Predictive modeling of turning operations under different cooling/ lubricating conditions for sustainable manufacturing with machine learning techniques

Djordje Cica, Branislav Sredanovic and Sasa Tesic
*Faculty of Mechanical Engineering, University of Banja Luka,
Bosnia and Herzegovina, and*
Davorin Kramar
Faculty of Mechanical Engineering, University of Ljubljana, Slovenia

Abstract

Sustainable manufacturing is one of the most important and most challenging issues in present industrial scenario. With the intention of diminish negative effects associated with cutting fluids, the machining industries are continuously developing technologies and systems for cooling/lubricating of the cutting zone while maintaining machining efficiency. In the present study, three regression based machine learning techniques, namely, polynomial regression (PR), support vector regression (SVR) and Gaussian process regression (GPR) were developed to predict machining force, cutting power and cutting pressure in the turning of AISI 1045. In the development of predictive models, machining parameters of cutting speed, depth of cut and feed rate were considered as control factors. Since cooling/lubricating techniques significantly affects the machining performance, prediction model development of quality characteristics was performed under minimum quantity lubrication (MQL) and high-pressure coolant (HPC) cutting conditions. The prediction accuracy of developed models was evaluated by statistical error analyzing methods. Results of regressions based machine learning techniques were also compared with probably one of the most frequently used machine learning method, namely artificial neural networks (ANN). Finally, a metaheuristic approach based on a neural network algorithm was utilized to perform an efficient multi-objective optimization of process parameters for both cutting environment.

Keywords Machine learning, Sustainable machining, Machining force, Cutting power, Cutting pressure

Paper type Research paper

1. Introduction

Cutting fluids are traditionally used in metal cutting operations to improve the tool life, surface quality as well as entire machining process productivity. However, cutting fluids have negative effects on the human health and environment due to presence of potentially harmful chemicals [1]. In addition, the use of cutting fluids represents a considerable amount of total manufacturing costs [2]. Weinert et al. [3] demonstrates that the estimated cost of the cutting fluids is around 7 to 17% of the aggregate machining costs. Nowadays, conventional flood cooling is the most common cooling/lubricating technique used to improve machining performance. However, high cutting fluid consumption as well as power consumption, poor

Declaration of competing interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

cooling/lubrication capability, excessive waste management and problems related to human health and environmental issues are some of the most important disadvantages of this method.

The growth of global production and consequently the increase of cutting fluids application caused intensive research concerning economic and environmental aspects of systems for cooling/lubricating the cutting zone. Thus, recently several cooling/lubrication techniques were developed in order to achieve sustainable manufacturing by reducing or eliminating of cutting fluids. Currently, the most widely used cooling/lubricating techniques with low negative effect on the environment and human operator's health are: dry cutting, cryogenic cooling, minimum quantity lubrication (MQL), high-pressure coolant (HPC), or utilization as a cooling/lubricating fluid the biodegradable oils. Apart from being more economical and environmentally sustainable, these new technologies are also more efficient as compared with traditional flood cooling. Considerable improvements have been reported in surface quality, tool life, productivity, total costs, etc.

Due to fact that manufacturing operations consume significant amounts of energy, competitive manufacturing industries require energy efficient machining processes to minimize negative effect on the environment as well as to reduce costs. For instance, according to Zhou et al. [4] about 20% of overall consumed annual worldwide energy is used in manufacturing. In addition to eco-friendly and clean machining process, waste management and reduction, conserving energy represents one of the important indicators of sustainable production. Therefore, industries are searching for alternative manufacturing methods in which less energy is consumed [5,6].

Machining processes are characterized by the presence of a large number of highly correlated parameters. Due to the high complexity and nonlinearity of metal cutting phenomena, it is very complicated or even impossible to formulate an adequate analytical model using traditional methods based on the process's physics. Nowadays, the trends are towards predictive modeling of these processes using machine learning methods. Over the last few decades, artificial neural networks (ANN) and multiple regression have been successfully implemented in the prediction of various quality characteristics in turning under dry and conventional coolant supply, such as surface roughness [7–11], cutting force [12–15], tool wear [16–19], specific cutting force [7,12], cutting power [10,15,19] and cutting temperature [20–22].

Nowadays, academic research as well as industrial efforts are being directed towards the challenge of elimination or, at least, minimization of cutting fluids use, while preserving process efficiency. However, there are significantly less studies dealing with application of the machine learning techniques in estimating machining responses in environmentally friendly processes, such as MQL and HPC-assisted machining environments. Cica et al. [23] formulate two models, namely, ANFIS and ANN-based, for prediction of main cutting force, feed force and passive force. Three different cooling and lubricating conditions (flood, MQL and HPC), depth of cut, feed rate, cutting speed were used as the variables for cutting force components modeling. They concluded that both models can be used effectively to predict the cutting force components in turning operations. Mia et al. [24] proposed ANN and support vector regression (SVR) models for surface roughness, cutting temperature and chip coefficient prediction when turning of Ti-6Al-4V alloy. The cutting speed, feed rate, cutting conditions (dry and HPC), and turning forces were the input variables. They demonstrated that both methods can be used successfully in predicting the machining responses. Cica et al. [25] modeled tool life and surface roughness in hard turning of the 100Cr6 steel under HPC environment using ANN and ANFIS. Experimental machining data such are feed, cutting speed and time were used in this study to train and evaluate both models. The estimation results obtained by both models are compared with experimental results and very good agreement is observed. Mia and Dhar [26] developed ANN-based predictive models of surface

roughness in hard turning of EN 24T steel under dry and high pressure coolant jet machining environments. Apart from cutting conditions, cutting speed, feed rate and material hardness were used as the input variables. Different ANN architectures and several training methods were employed to determine the best predictive model. Mia and Dhar [27] formulate two predictive models of surface roughness, namely support vector regression and response surface methodology (RSM) in turning of AISI 1060 steel under dry and HPC conditions. The cutting speed, feed rate and material hardness were considered as input variables for model formulation. The results indicated that both methods can be utilized to predict the roughness value in dry turning, while the support vector regression model is preferable over RSM in HPC assisted turning. Kamruzzaman et al. [28] formed ANN model of cutting temperature in terms of cutting speed, feed rate, depth of cut, workpiece materials (C-60, 17CrNiMo4 and 42CrMo4) and cutting environments (dry, wet and HPC), and found 97.3% accuracy. Mia et al. [29] formulated ANN-based predictive model of surface roughness model for MQL-assisted hard turning, wherein cutting speed, feed rate and MQL flow rate were inputs. Their results indicate that ANN model is capable of preserving 97.5% accuracy. Mia et al. [30] utilized the SVR for the prediction of average surface roughness parameter with respect to spindle speed, feed rate, depth of cut and time gap between pulsing in MQL assisted turning of high hardness steel. Their results show that the developed model is able to predict the output responses with 95.04% accuracy. Abbas et al. [31] developed the regression models for the surface roughness and power consumption under dry, wet and nanofluid MQL-assisted turning of AISI 1045. Nouioua et al. [32] utilized response surface methodology and ANN technique to search for optimal prediction of predicting surface roughness and cutting force in turning of X210Cr12 steel according to cutting speed, feed rate and cutting depth under dry, wet and MQL machining conditions. ANN were found to be better than the response surface methodology model in the prediction of cutting parameters.

Machine learning techniques have been extensively utilized in the prediction of different machining responses in turning. However, presented models in literature mainly dealt with dry and wet cutting. Furthermore, a very few utilizable information is provided regarding the prediction of the cutting energy and as well as the cutting pressure under different cooling/lubricating conditions. This study presents a prediction models development of machining force, cutting energy and cutting pressure in turning using three regression based machine learning techniques (polynomial regression, support vector machine and Gaussian process regression) as well as artificial neural networks. Contrary to other presented works, here the estimation of selected machining responses was carried out for different cooling/lubricating conditions. In particular, the study covered the MQL and HPC machining conditions. Selected machine learning techniques are moreover used for comparative assessment of machining responses in order to determine the best approach according to model accuracy and capability. In addition, multi-objective optimization problem was also carried out.

