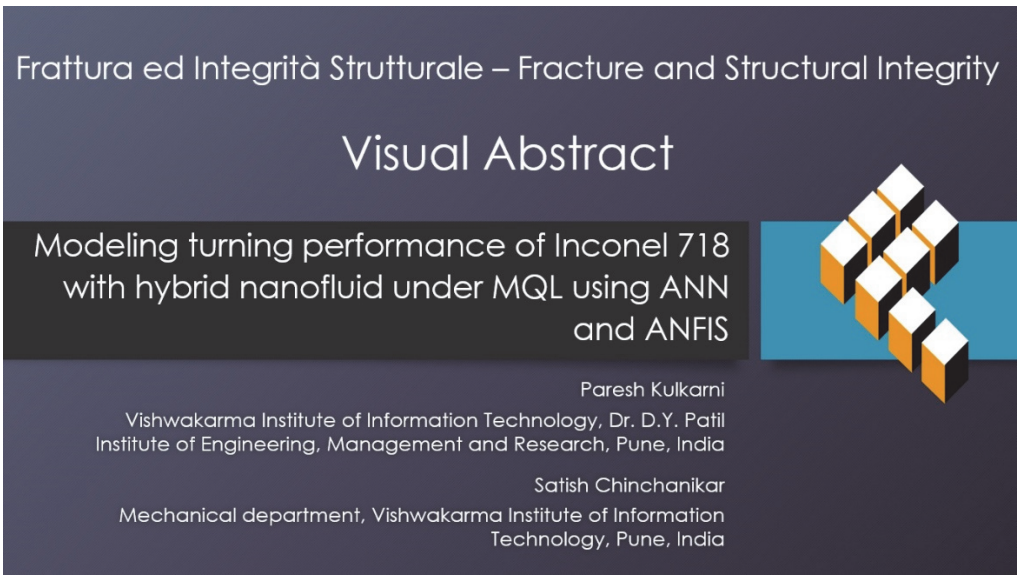
Modeling turning performance of Inconel 718 with hybrid nanofluid under MQL using ANN and ANFIS

Paresh Kulkarni
Vishwakarma Institute of Information Technology, Assistant Professor at Dr. D.Y. Patil Institute of Engineering, Management and Research, Pune-411044, India
paresh.219p0014@viit.ac.in, http://orcid.org/0000-0002-2761-8754

Satish Chinchanikar
Mechanical department, Vishwakarma Institute of Information Technology, Pune, India
satish.chinchanikar@viit.ac.in, http://orcid.org/0000-0002-4175-3098



Citation: Kulkarni, P., Chinchanikar, S., Modeling turning performance of Inconel 718 with hybrid nanofluid under MQL using ANN and ANFIS, Frattura ed Integrità Strutturale, 70 (2024) 71-90.

Received: 17.06.2024
Accepted: 16.07.2024
Published: 02.08.2024
Issue: 10.2024

KEYWORDS. ANN, ANFIS, Inconel 718, Subtractive manufacturing, Tool wear, Fracture, Modeling.

INTRODUCTION

The expansion of the manufacturing field during the previous decade has had an influence on the development of several countries. Precise and sustainable machining are critical to the industrial economy and environmental sustainability. With the goal of enhancing the cutting of nickel alloys using a sustainable strategy, significant research was conducted on cutting and Minimum Quantity Lubrication (MQL) variables. [1]. The Inconel 718 nickel alloy is difficult

to cut due to its high temperature strength, work hardening inclination, and restricted thermal conductivity. As a result, it offers a variety of challenges in the machining space. Various sustainable techniques were used to improve the machining of Inconel 718 material while also considering environmental concerns [2-3]. Nanoparticle-based materials like multi-walled carbon nanotubes (MWCNTs) and aluminum oxide (Al_2O_3) nanofluids were utilized during cutting to enhance machining performance and safeguard operator health [4].

Manufacturing in prior decades relied exclusively on massive imperial databases produced by previous researchers. Efficient machining can be achieved through the development of suitable cutting techniques and the application of scientifically proven methods like artificial intelligence (AI) and soft computing. Accurate modeling techniques are vital in modern manufacturing for designing efficient machining processes, reducing production costs, enhancing product quality, and enhancing operational efficiency. These techniques aid engineers in simulated scenarios, optimizing parameters, saving time and cost, and predicting issues, thereby preventing costly errors during manufacturing. Soft computing techniques are being increasingly used for modeling owing to their self-learning capabilities, fuzzy principles, and evolutionary computational philosophy, which computes the total of all incoming signals. These techniques offer versatile solutions for complex and uncertain data, adapting and learning from new information, making them ideal for a wide range of modeling applications. These methods work especially well in areas like optimization, pattern recognition, and artificial intelligence, where more traditional approaches could have trouble yielding reliable results. Another benefit of soft computing techniques is their capacity for continuous improvement and adaptation, which makes them ideal for dynamic and changing systems.

The integration of AI techniques in the manufacturing field has evolved in recent years, like artificial neural networks (ANN) and adaptive neuro-fuzzy inference systems (ANFIS). The ANFIS, was used in the die sinking electric discharge machining (EDM) process to predict surface roughness and metal removal rate for AISI D2 tool steel [5]. The Genetic programming (GP) models predicted the machining characteristics with the highest accuracy and are recognized for their ability to automatically find complicated correlations and patterns in data without prior assumptions [6]. The tool wear behavior in turning AISI 304 stainless steel was investigated using an empirical and ANN modeling technique. The ANN model showed better prediction precision than the traditional empirical models [7]. The hybrid machine learning (ML) models were developed to predict induced residual stresses during Inconel 718 alloy turning. The hybrid pigeon optimization algorithm (POA) and particle swarm optimization (PSO) performed better than the standard unitary ANN model in terms of output response predictability [8].

The ANN model was trained using the Levenberg-Marquardt (LM) prediction model for forecasting the machining rate and tool wear of Inconel 718 [9]. Finite element modeling (FEM) was utilized to estimate the cutting force, while GP was used to determine the mathematical relationship between the process factors to conserve power consumption while machining hard-to-cut Inconel 718 [10]. The performance of an ANN model for tool wear agreed well with measured flank wear while turning Inconel 718 [11]. The wear rate of friction stir processed surface composites was analyzed using ANN, with the feedforward back propagation approach adjusted to minimize mean squared error [12].

The ANFIS was used with different membership functions to forecast the mechanical properties of friction steel welded AA7075-T651 joints. When compared to the other functions, the triangular membership function (Trimf) had the lowest level of inaccuracy [13]. An attempt was made to establish a prediction model for machining time during milling of Inconel 718 with the help of Box-Behnken designs (BBD)-response surface methodology (RSM) and ANN techniques, and it was discovered that ANN based modeling methods outperform the BBD-RSM methods in predicting machining time [14]. Back-propagation techniques were utilized to develop ANN models for electric discharge machining of Inconel-718 [15].

The ANN model performed exceedingly well statistically, with a solid correlation and a very low error ratio between actual and anticipated flank wear data while minimizing tool wear during the milling of Inconel 718 [16]. ANFIS technology was used for EDM Inconel 625 and showed good prediction accuracy for surface roughness, tool wear, and metal removal rate [17]. ANN models are built utilizing feed-forward back propagation algorithms and showed good prediction accuracy for machining Inconel 718 [18]. The study utilized Mamdani-based fuzzy logic to optimize settings, followed by a desirability function technique from RSM to maximize material removal rate (MRR) and minimize surface roughness. In comparison to the experimental settings, the fuzzy system improves prediction accuracy [19]. ANN had the highest prediction efficiency compared to gene expression programming (GEP) during the EDM of Inconel 718 [20]. The ANN model, which makes use of a back-propagation neural network (BPNN), showed excellent prediction accuracy. Overall, the use of ANN has shown promising results in predicting and optimizing machining parameters. This approach can lead to improved efficiency and accuracy in achieving desired outcomes [21–24].

A group of researchers developed RSM, ANN, ANFIS, and GEP models for precise prediction of responses during the machining of nickel alloys. Researchers found that GEP emerged to be superior to ANN, some observed better prediction using ANFIS, and some studies found that ANN and ANFIS outperformed RSM [25–29]. Some studies reported better predicting accuracy by ANN than ANFIS [30]. Overall, the comparison of these models showed that each had its strengths

and weaknesses in predicting responses during the machining of nickel alloys. The choice of model may depend on the specific requirements and characteristics of the machining process being studied.
From the literature review, it has been observed that ANN and ANFIS, being popular machine learning techniques, were mostly used to predict machining performance because of their ability to handle complex and non-linear relationships in data. However, very few attempts have been observed to model the turning performance of Inconel 718 with hybrid nanofluid under MQL. With this view, this study develops ANN and ANFIS models for cutting force, surface roughness, and tool life during the turning of Inconel 718 with hybrid nanofluid under MQL. Hybrid nanofluid was prepared by mixing aluminum oxide (Al_2O_3) and multi-walled carbon nanotubes (MWCNTs) at constant proportions in vegetable-based palm oil. The worn-out tools were analyzed through images captured using optical and scanning electron microscopes. This investigation aims to evaluate the effectiveness of ANN and ANFIS models in predicting the machining process. By comparing the performance of these models in predicting machining outcomes, this study seeks to provide valuable insights for improving process efficiency and accuracy.

METHODOLOGY

In this section, experimental details to develop predictive ANN and ANFIS models for cutting force, surface roughness, and tool life during the turning of Inconel 718 with hybrid nanofluid under MQL are presented. The preparation of a hybrid nanofluid, its properties, and machining and MQL parameters are discussed. Further, methodology to develop ANN and ANFIS models is presented.

Experimental details

The Inconel 718 cylindrical bar, 400 mm in length and 70 mm in diameter, was used as the specimen for the turning with an AlTiN-coated CNMG120408MS carbide cutting insert. A robust and accurate CNC lathe machine was used to conduct cutting tests, maintaining constant tool height, overhang, and geometry throughout the turning process (Fig. 1).

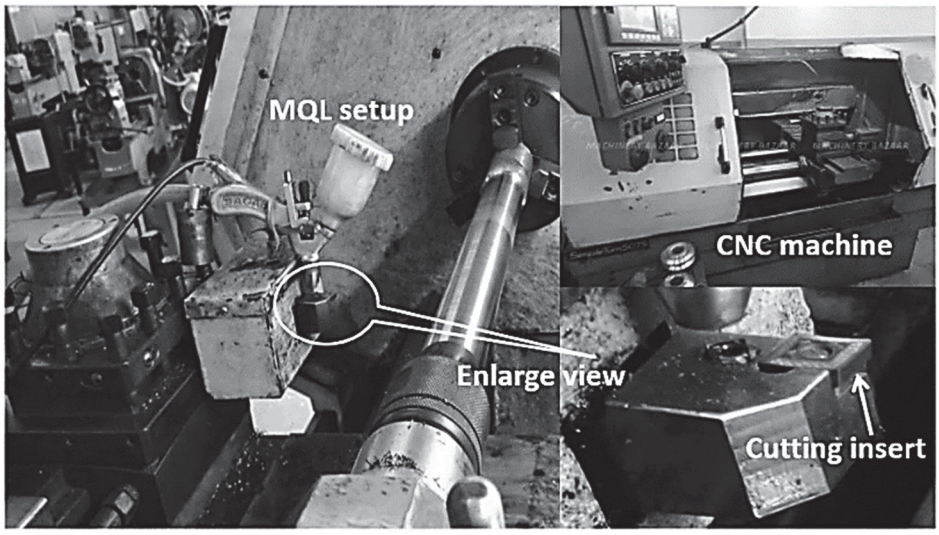


Figure 1: Experimental set-up.

