© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

Journal homepage: http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

Surface Roughness Prediction and Optimisation using Novel Joint Artificial Neural Network and Bat Algorithm

Desmond Eseoghene Ighravwe[1], Sunday Ayoola Oke[2*]

[1]Department of Mechanical and Biomedical Engineering, Bells University of Technology, Ota, NIGERIA

[2]Department of Mechanical Engineering, Faculty of Engineering, University of Lagos, Room 10, Mezzanine Complex, Akoka-Yaba, Lagos, 100213, NIGERIA

*Corresponding Author

DOI: https://doi.org/10.30880/ijie.2022.14.04.003
Received 17 February 2020; Accepted 17 May 2021; Available online 20 June 2022

Abstract: This paper targets the surface roughness concept in end milling in which the tool-work material combination is central to its success. At present, sufficient optimal surface roughness information is repeatedly not accessible to CNC end milling operators and this problem is anticipated to grow worse in the forthcoming years. Consequently, the unique development and validation of optimisation tools are interventions to tackle access to optimal roughness information problems. This paper examined two novel models, the combined artificial neural network and bat algorithm as well as joint artificial neural network and particle swarm optimisation to predict and optimise the process parameters of an end milling scheme. Both models were tested with literature data. Additionally, the work investigates machining time and introduces a bi-objective fuzzy goal programming optimisation model. The striking results revealed the optimal values as 0.8816 and 0.8088 for the particle swarm optimisation procedure while the bat procedure yielded 0.275 and 0.178, which places the bat procedure ahead of the counterpart, particle swarm optimization procedure.

Keywords: Surface finish, optimisation, artificial neural network, machining time, bat algorithm, end milling, fuzzy goal programming, bi-objective

Nomenclature

X_1 Cutting speed (m/min)
X_2 Depth-of-cut (m/min)
X_3 Feed rate (mm/sec)
X_4 Immersion angle (°)
W_1 Weight for surface roughness
W_2 Weight for machining time
F_1 Predicted value for surface roughness (μm)
F_2 Predicted value for machining time (min)
λ_1 Membership function for surface roughness
λ_2 Membership function for machining time

1. Introduction

The surface roughness concept in end milling refers to a quantified deviation of the texture for milled specimens from their ultimate appearance [1,2]. Obtaining large deviations implies surface roughness while small deviations signify

*Corresponding author: sa_oke@yahoo.com

surface smoothness[1]. However, a value in-between these two extremes, the optimal value, are desirable in end milling operations [3-9]. Certainly, at present, general studies in end milling optimisation for computer numeric controlled (CNC) machines is still in its immaturity. End milling is one of the most expensive [10-12] and significantly time-consuming machining, requiring precision, skill and expertise of the milling operator [13]. Furthermore, end milling is a specialized aspect with many jobs from the assembly operations such as automobile systems [14], and any mistake on the machining process could result in the wasteful dumping of worked materials for new ones or re-processing [15]. Therefore, the optimisation of the process is critical for judicious resource utilization [16-18]. As such, surface finish is a common benchmark which should attract scientific attention for greater scrutiny and analysis [19,20]. At present, in practice, sufficient optimal surface roughness information is repeatedly not accessible to CNC end milling operators and this problem is anticipated to grow worse in the forthcoming years [21]. This research gap triggered the current investigators to optimise end milling parameters using novel and innovative schemes [22-28]. Consequently, this paper examined two novel models, combined artificial neural network and bat algorithm (ANN-BA) as well as joint artificial neural network and particle swarm optimisation (ANN-PSO) to concurrently predict and optimise the process parameters of an end milling scheme. Both models were tested with literature data from Prajina's work [29]. The principal advantages of both the artificial neural network and the bat algorithm are exploited for the benefit of the work. The value brought by the artificial neural network into the union is its aptitude to perform through partial information; surface roughness requires expensive equipment and inability to use this equipment poses a threat in measurement activities. Thus, the artificial neural network is a substitute for this concern. The complementary value brought by the bat algorithm is its incredibility fast convergence at the actual phase by swapping from voyaging to utilization. Interestingly, the combination of these advantages has not been exploited to solve the surface roughness optimisation problem of machined parts to date.

This research is innovative and a rare method to machinery practice by tackling a research problem that is sparsely treated in machining science and technology studies: combined prediction and optimisation of surface roughness. Learning about this novel method for end milling is critical for customer retention and cost minimisation purposes [17,18]. The outcome of this work will stimulate understanding into how to predict with precision and optimise the surface roughness of machined parts in the machine shops, and by extension in general milling processes. This will influence the competence and reliability of judgment for machine shop engineers, their consciousness to attain optimal surface roughness for customer retention, company's sustenance and the development of enhanced training and operational policies.

2. Research Methodology

The research methodology presented in the current study is based on the information in Figures 1 and 2. One of the reasons for computing the BA and particle swarm optimisation (PSO) algorithms is that they both possess explorative and exploitative ability for searching the able solution spaces for a problem of interest [30]. The first two stages in Figures 1 and 2 have been addressed by different authors in literature, and an investigation was by Prajina [29]. Thus, this work is limited to the remaining stages in Figure 1. First, the development of predictive models for surface roughness and machining time is considered and an artificial neural network, ANN, is selected judged by its excellent predictive power [31,32]. This step is then followed by proposing the bi-objective fuzzy goal programming model. Finally, the description of bat algorithm is offered.

In establishing the proposed framework, the ANN-cum-BA scheme, the following nomenclature is employed to explain the non-linear fuzzy bi-objective optimisation scheme. The characteristic of ANN model involves modification of the values of the connecting weights among the various layers in a selected ANN architecture, to minimise a cost function (Equation 1) that consider the differences between predicted and actual values. In choosing the architecture for the ANN scheme (Figure 3), the preferred structure is a 2-layer hidden platform that accommodates the parameters of immersion angle, feed rate, depth-of-cut and cutting speed, being considered as the inputs whereas the surface roughness or machining time was considered as the output(s). Interestingly, as the input signals merge with the linking weights of the ANN platform, the final results produce a summation outcome (Equation 1) or a product outcome (Equation 2) [33-36].