2. Experimental details
Straight turning of AISI 1045 (C45E) steel supplied as bars 120 mm in diameter and 300 mm long in a lathe Boehringer that develops a spindle power of 8 kW have been carried out by standard carbide inserts SNMG 1204 08 NMX. In this study, focus is placed on the application of various cooling/lubricating techniques in machining. Therefore, the experiments are conducted under different machining environments, namely MQL and HPC. The MQL and HPC systems were attached in the experimental setup during the machining trials.

For MQL assisted turning, cutting fluid was supplied to spray gun at the rate of 30 ml/h, which is mixed with compressed air (3 bar) in the mixing chamber of spray gun. Then the mixture of air and cutting fluid is supplied at the cutting zone by spray gun nozzle located

30 mm away from tool tip, at an angles of 90° and 30°, from the cutting edge and clearance face, respectively.

During HPC assisted turning, the cutting fluid was supplied at a constant pressure of 50 bar and flow rate of 2 l/min through 0.4 mm nozzle (diameter) normal to the cutting edge at a low angle (about 5–6°) with the cutting tool rake face. The nozzle was positioned 30 mm away from the tool tip with purpose to achieve fairly close to the tool-chip contact zone as well as to reduce the interference of the nozzle with the flowing chips.

Apart from different machining environments, three cutting parameters, that are cutting speed (v), depth of cut (a) and feed rate (f), were also selected as control factors. Referring to Table 1, the three levels of cutting speed, three levels of depth of cut and four levels of feed rate generate 36 ($3 \times 3 \times 4$) number of experimental runs for each of machining environment. The ranges of these parameters were selected based on the recommendations of the cutting tool manufacturer and in accordance with previous studies. Moreover, the parameter ranges were also extended in order to achieve higher productivity and to investigate machining responses in different machining environments.

The three components of the cutting force, namely, main cutting force (F_c), feed force (F_f) and passive force (F_p), were measured using the Kistler dynamometer type 9259A. The measurement chain further includes a charge amplifier (Kistler 5001), spectrum analyzer (HP3567A) and personal computer for data acquisition and analysis.

The machining force (F_R), cutting power (P_c) and cutting pressure (K_s) are computed from the following equations:

$$F_R = \sqrt{F_c^2 + F_f^2 + F_p^2} \tag{1}$$

$$P_c = F_c \cdot v \tag{2}$$

$$K_s = \frac{F_c}{a \cdot f} \tag{3}$$

The obtained experimental data were divided into two data sets, namely training data set for model development (75% of the entire data set) and test data set for model validation (25% of the entire data set). Thus, 27 sets of randomly selected experimental trials were used for model construction, leaving the remaining 9 sets of data to test model performance. Identical data partition scheme was utilized for MQL and HPC machining conditions. The detailed experimental conditions are listed in Table 2.

The experimental setup comprised of work material, cutting tool, MQL and HPC system and environment is shown in Figure 1.

3. Machine learning methods
3.1 Polynomial regression
Regression analysis is probably one of the most important aspects of statistical as well as machine learning based analysis. The objective of the regression analysis is to model the

Factor	Name	Units	Type	Levels			
				Level 1	Level 2	Level 3	Level 4
A	Environment		Categorical	MQL	HPC	–	–
B	Cutting speed, v	m/min	Numerical	210	320	400	–
C	Depth of cut, a	mm	Numerical	1.5	2	2.5	–
D	Feed rate, f	mm/rev	Numerical	0.224	0.28	0.355	0.4

Table 1.
Control factors along with their values and levels

	Cutting parameters				MQL Responses			HPC			
No.	v (m/min)	a (mm)	f (mm/rev)	F_R (N)	P_c (kW)	K_s (N/mm^2)	F_R (N)	P_c (kW)	K_s (N/mm^2)	Type
1.	210	1.5	0.224	985	2.89	2455.4	913	2.71	2303.6	Training
2.	210	1.5	0.28	1099	3.29	2235.7	1034	3.14	2138.1	Testing
3.	210	1.5	0.355	1346	4.15	2229.1	1268	3.94	2116.4	Training
4.	210	1.5	0.4	1448	4.48	2135.0	1398	4.37	2080.0	Training
5.	210	2	0.224	1160	3.48	2221.0	1165	3.55	2265.6	Training
6.	210	2	0.28	1325	4.05	2064.3	1322	4.09	2085.7	Training
7.	210	2	0.355	1645	5.14	2069.0	1635	5.15	2071.8	Testing
8.	210	2	0.4	1806	5.68	2028.8	1802	5.72	2042.5	Training
9.	210	2.5	0.224	1382	4.21	2148.2	1427	4.35	2217.9	Testing
10.	210	2.5	0.28	1588	4.91	2005.7	1608	4.98	2034.3	Training
11.	210	2.5	0.355	2007	6.30	2028.2	2006	6.33	2038.3	Training
12.	210	2.5	0.4	2234	7.04	2011.0	2228	7.05	2015.0	Training
13.	320	1.5	0.224	935	4.19	2336.3	896	4.07	2273.8	Training
14.	320	1.5	0.28	1055	4.84	2159.5	1011	4.68	2088.1	Training
15.	320	1.5	0.355	1288	6.06	2133.3	1241	5.91	2080.8	Training
16.	320	1.5	0.4	1412	6.70	2095.0	1385	6.61	2066.7	Testing
17.	320	2	0.224	1068	5.14	2151.8	1142	5.32	2227.7	Training
18.	320	2	0.28	1306	6.11	2044.6	1288	6.09	2039.3	Training
19.	320	2	0.355	1611	7.71	2036.6	1605	7.76	2049.3	Testing
20.	320	2	0.4	1781	8.55	2003.8	1727	8.37	1961.3	Training
21.	320	2.5	0.224	1373	6.37	2133.9	1381	6.46	2162.5	Testing
22.	320	2.5	0.28	1572	7.40	1982.9	1571	7.43	1991.4	Training
23.	320	2.5	0.355	1951	9.36	1977.5	1959	9.44	1994.4	Training
24.	320	2.5	0.4	2174	10.43	1955.0	2172	10.49	1966.0	Training
25.	400	1.5	0.224	849	4.90	2187.5	885	5.05	2253.0	Training
26.	400	1.5	0.28	969	5.66	2021.4	994	5.77	2059.5	Training
27.	400	1.5	0.355	1207	7.23	2037.6	1221	7.37	2077.0	Testing
28.	400	1.5	0.4	1316	7.94	1985.0	1364	8.16	2040.0	Training
29.	400	2	0.224	1115	6.48	2169.6	1117	6.49	2174.1	Training
30.	400	2	0.28	1278	7.53	2017.9	1280	7.57	2026.8	Training
31.	400	2	0.355	1573	9.48	2002.8	1583	9.54	2015.5	Training
32.	400	2	0.4	1742	10.54	1976.3	1757	10.59	1985.0	Testing
33.	400	2.5	0.224	1407	8.03	2151.8	1369	7.94	2126.8	Training
34.	400	2.5	0.28	1584	9.23	1977.1	1558	9.21	1974.3	Training
35.	400	2.5	0.355	1961	11.68	1974.1	1944	11.75	1986.5	Testing
36.	400	2.5	0.4	2167	12.92	1938.0	2191	13.23	1985.0	Training