A PCBNR2525M12 ISO-coded tool holder was utilized throughout the turning process. During the cutting process, a nanoparticle-based MQL environment was applied. To create the hybrid nanofluid, MWCNTs and Al_2O_3 nanoparticles were mixed with 50–50% vegetable-based palm oil. A typical two-step procedure was used to create a nanofluid. The nanofluid was mechanically stirred for 20 minutes at 700 rpm to ensure homogeneity, followed by probe sonication for 30 minutes at 50 kHz to dilute and stir the nanoparticles. To obtain further homogeneity and avoid sedimentation, the nanofluid underwent magnetic stirring for 20 minutes at 500 rpm to ensure homogeneity and prevent sedimentation, followed by settling for 10 minutes to eliminate any remaining air bubbles. Naphthyl sulfate was utilized as a surfactant to enhance nanofluid dispersion and stability, reducing agglomeration and preventing nanoparticles from settling over time. More details about the typical standard two-step method utilized for the preparation of nanofluids can be found in [31–32]. MWCNTs were chosen for their outstanding cooling and lubricating capacity, while Al_2O_3 was chosen for its better thermal

conductivity and excellent wetting effectiveness. The hybrid form will have a synergistic impact between the specific characteristics of MWCNTs and Al_2O_3. The measured characteristics of hybrid nanofluid are depicted in Tab. 1. The hybrid nanofluid exhibits enhanced thermal conductivity and heat transfer due to the combined action of Al_2O_3 and MWCNT nanoparticles in the base fluid.

Density (g/ml)	Viscosity (cP)	Acid value (KOH/g)	Surface tension (N/m)	Contact angle (°)	Thermal conductivity (W/m°C)
0.945	212	4.61	43.02	32.55	0.213

Table 1: Characteristics of hybrid Al_2O_3+MWCNT nanofluid.

The process parameters for the chosen workpiece-tool pair were carefully selected after a thorough literature review, pilot tests, machine capacity, and tool manufacturer advice. Based on a review of the relevant literature, pilot experiments, and advice from the tool's maker, the input variable ranges were chosen. These input variable ranges were selected to ensure optimal cutting performance and minimize tool wear. During the experiment, the MQL setup parameters were as follows: The flow rate was 60 ml/hour, the air pressure was 4 bar, the nozzle angle was 15 degrees, the standoff distance was 20 mm, and the nozzle diameter was 1 mm. The cutting speed ranged from 30 to 100 m/min, the feed ranged from 0.1 to 0.3 mm/rev, and the depth of cut ranged from 0.2 to 0.8 mm. The MQL parameters were optimized to ensure efficient cooling and lubrication during the cutting process. In total, 15 experiments were conducted using a central composite rotatable design test matrix with an alpha value of 1.6817, without any repetitions. The process parameters were altered at five different levels, which included the axial points of plus and minus alpha, factorial points of plus and minus one, and the center point. The set of cutting parameters used for turning Inconel 718 and the experimental results are presented in the results and discussion section (Tab. 2).
At each cutting parameter, the responses, namely cutting force, surface roughness, tool life, and tool wear analysis, were studied. The cutting force was measured using a previously calibrated strain gauge-type dynamometer. A Dino-Lite digital microscope was used to quantify the flank wear following each cutting pass, and a Mitutoyo SJ.201 surface roughness tester was used to determine the average surface roughness (Ra). In accordance with ISO 3685-1977(E) requirements, the tool life limits were established at 0.2 mm flank wear or a catastrophic failure. A carbide tool coated with PVD-AlTiN was used for the trials.

Artificial neural networks (ANN)

ANNs are computer programs that are highly helpful for data processing-related prediction and classification problems. ANNs, designed to mimic human brain processing, are effective for image recognition and natural language processing, using interconnected nodes to analyze data and learn patterns. They draw inspiration from the characteristics of biological neuron systems that resemble the human brain and acquire information via experience. This information is then utilized for data processing tasks like classification and prediction [33]. ANNs, through training, adapt and improve their performance over time, making them powerful tools for complex data analysis tasks in finance, healthcare, and marketing. By modifying the weights and biases (learning) in a network to capture the linear and non-linear structure of the data while keeping an acceptable error limit, ANNs can predict outcomes. Until the network displays a minimal error for each of the input and output values, the weights are iteratively changed. The choice of an appropriate network architecture, training techniques, and hyperparameters make this feasible [34]. The network training function utilizes Levenberg-Marquardt optimization to update weight and bias variables.
A multilayer perceptron (MLP) is a common feed forward artificial neural network, as seen in Fig. 2. The three levels that make up the MLP are the input layer, hidden layer, and output layer. Each layer is made up of a network of artificial neurons. Domain provides input to the input layer, from which neurons send the information to the hidden layer. The hidden layer performs many computations on the attributes provided by the input layer, and then sends the result to the output layer. The output layer essentially transforms the knowledge that has been learned in the hidden layers. On the other hand, the activation functions of the hidden layer and the output layer usually differ.
If more accurate output predictions are to be created, the network must be trained. To make sure the neural network generalizes successfully to new data, the training procedure necessitates monitoring performance metrics, carefully adjusting hyperparameters, and making necessary modifications. The most popular training algorithm is error back propagation. Initializing the neural network parameters is the first step in the typical ANN technique. Prior to the training process starting, this step comprises setting the initial settings for the network's weights and biases. Next, the weights are modified in accordance with the output node error, which is ascertained. When the ANN output for each set is sufficiently close to the

anticipated output, the training is considered to have ended. This process is done for every pair of input and output training data.

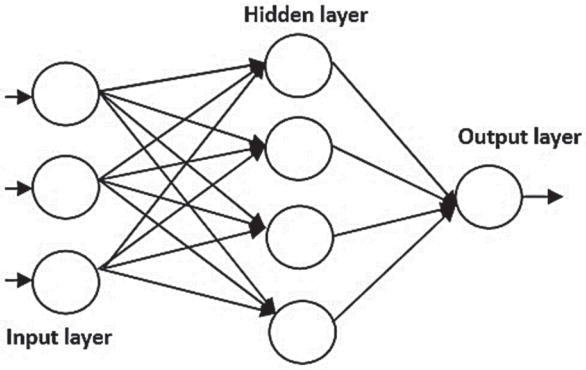


Figure 2: Typical ANN architecture.

Training of an ANN model

A neural network is a machine learning algorithm used to solve classification or regression problems. It involves preprocessing input features, initializing weights, adding bias, and choosing activation functions. The architecture is crucial for model building, aiming to minimize errors and generalize well to new data. Fig. 3 shows a flow chart for training the ANN model.

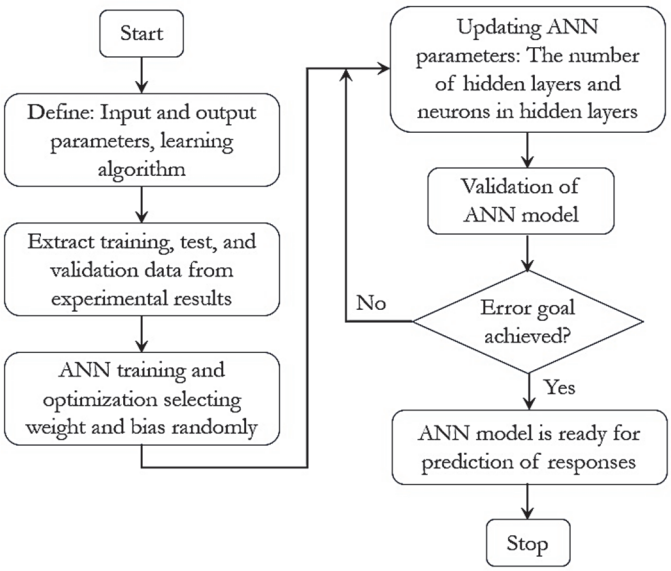


Figure 3: Flow chart of the training of the ANN model [7].

Parameters and hyperparameters used by ANN models are adjusted based on training data. To cut expenses, backpropagation optimizes parameters like weights and biases. Hyperparameters are established values that may be manually updated, although establishing ideal values might be problematic owing to dataset size and composition. An ANN's structure and training procedure are significantly influenced by hyperparameters like hidden layers, neurons, activation function, learning rate, loss function, epochs, optimizer type, batch size, etc. The calculation cost, algorithm execution time, and prediction accuracy are all significantly impacted by hyperparameter settings. The complexity of a dataset increases the number of hidden layers a neural network needs to identify significant non-linear patterns. The neural network's capacity for learning is thus determined. A smaller network with fewer hidden layers may not fit the training set, struggle to recognize intricate patterns, or effectively predict unknown data. A larger network with an excessive number of hidden layers could overfit the training set. As a result, that kind of network has poor generalization to unknown data. Hidden neurons in a network affect learning capacity, with too many creating large networks that overfit training data, and too few creating smaller networks that underfit.

An activation function applies mathematical processes to forecast a network and assesses if the input of a neuron is significant. It uses regression for a single neuron in the output layer and introduces non-linearity in hidden layers to derive output from input data. The optimizer's task is to reduce the loss function by updating network parameters using the popular gradient descent algorithm, which requires minimal steps to descend the error curve. The learning rate is a crucial hyperparameter in neural network training, determining step speed and direction, and should be started with a small value and gradually increased as needed.

The loss function is a tool used to evaluate the performance of a neural network during training by calculating the error between anticipated and actual values. The goal is to minimize the loss function by using an optimizer. The three types of loss functions that are possible are mean squared error (MSE), mean absolute error (MAE), and mean percentage error. The epoch is a crucial hyperparameter that represents the total number of times the model observes the entire dataset. An increase in the number of epochs is recommended when the network is trained at a very low learning rate or when the batch size is undersized.

Artificial neural fuzzy inference systems (ANFIS)

ANFIS is a system that integrates neural networks and fuzzy logic to handle uncertainties and imprecise data, enabling flexibility in complex system modeling while maintaining interpretability. It is a computer structure that establishes a hybrid system by fusing the ideas of neural networks with fuzzy logic. ANFIS is a machine learning method that creates a logical link between inputs and outputs by combining fuzzy logic (FL) theories with ANN rules inside an adaptive network architecture. The ANFIS system can effectively model complex, non-linear relationships between variables, making it interpretable and adaptable. By leveraging the strengths of both neural networks and fuzzy logic, ANFIS can provide accurate predictions and classifications in various applications. The ANFIS model has five levels, and there are many nodes connecting them. The layer before it extends each input node. Five different layers are used to create an inference system: the fuzzy layer, product layer, normalized layer, de-fuzzy layer, and the total output layer, which is the typical ANFIS architecture, as shown in Fig. 4.