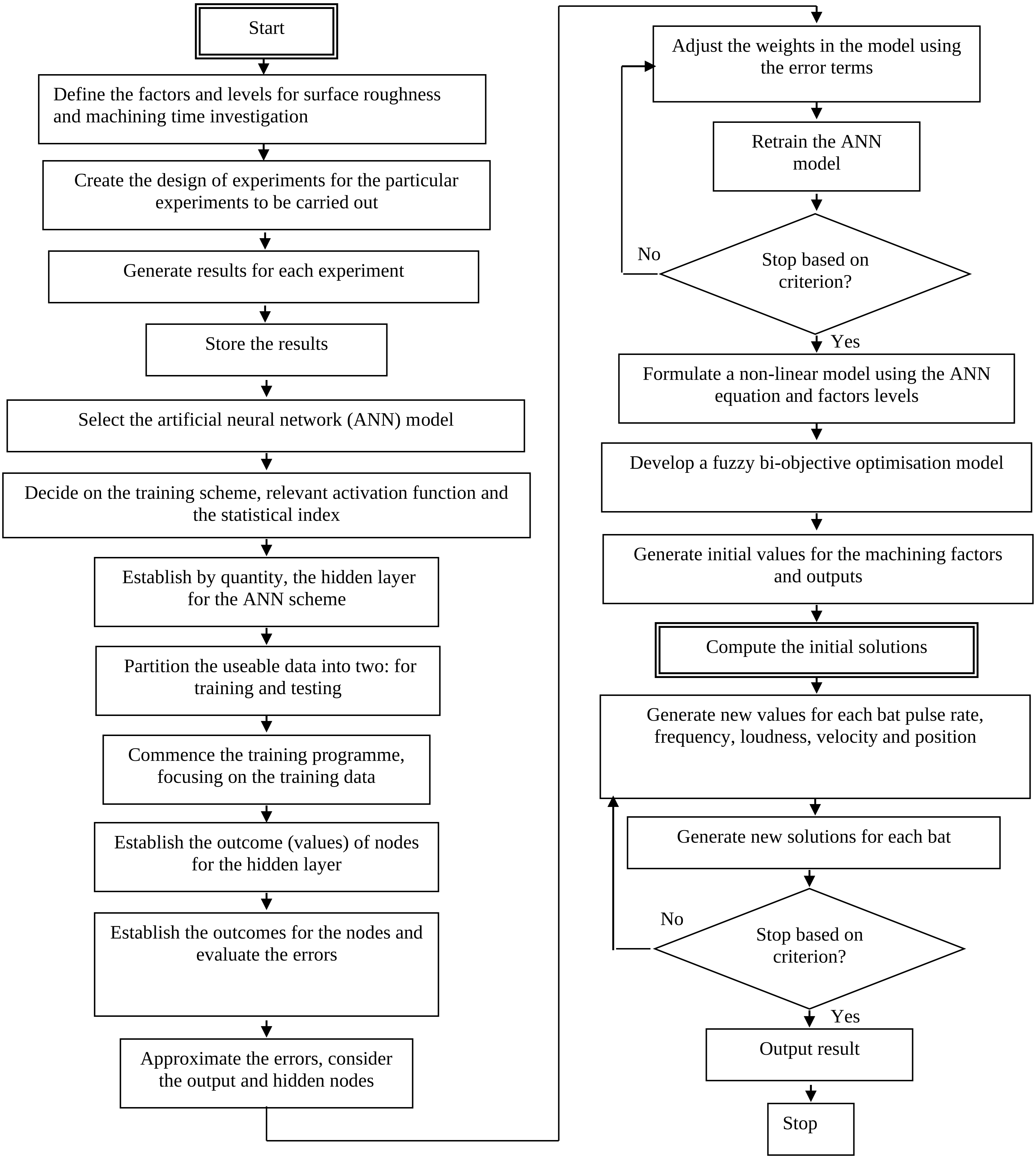

Fig. 1 - The proposed ANN-BA flowchart

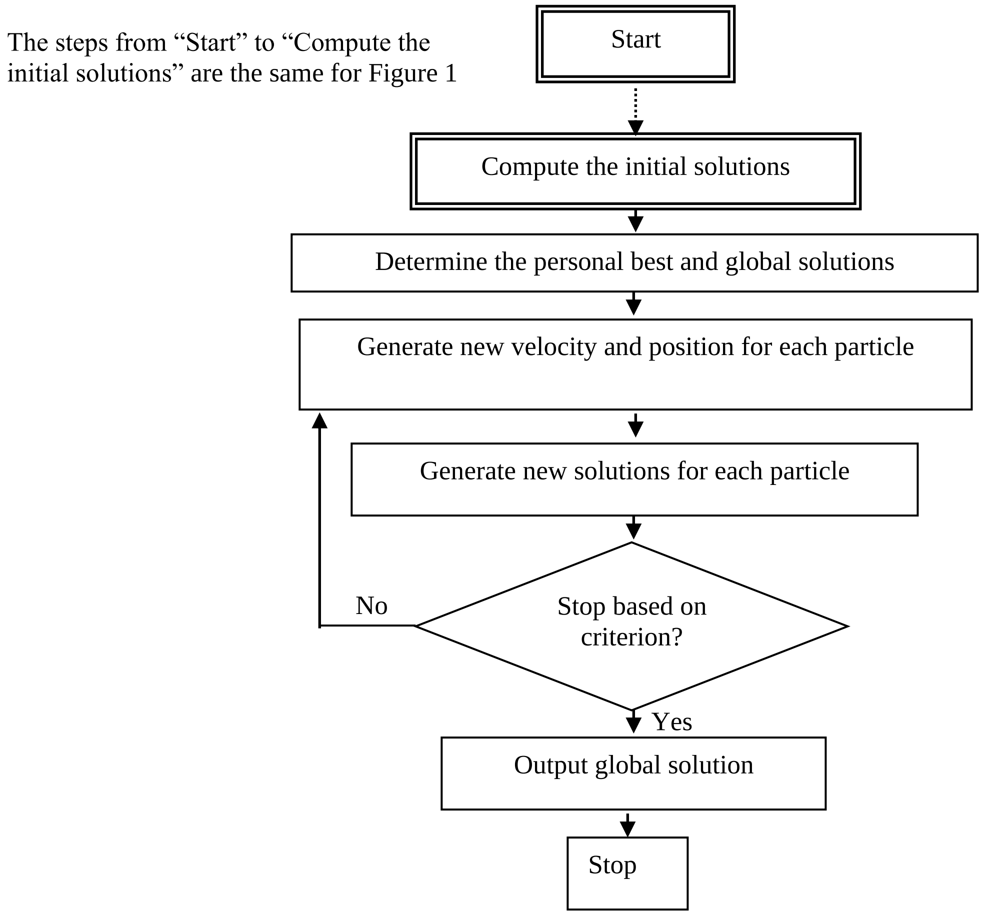

Fig. 2 - The ANN-PSO flowchart

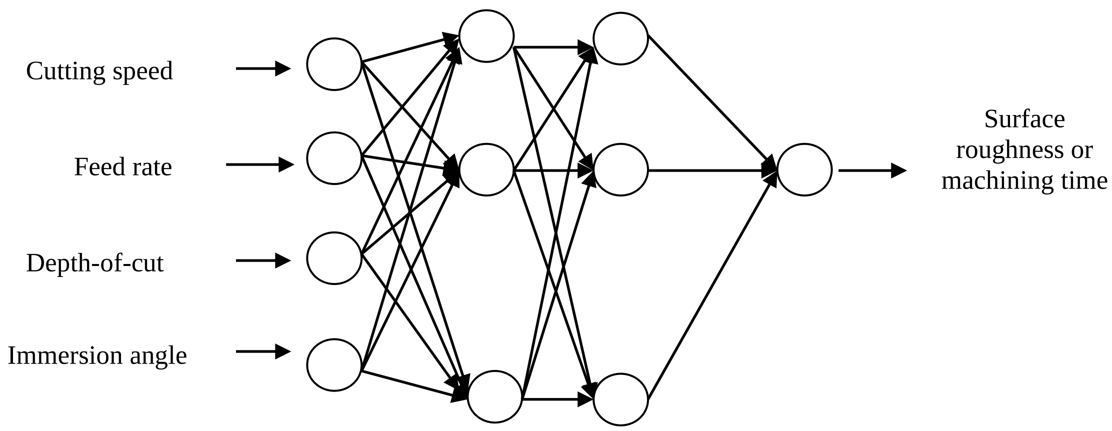

Fig. 3 - ANN architecture for surface roughness (or machining time) prediction

$$net_j = \sum_{i=1}^{m} x_i w_i + \theta_j \tag{1}$$

$$net_j = \prod_{i=1}^{m} x_i^{w_i} + \theta_j \tag{2}$$

where θ_j is the bias linked to node j

Equation (1) [31] was found useful to create values for nodes in the subsequent layers. The conversion of the input signal into output in a node is based on activation functions (hyperbolic tangent, Gaussian and sigmoid functions). The sigmoid function, also referred to in Equation (3) has the utility to map the input values from Equation (1) to and array between 0 and 1. The Equation (3) has been found to be very effective by researchers to train ANN schemes [31]. The incentive may be due to the nature of most outputs, which fall between 0 and 1. In choosing the transformation for the sigmoid function, the activation function was considered while keeping in view its success path in supervised ANN schemes [36]. Through the use of Equation (3) to establish the real value of an ANN's node, a transformation of Equation (1) was made through an indirect means to a non-linear structure (Equation 4).