Table 2.
The full data set consisting of 36 data points

relationship between among dependent and independent variables for the purpose of estimation of future values [33]. The simplest approach to the regression task is linear regression, where the dependent (response) variable is modeled as a linear combination of the independent (input) variables. More advanced regression models include multiple regression analysis where dependent variable is also linearly related to the independent variables. An assumed linear relationship among dependent and independent variables might be inadequate to describe the particular relationship. Therefore, this paper deals with the task of polynomial regression. In polynomial regression model, the relationship between the dependent variable and independent variables is modeled in the form of a polynomial equation. Since polynomial regression models are considered as special cases of multiple linear regression models, fitting these models with least squares does not introduce any new problem and analysis of residuals can be utilized to determine the adequacy of the model.
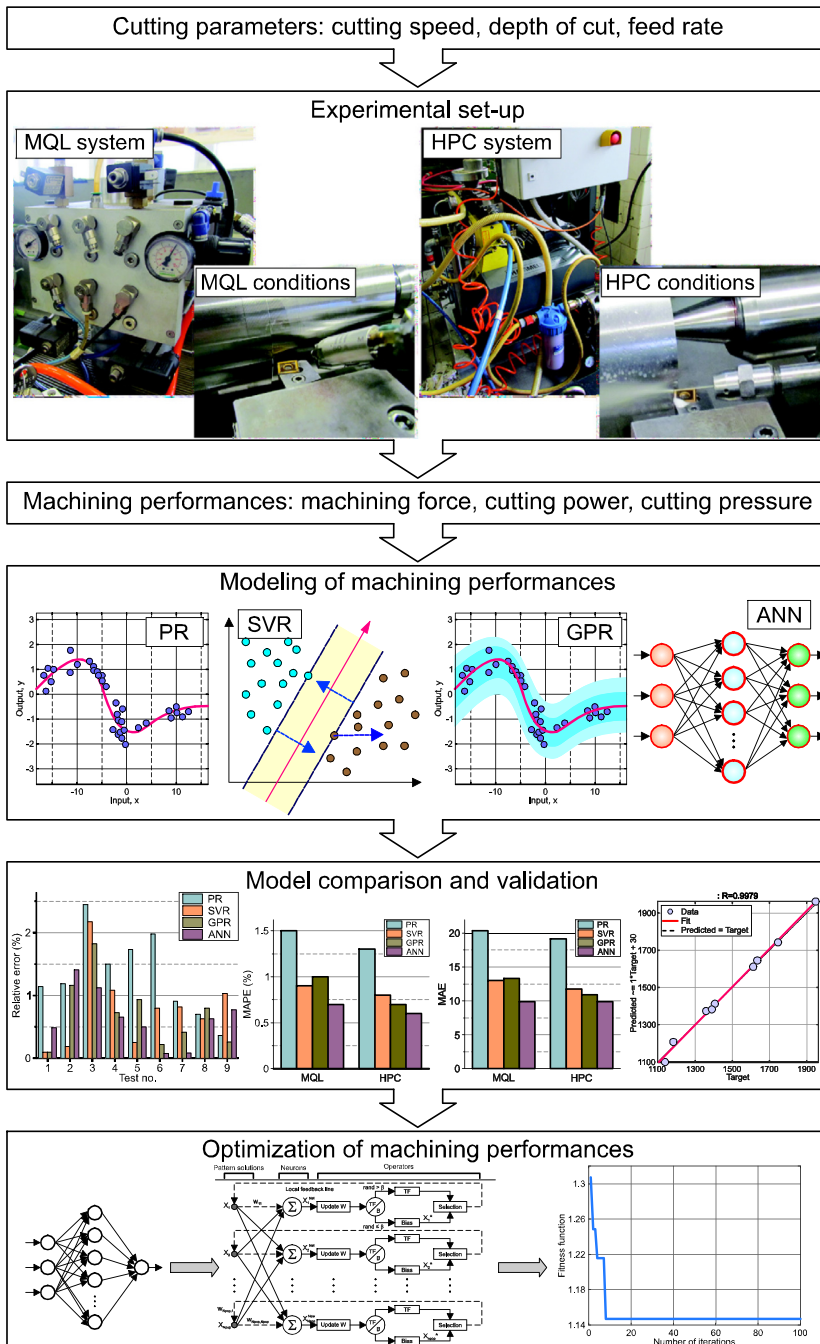
The general expression for a second-order polynomial (quadratic) model that represent the simplest extension of the straight-line model is given by

$$y = \beta_0 + \sum_{i=1}^{k} \beta_i x_i + \sum_{i=1}^{k} \beta_{ii} x_i^2 + \sum_i \sum_j \beta_{ij} x_i x_j + \varepsilon \tag{4}$$

where y is a dependent variable, x_i and x_j are independent variables, β_0 is fixed term, β_i, β_{ii} and β_{ij} are the coefficients of linear, quadratic and cross product terms, respectively, and ε is random error.

3.2 Support vector regression

The support vector machine (SVM) is relatively novel algorithm based on the theoretical foundation of statistical learning theory proposed by Vapnik [34]. Known for its excellent generalization ability, robustness, small number of adjusting parameters, single global optimum solutions and no necessity for experimentation to finding the learning machine architecture SVM is perhaps the most accepted machine learning approach for supervised learning.

SVM acts by producing a separating hyperplane maximizing the margin within two data sets in accordance to their classes which have been formerly mapped to a high dimensional space. The margin is established by creating two parallel hyperplanes on each side of the separating hyperplane. The larger the margin between the classes the better the generalization error of the classifier is achieved. Thus, an optimal separation (solution) is attained by the hyperplane which has the largest distance to the neighboring data points of two classes. The points on the boundary of the slab that are closest to the separating hyperplane are called support vectors. After the support vectors are selected, remain of the feature set can be excluded, because the support vectors involve all the indispensable information for the classifier (Figure 2).

SVM provides particular distinguishing features that make it an effective tool in modeling and prediction tasks with widespread application in many engineering areas. One of the major advantages of using SVM is that model can be determined by assigning a quite a few parameters, namely the kernel function, the loss function, the cost function etc. Furthermore, appropriate architecture does not have to be specified before training and SVM produce a unique solution after training.
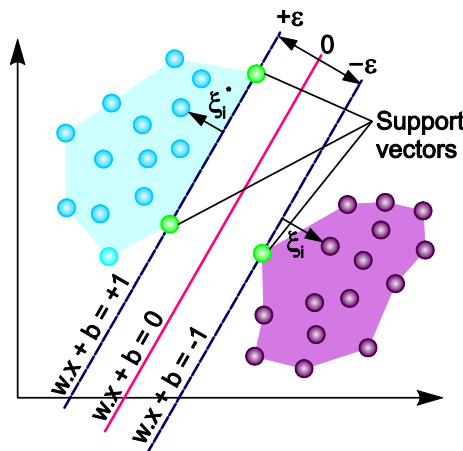

Figure 2.
The graphical
illustration of SVM

Although SVM was originally developed for the classification problems, this algorithm can be also implemented to regression problems by the introduction of a loss function that includes a distance measure. The method by which regression problems can be solved through SVM is known as support vector regression (SVR).

Considering a set of data points $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$ such that $\mathbf{x} \in \mathbb{R}^n$ is an input, $y \in \mathbb{R}$ is a corresponding target output to be estimated by the regression function, and N is the total number of data patterns. The idea of the regression problem is to find a function $f(\mathbf{x})$ that has at most ε deviation from the actually obtained targets y_i for all the training data. The nonlinear relationship between the output and the input can be described by a following regression function:

$$f(\mathbf{x}) = \mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}) + b \tag{5}$$

where $\mathbf{w}$ is the weight vector, $\phi(\mathbf{x})$ is a nonlinear function that map the input pattern $\mathbf{x}$ from $\mathbb{R}^n$ into a higher-dimensional feature space and b is the bias term. The objective is to find values of unknown parameters that are the weight vector $\mathbf{w}$ and bias term b such that the values of $\mathbf{x}$ can be determined by minimizing the regression risk. Based on it, the regression problem can be formulated as follows

$$\text{Minimize } C\frac{1}{N}\sum_{i=1}^{N}L_\varepsilon(y_i, f(\mathbf{x})) + \frac{1}{2}\|\mathbf{w}\|^2 \tag{6}$$

where

$$L_\varepsilon(y, f(\mathbf{x})) = \begin{cases} 0, & \text{if } |y - f(\mathbf{x})| \le \varepsilon, \\ |y - f(\mathbf{x})| - \varepsilon, & \text{otherwise.} \end{cases} \tag{7}$$

In the regularized risk function given by Eq. (6), C is regularized constant, $L_\varepsilon(y, f(\mathbf{x}))$ is the loss function. The most common loss function, namely the linear loss function with ε-insensitivity zone, was proposed by Vapnik [34] s given by Eq. (7). The parameter ε is the difference between actual values and values computed from the regression function.