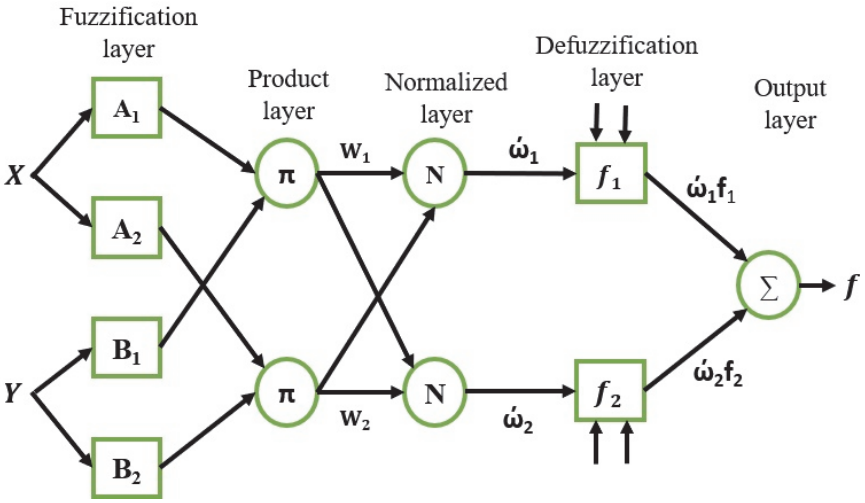


Figure 4: General ANFIS architecture.

The fuzzification layer (layer 1) converts the input space into fuzzy membership functions that are subsequently translated into linguistic categories. The product layer (layer 2) defines the level of membership for each rule. Each node in this layer computes the firing strength of a particular rule by multiplying the membership grades of the input variables associated with it. The normalization layer (layer 3) normalizes the firing strengths obtained from the product layer. It ensures that the sum of all firing strengths equals one, yielding a proper probability distribution. The defuzzification layer (layer 4) calculates the crisp output depending on the firing strengths of the rules. The output layer (layer 5) generates the ANFIS model's final output. This layer is just a linear combination of the normalized firing strengths and the corresponding rule outputs.

In summary, the product layer determines each rule's firing strength, whereas the fuzzy layer determines the membership grades of the input variables. The final output is achieved by combining firing strengths with the de-fuzzy layer after normalization by the normalized layer. ANFIS can efficiently capture and depict complicated relationships in data because of its multi-layered methodology.

Training of an ANFIS model

The ANFIS process involves defining input variables, normalizing the data, dividing the dataset into training and testing subsets, and defining the initial fuzzy inference system. The ANFIS model is trained, with output calculated using fuzzy rules and parameters updated using learning algorithms. The model is evaluated, with the mean squared error and the root mean squared error. By fine-tuning its parameters, the model is verified using the testing dataset before being used to make predictions on fresh data. Training an ANFIS model involves the following steps as shown in Fig. 5.

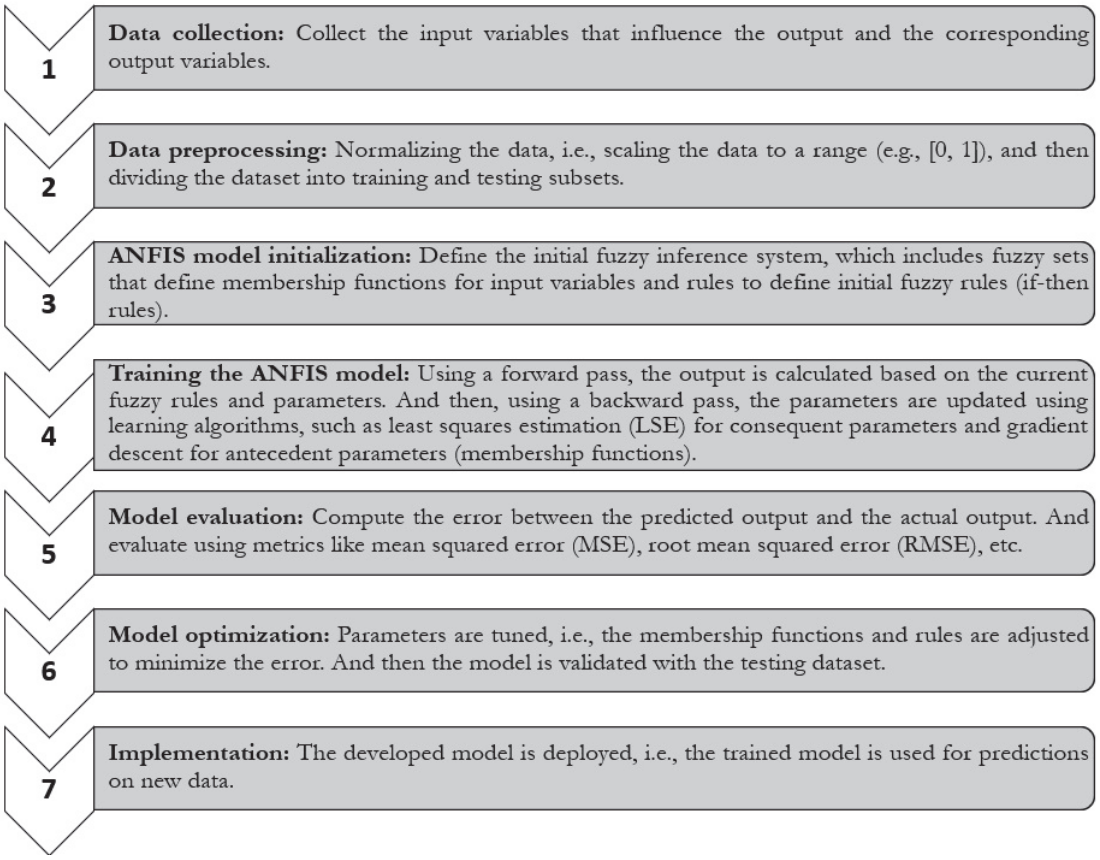


Figure 5: Steps for training of an ANFIS model.

Determining membership functions (MFs), or the form of fuzzy sets, creating fuzzy rules based on domain knowledge, training an ANFIS model with the right algorithms to modify parameters, and lastly evaluating an ANFIS model to periodically evaluate model performance are crucial steps in the training process. These steps are essential in ensuring that the ANFIS model accurately captures the underlying relationships in the data and makes accurate predictions. Properly tuning the membership functions and fuzzy rules is key to improving the model's performance and predictive capabilities. Furthermore, the accuracy and generalization capacity of the ANFIS model may be improved by iteratively training it and fine-tuning its parameters. The predictive capacity and dependability of the model in practical applications may be further increased by routinely updating and improving it considering fresh information or insights. A typical proposed ANFIS architecture in the present study is as shown in Fig. 6.

RESULTS AND DISCUSSION

Shear instability, localized deformation, and intermetallic phases all pose problems for Inconel 718 machining, affecting tool wear, surface integrity, cutting forces, and overall machinability. For sustainable and effective machining, research on the machinability of nickel alloys highlights the importance of choosing the right process parameters and considering elements like tool shape, material, and cooling methods. This section explores the development of an ANN and ANFIS models for cutting force, surface roughness, and tool life during the turning of Inconel 718 with hybrid nanofluid under MQL. Initially, experimental results varying with cutting parameters and the assessment of tool wear, its forms, and

wear mechanisms are discussed. Further, this section evaluates the effectiveness of ANN and ANFIS models in predicting the machining process. By comparing the performance of these models in predicting machining outcomes, this study seeks to provide valuable insights for improving process efficiency and accuracy.

The set of cutting parameters used for turning Inconel 718 and the experimental results are presented in the Tab. 2. Hybrid nanofluid was prepared by mixing aluminum oxide (Al_2O_3) and multi-walled carbon nanotubes (MWCNTs) at constant proportions in vegetable-based palm oil. As depicted in Tab. 2, the study examined the responses at each cutting parameter, including cutting force, surface roughness, and tool life. The cutting force, flank wear, and average surface roughness were measured using a strain gauge-type dynamometer, a Dino-Lite digital microscope, and a Mitutoyo SJ.201 surface roughness tester, respectively. The tool life limits were set at 0.2 mm flank wear or a catastrophic failure in accordance with ISO 3685-1977(E) requirements.

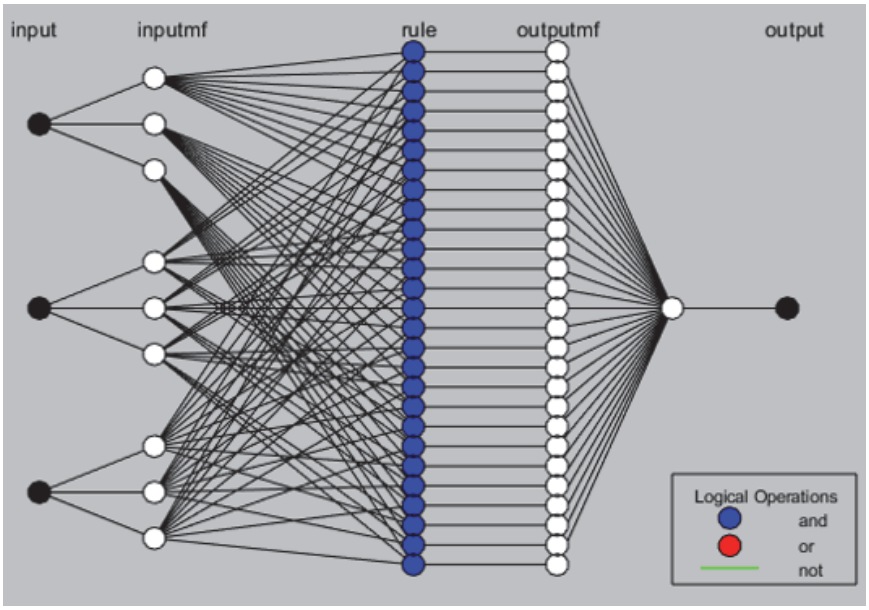


Figure 6: Typical proposed ANFIS architecture.

As cutting speed increased, a reduction in cutting force was seen. The lower cutting forces observed at faster cutting speeds might be the result of the material being softer due to the increased cutting temperature during machining. This softening reduces the material's resistance to deformation, leading to lower cutting forces. Lower cutting forces are further contributed to by the increased cutting speed, which further encourages the creation of a thinner, more stable chip and improves chip evacuation by minimizing tool-workpiece contact. It is observed that as the feed and depth of cut rise, so do the cutting forces. This is because cutting forces are more greatly impacted by the depth of cut, which directly influences the amount of material removed. The thickness and contact area of the chip can also change because of variations in the depth of cut, which further affect the cutting forces. On the other hand, since feed and cutting speed modifications mainly impact the rate of material removal, their impact on cutting forces is comparatively smaller [31-32].

