIRO DA SILVA, M., ANTUNES, C.H., DIAS, L. and GLICKSMAN, L. Multi-objective optimization for building retrofit: A model using genetic algorithm and artificial neural network and an application, *Energ Buildings*, 2014, **81**, pp 444–456.
24. CHIU, S.L. Fuzzy model identification based on cluster estimation, *J Intell Fuzzy Syst*, 1994, **2**, (3), pp 267–278.