Introducing two positive slack variables $(\xi_i + \xi_i^*)$ that represent the distance from actual values to the corresponding boundary values of the ε-tube, it is possible to transform Eq. (6) in a primal objective function given by following equation:

$$\text{Minimize } C\frac{1}{N}\sum_{i=1}^{N}(\xi_i + \xi_i^*) + \frac{1}{2}\|\mathbf{w}\|^2 \tag{8}$$

subject to

$$\begin{aligned}
&y_i - \mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}_i) - b \le \varepsilon + \xi_i \\
&\mathbf{w}^{\mathbf{T}}\phi(\mathbf{x}_i) + b - y_i \le \varepsilon + \xi_i^* \\
&\xi_i, \ \xi_i^* \ge 0, \ i = 1, 2, ..., N.
\end{aligned} \tag{9}$$

By adding Lagrangian multipliers this constrained optimization problem can be solved using the Eq. (10)

$$\text{Minimize } \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) - \sum_{i=1}^{N}\beta_i[y_i - \mathbf{w}\phi(\mathbf{x}_i) - b + \varepsilon + \xi_i]$$

$$- \sum_{i=1}^{N}\beta_i^*\left[\mathbf{w}\phi(\mathbf{x}_i) + b - y_i + \varepsilon + \xi_i^*\right] - \sum_{i=1}^{N}(\alpha_i\xi_i + \alpha_i^*\xi_i^*) \tag{10}$$

Eq. (10) is minimized with respect to primal variables $\mathbf{w}$, b, ξ and ξ^*, and is maximized with regard to non-negative Lagrangian multipliers α_i, α_i^*, β_i and β_i^*. Finally, using the appropriate Karush-Kuhn-Tucker conditions on Eq. (9) yields the following dual Lagrangian form of the optimization problem:

$$\text{Maximize} \sum_{i=1}^{N} y_i \left(\beta_i - \beta_i^*\right) - \varepsilon \sum_{i=1}^{N} \left(\beta_i + \beta_i^*\right) - \frac{1}{2} \sum_{i=1}^{N}$$
$$\times \sum_{j=1}^{N} \left(\beta_i - \beta_i^*\right)\left(\beta_j - \beta_j^*\right) K(\mathbf{x}_i, \mathbf{x}_i) \tag{11}$$

subject to

$$\sum_{i=1}^{N} \left(\beta_i - \beta_i^*\right) = 0 \text{ and } \beta_i, \, \beta_i^* \in [0, \, C], \, i = 1, \, 2, ..., N. \tag{12}$$

After obtaining values of the Lagrange multipliers, an optimal solution of weight vector of the regression is represented by Eq. (13)

$$\mathbf{w} = \sum_{i=1}^{N} \left(\beta_i - \beta_i^*\right) \phi(\mathbf{x}_i) \tag{13}$$

Thus, the regression function can be written as follows:

$$f(\mathbf{x}) = \sum_{i=1}^{N} \left(\beta_i - \beta_i^*\right) K(\mathbf{x}_i, \mathbf{x}_j) + b \tag{14}$$

where $K(\mathbf{x}_i, \mathbf{x}_j)$ is the Kernel function which can be described in the feature space as following:
$$K(\mathbf{x}_i, \, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \tag{15}$$

The Kernel function can be substituted by any function satisfying the Mercer's condition. Several kernel functions can be used, such as the following: Gaussian or radial basis function (RBF) kernel, polynomial kernel and linear kernel. In this study, the radial basis kernel function has been chosen and this is given by:

$$K(\mathbf{x}_i, \, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) \tag{16}$$

where σ is the kernel width parameter.

3.3 Gaussian process regression
Gaussian process regression (GPR) for machine learning was initially proposed by Williams and Rasmussen [35]. Compared with other regression techniques based on the kernel method, such as SVM, GPR is a probabilistic model based on the standard Bayesian approaches. GPR is very convenient to deal with complex problems of high dimensionality, nonlinearity and small number of training parameters. As a result of its very good performance, GPR has been widely applied in recent years in various fields of engineering.

Assume that $\mathbf{D} = \{\mathbf{X}, \, \mathbf{y}\}$ is a set of training data, where $\mathbf{X} = [\mathbf{x}_1, \, \mathbf{x}_2, \dots, \mathbf{x}_n]$ is an input vector in R^d and $\mathbf{y} = [y_1, \, y_2, \dots, y_n]$ is a vector containing scalar training outputs y_i in R (extension to multiple outputs is possible). The output y_i could be assumed to contain mean-zero additive Gaussian noise with variance σ_n^2, which is also described $p(\varepsilon_i) = N(0, \, \sigma_n^2)$.

Furthermore, assuming that the outputs are independent and identically distributed, each observation y_i can be thought of as related to an underlying function $f(\mathbf{x})$ through a Gaussian noise model

$$y_i = f(\mathbf{x}_i) + \varepsilon_i \tag{17}$$

The joint distribution over the (noisy) outputs is a zero-mean Gaussian and has the following form

$$p(f(\mathbf{x})|\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n) = N(0, \mathbf{K}(\mathbf{x}, \mathbf{x}') + \sigma_n^2\mathbf{I}) \tag{18}$$

where $f(\mathbf{x}) = [f_1, f_2, ..., f_n]^T$ s a vector of latent function values, $\mathbf{K}(\mathbf{x}, \mathbf{x}')$ is the covariance (kernel) matrix with elements $\mathbf{K}_{ij}(\mathbf{x}_i, \mathbf{x}_j)$, the term $\sigma_n^2\mathbf{I}$ introduces the Gaussian noise, and $\mathbf{I}$ is the identity matrix.

Given the training samples and a set of test points $\mathbf{X}^*$, the goal of GPR is to find the predictive outputs f^* with probabilistic confidence levels. According to the definition of Gaussian process, a prior joint distribution of the training outputs f and test output f^* can be formulated by following equation

$$p(f, f^*) = N\left[0, \begin{matrix} \mathbf{K}_{f,f} & \mathbf{K}_{f^*,f} \\ \mathbf{K}_{f,f^*} & \mathbf{K}_{f^*,f^*} \end{matrix}\right] \tag{19}$$

The independent likelihood can be formulated as follows

$$p(y|f) = N(f, \sigma_n^2\mathbf{I}) \tag{20}$$

Assuming that the hyper-parameters involved in $\mathbf{K}$ were learned from the training data in advance, the posterior distribution can be obtained to give the Gaussian predictive distribution

$$p(f^*|y) = N(\mu_*, \sigma_*^2) \tag{21}$$

where the mean μ and the variance Σ are given by

$$\mu_* = \mathbf{K}_{f^*f}(\mathbf{K}_{ff} + \sigma_n^2\mathbf{I})^{-1}y \tag{22}$$

$$\sigma_*^2 = \mathbf{K}_{f^*f^*} - \mathbf{K}_{f^*f}(\mathbf{K}_{ff} + \sigma_n^2\mathbf{I})^{-1}\mathbf{K}_{ff^*} \tag{23}$$

The squared exponential kernel function evaluates the covariance between the two input feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ as [36]

$$\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_s^2\exp\left(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x}_j)\mathbf{M}^{-2}(\mathbf{x}_i - \mathbf{x}_j)\right) + \sigma_n^2\delta_{ij} \tag{24}$$

where σ_s^2 is the signal variance that quantifies the overall magnitude of the covariance value, usually initialized to 1, $\mathbf{M}$ is a diagonal matrix $\mathbf{M} = \text{diag}\{l_1, l_2, ..., l_m\}$, l are scaling factors and δ_{ij} is Kronecker's delta function that serves to selectively specify the noise variance σ_n^2 to the covariance value.