The analysis of worn-out tools at different cutting conditions is discussed with the images captured using scanning electron microscopes as shown in Fig. 7. The photographs depict the tool's rake and flank faces upon turning at the end of the tool wear criterion, which was set at 0.2 mm of flank wear, or in the event of a catastrophic failure. A micrograph of the tools employing hybrid nanofluids at experimental runs R1, R7, R8, R10, and R15 is displayed in Fig. 7. Severe damage to the cutting tool, coating delamination, and pitting on the substrate can be prominently seen; The tool failure can be seen as occurred because of metal adhesion and chipping off the cutting edge due to the breaking of the unstable piled-up adhered material during machining. The pitting on the substrate of the tool and notch wear can be seen at experimental runs 7 and 8 and the catastrophic tool failure and chipping off the cutting edge at higher cutting speeds (experiment index 10) can be seen.

ANN modeling

The study used the neural network toolbox in MATLAB software to train and forecast using artificial neural networks (ANN). By varying the number of hidden layers and neurons inside those layers, several configurations of ANNs were produced. Plotting training and test errors versus the total number of epochs allowed for the evaluation of these networks'

performance. The predefined epoch count ensured that every piece of data was used precisely once per training session. While having too many epochs might cause overfitting and needless use of time and computing resources, setting the number of epochs too low can cause the model to terminate before it converges. With a zero-error target, the study's default parameters for the maximum number of epochs and learning rate were set to 1000 and 0.01, respectively. The model's development time was not limited by computational time. Tab. 3 provides specifics on the ANN parameters and system setups that were employed in the investigation.

Expt. Run	Input variables			Output responses (variables)		
	Cutting speed (V) (m/min)	Feed (f) (mm/rev)	Depth of cut (d) mm	Cutting force (Fc) (N)	Surface roughness (Ra) (μm)	Tool life (TL) (min)
R1	65	0.2	0.8	537	1.49	4.21
R2	45	0.15	0.7	431	1.36	7.7
R3	45	0.25	0.7	617	1.9	5.46
R4	85	0.15	0.7	374	1.15	3.01
R5	85	0.25	0.7	568	1.78	2.14
R6	65	0.2	0.5	277	1.42	4.06
R7	65	0.1	0.5	168	0.93	7.18
R8	65	0.3	0.5	473	1.88	3.29
R9	30	0.2	0.5	343	2.06	14.14
R10	100	0.2	0.5	313	1.15	2.08
R11	45	0.15	0.3	166	1.02	12.91
R12	45	0.25	0.3	248	1.82	9.15
R13	85	0.15	0.3	136	0.94	5.05
R14	85	0.25	0.3	234	1.32	3.58
R15	65	0.2	0.2	107	1.02	8.09

Table 2: Experimental variable matrix of turning process.

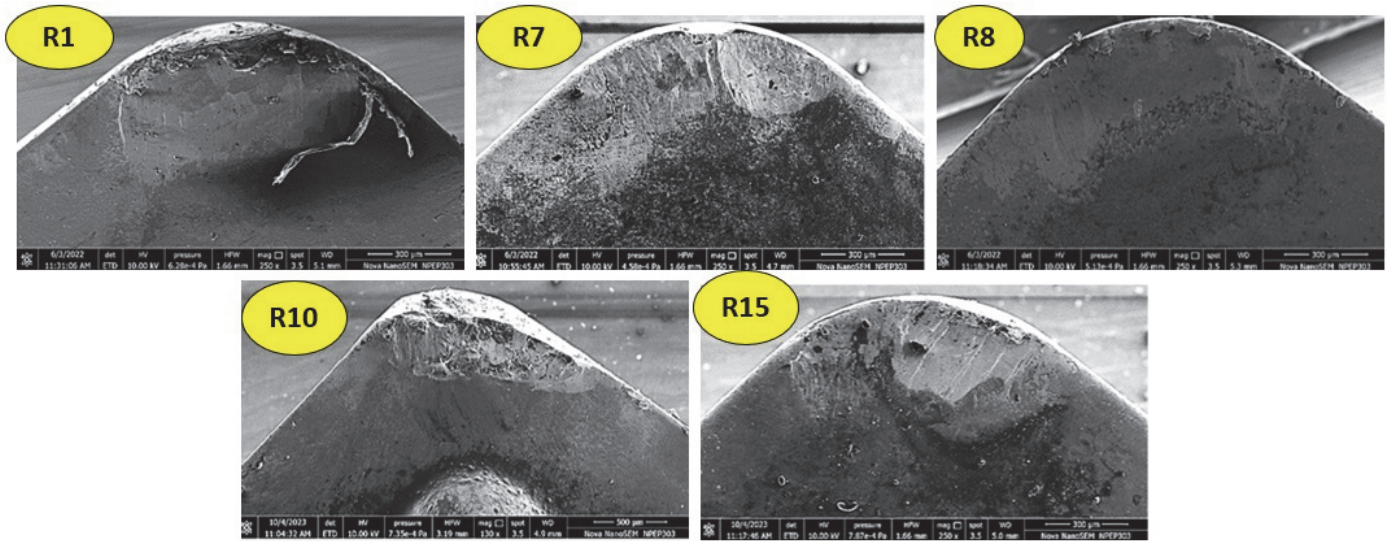


Figure 7: Tool images at experiment run R1, R7, R8, R10, and R15.

An ANN model is created using the MATLAB Toolbox to forecast the cutting force, surface roughness, and tool life depending on varying cutting parameters. The input, hidden, and output layers make up the three tiers of the ANN architecture (Fig. 8). Three neurons make up the input layer, which represents input variables including feed, depth of cut, and cutting speed. One neuron in the output layer is responsible for estimating either tool life, surface roughness, or cutting force. The number of neurons required to use a feed-forward neural network to map an array of numerical inputs to an array of numerical targets is present in the hidden layer(s). The program MATLAB Toolbox Neural Fitting helps with data

selection, network construction and training, and performance evaluation with regression analysis and mean squared error. A two-layer feed-forward network with sigmoid hidden neurons and linear output neurons is chosen to address multi-dimensional issues using reliable data and sufficient hidden layer neurons. The transfer function is the hyperbolic tangent sigmoid transfer function (tansig), as defined by Eqn. (1).

ANN parameter	Characteristics
Type of network	Feed-forward backpropagation
Type of training function	TRAINLM
Type of learning function	LEARNGDM
Performance function	MSE (Mean Squared Error)
Number of hidden layer(s)	1
Number of neurons on hidden layer	10
Number of epochs (max)	1000
Learning rate	0.001
Rate of train data (random)	70%
Rate of test data (random)	30%
Learning algorithm	Levenberg-Marquardt backpropagation technique
Transfer function	Tansig (tangent sigmoid)

Table 3: ANN model parameters and system configuration in the construction and analysis.

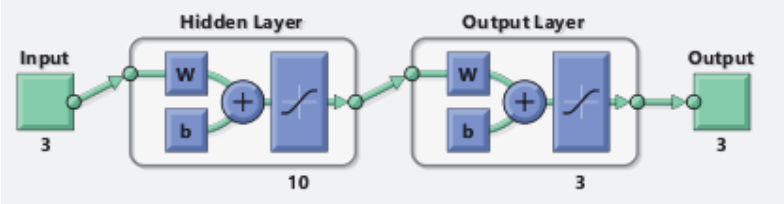


Figure 8: ANN architecture to obtain cutting force, surface roughness, and tool life.

$$f(N) = \tanh(N) = \frac{2}{1+e^{-2N}} - 1 \quad (1)$$

where $f(N)$ is the hyperbolic tangent sigmoid transfer function.

Scaled conjugate gradient algorithms, Bayesian regularization, and the Levenberg-Marquardt technique can all be used to train the network. Despite requiring more memory, the Levenberg-Marquardt technique was chosen for this investigation mainly because it converges more quickly than the other methods. Three types of data samples are used by the neural network for testing, validation, and training. Roughly 15% of the data is reserved for validation, or confirming the expected results, while the remaining 70% is used to train the neural network. The network's generalization is assessed using these validation datasets, and the training process is terminated when the network reaches the optimal level of generalization. An unbiased evaluation of the network's performance both during and after training is provided by the last 15% of the data, which is set aside for testing. The ANN models for cutting force, surface roughness, and tool life were created by analyzing experimental results from eleven different cutting conditions, as presented in Tab. 2. Experimental results from four cutting different conditions other than those used for training were selected for validation, and experimental results from the five different cutting conditions other than those used for training and validation were selected for testing and will be discussed in the testing of an ANN and ANFIS subsection.

The study evaluated the effectiveness of ANN models in forecasting cutting force, surface roughness, and tool life, varying with the number of hidden layers and neurons. One popular technique for assessing the convergence of neural network models is to analyze learning curve graphs, which usually involve a graph of loss (or error) versus epoch. Accuracy is predicted to rise as the training epoch count rises, whereas loss is predicted to fall and finally stabilize. A neural network should finally converge across several epochs. It has been shown that adding additional hidden layers and neurons lengthens the computational time, or the amount of time it takes for an ANN model to converge. The optimal results, with the lowest mean squared error and the highest regression coefficient for the entire data set, were achieved with one hidden layer and

10 neurons. This study concluded that cutting force, surface roughness, and tool life can be predicted accurately using an ANN model with one hidden layer and 10 neurons.

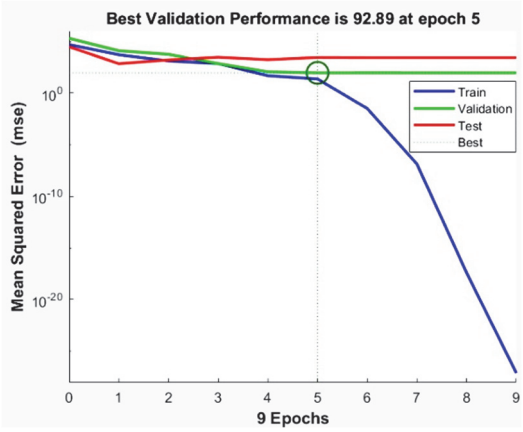


Figure 9: Neural network training performance for cutting force model.