$$y_j = \frac{1}{1 + \exp^{-net_j}} \tag{3}$$

$$y_j = \frac{1}{1 + \exp^{-\left(\sum_{i=1}^{m} x_i w_i + \theta_j\right)}} \tag{4}$$

The error terms for the output layer (δ_j^o) is defined as Equation (5) and Equation (6) expressed the error terms at the hidden layer (δ_j^h) in the ANN model [37].

$$\delta_j^o = \bar{y}_{j,actual} - \bar{y}_{j,predicted} \tag{5}$$

$$\delta_j^h = y_{ij}^h \left(1 - y_{ij}^h\right) \sum_{j=1}^{N} \delta_j^o w_{ij}^h \tag{6}$$

where, $y_{j,actual}$ denotes the target output *j* value and $y_{j,predicted}$ denotes the predicted output *j* output value. y_{ij}^h is the output between layer *i* and *j* in the ANN model and w_{ij}^h is the connecting weight between one hidden layer and the other.

The values of δ_j^o and δ_j^h are used in updating of the connecting weights at the output and hidden layers in the ANN architecture as shown in Equations (7) and (8), respectively.

$$w_{ij}^o(t) = w_{ij}(t-1) + \eta \delta_i x_j \tag{7}$$

$$w_{ij}^h(t) = w_{ij}(t-1) + \eta \delta_j^h y_{ij}^h \tag{8}$$

where, w_{ij}^o is the weight connecting weight between a hidden and the output layer in the ANN model. η is known as the learning rate.

To monitor the deviation of the newly trained data from the actual, the researchers concluded to use the mean square error, MSE (Equation 9) as a lens to control the deviations.

$$\text{MSE} = \frac{1}{n} \sum_{j=1}^{n} \left(\bar{y}_{j,actual} - \bar{y}_{j,predicted}\right)^2 \tag{9}$$

On a general note, in discussing the artificial neural network optimization, the objective function formulated is the mean square error function as shown in Equation (9).

2.1 Bi-objective fuzzy goal programming (FGP) optimisation model

The generated equations for computing the outputs from the trained ANN models for surface roughness and machining time are considered as equations that are required to be minimised, given as Equations (10) and (11), respectively.

$$\text{Min } Z_1 = f_1(x) \tag{10}$$

$$\text{Min } Z_2 = f_2(x) \tag{11}$$

To incorporate the decision from a workshop manager, fuzzy logic is considered as a tool in establishing the interrelationships between workshop manager's desires for surface roughness and machining time that will result in optimal use of computer numerically control (CNC) end milling machines in their workshop. The design of membership functions, surface roughness (λ_1) and machining time (λ_2) is carried out using the maximum values for surface roughness (Ra_{max}) and machining time (Tm_{max}) and as obtained from Prajina[29], is the expected value at the margin between the core and the boundary membership functions (χ_i). By applying the knowledge gained from the work of Amid and Ghodsypour [38], we present the membership functions for surface roughness and machining time minimisation as Figures 4a and 4b, respectively.

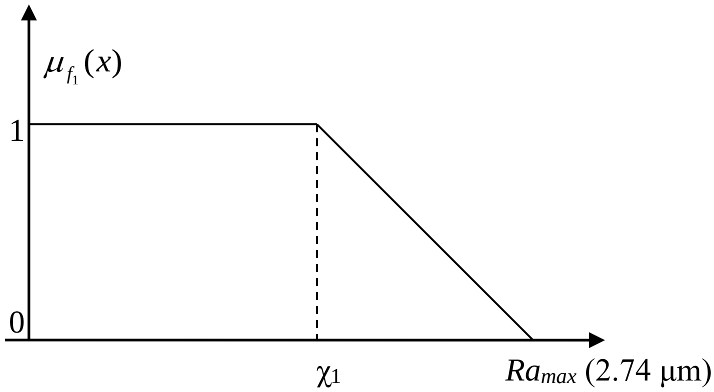

Fig. 4 (a) - Membership function for surface roughness

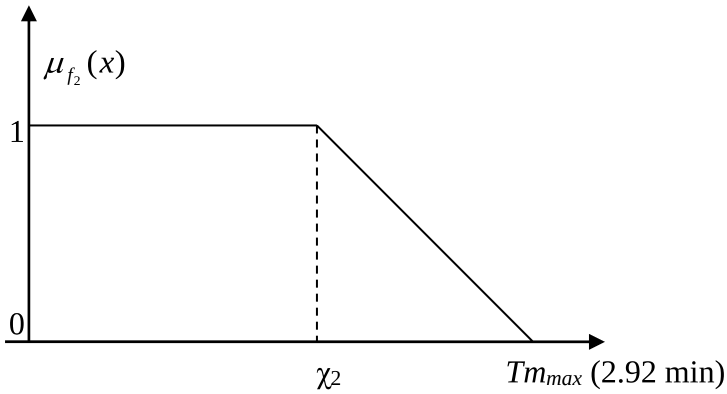

Fig. 4 (b) - Membership for machining time

By considering the information in Figures 4a and 4b, the soft constraints in the proposed model are expressed as Equations (12) and (13)

$$\lambda_1 \leq \frac{Ra_{\max} - f_1(x)}{Ra_{\max} - \chi_1} \tag{12}$$

$$\lambda_2 \leq \frac{Tm_{\max} - f_2(x)}{Tm_{\max} - \chi_2} \tag{13}$$

where λ_1 is the membership function for surface roughness and λ_2 is a membership function for machining time

The literature approach to tackling the problem of assigning weights to surface roughness and machining time is to consider fixing bounds on the ratio of surface roughness to machining time weights [39]. To achieve this, Equations (14) and (15) are included in the proposed model. One benefit of this approach is that it will reduce the effect of subjective weight allocation by a decision marker (workshop manager) on the optimal values of end milling process variables.

$$L \leq \frac{w_1}{w_2} \leq U \tag{14}$$

$$w_1 + w_2 = 1 \tag{15}$$

With the expected limits on the weight ratio and the expected membership functions for the objective functions, Amid and Ghodsypour [38] point out that a single objective that will be maximised may be stated in Equation (16).

$$\text{Max } G = \sum_{i=1}^{2} w_i \lambda_i \tag{16}$$

The optimal values for end milling parameters may be obtained using the limit of parameters as constraints [40]. Given this assertion, Equations (17) to (20) are considered.