The parameters of the kernel function denoted by $\theta = [\mathbf{M}, \sigma_s, \sigma_n]$ are called the hyper-parameters of the Gaussian process. These parameters can be learned by maximizing the log likelihood of the training outputs given the inputs

$$\theta_{\max} = \arg \max_\theta\{\log(p(y|X, \theta))\} \tag{25}$$

where the log term in Eq. (25) can be expressed as

$$\log(p(y|X,\,\theta)) = -\frac{1}{2}y^T\left(K + \sigma^2 I\right)^{-1}y - \frac{1}{2}\log|K + \sigma^2 I| - \frac{n}{2}\log 2\pi \tag{26}$$

This nonlinear optimization problem can be solved using numerical optimization techniques, such as gradient-based methods [37].

3.4 Artificial neural networks

In last few decades, machine learning techniques, mostly artificial neural networks (ANN), have caught the interest of many researchers in practically all engineering fields [38]. ANN have been inspired by the human brain information processing in an effort to achieve human-like performance [39]. Because of nonlinear function approximation capability, noise resistance, adaptability and good generalization capability, ANN are especially useful for modeling of the machining processes characterized with many highly interrelated parameters.

Different ANN models have been proposed in the literature, but the multi-layer perceptron (MLP) is the most widely used. MLP is a kind of feed-forward ANN consisting neurons divided into three type of layers: (i) input layer, (ii) output layer and (iii) hidden layers (one or more). Each layer containing a group of neurons (nodes) that are linked with neurons from other layers by connections between the neurons. Each neuron within the network is typically a simple processing unit where the basic calculations are performed to process one or more inputs and produced the proper outputs. Links between neurons, or a synapses, have an associated weight which control the output of the neuron. The outputs of the ANN can be modified by adjusting the values of synaptic weights. These adjustment of weights is designed to be in a direction that minimize the difference among ANN output and present response vectors. The back-propagation (BP) algorithm is probably one of the most popular techniques in the field of ANN. Thus, in the present study, multi-layer feed forward ANN based on BP algorithm is selected to develop the prediction of the model.

Figure 3 shows a typical BP network architecture containing one input layer, one hidden layer and one output layer. Input layer consists of a set of neurons representing the process inputs features. The number of hidden layers as well as the number of nodes per hidden layer is usually determined through a trial and error method, by increasing or the number of hidden layer and neurons during training. The last layer acts as the network output layer. The number of neurons in the last layer that acts as the network output layer is equal to the number of functions being approximated by the model.
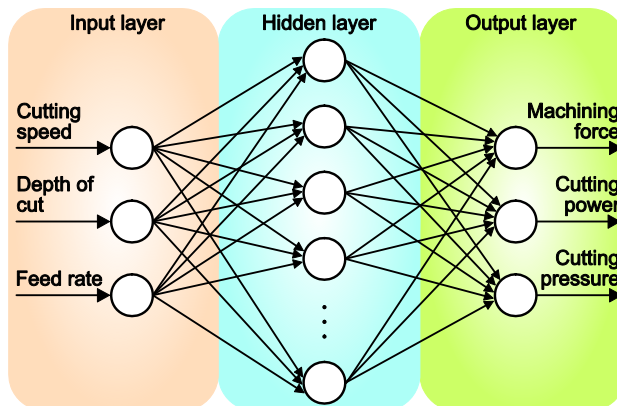

Figure 3.
The ANN architecture

4. Analysis and discussion

4.1 Predicting responses using machine learning techniques

In this section, four predictive models, namely polynomial regression (PR), support vector regression (SVR), Gaussian process regression (GPR) and artificial neural networks (ANN) were developed and compared on the basis of their prediction accuracy.

First, from a set of experimentally obtained data, a polynomial regression models to estimate the machining force, cutting power and cutting pressure as a function of machining parameters such as cutting speed (v), depth of cut (a) and feed rate (f) in MQL assisted turning are determined by

$$F_{R_{\text{MQL}}} = 929.3217 - 1.7317v - 266.9927a - 532.4534f + 0.6772va + 2146.3422af \quad (27)$$

$$P_{c_{\text{MQL}}} = 7.3469 - 0.0192v - 3.9496a - 17.379f + 0.0118va + 0.0502vf + 10.1892af \quad (28)$$

$$K_{s_{\text{MQL}}} = 4997.2191 - 2.3269v - 1058.7474a - 7470.6688f + 0.8768va + 160.6979a^2 + 10250.0764f^2$$
$$\quad (29)$$

Similarly, following polynomial regression models in HPC assisted turning have been established

$$F_{R_{\text{HPC}}} = 928.91 - 1.1937v + 14.0408a - 3418.2f + 2008.56af + 0.00159v^2 + 4970.8f^2 \quad (30)$$

$$P_{c_{\text{HPC}}} = 8.2151 - 0.01723v - 3.1323a - 32.0446f + 0.0099va + 0.0578vf + 9.8238af$$
$$+ 20.9642f^2 \quad (31)$$

$$K_{s_{\text{HPC}}} = 3863.375 - 0.34295v - 93.5567a - 8861.9024f + 12519.3723f^2 \quad (32)$$

Among 36 data sets of process variables related to MQL machining condition, 27 data sets have been utilized with the purpose to create regression equations, while 9 data sets were reserved to test the established equation's predictive capacity. Identical approach was applied in the case of experimental data sets found on HPC machining condition. The analysis of variance (ANOVA) was utilized to justify the significance of the developed regression models. This analysis was carried out for a significance level of $\alpha = 0.05$, i.e. for a confidence level of 95%. ANOVA results shows that the developed mathematical models for machining force, cutting power and cutting pressure are adequate, irrespective of the cutting condition (MQL or HPC).

For SVR-based models Statistics and Machine Learning Toolbox from Matlab was used. The selected machining parameters for SVR inputs were the cutting speed, depth of cut and feed rate, whereas the quality responses were machining force, cutting power and cutting pressure. The type of kernel function, kernel function parameters, the value of C, and the value of ε for the ε-insensitive loss function are the most important factors which have significant effects on the performance of the SVR-based model. Optimized values of these parameters can be determined by using various methods, such as cross validation, grid search, genetic algorithm, particle swarm optimization, etc. In present work, RBF kernel was selected for developing all SVR-based models. The kernel scale was estimated automatically using a heuristic subsampling procedure, which was empirically determined by built-in Matlab functions. For median-sized problems the grid search method is an efficient technique for determination the optimum values of the SVR-based model. In this method, the parameters are varied by fixed step-sizes across a range of values, and the performance of each set of parameters is compared by using different statistical measures such as maximizing correlation coefficient, minimizing normalized root mean square error or mean absolute percentage error, etc. With purpose to improve the generalization ability, this method can use cross validation process. In the present study, the selection of the optimal values of C and ε for each SVR-based model was determined employing grid search method.

The Matlab Statistics and Machine Learning Toolbox was also used for creating and testing the different GPR-based models of the machining force, cutting power and cutting pressure in MQL and HPC machining condition. The determination of appropriate values of hyper-parameters is crucial step for the prediction capabilities of any GPR-based model. Therefore, the grid search method was utilized in order to solve that problem. There are several key parameters and functions influencing the GPR-based model. For this study, the parameters such as kernel function, basis function and initial value for the noise standard deviation of the GPR model were optimized to obtain the best model configuration for the prediction of selected quality responses.

In ANN model, a back-propagation algorithm was used to predict the machining responses, where the error for hidden layers is determined by propagating back the error determined for the output layer. As previously mentioned, the numbers of the neurons in the input and output layer are governed by the numbers of input and output variables, respectively. However, the number of hidden layers and the favorable number of neurons in each hidden layer are subject to the complexity of the target function, generalization capabilities, computation time required for training, the risk of over-fitting, etc. Therefore, network optimization was performed by adjusting the number of hidden layers and the number of nodes in these layers through a trial and error method, in order to adjust the converged error. After examining different neural networks architectures, the result showed that network structure with one hidden layer and nine neurons was found to be accurate and reliable in the present investigation. Among different training methods, the Levenberg-Marquadt was selected as training algorithm, because it consumes less memory and require fewer computation time. The hyperbolic tangent sigmoid transfer function has been used between the input and hidden layers and a linear transfer function have been utilized between the hidden and output layers. The network was trained for 10,000 epochs with a learning rate of 0.03 and a momentum term of 0.1, while error between the desired and the actual outputs is less than 0.001 at the end of training process.