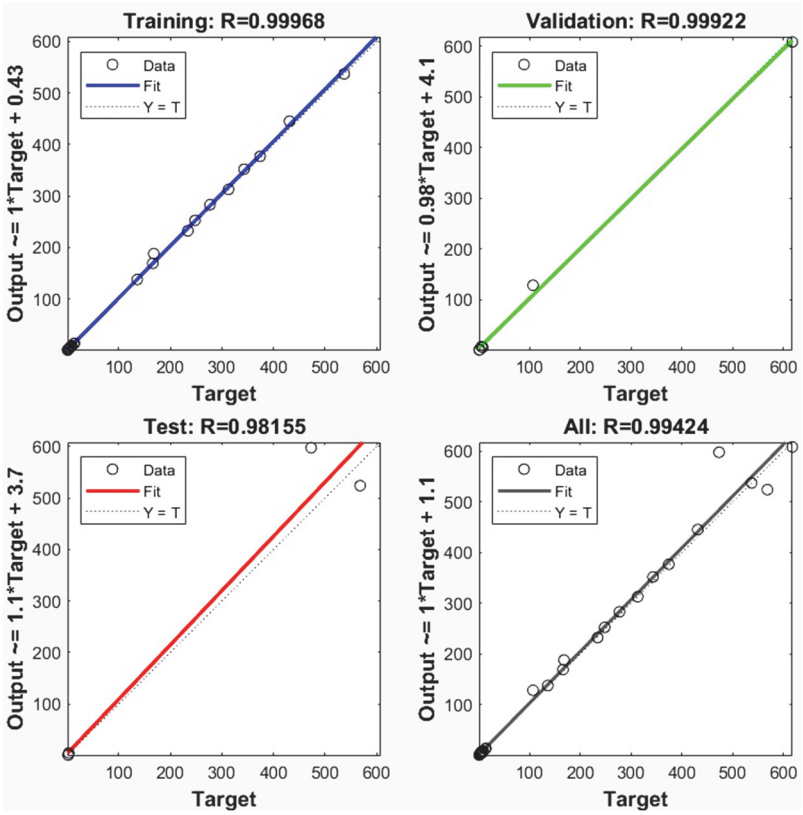


Figure 10: Prediction accuracy of developed neural network model (a) Training, (b) Validation, (c) Test, (d) All data set.

Fig. 9 depicts the training performance of an ANN model using one hidden layer with ten neurons. The optimal validation performance was achieved at epoch 5, with a score of 92.89 with a prediction accuracy of 0.9942. The average squared error between objectives and outputs, or mean squared error, is used to assess the effectiveness of neural network training. Lesser values are preferable. The correlation between outputs (predicted values) and goals (inputs) is measured by R values. Figs. 10(a), (b), (c), and (d) illustrate neural network regression graphs for cutting force with R values discovered close to one during model training, validation, and testing, as well as for the complete data set. Regression coefficients close to one demonstrates that the developed neural network models could be accurately applied to forecast cutting force, surface roughness, and tool life of PVD-coated AlTiN carbide tools when turning Inconel 718 using hybrid MWCNT+Al_2O_3 nanofluid under MQL.

ANFIS modeling

ANFIS is a hybrid intelligent system that combines neural network learning capabilities with fuzzy logic principles, making it an effective tool for modeling complex, nonlinear systems. The ANFIS system utilizes data to enhance performance, with fuzzy rules and membership functions providing a transparent understanding of its decision-making process. Sugeno and Mamdani approaches are widely used in fuzzy logic for designing fuzzy inference systems (FIS), with choices based on application requirements like interpretability vs. computational efficiency and precision. In this work, an ANFIS model is developed using the Sugeno FIS. The Sugeno technique yields greater computing efficiency since it employs weighted averages for the defuzzification process, which makes it simpler and more effective. Moreover, its lower computational cost with more precise outputs, making it ideal for precise control and optimization applications. One especially useful application of the Sugeno approach is to represent nonlinear systems using a set of linear equations.

MATLAB provides a robust toolbox for ANFIS. MATLAB's ANFIS toolbox offers a user-friendly interface for creating and refining fuzzy inference systems, benefiting researchers and practitioners in fuzzy logic and artificial intelligence. The ability to train fuzzy inference systems using a hybrid learning algorithm that blends backpropagation with the least-squares approach is one of ANFIS's primary MATLAB features. Numerous membership functions are supported, including generalized bell, triangular, Gaussian, and trapezoidal. In addition, the visualization tools help with surface plots, rules, and membership functions.

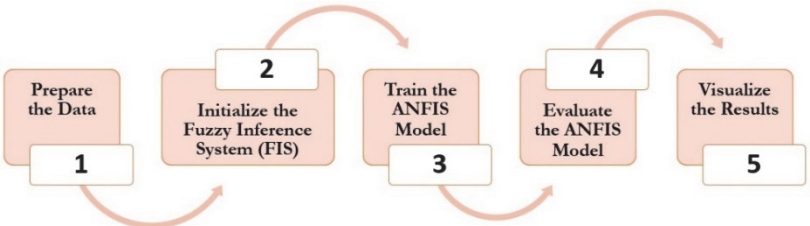


Figure 11: Steps to develop ANFIS model in MATLAB

Fig. 11 depicts the basic steps to develop ANFIS model in MATLAB. The first step in developing an ANFIS model is to identify input and output variables, their ranges, and the availability of data for training and testing of the model. The experimental data listed in Tab. 2 was used for creating the ANFIS model. The data used for testing the developed model is discussed in the coming subsections. As the ANFIS analysis considers only one output value per model, the models for each of the responses are considered separately. Three distinct ANFIS models to predict the responses, namely cutting force ($\boldsymbol{Fc}$), surface roughness ($\boldsymbol{Ra}$), and tool life ($\boldsymbol{TL}$), were developed considering the input parameters, such as cutting speed ($\boldsymbol{V}$), feed ($\boldsymbol{f}$), and depth of cut ($\boldsymbol{d}$).

FIS is generated using grid partitioning option in MATLAB's ANFIS toolbox. Grid partitioning in the context of FIS entails splitting the input space into a grid, with each cell denoting a distinct fuzzy rule. FIS can handle several input variables and their interactions effectively by partitioning the input space into a grid. Grid partitioning enhances the FIS model's interpretability and helps to streamline the rule base. Determining the input space and the range of each input variable, together with the corresponding partition for each range, is the first stage in the grid partitioning process. Next, create membership functions (triangular, trapezoidal, or Gaussian functions) for each input variable to define fuzzy sets (e.g., low, medium, and high). Next, each input variable's fuzzy sets are combined to generate a grid. Every grid cell represents a distinct collection of fuzzy sets derived from the input data. Ultimately, each grid cell is given a set of fuzzy rules. Based on the arrangement of fuzzy sets in that cell, each rule denotes an inference.

The next step in a FIS is to define membership functions (MFs) to specify how each point in the input space is mapped to a degree of membership between 0 and 1. The choice of membership functions significantly affects the performance and behavior of a FIS. The choice of membership function depends on the specific application and the nature of the data. Triangular and trapezoidal MFs are simple to compute and interpret, making them good for applications with linear characteristics. Gaussian and generalized bell MFs are smooth and continuous, making them better for applications requiring smooth transitions. In the present study, ANFIS models are developed using triangular, trapezoidal, Gaussian, and generalized bell membership functions. The output membership functions of this kind of FIS are either constant or linear. In this work, output membership functions are selected as constant.

The next stage is training a FIS using either hybrid or backpropagation optimization methods. In this study, hybrid optimization is opted for with a view to combining different optimization techniques for fine-tuning the parameters of the FIS, such as membership functions and rule weights. Hybrid methods can leverage the strengths of various optimization algorithms to achieve better performance and faster convergence. Optimizing the parameters of these membership

functions can significantly enhance the performance of the FIS. Hyperparameters used to develop the ANFIS models for cutting force (Fc), surface roughness (Ra), and tool life (TL)are as shown in Tab. 4.

ANFIS parameter	Characteristics
Membership functions (MFs)	Triangular (trimf), Trapezoidal (Trapmf), Gaussian (Gaussmf), and Generalized Bell (Gbellmf)
Membership function Output	Constant
Fuzzy Inference System (FIS)	Takagi-Sugeno Fuzzy Model
No of epoch	10
No of fuzzy rules	27
Weight of rules	1
No of nodes	78
No of linear parameters	27
No of non-linear parameters	27
No of training data pairs	15

Table 4: ANFIS model parameters.

FIS for each of the response, namely Fc, Ra, and TL was trained using the training data as depicted in Tab. 2. Error tolerance was set to zero. The data was trained using a fixed number of epochs, i.e., 10. FIS training was stopped once the designated epoch number is reached and minimal training root mean squared error (RMSE) was noted for the trained model. Training error, i.e., root mean squared error (RMSE) for an ANFIS cutting force, surface roughness, and tool life models when using triangular MF are displayed in Fig. 12. The training errors for Fc, Ra, and TL can be seen as 0.000796, 0.000003, and 0.000024, respectively.

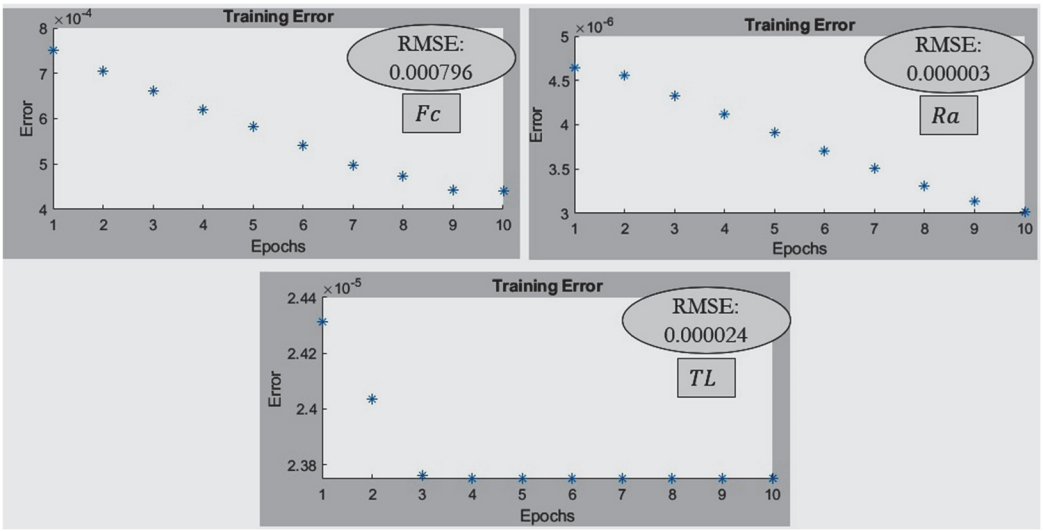


Figure 12: Training error plots for Fc, Ra, and TL.

The ANFIS models of cutting force, surface roughness, and tool life are developed for each of the responses by establishing a FIS, specifying membership functions, and including fuzzy rules. In this investigation, twenty-seven fuzzy rules are employed. Fuzzy rules, typically "If-Then" statements, form the foundation of a fuzzy inference system (FIS), defining the relationship between input and output variables. Each rule is constructed using linguistic variables and fuzzy sets, representing expert knowledge, or learned patterns in the data. MATLAB's fuzzy rules offer a robust method for defining and manipulating fuzzy logic systems. Fig. 13 depicts the structure for ANFIS model developed for cutting force (Fc), surface roughness (Ra), and tool life (TL).