$$x_{\min,1} \leq x_1 \leq x_{\max,1} \tag{17}$$

$$x_{\min,2} \leq x_2 \leq x_{\max,2} \tag{18}$$

$$x_{\min,3} \leq x_3 \leq x_{\max,3} \tag{19}$$

$$x_{\min,4} \leq x_4 \leq x_{\max,4} \tag{20}$$

The proposed bi-objective weighted fuzzy goal programming model for surface roughness and machining time minimisation recalled from previously stated equations is presented as follows:

$$\text{Max } G = \sum_{i=1}^{2} w_i \lambda_i \tag{16}$$

Subject to

$$\lambda_1 \leq \frac{Ra_{\max} - f_1(x)}{Ra_{\max} - \chi_1} \tag{12}$$

$$\lambda_2 \leq \frac{Tm_{\max} - f_2(x)}{Tm_{\max} - \chi_2} \tag{13}$$

$$x_{\min,1} \leq x_1 \leq x_{\max,1} \tag{17}$$

$$x_{\min,2} \leq x_2 \leq x_{\max,2} \tag{18}$$

$$x_{\min,3} \leq x_3 \leq x_{\max,3} \tag{19}$$

$$x_{\min,4} \leq x_4 \leq x_{\max,4} \tag{20}$$

$$L \leq \frac{w_1}{w_2} \leq U \tag{14}$$

$$w_1 + w_2 = 1 \tag{15}$$

To implement a BA, the initial values for decision variables are generated and the frequency for each bat is also generated using Equation (21). After this, the quality of the solution from each bat will be determined and the global solution at $t = 0$ is also determined.

$$Q_i = Q_{\min} + \left(Q_{\max} - Q_{\min}\right)\alpha_b \tag{21}$$

where Q_i represents bat *i* frequency, $Q_{\min}$ and $Q_{\max}$ represents the minimum and maximum frequency of the bats, and α_b represents a random number having an array from 0 to 1.

The value of velocity for each virtual bat changes as the iteration-size of a BA increases. To evaluate the value of a virtual bat velocity at step *t*, the current position of a bat, the global solution for the bats and current frequency of a bat at iteration-size *t*-1 are combined as Equation (22).

$$v_{ij}^t = v_{ij}^{t-1} + (x_{ij}^{t-1} - x_j^*)Q_i \tag{22}$$

where v_{ij}^t represents the velocity of bat *i* for decision variable *j* at step *t*, x_{ij}^t represents the position of bat *i* for decision variable *j* at step *t* and x_j^* represents the global value of decision variable *j*.

This new velocity (v_{ij}^t) is used in flying a virtual bat to a new position by combining it with its position at iteration-size *t*-1. The decision on whether to accept a new position, which is the result obtained from Equation (23), depends on the pulse rate (r_i^t) of a bat and a random number (V_i) that lies between $(0,1)$. If $r_i^t < V_i$, the bat is allowed to walk randomly around its old solution at iteration-size *t*-1. This enables any bat that experiences this problem to carry out local searches based on its previous best solution at iteration size *t*-1. This is achieved using the average loudness (A^t) of at the bats at iteration-size *t* and within the range of -1 and 1. This computation can be achieved using Equation (24).

$$x_{ij}^t = v_{ij}^t + x_{ij}^{t-1} \tag{23}$$

$$x_{ij}^t = x_{ij}^{t-1} + \varepsilon A^t \tag{24}$$

As the BA iteration-size increases, the values of r_i^t and A_i^t are subject to modifications in a way that the value of r_i^t for bat *i* increases and the value of A_i^t for bat *i* decrease as shown in Equations (25) and (26), respectively. The decision on when to modify the values of r_i^t and A_i^t depends on the difference between the quality of the solution at iteration-size *t* (f_t) and iteration-size *t*-1 (f_{t-1}). Also, consideration is given to the difference between the current value of A_i^t and a random number (U_i) the lies between $(0,1)$. If $f_t < f_{t-1}$ and $U_i < A_i^t$, the values of r_i^t and A_i^t increase and decrease, respectively. If the conditions are not satisfied, the values of r_i^t and A_i^t at iteration *t* are retained.

$$A_i^t = \beta A_i^{t-1} \tag{25}$$

$$r_i^t = r_i^0\left(1 - \exp(-\ddot{\alpha} t)\right) \tag{26}$$

where $\ddot{\alpha}$ and β represent constant parameters.

To generate an unbiased solution (optimal solution), PSO algorithm combines the cognitive knowledge (personal best solution) and social (global solution) knowledge at a particular iteration-size *t*, as shown in Equation (27) [41]. The directions which particles move towards are controlled by the current velocity (v_{ij}^t) and the previous position (x_{ij}^{t-1}), as defined in Equation (28). In order to reduce the influence of v_{ij}^t on a particle's new position, the use of velocity clamping is proposed by Eberhart *et al*. [42].

$$v_{ij}^t = \bar{\alpha} v_{ij}^{t-1} + c_1 r_{1j}(x_{ij}^{lbest} - x_{ij}^{t-1}) + c_2 r_{2j}(x_{ij}^{gbest} - x_{ij}^{t-1}) \tag{27}$$

$$x_{ij}^t = v_{ij}^t + x_{ij}^{t-1} \tag{28}$$

where $\bar{\alpha}$ is known as weight, c_1 and c_2 are constants, and r_1 and r_2 are uniform random numbers within the range of $(0,1)$, and $lbest$ and $gbest$ represents personal and global best solutions, respectively.

3. Application, Results and Discussion

The datasets used in demonstrating the applicability of the proposed model is obtained from Prajina's work [29]. The information on the ANN architecture used is presented in Table 1.

Table 1 - ANN parametric settings

Parameters	Values
Number of inputs	4
Number of output	1
Performance (Mean square error)	$1 \, x \, 10^{-3}$
Number of hidden layers	2
Number of epochs	6000
Number of neurons in hidden layer	11-4
Training algorithm	Gradient descent
Transfer function	Sigmoid function

In this article, four input features are available for the prediction network while the size of the output layer is one. However, the question of interest is whether there are criteria used to decide on the number of hidden layers for the problem as well as the adequate number of nodes that should be in the hidden layer. To tackle this problem, there are two schools of thought: One believes that no specific approach exists but a trial and error method will help. However, the second school of thought believes that some general rules exist. While studying the techniques in the two schools of thought, the adopted approach was approved by the results obtained by the two techniques. To align with the first school of thought, cross-validation was used to obtain the accuracy of the test set. The correlation coefficient obtained for testing the data were 0.81 and 0.52 for the developed model of the artificial neural network. The proponent of the first school of thought noted that the optimum number of hidden units may be lower than the number of inputs. But the inputs are the cutting speed, feed rate, depth of cut and immersion angel, which are four items. Consequent, the chosen number of hidden units is two, which is lower than four, and in concurrence with the recommendation of the first proponent. Furthermore, the proponent asserted that occasionally two hidden units are adequate and work best when the data is little, which is the present situation. For the second school of thought that believes in some general rules, there are many variants. Some researchers proposed the formula as (no of inputs + no of outputs) 0.5 + (1 to 10). For the present case, the number of inputs is 4, the number of output is 1, and the results of the application of the formula yield 2.236 + (1 to 10), which means between 3.236 and 12.236. However, the chosen number of layers, although outside this range, is still close to the lower limit of 3.236, which is 2.

For the school of thought that follows quantitative rules, a second variant proposed 2/3 of the total number of inputs. As there are four features, the 2/3 of 4 is $2\,^2/_3$ and two hidden layers were chosen based on this, which makes the obtained results in alignment with the proposal by this second school of thought, and supported by the 1988 technical report given by Cybenko. In the literature, it was suggested that the use of a large number of neurons in each hidden layer is appropriate. Consequently, in this article, eleven neurons were used for the first hidden layer while four neurons were used for the second hidden layer. Thus, these choices concurred with the literature suggests.