The prediction accuracy of the developed models was analyzed in terms four statistical measures: mean absolute percentage error (MAPE), maximum absolute percentage error (MaxAPE), the mean absolute error (MAE) and normalized root mean square error (NRMSE). Furthermore, coefficient of determination (R^2) was also used as performance metric. These statistical metrics are defined as follows:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{|T_i - Y_i|}{T_i} \right) \cdot 100 \tag{33}$$

$$\text{MaxAPE} = \max \left(\frac{|T - Y|}{T} \cdot 100 \right) \tag{34}$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |T_i - Y_i| \tag{35}$$

$$\text{NRMSE} = \frac{\frac{\sqrt{\sum_{i=1}^{n}(T_i - Y_i)^2}}{n}}{\sigma} \tag{36}$$

$$R^2 = \left(\frac{\sum_{i=1}^{n}(T_i - \bar{T}_i)(Y_i - \bar{Y}_i)}{\sqrt{\sum_{i=1}^{n}(T_i - \bar{T}_i)^2 \sum_{i=1}^{n}(Y_i - \bar{Y}_i)^2}} \right)^2 \tag{37}$$

where n is number of training pattern, T_i and Y_i are experimental and predicted result of ith training pattern, respectively, $\bar{T}_i$ and $\bar{Y}_i$ are the average values of experimental and predicted result, respectively, and σ is standard deviation of T_i.

MAPE, MaxAPE, MAE and NRMSE were used to measure the deviation between the observed and predicted values. The smaller the values of these performance metrics, the closer were the predicted values to the observed values. The statistical metric R^2 was used to measure the correlation between the observed and the predicted values. A value of 1 indicates a perfect relationship between the two variables. Performance evaluations of different machine learning models in terms of these five statistical measures for the test data set are shown in Table 3. Furthermore, to illustrate obtained results more clearly, a performance comparison of machine learning techniques for machining force, cutting power and cutting pressure prediction are also shown in Figures 4–6, respectively.

From the comparative results of Table 3 considering 9 trials for the test data set it is evident that all methods provide reliable accurate prediction accuracy. For MQL cutting

Cutting environment	Response	Method	Statistical metrics				
			MAPE	MaxAPE	MAE	NRMSE	R^2
MQL	$F_{R\text{MQL}}$	PR	1.5	3.3	20.33	0.098	0.9955
		SVR	0.9	1.8	13.00	0.055	0.9983
		GPR	0.9	2.4	12.05	0.056	0.9985
		ANN	0.7	1.2	9.89	0.044	0.9992
	$P_{c\text{MQL}}$	PR	2.7	9.5	0.142	0.065	0.9983
		SVR	0.9	1.7	0.057	0.024	0.9997
		GPR	0.9	2.7	0.053	0.023	0.9998
		ANN	0.8	2.4	0.047	0.022	0.9999
	$K_{s\text{MQL}}$	PR	1.4	2.8	28.63	0.375	0.9906
		SVR	1.1	2.2	23.99	0.339	0.9601
		GPR	0.9	2.9	19.41	0.315	0.9750
		ANN	0.8	2.3	16.96	0.265	0.9799
HPC	$F_{R\text{HPC}}$	PR	1.3	2.4	19.11	0.076	0.9981
		SVR	0.8	2.2	11.78	0.053	0.9991
		GPR	0.7	1.8	10.89	0.048	0.9992
		ANN	0.6	1.4	9.89	0.043	0.9992
	$P_{c\text{HPC}}$	PR	2.1	6.7	0.126	0.053	0.9992
		SVR	0.9	2.7	0.065	0.032	0.9995
		GPR	0.7	2.7	0.054	0.030	0.9996
		ANN	0.6	1.3	0.043	0.018	0.9999
	$K_{s\text{HPC}}$	PR	1.4	3.1	28.51	0.470	0.9464
		SVR	0.8	2.0	16.13	0.300	0.9713
		GPR	0.8	2.3	15.86	0.265	0.9678
		ANN	0.7	2.3	13.51	0.245	0.9739

Table 3.
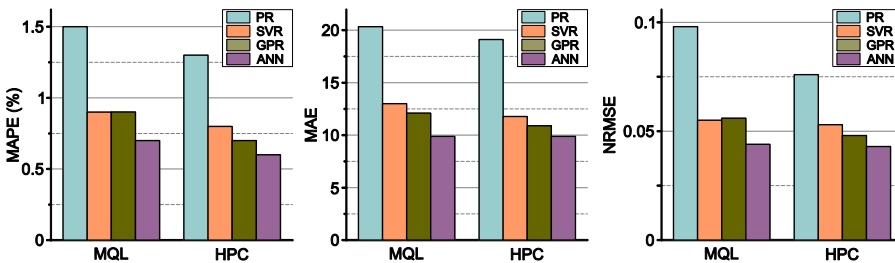Performance evaluations of different models for the test data set

Figure 4.
Performance comparison of machine learning techniques for machining force prediction

environment, MAPE values varies from 0.7 to 2.7, whereas MaxAPE values ranging from 1.2 to 9.5%. The highest values of these two metrics were found in modeling of cutting power using PR. It is evident that SVR and GPR method outperformed PR, while ANN slightly outperformed both SVR and GPR models for all performance characteristics. A similar conclusion can be drawn also with regard to MAE and NRMSE analysis. It was also found that the coefficient of determination for the predicted machining force and cutting power values is found to be a very high (above 0.99), noting slightly lower values in modeling of cutting pressure (above 0.96).

According to presented results for HPC assisted turning, MAPE values are within the range 0.6–2.1% and MaxAPE values are identified to be between 1.3 and 6.7%. The highest values of MAPE and MaxAPE were found in modeling of cutting power when PR is employed, just as in the MQL cutting condition. The results of these two metrics, as well as MAE and NRMSE, again confirm the superiority of using SVR and GPR over PR method. In addition, it was found that ANN is even slightly accurate compared to SVR and GPR methods when estimating all responses. The determination coefficient for the estimated machining force and cutting power values were both very close to 1 (above 0.99), while somewhat lower values (above 0.94) were observed in estimating of cutting pressure.

Based on the analysis of the results, it is obvious that all considered machine learning techniques are accurate, efficient and practical tool for estimation of machining force, cutting energy and cutting pressure under different cooling/lubricating conditions. The comparison of regression based machine learning techniques shows that SVR and GPR models have similar performances and have better outcomes in terms of accuracy than PR-based models. Moreover, the ANN-based models were even somewhat more accurate compared to the regression based machine learning techniques for all datasets. Despite the fact that the SVR, GPR and ANN outperformed PR-based method, results produced by this method have been quite satisfactory. Additionally, in terms of computational time, PR method is the fastest because the training does not require much parameter tuning.

Figure 5.
Performance comparison of machine learning techniques for cutting power prediction
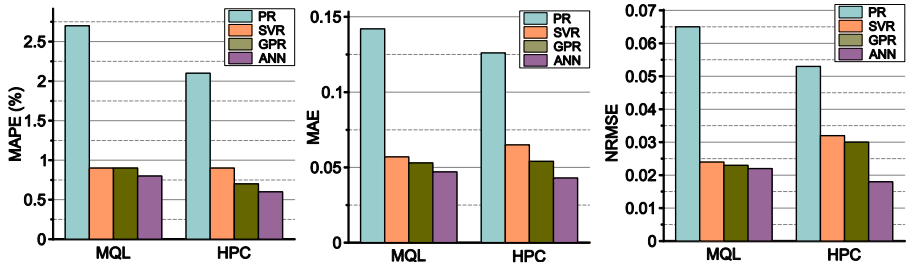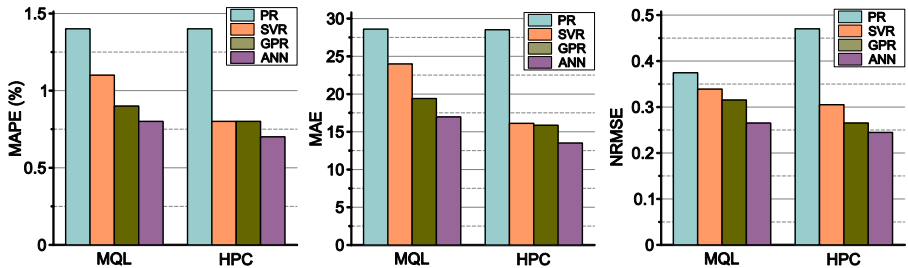