The developed models were tested, and the testing errors for training data and testing data for Fc, Ra, and TL were evaluated. Fig. 14 depicts the mapping of training data and testing data with the FIS output for Fc, Ra, and TL when using

triangular MF. Additionally, Tab. 5 displays the testing error for the training and testing data for each response, along with the corresponding R-squared values, when using the triangular, trapezoidal, Gaussian, and generalized bell membership functions. The lowest testing error with the better R-squared value close to one can be seen for the triangular membership function, followed by the Gaussian (Gaussmf) membership function for the ANFIS cutting force and surface roughness models. However, no significant difference can be seen in the testing error or R-squared value for the different MFs considered in the present study. The highest R-squared value and the lowest testing error can be seen with the triangular membership function. In general, this study found better prediction accuracy for the responses with the triangular membership function. However, this study finds scope for further improvement in the prediction accuracy of the ANFIS surface roughness model, considering hybrid optimization methods for fine-tuning the FIS parameters, such as membership functions and rule weights.

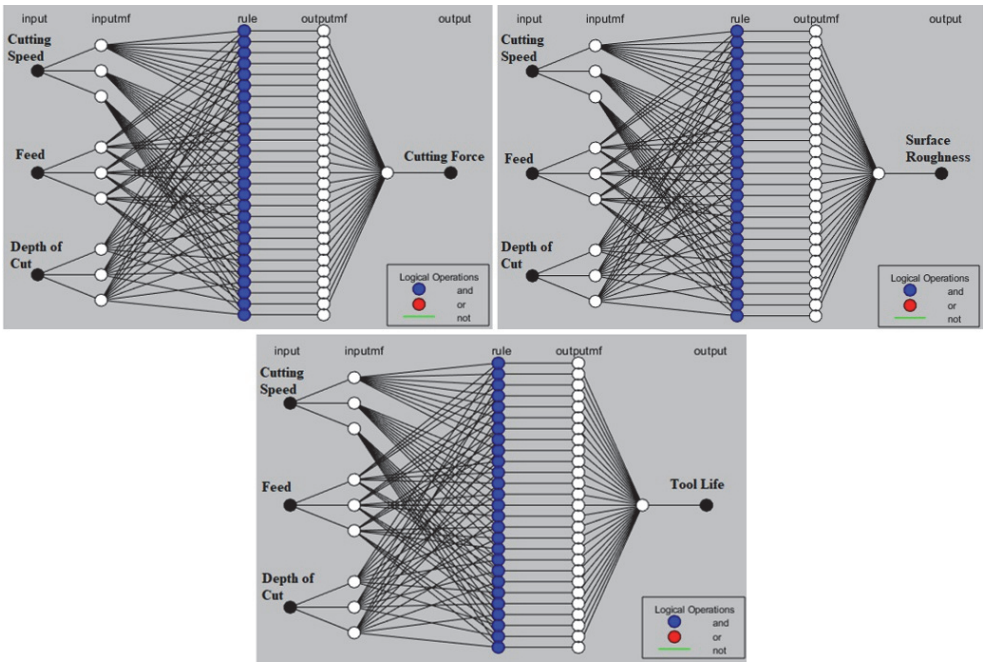


Figure 13: Developed ANFIS models for Fc, Ra, and TL.

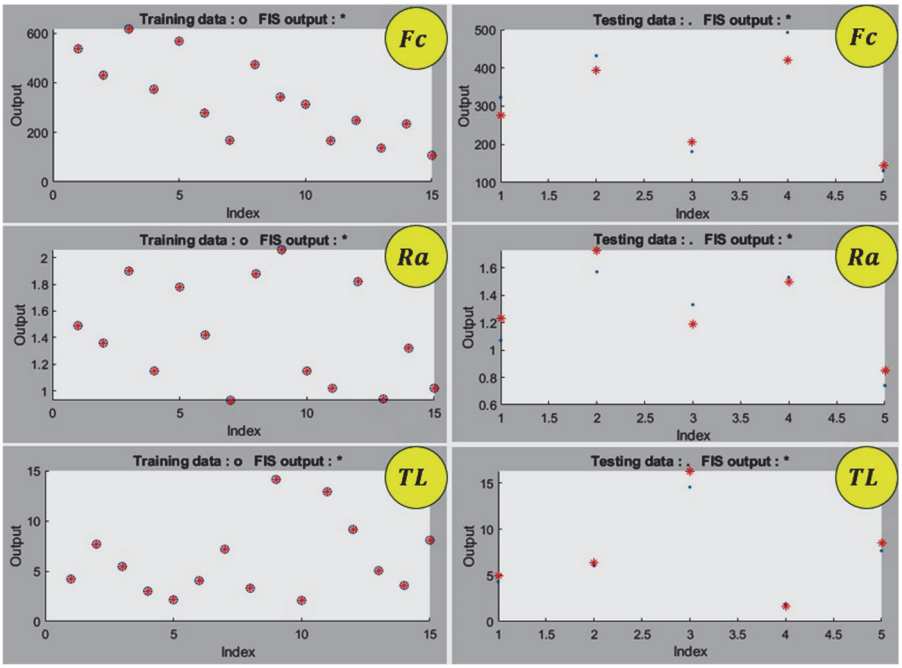


Figure 14: Mapping of training data and testing data with FIS output for Fc, Ra, and TL.

Type of membership function (MF)	ANFIS for cutting force (Fc) model			ANFIS for Surface roughness (Ra) model			ANFIS for tool life model (TL)		
	Average testing error for		R^2 for test data	Average testing error for		R^2 for test data	Average testing error for		R^2 for test data
	Training data	Testing data		Training data	Testing data		Training data	Testing data	
Triangular (Trimf)	0.000751	44.3491	0.9815	0.000003	0.1291	0.8543	0.000024	0.9198	0.9986
Trapezoidal (Trapmf)	0.000741	215.773	0.1594	0.000003	0.6521	0.0052	0.000007	1.2524	0.9841
Gaussian (Gaussmf)	0.000479	45.7215	0.9322	0.000002	0.1744	0.7324	0.000022	1.3734	0.9905
Generalized Bell (Gbellmf)	0.001051	94.787	0.7667	0.000004	0.2667	0.4971	0.000019	1.0856	0.9796

Table 5: Testing error for training and testing data for responses with different MFs

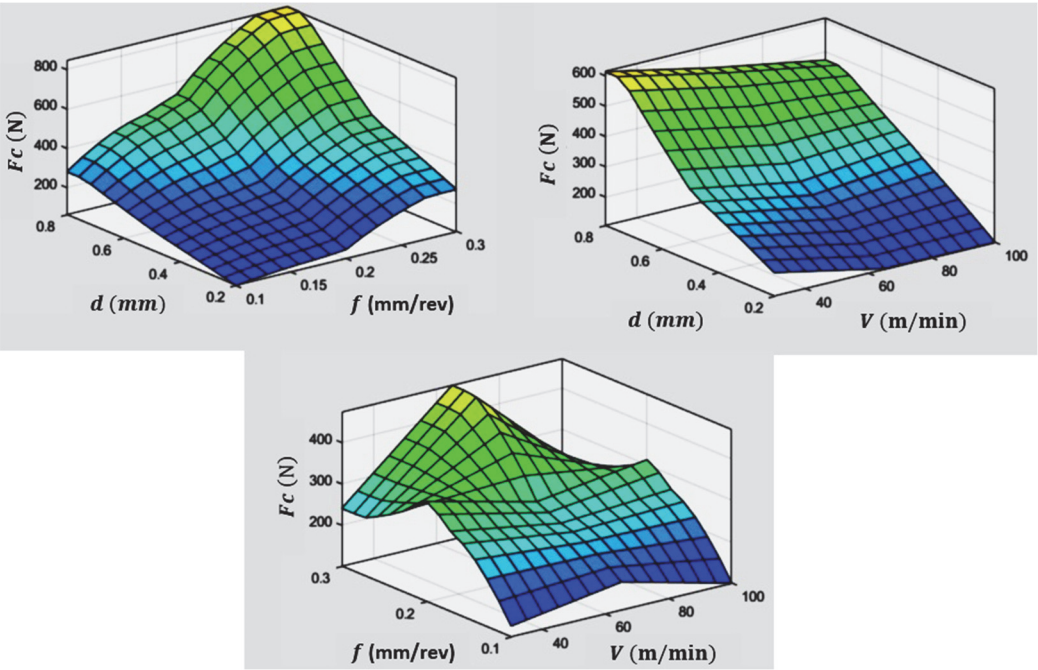


Figure 15: Surface plots for cutting force (*Fc*)

Surface graphs were plotted for cutting force, surface roughness, and tool life, respectively, as illustrated in Figs. 15–17, to better comprehend the simultaneous influence of input parameters over output parameters. Graphs offer a visual representation of input parameters' impact on output parameters, aiding in a comprehensive analysis of process performance, thereby enhancing efficiency and quality. Fig. 15 shows surface graphs of cutting force that vary with two parameters and a constant third parameter at the center level. The center level of cutting parameters, namely cutting speed, feed, and depth of cut, is 65 m/min, 0.2 mm/rev, and 0.5 mm, respectively. The depth of cut and feed are found to have a significant impact on cutting force. The cutting force found to be higher, reaching up to 800 N at higher values of feed and depth of cut of 0.3 mm/rev and 0.8 mm. At a lower cutting speed of 30 m/min and a higher depth of cut of 0.8 mm, the cutting force can be seen reaching up to 600 N. Higher cutting speeds and lower feed and depth of cut values can be used to achieve a lower cutting force.

Figs. 16-17 show surface graphs of surface roughness and tool life that vary with two parameters and a constant third parameter at the center level. As stated above, the center level of cutting parameters, namely cutting speed, feed, and depth of cut, is 65 m/min, 0.2 mm/rev, and 0.5 mm, respectively. The feed and depth of cut are found to have a significant impact on surface roughness. And cutting speed, followed by feed, is found to have a significant impact on tool life. The surface roughness was found to be higher, reaching beyond 2 μm at higher values of feed and depth of cut of 0.3 mm/rev and 0.8 mm. Cutting speed can be seen as having a lower effect on the surface roughness in comparison to feed and depth of cut. On the other hand, from Fig. 17, cutting speed can be seen as having a prominent effect on the tool life in comparison to

feed and depth of cut. The better tool life in the range of 8-10 min was found to be obtained using a lower cutting speed of 30 m/min, and a feed in the range of 0.2-0.25 mm/rev, and a lower depth of cut.
Results suggested that increasing cutting speed and lowering feed and depth of cut values resulted in decreased cutting force and surface roughness. The selection of feed and depth of cut turned out to be more critical in achieving a lower cutting force. Feed significantly impacts surface roughness. The selection of lower cutting parameters led to better tool life estimates.