Furthermore, the learning algorithm incorporated in the development of the artificial neural network model is gradient descent, which is a straight forward approach to minimize the objective function of the artificial neural network model. The gradient descent, while recognized as an optimization algorithm is effective in the literature to iteratively minimize some function while navigating in the path of the steepest descent, known as the negative of the gradient. The gradient descent was effectively used to train the artificial neural network model.

Based on the implementation of the parametric settings for the ANN architecture in Table 1, we use the literature datasets and the relationship between the predicted and experimental values for surface roughness and machining time during the training and testing stages of the ANN models, presented in scatter plots in Figures 5 to 8.

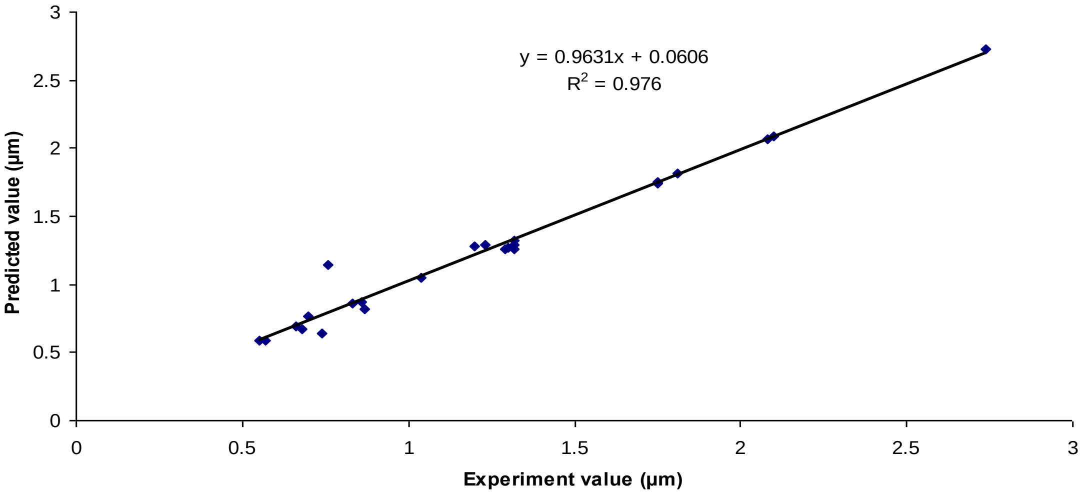

Fig. 5 - Scatter plot for experimental and predicted values of surface roughness using training datasets

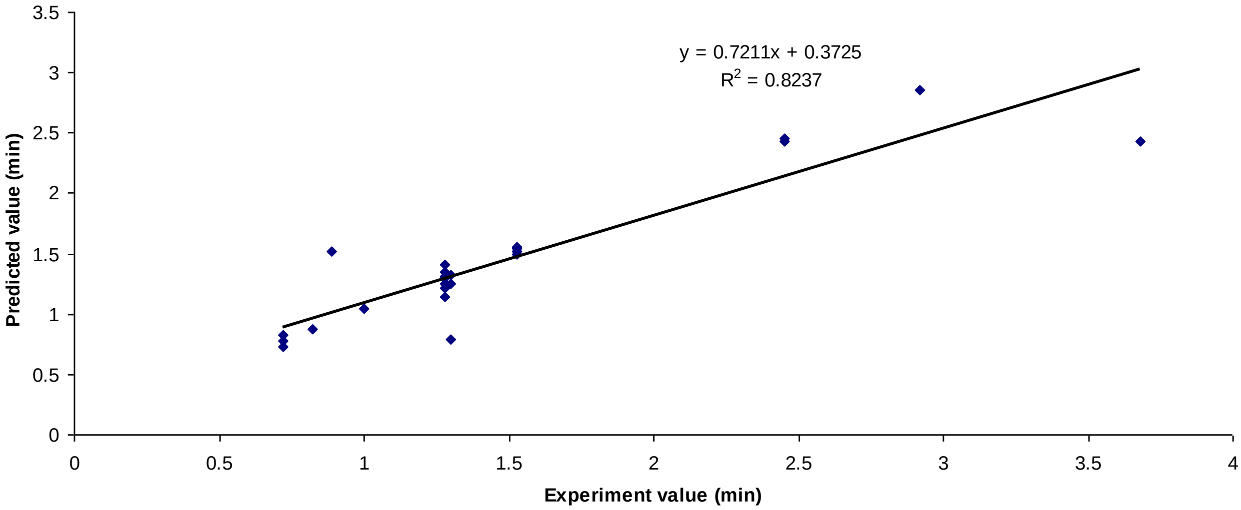

Fig. 6 - Scatter plot for experimental and predicted values of machining time using training datasets

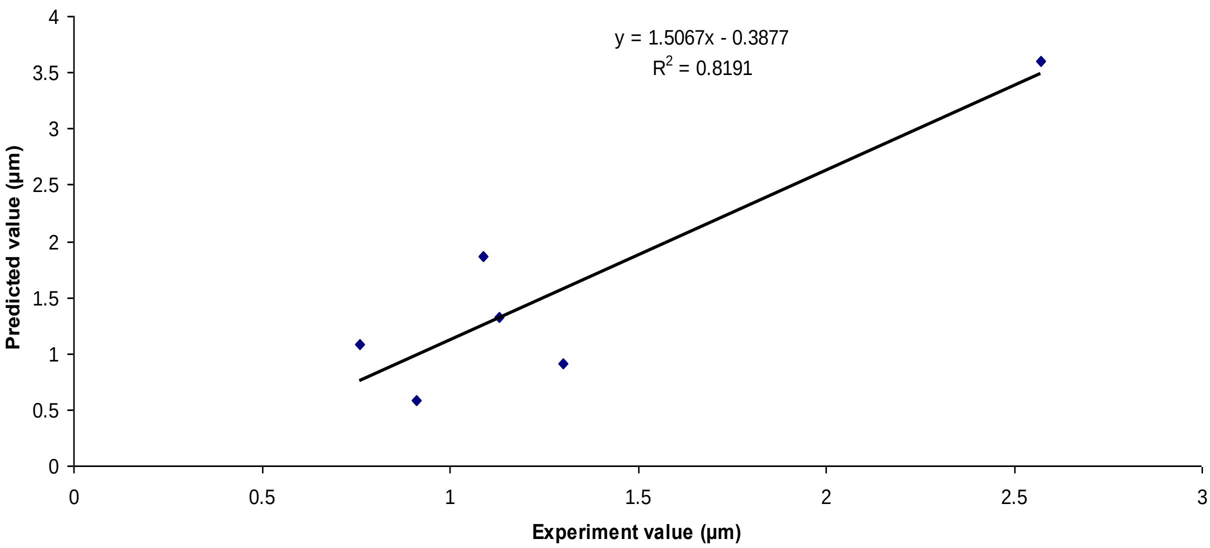

Fig. 7 - Scatter plot for experimental and predicted values of surface roughness using testing datasets