Figure 6.
Performance comparison of machine learning techniques for cutting pressure prediction

4.2 Multi-objective optimization
The relationships between machining parameters and responses have been established using different machine learning methods. However, SVR, GPR and ANN-based have been characterized by complex non-linear functions, thus conventional optimization methods are difficult to use effectively and consistently. On the contrary, PR-based models are simple, yet effective, easy to interpret and broadly applicable to relate input parameters and output parameters. Therefore, the mathematical relation between machining variables and responses for MQL and HPC machining environment, given in Eqs. (27)–(29) and Eqs. (30)–(32), respectively, were used as functional equations to establish the objective functions. The objective has been to find the optimal machining condition for minimizing the machining force, cutting power and cutting pressure simultaneously. Hence, for multi-objective optimization of the turning under MQL and HPC cutting conditions, the following complex objective functions (COF) were developed

$$\text{COF}_{\text{MQL}}(v,\,a,\,f) = w_1\frac{F_{R_{\text{MQL}}}}{F_{R_{\text{MQLmin}}}} + w_2\frac{P_{c_{\text{MQL}}}}{P_{c_{\text{MQLmin}}}} + w_3\frac{K_{s_{\text{MQL}}}}{K_{s_{\text{MQLmin}}}} \tag{38}$$

$$\text{COF}_{\text{HPC}}(v,\,a,\,f) = w_1\frac{F_{R_{\text{HPC}}}}{F_{R_{\text{HPCmin}}}} + w_2\frac{P_{c_{\text{HPC}}}}{P_{c_{\text{HPCmin}}}} + w_3\frac{K_{s_{\text{HPC}}}}{K_{s_{\text{HPCmin}}}} \tag{39}$$

where w_1, w_2 and w_3 are the weight values of the machining force, cutting power and cutting pressure, respectively. In this study, equal weights for all responses were selected, i.e. $w_1 = w_2 = w_2 = 1/3$.

The minimization of the developed complex objective functions should be performed on the basis of operation constraints. These constraints impose the lower and upper limits of the experimental parameters and are given as follows: $200 \leq v \leq 400$, $1.5 \leq a \leq 2.5$, $0.224 \leq f \leq 0.4$.

For the purpose of solve such types of problems, different metaheuristic methods such as the genetic algorithm, particle swarm optimization, differential evolution, simulated annealing, etc. are commonly applied. In present research, a relatively new optimization algorithm, namely neural network algorithm was employed for the multi-objective optimization of the MQL and HPC assisted turning. This method is developed based on the structure and concept of artificial neural networks in terms of generating new candidate solutions and also employing other operators used in the conventional ANN. More details about this optimization algorithm can be found in [40].

The minimum values of the complex objective function $\text{COF}_{\text{MQL}} = 1.147$ and $\text{COF}_{\text{HPC}} = 1.095$ were found after 8 iterations and 5 iterations, respectively. The optimization algorithm results revealed that for both cutting environments the best combination of machining parameters in simultaneously optimizing the machining force, cutting power and cutting pressure was: 210 m/min for cutting speed, 1.5 for depth of cut and 0.224 mm/rev for feed rate. Thus, the experimental trial 1 has the optimal combination of machining parameters for MQL as well as for HPC assisted turning.

5. Conclusion
The present study revealed comparative analysis of four machine learning methods: polynomial regression, support vector regression, Gaussian process regression and artificial neural networks, for machining force, cutting power and cutting pressure prediction in the turning of AISI 1045 using coated carbide tools. In the developed models, the input data include cutting parameters, such as cutting speed, depth of cut and feed rate. The prediction of selected quality characteristics was carried out for two different machining environments. To be specific, the study covered the minimum quantity lubrication and high-pressure coolant assisted turning. The performance of four methods were evaluated in terms of

different statistical measures such as mean absolute percentage error, maximum absolute percentage error, the mean absolute error, normalized root mean square error and correlation coefficient and very good agreements with experimental results were observed.

The developed prediction models of machining force, cutting power and cutting pressure have exhibited a very high accuracy in prediction for MQL as well as for HPC machining environment. According to presented results, MAPE values are within the range 0.7–2.7% and 0.6–2.1%, whereas MaxAPE values varies from 1.2 to 9.5% and from 1.3 to 6.7%, for MQL and HPC cutting environment, respectively. The highest values of these two statistical measures were observed in modeling of cutting power in both cutting environment when PR is employed. The comparison was also done using MAE and NRMSE as the performance measures. When regression based machine learning techniques are compared, it is found that SVR and GPR models have comparable performances and that these models obtain relatively better accuracy than those achieved by PR-based model. Moreover, the results revealed that ANN-based model have slightly better outcomes in terms of accuracy than regression based machine learning methods when estimating quality characteristic for both cutting environment. Summarizing the main features of the statistical results, it can be concluded that selected machine learning techniques produce adequate results when compared to the experimental outcomes. The estimated machining parameters on test data set were found to be closely correlated with the real performance results. Thus, by using developed models acceptable results can be estimated rather than experimentally obtained which consequently reduces the testing cost and time.

Moreover, mathematical models of the multi-objective optimization were established based on the polynomial regression method and metaheuristic approach based on a neural network algorithm was used to obtain optimal solutions. The optimal combination of machining parameters for both cutting environment based on the studied performance criteria (i.e. machining force, cutting power and cutting pressure) was found to be 210 m/min for cutting speed, 1.5 for depth of cut and 0.224 mm/rev for feed rate.

The performance can additional be enhanced with a wide range of cutting conditions, taking into account other major aspects of cutting operations such as tool coatings, tool geometry, workpiece materials, etc., as well as considering the additional quality characteristics.

References

[1]. M. Mia, M.K. Gupta, G. Singh, G. Królczyk, D.Y. Pimenov, An approach to cleaner production for machining hardened steel using different cooling-lubrication condition, J. Cleaner Prod. 187 (2018) 1069–1081.

[2]. S. Ghosh, P.V. Rao, Application of sustainable techniques in metal cutting for enhanced machinability: a review, J. Cleaner Prod. 100 (2015) 17–34.

[3]. K. Weinert, I. Inasaki, J.W. Sutherland, T. Wakabayashi, Dry machining and minimum quantity lubrication, CIRP Ann-Manuf. Techn. 53 (2) (2004) 511–537.

[4]. L. Zhou, J. Li, F. Li, Q. Meng, J. Li, X. Xu, Energy consumption model and energy efficiency of machine tools: a comprehensive literature review, J. Cleaner Prod. 112 (2016) 3721–3734.

[5]. J.K. Watson, K.M.B. Taminger, A decision-support model for selecting additive manufacturing versus subtractive manufacturing based on energy consumption, J. Cleaner Prod. 176 (2018) 1316–1322.

[6]. V. Vijayaraghavan, S. Castagne, Sustainable manufacturing models for mass finishing process, Int. J. Adv. Manuf. Technol. 86 (1–4) (2016) 49–57.

[7]. V.N. Gaitonde, S.R. Karnik, L. Figueira, J.P. Davim, Performance comparison of conventional and wiper ceramic inserts in hard turning through artificial neural network modeling, Int. J. Adv. Manuf. Technol. 52 (1–4) (2011) 101–114.

[8]. A. Khan, K. Maity, A comprehensive GRNN model for the prediction of cutting force, surface roughness and tool wear during turning of CP-Ti grade 2, Silicon 10 (5) (2018) 2181–2219.

[9]. R. Kumar, S. Chauhan, Study on surface roughness measurement for turning of Al 7075/10/SiCp and Al 7075 hybrid composites by using response surface methodology (RSM) and artificial neural networking (ANN), Measurement 65 (2015) 166–180.