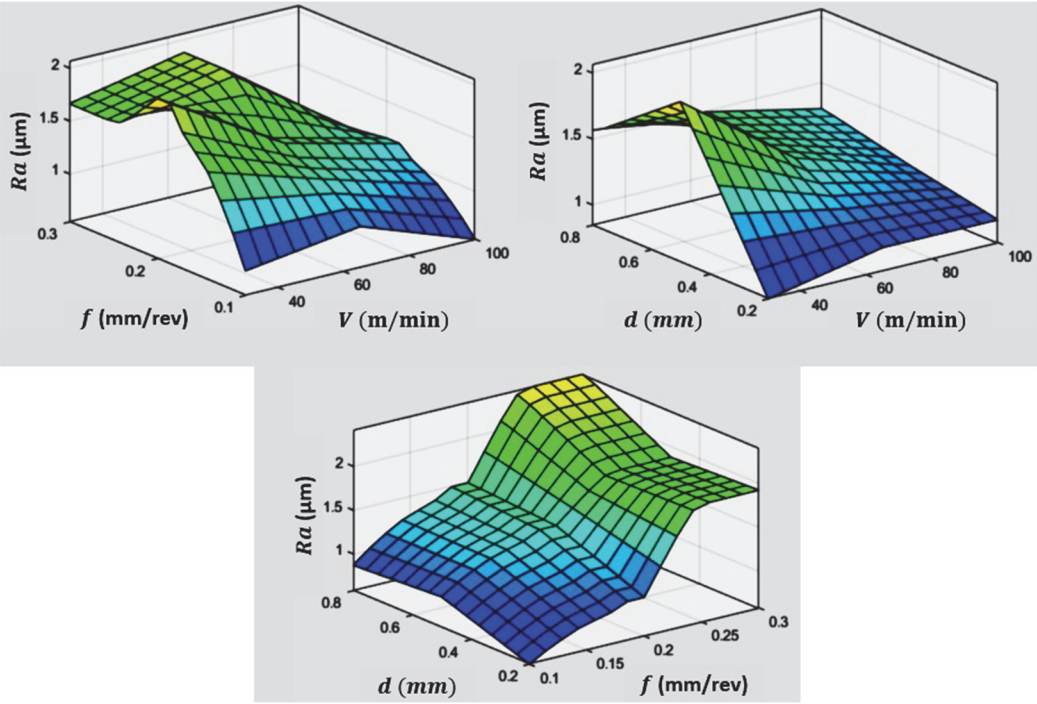


Figure 16: Surface plots for surface roughness (*Ra*).

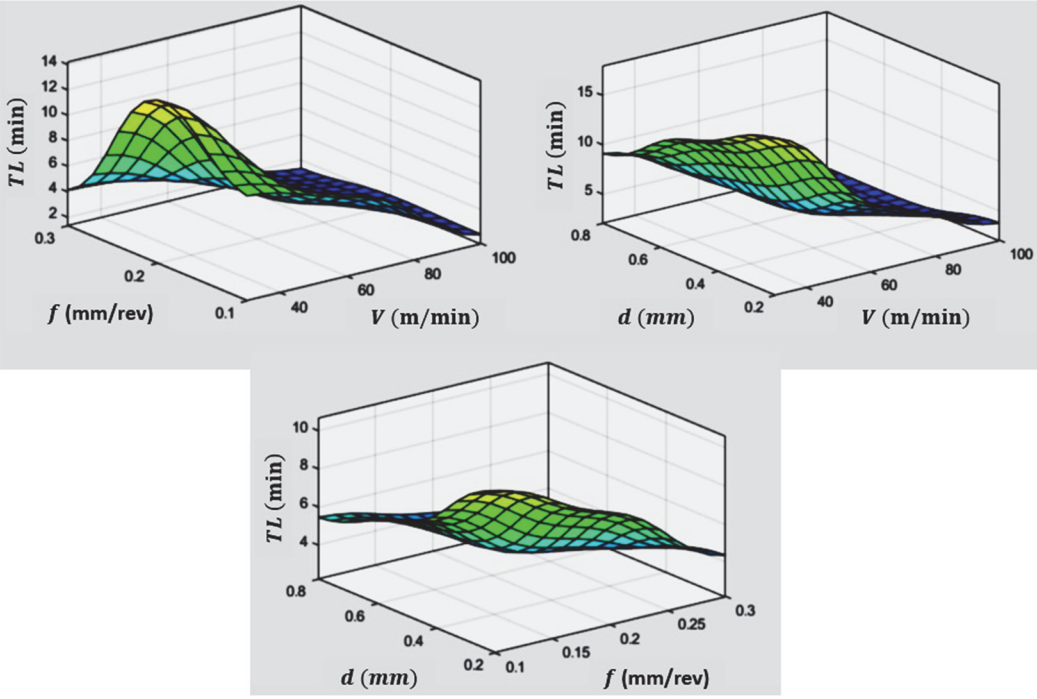


Figure 17: Surface plots for tool life (*TL*)

Validatory experiments

Turning experiments are conducted to validate the ANN and ANFIS predicted responses at process parameters than those were used while training the ANN and ANFIS models. Tab. 6 compares the experimental findings with the anticipated ANN and ANFIS predicted responses. This additional experiment provided a robust assessment of the models' performance across a wider range of input parameters. The experimental values shown for cutting force, surface roughness, and tool life are averages of measurements taken at three repeated trials for a tool, aiming to minimize outliers before analyzing the data. The percentage error in predicted responses by ANN and ANFIS from validatory experimental findings is as shown in Tab. 7. With an average error of less than 15%, there is a good agreement between the experimental findings and the ANN and ANFIS predicted results.

Expt. run	Cutting speed (V) (m/min)	Feed (f) (mm/rev)	Depth of cut (d) (mm)	Cutting force (Fc) (N)			Surface roughness (Ra) (μm)			Tool life (TL) (min)		
				Expt.	ANN	ANFIS	Expt.	ANN	ANFIS	Expt.	ANN	ANFIS
1	55	0.22	0.55	426	464.10	394.01	1.65	1.48	1.73	6.13	6.69	6.36
2	35	0.18	0.3	182	189.83	205.93	1.29	1.38	1.19	14.54	13.08	16.31
3	60	0.1	0.45	178	213.26	144.57	0.84	0.98	0.85	8.66	9.56	8.47

Table 6: Validation experiments.

Expt. run	Cutting speed (V) (m/min)	Feed (f) (mm/rev)	Depth of cut (d) (mm)	% absolute error in cutting force (Fc) (N)		% absolute error in surface roughness (Ra) (μm)		% absolute error in tool life (TL) (min)	
				ANN	ANFIS	ANN	ANFIS	ANN	ANFIS
1	55	0.22	0.55	8.94	7.51	10.07	4.89	9.15	3.81
2	35	0.18	0.3	4.30	13.15	6.78	7.71	10.07	12.15
3	60	0.1	0.45	19.81	18.78	16.07	0.81	10.35	2.21
% absolute average error				11.02	13.15	10.97	4.47	9.86	6.05

Table 7: % absolute error in ANN and ANFIS predicted results from the validatory experimental results.

The results of the validatory experiments showed that both models were able to accurately predict outcomes within the specified range, further validating their predictive capabilities. This study finds that the soft computing techniques such as ANN and ANFIS could be reliably used to model turning of Inconel 718 under NFMQL cutting conditions. In comparison to ANN, the ANFIS technique proved to be more accurate in forecasting surface roughness and tool life within a restricted range of process parameters and limited experimental findings. The accuracy of cutting force prediction with both models did not, however, differ much.

Overall, the results demonstrate the effectiveness of both models in accurately predicting outcomes across various input parameters. This comprehensive analysis enhances our understanding of the predictive capabilities of these models in real-world applications. However, by using hybrid optimization techniques to optimize membership function parameters, this study recommends more research on the modeling of Inconel 718 machining using ANFIS. Further exploration of different machining parameters and their effects on the ANFIS model could provide valuable insights for improving the accuracy and efficiency of Inconel 718 machining processes. Additionally, comparing the performance of ANFIS with other machine learning techniques in this context may offer a comprehensive understanding of its capabilities and limitations.

CONCLUSIONS

Nickel alloys' low heat conductivity and poor machinability cause severe damage to cutting tools during machining, leading to increased manufacturing costs. It is essential to develop models to predict machining parameters to increase production efficiency, save costs, and ensure quality. Soft computing techniques, with their self-learning

capabilities, fuzzy principles, and evolutionary computational philosophy, are being increasingly utilized in modeling such complex machining processes.

Utilizing well-designed NFMQL machining trials, this study builds models for adaptive neuro-fuzzy inference systems (ANFIS) and artificial neural networks (ANN) to forecast the machining performance of Inconel 718 turning. The study used the neural network toolbox and ANFIS toolbox in MATLAB software to train and forecast the cutting force, surface roughness, and tool life. From the current study, the following conclusions could be drawn:

- The selection of feed and depth of cut turned out to be more critical in achieving a lower cutting force. Feed significantly impacts surface roughness. The selection of lower cutting parameters led to better tool life estimates.
- The highest tool life of 8-10 min was obtained at a lower cutting speed of 30 m/min, a feed in the range of 0.2-0.25 mm/rev, and a lower depth of cut. The surface roughness was found to reach beyond 2 μm at higher values of feed and depth of cut of 0.3 mm/rev and 0.8 mm. The cutting force found to be higher, reaching up to 800 N at higher values of feed and depth of cut of 0.3 mm/rev and 0.8 mm.
- The ANN model developed to predict cutting force, surface roughness, and tool life using one hidden layer with ten neurons showed optimal validation performance at epoch 5, with a score of 92.89 and a prediction accuracy of 0.9942.
- The ANFIS models were developed using triangular, trapezoidal, Gaussian, and generalized bell membership functions. ANFIS models developed for cutting force and surface roughness showed better prediction accuracy with the triangular membership function, followed by the Gaussian (Gaussmf) membership function. However, better prediction accuracy of the ANFIS tool life model was observed with almost all the different membership functions considered in the present study.
- The ANFIS models developed using the triangular membership function showed minimum testing error with better R-squared values of 0.9815, 0.8543, and 0.9986 for cutting force, surface roughness, and tool life, respectively.
- This study found a good agreement between the experimental findings and the ANN and ANFIS predicted results, with an average error of less than 15%. However, ANFIS outperforms ANN in terms of accuracy, with prediction errors of 4.47% and 10.97% for surface roughness, and 6.05% and 9.86% for tool life, respectively. However, the accuracy of cutting force prediction was slightly higher with the ANN, with prediction errors of 11.02% against 13.15% for ANFIS. This shows that ANFIS could be a better option for forecasting the machining performance while turning Inconel 718.
- This study suggests further investigation into ANFIS modeling, with a focus on membership function parameter optimization through hybrid optimization techniques.