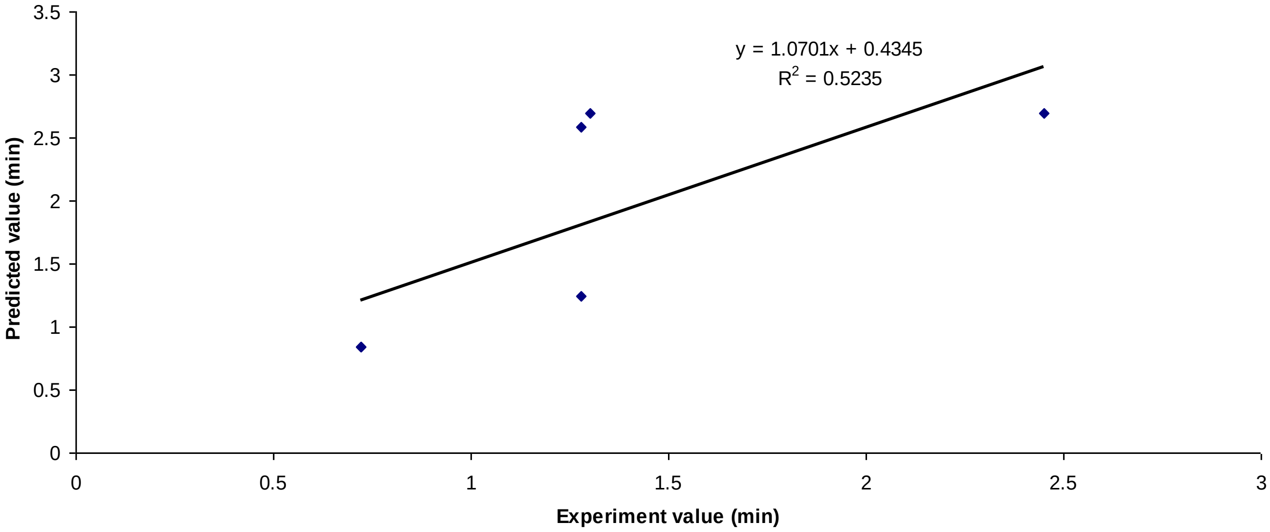

Fig. 8 - Scatter plot for experimental and predicted values of machining time using testing datasets

Interestingly, from the correlation coefficient testing data, 0.8191 was obtained for surface roughness while 0.5235 was the result for the machining time. From the literature, a correlation coefficient of more than 0.8 is usually portrayed as strong and since 0.8191 is greater than 0.8, it is strong and used for further analysis in this article. Moreover, the literature describes a correction coefficient of less than 0.5 as weak. While the 0.5235 value obtained for the machining time is comparatively low, it is above the threshold of 0.5 and may not be described as weak. Consequently, this value was used for further analysis in this work.

Furthermore, experimental data in machining science and technology refers to data obtained through different means, including experimental design, which is used for the case study under investigation. In this article Table 2, which describes the central composite design matrix from Prajina [29] was used as the reference data upon which further analysis on the ANN, its integration with the bat algorithm and the particle swarm optimization were made. The experimental data consist of four inputs, namely depth of cut, feed rate, cutting speed and immersion angle from which five output terms were generated (F_x, F_y, F_z, R_a and T_m). These outputs are the forces in the x, y and z directions as well as the roughness average and the machining time. However, the focus of this article is only on two output measures, namely surface roughness (measured as roughness average) and the machining time. Machining is the processes involved in transforming metal pieces into final products through a metal removal process. But the machining time is crucial to the process engineer, which is the time when the machine processes the metal pieces. Furthermore, surface roughness evaluates the texture of a metal's surface, revealing the vertical deviation of an evaluated surface from its ultimate appearance. As surface roughness offers a reliable prediction of the performance of metal pieces during processing it is the focus of attention in the present article. Consequently, the machining time and surface roughness were chosen as the outputs reported in Prajina [29]. Besides, it is interesting to know how much percentage of data is used for training, testing and validation in this article. To address this issue, it should be noted that the training set was used to build up the artificial neural network model while the test (or validation) set serves as a useful purpose to validate the model built. Furthermore, while the process of training was successfully achieved, the data set was excluded from the validation set to avoid interference and permit the independence of judgments. Consequently, according to literature [43, 44], the 70% data proportion recommended for training and 30% data proportion suggested for testing has been adopted in the present article.

Table 2 presents the bounds for cutting speed, depth-of-cut, feed rate and immersion angle.

Table 2 - The bounds of selected end milling parameters

Parameter	Minimum	Maximum
Cutting speed (m/min)	56	224
Depth-of-cut (mm)	0.4	2
Feed rate (mm/sec)	0.3	1.5
Immersion angle (°)	90	360

The parametric settings for the BA and the PSO used in solving the proposed optimisation model are presented in Table 3. Based on these parametric settings, convergence plots were generated for the BA and PSO algorithms (Figure 9). These plots showed that the PSO algorithm converged faster than the BA algorithm (Figure 9). Furthermore, the computational time of the PSO algorithm is less than that of the BA algorithm. However, the BA algorithm solution qualities are better than the PSO algorithm (Table 4).

Table 3 - Parametric settings for the solution methods

Parameter	PSO	BA
Number of iteration	100	100
C_1 and C_2	1.5 and 1.2	-
W	0.5	-
α and β	-	0.9 and 0.9
Particle size	30	-
Number of bats	-	30

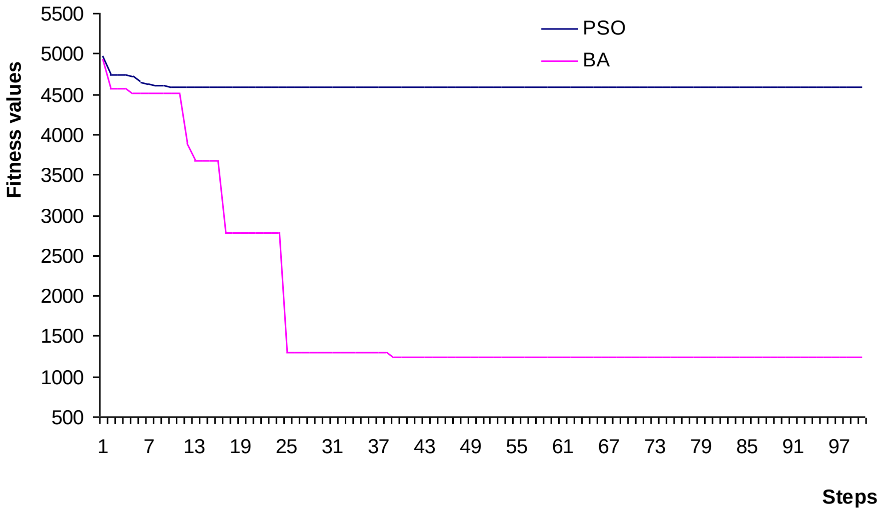

Fig. 9 - Convergence plots for the meta-heuristics

Table 4 - Analysis of the PSO and BA algorithm fitness results

Algorithm	Time (sec)	Maximum Value	Minimum value	Average Value	Confidence level at α = 99%
PSO	149.07	4961.85	4582.27	4594.16	2596.20 - 6156.47
BA	256.05	4924.14	1245.29	1865.41	1134.62 - 3031.85

The results obtained from using the BA and the PSO algorithm in solving the developed mathematical model for the minimisation of surface roughness and machining time are presented in Table 5.