[10]. A. Laouissi, M.A. Yallese, A. Belbah, S. Belhadi, A. Haddad, Investigation, modelling, and optimization of cutting parameters in turning of gray cast iron using coated and uncoated silicon nitride ceramic tools. Based on ANN, RSM, and GA optimization, Int. J. Adv. Manuf. Technol. 101 (1–4) (2019) 523–548.

[11]. A. Agrawal, S. Goel, W.B. Rashid, M. Price, Prediction of surface roughness during hard turning of AISI 4340 steel (69 HRC), Appl. Soft Comp. 30 (2015) 279–286.

[12]. H. Aouici, H. Bouchelaghem, M.A. Yallese, M. Elbah, B. Fnides, Machinability investigation in hard turning of AISI D3 cold work steel with ceramic tool using response surface methodology, Int. J. Adv. Manuf. Technol. 73 (9–12) (2014) 1775–2178.

[13]. V.S. Sharma, S. Dhiman, R. Sehgal, S. Sharma, Estimation of cutting forces and surface roughness for hard turning using neural networks, J. Intell. Manuf. 19 (4) (2008) 473–483.

[14]. I. Meddour, M.A. Yallese, H. Bensouilah, A. Khellaf, M. Elbah, Prediction of surface roughness and cutting forces using RSM, ANN, and NSGA-II in finish turning of AISI 4140 hardened steel with mixed ceramic tool, Int. J. Adv. Manuf. Technol. 97 (5–8) (2018) 1931–1949.

[15]. A. Zerti, M.A. Yallese, I. Meddour, S. Belhadi, A. Haddad, T. Mabrouki, Modeling and multi-objective optimization for minimizing surface roughness, cutting force, and power, and maximizing productivity for tempered stainless steel AISI 420 in turning operations, Int. J. Adv. Manuf. Technol. 102 (1–4) (2019) 135–157.

[16]. L.C.S. Rocha, A.P. Paiva, J.P. Rotela, P.P. Balestrassi, P.H.S. Campos, Robust multiple criteria decision making applied to optimization of AISI H13 hardened steel turning with PCBN wiper tool, Int. J. Adv. Manuf. Technol. 89 (5–8) (2017) 2251–2268.

[17]. R. Quiza, L. Figueira, J.P. Davim, Comparing statistical models and artificial neural networks on predicting the tool wear in hard machining D2 AISI steel, Int. J. Adv. Manuf. Technol. 37 (7–8) (2008) 641–648.

[18]. A. Salimiasl, A. Özdemir, Analyzing the performance of artificial neural network (ANN)-, fuzzy logic (FL)-, and least square (LS)-based models for online tool condition monitoring, Int. J. Adv. Manuf. Technol. 87 (1–4) (2016) 1145–1158.

[19]. A.K. Gupta, S.C. Guntuku, R.K. Desu, A. Balu, Optimisation of turning parameters by integrating genetic algorithm with support vector regression and artificial neural networks, Int. J. Adv. Manuf. Technol. 77 (1–4) (2015) 331–339.

[20]. M. Mia, N.R. Dhar, Response surface and neural network based Predictive models of cutting temperature in hard turning, J. Adv. Res. 7 (6) (2016) 1035–1044.

[21]. F. Kara, K. Aslantaş, A. Çiçek, Prediction of cutting temperature in orthogonal machining of AISI 316 L using artificial neural network, Appl. Soft Comp. 38 (2016) 4–74.

[22]. I. Korkut, A. Acır, M. Boy, Application of regression and artificial neural network analysis in modelling of tool–chip interface temperature in machining, Expert Syst. Appl. 38 (9) (2011) 11651–11656.

[23]. D.J. Cica, B. Sredanovic, G. Lakic-Globocki, D. Kramar, Modeling of the cutting forces in turning process using various methods of cooling and lubricating: an artificial intelligence approach, Adv. Mech. Eng. 5 (2013) 1–8.

[24]. M. Mia, M.A. Khan, N.R. Dhar, Performance prediction of high-pressure coolant assisted turning of Ti-6Al-4V, Int. J. Adv. Manuf. Technol. 90 (5–8) (2017) 1433–1445.

[25]. D. Cica, B. Sredanovic, D. Kramar, Modelling of tool life and surface roughness in hard turning using soft computing techniques: a comparative study, Int. J. Mater. Prod. Tech. 50 (1) (2015) 49–64.

[26]. M. Mia, N.R. Dhar, Prediction of surface roughness in hard turning under high pressure coolant using Artificial Neural Network, Measurement 92 (2016) 464–474.

[27]. M. Mia, N.R. Dhar, Prediction and optimization by using SVR, RSM and GA in hard turning of tempered AISI 1060 steel under effective cooling condition, Neural Comput. Appl. 31 (7) (2019) 2349–2370.

[28]. M. Kamruzzaman, S.S. Rahman, M.Z.I. Ashraf, N.R. Dhar, Modeling of chip–tool interface temperature using response surface methodology and artificial neural network in HPC-assisted turning and tool life investigation, Int. J. Adv. Manuf. Technol. 90 (5–8) (2017) 1547–1568.

[29]. M. Mia, M.H. Razi, I. Ahmad, R. Mostafa, S.M.S. Rahman, D.H. Ahmed, P.R. Dey, N.R. Dhar, Effect of time-controlled MQL pulsing on surface roughness in hard turning by statistical analysis and artificial neural network, Int. J. Adv. Manuf. Technol. 91 (9–12) (2017) 3211–3223.

[30]. M. Mia, M.S. Morshed, Md Kharshiduzzaman, M.H. Razi, Md R. Mostafa, S.M.S. Rahman, I. Ahmad, M.T. Hafiz, A.M. Kamal, Prediction and optimization of surface roughness in minimum quantity coolant lubrication applied turning of high hardness steel, Measurement 118 (2018) 43–51.

[31]. A.T. Abbas, M.K. Gupta, M.S. Soliman, M. Mia, H. Hegab, M. Luqman, D.Y. Pimenov, Sustainability assessment associated with surface roughness and power consumption characteristics in nanofluid MQL-assisted turning of AISI 1045 steel, Int. J. Adv. Manuf. Technol. 1–17 (2019).

[32]. M. Nouioua, M.A. Yallese, R. Khettabi, S. Belhadi, M.L. Bouhalais, F. Girardin, Investigation of the performance of the MQL, dry, and wet turning by response surface methodology (RSM) and artificial neural network (ANN), Int. J. Adv. Manuf. Technol. 93 (5–8) (2017) 2485–2504.

[33]. D.A. Freedman, Statistical Models: Theory and Practice, Cambridge University Press, 2005.

[34]. V.N. Vapnik, Statistical learning theory, J. Assoc. Comp. Mach. 40 (1993) 741–764.

[35]. C.K. Williams, C.E. Rasmussen, Gaussian Processes for Regression, MIT, Cambridge, 1996.

[36]. R.M. Neal, Bayesian Learning for Neural Networks, Springer, 1996.

[37]. C.E. Rasmussen, Gaussian processes in machine learning, in: Advanced Lectures on Machine Learning, Springer, Berlin Heidelberg New York, 2004, pp. 63–71.

[38]. J.V. Abellan-Nebot, F. Romero Subirón, A review of machining monitoring systems based on artificial intelligence process model, Int. J. Adv. Manuf. Technol. 47 (1–4) (2010) 237–257.

[39]. V. Vijayaraghavan, A. Garg, J.S.L. Lam, B. Panda, S.S. Mahapatra, Process characterisation of 3D-printed FDM components using improved evolutionary computational approach, Int. J. Adv. Manuf. Technol. 78 (5–8) (2015) 4781–4793.

[40]. A. Sadollah, H. Sayyaadi, A. Yadav, A dynamic metaheuristic optimization model inspired by biological nervous systems: neural network algorithm, Appl. Soft Comput. 71 (2018) 747–782.

Corresponding author
Djordje Cica can be contacted at: djordje.cica@mf.unibl.org