REFERENCES

[1] Kulkarni, P. and Chinchanikar, S. (2023). A Review on Machining of Nickel-Based Superalloys Using Nanofluids Under Minimum Quantity Lubrication (NFMQL). J. Inst. Eng. India Ser. C, 104(1), pp. 183-199. DOI: 10.1007/s40032-022-00905-w

[2] Saleem, M.Q. and Mehmood, A. (2022). Eco-friendly precision turning of superalloy Inconel 718 using MQL based vegetable oils: tool wear and surface integrity evaluation. J. Manuf. Processes, 73, pp. 112-127. DOI: 10.1016/j.jmapro.2021.10.059

[3] Airao, J., Khanna, N. and Nirala, C.K. (2022). Tool wear reduction in machining Inconel 718 by using novel sustainable cryo-lubrication techniques. Tribol. Int., 175, p. 107813. DOI: 10.1016/j.triboint.2022.107813

[4] Pandey, K. and Datta, S. (2021). Machinability study of Inconel 825 superalloy under nanofluid MQL: Application of sunflower oil as a base cutting fluid with MWCNTs and nano-Al_2O_3 as additives. In Sustainable Manufacturing and Design (pp. 151-197). Woodhead Publishing. DOI: 10.1016/B978-0-12-822124-2.00008-1

[5] Prabhu, S. and Vinayagam, B.K. (2015). Adaptive neuro fuzzy inference system modelling of multi-objective optimisation of electrical discharge machining process using single-wall carbon nanotubes. Aust. J. Mech. Eng., 13(2), pp. 97-117. DOI: 10.7158/M13-074.2015.13.2

[6] Hegab, H., Salem, A., Rahnamayan, S. and Kishawy, H.A. (2021). Analysis, modeling, and multi-objective optimization of machining Inconel 718 with nano-additives based minimum quantity coolant. Appl. Soft Comput., 108, p. 107416. DOI: 10.1016/j.asoc.2021.107416

[7] Chinchanikar, S. and Gadge, M. (2024). Investigations on tool wear behavior in turning AISI 304 stainless steel: An empirical and neural network modeling approach. Frattura ed Integrità Strutturale, 18(67), 176-191. DOI:10.3221/IGF-ESIS.67.13

[8] Elsheikh, A.H., Muthuramalingam, T., Shanmugan, S., Ibrahim, A.M.M., Ramesh, B., Khoshaim, A.B. and Sathyamurthy, R. (2021). Fine-tuned artificial intelligence model using pigeon optimizer for prediction of residual stresses during turning of Inconel 718. J. Mater. Res. Technol., 15, pp. 3622-3634. DOI: 10.1016/j.jmrt.2021.09.119
[9] Niyas, S., Jappes, J.W., Adamkhan, M., and Brintha, N.C. (2022). An effective approach to predict the minimum tool wear of machining process of Inconel 718. Mater. Today Proc., 60, pp. 1819-1834. DOI: 10.1016/j.matpr.2021.12.501
[10] Vijayaraghavan, V., Garg, A., Gao, L., Vijayaraghavan, R. and Lu, G. (2016). A finite element based data analytics approach for modeling turning process of Inconel 718 alloys. J. Cleaner Prod., 137, pp. 1619-1627. DOI: 10.1016/j.jclepro.2016.04.010
[11] Segreto, T., D'Addona, D. and Teti, R. (2020). Tool wear estimation in turning of Inconel 718 based on wavelet sensor signal analysis and machine learning paradigms. Prod. Eng., 14, pp. 693-705. DOI: 10.1007/s11740-020-00989-2
[12] Dinaharan, I., Palanivel, R., Murugan, N. and Laubscher, R.F. (2022). Application of artificial neural network in predicting the wear rate of copper surface composites produced using friction stir processing. Aust. J. Mech. Eng., 20(4), pp. 1079-1090. DOI: 10.1080/14484846.2020.1769803
[13] Gaikwad, V.S. and Chinchanikar, S.S. (2022). Adaptive Neuro Fuzzy Inference System to Predict the Mechanical Properties of Friction Stir Welded AA7075-T651 Joints. Jordan Journal of Mechanical & Industrial Engineering, 16(3).
[14] Mr, G.K. and Gupta, A.K. (2023). Mathematical modeling to estimate machining time during milling of Inconel 718 workpiece using ANN. Mater. Today Proc., 78, pp. 546-554. DOI: 10.1016/j.matpr.2022.11.314
[15] Vishnu, P., Kumar, N.S. and Manohar, M. (2018). Performance prediction of electric discharge machining of Inconel-718 using artificial neural network. Mater. Today Proc., 5(2), pp. 3770-3780. DOI: 10.1016/j.matpr.2017.11.630
[16] Kaya, B., Oysu, C. and Ertunc, H.M. (2011). Force-torque based on-line tool wear estimation system for CNC milling of Inconel 718 using neural networks. Adv. Eng. Software, 42(3), pp. 76-84. DOI: 10.1016/j.advengsoft.2010.12.002
[17] Sharma, D., Bhowmick, A. and Goyal, A. (2022). Enhancing EDM performance characteristics of Inconel 625 superalloy using response surface methodology and ANFIS integrated approach. CIRP J. Manuf. Sci. Technol., 37, pp. 155-173. DOI: 10.1016/j.cirpj.2022.01.005
[18] Bhandare, A. S., & Dabade, U. A. (2023). Modeling of Dry EDM process parameters during machining of Inconel 718 using artificial neural network. Mater. Today Proc. DOI: 10.1016/j.matpr.2023.08.293
[19] Bhowmick, S., Mondal, R., Sarkar, S., Biswas, N., De, J. and Majumdar, G. (2023). Parametric optimization and prediction of MRR and surface roughness of titanium mixed EDM for Inconel 718 using RSM and fuzzy logic. CIRP J. Manuf. Sci. Technol., 40, pp. 10-28. DOI: 10.1016/j.cirpj.2022.11.002
[20] Ozkavak, H.V., Sofu, M.M., Duman, B. and Bacak, S. (2021). Estimating surface roughness for different EDM processing parameters on Inconel 718 using GEP and ANN. CIRP J. Manuf. Sci. Technol., 33, pp. 306-314. DOI: 10.1016/j.cirpj.2021.04.007
[21] Paturi, U.M.R., Devarasetti, H., Reddy, N.S., Kotkunde, N. and Patle, B.K. (2021). Modeling of surface roughness in wire electrical discharge machining of Inconel 718 using artificial neural network. Mater. Today Proc., 38, pp. 3142-3148. DOI: 10.1016/j.matpr.2020.09.503
[22] Natarajan, C., Muthu, S. and Karuppuswamy, P. (2012). Investigation of cutting parameters of surface roughness for brass using artificial neural networks in computer numerical control turning. Aust. J. Mech. Eng., 9(1), pp. 35-45. DOI: 10.1080/14484846.2012.11464616
[23] Abhilash, P.M. and Chakradhar, D. (2020). Prediction and analysis of process failures by ANN classification during wire-EDM of Inconel 718. Adv. Manuf., 8, pp. 519-536. DOI: 10.1007/s40436-020-00327-w
[24] Yusoff, Y., Mohd Zain, A., Sharif, S., Sallehuddin, R. and Ngadiman, M.S. (2018). Potential ANN prediction model for multiperformances WEDM on Inconel 718. Neural Comput. Appl., 30, pp. 2113-2127. DOI: 10.1007/s00521-016-2796-4
[25] Hewidy, M. and Salem, O. (2023). Integrating experimental modeling techniques with the Pareto search algorithm for multiobjective optimization in the WEDM of Inconel 718. Int. J. Adv. Manuf. Technol., 129(1-2), pp. 299-319. DOI: 10.1007/s00170-023-12200-8
[26] Sen, B., Mia, M., Mandal, U. K. and Mondal, S. P. (2019). GEP-and ANN-based tool wear monitoring: a virtually sensing predictive platform for MQL-assisted milling of Inconel 690. Int. J. Adv. Manuf. Technol., 105, pp. 395-410. DOI: 10.1007/s00170-019-04187-y
[27] Kumar, A. and Pradhan, M.K. (2023). An ANFIS modelling and genetic algorithm-based optimization of through-hole electrical discharge drilling of Inconel-825 alloy. J. Mater. Res., 38(2), pp. 312-327. DOI: 10.1557/s43578-022-00728-6
[28] Premnath, A.A., Alwarsamy, T. and Sugapriya, K. (2014). A comparative analysis of tool wear prediction using response surface methodology and artificial neural networks. Aust. J. Mech. Eng., 12(1), pp. 38-48. DOI: 10.7158/M12-075.2014.12.1

[29] Babu, K.N., Karthikeyan, R. and Punitha, A. (2019). An integrated ANN–PSO approach to optimize the material removal rate and surface roughness of wire cut EDM on INCONEL 750. Mater. Today Proc., 19, pp. 501-505. DOI: 10.1016/j.matpr.2019.07.643
[30] Sada, S.O. and Ikpeseni, S.C. (2021). Evaluation of ANN and ANFIS modeling ability in the prediction of AISI 1050 steel machining performance. Heliyon, 7(2). DOI: 10.1016/j.heliyon.2021.e06136
[31] Kulkarni, P. and Chinchanikar, S. (2024). Machining effects and multi-objective optimization in Inconel 718 turning with unitary and hybrid nanofluids under MQL. Frattura ed Integrità Strutturale, (68). DOI: DOI:10.3221/IGF-ESIS.68.15
[32] Kulkarni, P. and Chinchanikar, S. (2024). Machinability of Inconel 718 using unitary and hybrid nanofluids under minimum quantity lubrication. Adv. Mater. Process. Technol., pp. 1-29. DOI: 10.1080/10426914.2020.1711924
[33] Solyali, D. (2020). A comparative analysis of machine learning approaches for short-/longterm electricity load forecasting in Cyprus. Sustainability, 12(9), p. 3612. DOI: 10.3390/su12093612
[34] Wen, L., Ye, X. and Gao, L. (2020). A new automatic machine learning based hyperparameter optimization for workpiece quality prediction. Meas. Control, 53(7-8), pp. 1088-1098. DOI: 10.1177/002029402093234

NOMENCLATURE

Al_2O_3:	Aluminum oxide
ANN:	Artificial neural networks
ANFIS:	Adaptive neuro-fuzzy inference system
BBD:	Box-Behnken designs
BPNN:	Back-propagation neural network
EDM:	Electric discharge machining
Fc :	Cutting force
FIS:	Fuzzy inference system
FL:	Fuzzy logic
Gaussmf:	Gaussian membership function
Gbellmf:	Generalized Bell membership function
GEP:	Gene expression programming
GP:	Genetic programming
LM:	Levenberg-Marquardt
LSE:	Least squares estimation
MFs:	Membership functions
MQL:	Minimum quantity lubrication
ML:	Machine Learning
MLP:	Multilayer perceptron
MRR:	Material Removal Rate
MSE:	Mean Squared Error
MWCNTs:	Multi-Walled Carbon Nanotubes
POA:	Pigeon Optimization Algorithm
PSO:	Particle Swarm Optimization
Ra :	Surface roughness
RMSE:	Root Mean Squared Error
RSM:	Response Surface Methodology
TL :	Tool Life
Trapmf:	Trapezoidal membership function
Trimf:	Triangular membership function