Table 5 - Optimal value for the selected ending milling parameters

Parameter	PSO	BA
Surface roughness (μm)	1.68	0.78
Machining time (min)	1.28	2.46
Depth-of-cut (mm)	1.21	0.87
Feed rate (mm/sec)	0.86	0.99
Cutting speed (m/min)	112	105
Immersion angle (^{o})	108	110

The decision to predict surface roughness and machining time using different ANN models are to ensure that the differences between experimental and predicted values for either surface roughness and machining time are extremely low. The number of training and testing data sets was 24 and 6, respectively. A joint prediction may affect the quality of the ANN model produced. With low prediction errors, the values from the proposed optimisation model may be considered practicable to implement. The ANN model and BA algorithm are implemented using VB.Net programming language on a 1.80 GHz processor, installed memory of 4.0 GB and Windows 8 pro-personal computer. However, existing optimisation packages like GAMS and CPLEX were not used to verify the quality of the solution from the BA and PSO because of the difficulty in integrating the generated non-linear equations from the developed ANN model into optimisation package (GAMS or CPLEX). Furthermore, the work has been based on the design of experiments (DOE) and specifically, the Taguchi approach is used in complement with predictive models (surface response method and ANN) and optimisation (linear and non-linear) model to fully understand the interrelationships among machining parameters.

Figures 5 to 8 verify the use of ANN as a suitable predictive model for surface roughness and machining time predictions using end milling datasets (R^2 are suitable). The values of the mean (μ) and standard deviation (σ) for surface roughness and machining time are used in fixing the values of χ_1 and χ_2 for surface roughness and machining time, respectively. At 1σ from the μ (i.e., μ - σ), the values for χ_1 and χ_2 are 1.82 μm and 2.13 min, respectively. The fixing of the numbers of σ from μ should be done with due consideration to the CNC process capability. Since product quality is assumed to be more important than the machining time requirement in the production system, according to customers' demand, then a decision maker is faced with two causes-of-actions: to assign equal weights to surface roughness and machining time (i.e., $w_1 = w_2$), and to assign surface roughness objective a higher weight value than the machining time ($w_1 \geq w_2$). Both options are considered here (for the first cause-of-action, the left-hand side of Equation (3) will be equal to 1). For the second cause-of-action, it was assumed that the maximum weight for surface roughness objective should be 0.7 (i.e. the right-hand side of the equation will be 2.33). Equations (29) and (30) were used for the confidence interval of the BA and PSO algorithms [45].

$$S = \bar{e} \pm t_{\alpha,n-1}\sigma \tag{29}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{m}\left(e_i - \bar{e}\right)^2}{n\left(n-1\right)}} \tag{30}$$

where S, $\bar{e}$ and σ represent confidence interval, mean value and standard deviation of an algorithm solution, respectively.

The optimal value for λ_1 and λ_2 using the PSO algorithm as a solution method are 0.8816 and 0.8088, respectively, while 0.557 and 0.443 are obtained as the optimal value of w_1 and w_2, respectively. The results generated for the proposed model using BA as a solution method yield optimal values of 0.275 and 0.178 for λ_1 and λ_2, respectively, while the optimal values for w_1 and w_2 are 0.681 and 0.319, respectively. The variation in the values w_1 and w_2 obtained using the BA and PSO parametric settings may be attributed to the differences in the optimal values for surface roughness and machining time. Based on the assumption that surface roughness objective is more important than machining time, it can be inferred that the BA solution is superior to the PSO solution. Furthermore, the large value of machining time in Table 5 has direct relationships with the low values of cutting speed and depth-of-cut those are obtained from the BA solution.

This work reveals certain novel features. First, it builds up an optimisation structure that offers interfaces among the significant parameters of the bi-objective weighted fuzzy goal programming model and enhances the surface roughness by reducing the roughness parameter while reducing the machining time for processing the work-piece. Consequently, this paper advances machining practice since the presented framework can offer valuable budgetary information for end milling machine. It thus aids production planning and control and machining cost monitoring. Furthermore, the work has unique features of bats and particle orientations. The advantage of building up new knowledge in machining practices related to the capture of uncertainties. Also, the work uses the new framework of FGP-BA as well as FGP-PSO for end milling for the first time.

Model Validation

In this article, the term model validation is viewed as an activity of confirming that the output of the models proposed is acceptable regarding the real data-creation process. Consequently, the artificial neural network model was validated by using the data produced in Prajina [29]. The developed mathematical model for experimental and predicted values of surface roughness and machining time using testing datasets are given as:

$$y = 1.5067x - 0.3877 \tag{29}$$

and

$$y = 1.0701x + 0.4345 \tag{30}$$

respectively. But values could be generated based on Equations (29) and (30) and compared with the training data (last 30% of total data), which is present in experimental trails no. 23 to 30, the correlation could then be obtained for the relationship. Consequently, new data were generated using Equations (29) and (30) while X varies from 0.5 to 4 as 0.5, 1, 1.5, 2, 2.5, 3, 3.5, and 4, respectively. For Equation (29), which represents the surface roughness expression, the progressive values (predicted) obtained ranged from 0.36565 to 5.6391. However, the counterpart values of the surface roughness based on experimental trials 23 to 30 offered values that ranged from 0.9 to 2.1. The values are correlated and a value of -0.62131, which is relatively acceptable. For Equation (30), which represents the machining time, the progressive values (predicted) obtained ranged from 0.96955 to 4.7149. Nonetheless, the counterpart values of the machining time based on experimental trials 23 to 30 provided values that ranged from 0.89 to 2.92. The values are correlation and a value of 0.569884, which is weaker than the correlation obtained for surface roughness. However, this value exceeds 50% and could be said to be acceptable. Thus, the ANN model has been validated.

4. Conclusions

In this study, BA and PSO algorithms were innovatively applied to surface roughness and machining time using a weighted fuzzy bi-objective optimisation and literature data. The optimum value for surface roughness is given by bat algorithm and hence the ANN-BA model is recommended for surface roughness. Besides, the optimum value for machining time as obtained by the PSO algorithm as ANN-PSO, therefore, revealing that the ANN-PSO algorithm is suitable for optimizing the process parameters. In the present investigation, a new methodological framework has been developed and tested for the end milling process optimisation. There is a concurrence of the literature data with the predictions. Further studies can be carried out using other meta-heuristics (colliding bodies optimisation, big-bang big-crunch algorithm) when applying the proposed model using a different set of data. The proposed model shows that it can be used in addressing the problem of weight assignments for different objectives during end milling operations.

Acknowledgement

The author would like to acknowledge the Department of Mechanical and Biomedical Engineering, Bells University of Technology, Ota, and Department of Mechanical Engineering, Faculty of Engineering, University of Lagos, Room 10, Mezzanine Complex, Akoka-Yaba, Lagos, Nigeria.

References

[1] Wojciechowski S, Maruda R.W, Barrans S. et al. Optimization of machining parameters during ball end milling of hardened steel with various surface inclinations. *Measurement* 2017; 111: 18-28

[2] Liu Q, Cheng J, Xiao Y et al., Effect of milling modes on surface integrity of KDP crystal processed by micro-ball-end milling. *Procedia CIRP*, 2018; 71: 260-266

[3] Lu Y.A., Ding Y. and Zhu L.M., Tool path generation via the multi-criteria optimisation for flat-end milling of sculptured surfaces. *Int J Prod Res.*, 2017; 55(15): 4261-4282

[4] Moufki A, Le Coz G., Dudzinski D, End-milling of Inconel 718 superallory - An analytical modelling. *Procedia CIRP*, 2017; 58: 358-363

[5] Tehranizadeh F and Budak E., Design of serrated end mills for unproved productivity, *Procedia CIRP*, 2017; 58: 493-498

[6] Altintas Y, Tuysuz O, Habibi Li Z.l., virtual compensation of deflection errors in ball end milling of flexible blades, *CIRP Ann-Manuf Techn.*, 2018; 67: 365-368

[7] Yuan Y, Jing X, Ehmann KF. et al. Modeling of cutting forces in micro-end milling, *J Manuf Process*, 2018; 31: 844-858

[8] Nghiep T.N, Sarhan AAD, Aoyama E., Analysis of tool deflection errors in precision CNC end milling of aerospace aluminum 6061-T6 alloy, *Measurement*, 2018; 125: 476-495

[9] Gao Y, Ko JH, Lee H.P. 3D Eulerian finite element modeling of end milling, *Procedia CIRP*, 2018; 7: 159-162

[10] Quintana G., Ciurana J. and Ribatallada J., Modelling power consumption in ball-end milling operations, *Mater Manuf Processes*, 2011; 26(5): 746-756

[11] Rao K.V., Power consumption optimization strategy in micro ball-end milling of D2 steel via TLBO coupled with 3D FEM simulation. *Measurement*, 2019; 132: 68-78

[12] Shi K. N., Ren J. X., Wang S. B. et al., An improved cutting power-based model for evaluating total energy consumption in general end milling process, *J Clean Prod*, 2019; 231: 1330-1341

[13] Xu G, Zhou H, Chen J., CNC internal data based incremental cost-sensitive support vector machine method for tool breakage monitoring in end milling, *Eng Appl Artificial Intellig*, 2018; 74:90-103.

[14] Wang P, Zhang S, Li Z. et al., Tool path planning and milling surface simulation for vehicle rear bumper mold, *Adv Mech Eng*, 2016; 8(3):1-10

[15] Zain A.M., Haron H. and Sharif S., Simulated annealing to estimate the optimal cutting conditions for minimizing surface roughness in end milling Ti-6AL-4V, *Mach Sci Tech*, 2010; 14,(1): 43-62.

[16] Moshat S., Datta S., Bandyopadhyay A. et al., Optimisation of CNC end milling process parameters using PCA-based Taguchi method, *Int J Eng, Sci Tech*, 2010; 2(1):92-102.

[17] Maiyar LM, Ramanujam R, Venkatesan K. et al., Optimization of machining parameters for end milling of incorel 718 super alloy using Taguchi-based grey relational analysis, *Procedia Eng*, 2013; Vol. 64, pp. 1276-1282

[18] Vijay S. and Krishnaray V., Machining parameters optimization in end milling of Ti6Al-4V, *Procedia Eng*, 2013; 64: 1079-1088

[19] Routara B.C., Bandyopadhyay A. and Sahoo P., Roughness modeling and optimization in CNC end milling using response surface method: Effect on workpiece material variation, *Int J Manuf Tech*, 2009; 40: 1166-1180.

[20] Routara B.C., Mohanty S.D., Datta S. et al., Optimisation of CNC end milling of UNS C34000 medium leaded brass with multiple surface roughness characteristics. *Sadhana*, 2010; 35(5): 619-629.

[21] Ighravwe D.E. and Oke S.A., Machining performance analysis in end milling: prediction using ANN and a comparative optimisation study of ANN/BB-BC and ANN/PSO, *Engineering J*, 2015; 19(5):121-137.

[22] Para V., Agnihotri, G. and Krishna, C.M., Optimisation of cutting in end milling process with the approach of particle swarm optimisation. *Int J Mech Indust Eng*, 2011; 1(2): 21- 25.

[23] Yang X.S., A new metaheuristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimisation (NICSO2010) (Eds. J. R. Gonzalez *et al*.), Studies in Comput Intellig, Springer Berlin, 2010; 284: 65-74.

[24] Yang X-S., Bat algorithm for multi-objective optimisation. *Int J Bio-inspired Comput*, 2011; 3(5): 267-274.

[25] Yang X-S. Bat algorithm: Literature review and applications, *Int J Bio-inspired Comput*, 2013; 5(3): 141-149.

[26] Wulandhari LA, Komsiyah S, Wicaksomo W, Bat algorithm implementation on economic dispatch optimization problem, *Procedia Computer Sci*, 2018; 135: 275-282

[27] Urbikain C and Lopez de Lacalle LN, Modeling of surface roughness in inclined milling operations with circle segments and mills, *Simulat Modeling Pract Theory*, 2018; 84:161-176

[28] Lofti S, Rami B, Maher B. et al. An approach to modeling the chip thickness and cutter work piece engagement region in 3 and 5 axis ball end milling, *J Manuf Procesess*, 2018; 34: 7-17

[29] Prajina N.V. Multi response optimisation of CNC end milling using response surface methodology and desirability function. *Int J Eng Res and Tech*, 2013; 6(6): 739-746.

[30] Eberhart R.C. and Kennedy J. A new optimiser using particle swarm theory. *In Proc Sixth Int Symp Micromach Human Sci*, 1995; 39-43.

[31] Niaki S.T.A. and Hoseinzade S. Forecasting S&P 500 index using artificial neural networks and design of experiments. *J Ind Eng Int*, 2013; 1-9.

[32] Theja K.D., Gowd G.H. and Kareemulla S. Prediction and optimisation of end milling process parameters using artificial neural network. *Int J Emerg Tech Adv Eng*, 2013; 3(9): 117-122.

[33] Rumelhart D.E., McClelland J.L. and the PDF Research Group, Parallel Distributed Processing, MA: MIT Press, Cambridge, 1986.

[34] Lippmann R.P., An introduction to computing with neural nets. *IEEE ASSP Magazine*, 1987; 4: 4-22.

[35] Zhang X., Time series analysis and prediction by neural In: Casdagli, M., Eubank, S. (Eds.), Nonlinear Modelling and networks. *Optimisation Methods and Software*, 1994; 4: 151-170.

[36] Zhang G., Patuwo B.E. and Hu M.Y., Forecasting with artificial neural networks: The state of the art. *Int J Forecast*, 1998; 14, 35-62.

[37] Ross T.J., *Fuzzy Logic with Eng Appl*, 2nd, John Wiley and Sons, Ltd. England, 2005.

[38] Amid A. and Ghodsypour S.H. An additive weighted fuzzy programming for supplier selection problem in a supply chain. *Int J Ind Eng Prod Res*, 2008; 19(4): 1-8.

[39] Tzeng G-H. and Huang J-J. *Multiple attribute decision making methods and applications*. CRC Press, New York, 2011.

[40] Yusoff A.R. and Sims N.D. Optimisation of variable helix end milling tools by minimising self-excited vibration. 7th *International Conference on Modern Practice in Stress and Vibration Analysis*, 2009.

[41] Millonas M.M. Swarms, phase transitions, and collective intelligence. Artificial Life III, Addison-Wisley, 1994; 417-445.

[42] Eberhart R.C., Simpson P.K. and Dobbins R.W. Computation Intelligence PC Tools. Academic Press Professional, 1st ed., 1996

[43] Ighravwe D.E., Oke S.A. The application of artificial neural network and fuzzy inference system for machining process with respect to material removal rate and surface roughness, *Kenya J Mech Eng*, 2015a; 8(1): 9-23

[44] Ighravwe D.E. and Oke S.A. Machining performance analysis in end milling: Predicting using ANN and a comparative optimisation study of ANN/BB-BC and ANN/PSO, *Eng J*, 2015b; 19(5): 121-137

[45] Engelbrecht A.P. *Artificial Intelligence: An Introduction*, John Wiley, England, 2007.