



Non-dominated sorting modified teaching–learning-based optimization for multi-objective machining of polytetrafluoroethylene (PTFE)

Elango Natarajan¹ · Varadaraju Kaviarasan² · Wei Hong Lim¹ · Sew Sun Tiang¹ · S. Parasuraman³ · Sangeetha Elango⁴

Received: 11 April 2019 / Accepted: 29 July 2019 / Published online: 2 August 2019
© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

A non-dominated sorting modified teaching–learning-based optimization (NSMTLBO) is proposed to obtain the optimum solution for a multi-objective problem related to machining Polytetrafluoroethylene. Firstly, an experimental design is done and the L27 orthogonal array with three-level of cutting speed (V_c), feed rate (f), depth of cut (ap) and nose radius (N_r) is formulated. A CNC turning machine is used to perform experiments with cemented carbide tool at an insert angle of 80° and the response variables known as surface finish and material removal rate are measured. A response surface model is rendered from the experimental results to derive the minimization function of surface roughness (R_a) and maximization function of material removal rate (MRR). Both optimization functions are solved simultaneously using NSMTLBO. A fuzzy decision maker is also integrated with NSMTLBO to determine the preferred optimum machining parameters from Pareto-front based on the relative importance level of each objective function. The best responses $R_a = 2.2347 \mu\text{m}$ and $MRR = 96.835 \text{ cm}^3/\text{min}$ are predicted at the optimum machining parameters of $V_c = 160 \text{ mm/min}$, $f = 0.5 \text{ mm/rev}$, $ap = 0.98 \text{ mm}$ and $N_r = 0.8 \text{ mm}$. The proposed NSMTLBO is reported to outperform other six peer algorithms due to its excellent capability in generating the Pareto-fronts which are more uniformly distributed and resulted higher percentage of non-dominated solutions. Furthermore, the prediction results of NSMTLBO are validated experimentally and it is reported that the performance deviations between the predicted and actual results are lower than 3.7%, implying the applicability of proposed work in real-world machining applications.

Keywords Design of experiments · Multi-response · Non-dominated sorting modified teaching–learning-based optimization · Response surface model · Surface roughness

List of symbols

Acronyms

CFRP	Carbon fibre reinforced polymer
DOE	Design of experiment
ECM	Electrochemical machining

EDM	Electric discharge machining
EMOTLBO	Enhanced multi-objective teaching–learning-based optimization
FIB	Focused ion beam
ICA	Imperialist competitive algorithm
microEDM	Micro-electric discharge machining
MOEA	Multi-objective evolutionary algorithms
MOGWO	Multi-objective grey wolf optimizer
MO-ITLBO	Multi-objective improved teaching–learning based optimization
MOP	Multi-objective optimization problem
MOPSO	Multi-objective particle swarm optimization
MOTLBO	Multi-objective teaching–learning-based optimization
NSGA-II	Non-dominated sorting genetic algorithm II
NSMTLBO	Non-dominated sorting modified teaching–learning-based optimization

✉ Elango Natarajan
cad.elango.n@gmail.com

¹ Faculty of Engineering, Technology and Built Environment, UCSI University, 56000 Kuala Lumpur, Malaysia

² Department of Mechanical Engineering, Sona College of Technology, Salem, Tamil Nadu 636005, India

³ School of Engineering (Mechatronics), Monash University, Subang Jaya, Malaysia

⁴ FTMS College, Cyber Jaya, Malaysia

NSTLBO	Non-dominated sorting teaching–learning-based optimization	W_P	Set containing the Pareto non-dominated solution set of MO-ITLBO
PSO	Particle swarm optimization	X_P	Set containing the Pareto non-dominated solution set of NSTLBO
PCD	Polycrystalline diamond		
PTFE	Polytetrafluoroethylene		
RSM	Response surface model	Operators	
TLBO	Teaching-learning-based optimization	$\psi(\cdot, \cdot, \cdot, \cdot)$	An operator that returns the response variable value based on the given control variables
WEDM	Wire-electric discharge machining	$\Psi_m(\cdot)$	An operator that returns the value of the m -th objective function based on the given individual solution
Indices and sets		$Trunc(\cdot, \cdot)$	An operator that returns the best N members with the lowest ranking and highest crowding distance values
d	Index of each dimension component of learner	$C(\cdot, \cdot)$	An operator that returns the percentage of solution set from one Pareto front that is dominated by solution set from another Pareto front
m	Index of objective function	\prec_{cco}	Crowding-comparison operator to compare the superiority of two solutions
n	Index of learner		
r	Index to indicate the rank of a given front		
s	Index of learner that is randomly selected for comparison		
F_r	Set containing all learners with the r -th rank value		
$L_{m,r}$	Set containing all sorted members of the r -th front for the m -th objective		
Q	Set containing all learners to create the next front		
S_n	Set containing all solutions dominated by the n -th learner		
Rank	Set containing all non-domination rank values of N learners		
F	Set containing all members from R fronts		
Δ	Set containing the crowding distances of all N learners		
P	Set containing all population members		
P^{off}	Set containing all offspring members		
P^{comb}	Set containing the combination of both population and offspring members		
Ψ^U	Set containing the utopia point of a multi-objective optimization problem with M objective functions		
Ψ^{SN}	Set containing the pseudo nadir point of a multi-objective optimization problem with M objective functions		
R_P	Set containing the Pareto non-dominated solution set of NSMTLBO		
S_P	Set containing the Pareto non-dominated solution set of MOPSO		
T_P	Set containing the Pareto non-dominated solution set of NSGA-II		
U_P	Set containing the Pareto non-dominated solution set of MOGWO		
V_P	Set containing the Pareto non-dominated solution set of MOTLBO		
		Parameters and variables	
		V_c	Cutting speed
		f	Feed rate
		ap	Depth of cut
		N_r	Nose radius
		R_a	Surface roughness
		ΔR_a	Error rate of surface roughness
		MRR	Material removal rate
		ΔMRR	Error rate of material removal rate
		$D_{initial}$	Initial diameter of PTFE sample before machining process
		D_{final}	Final diameter of PTFE sample after machining process
		L	Length of cut of PTFE sample
		T	Time taken to cut PTFE sample
		Y	Response term of regression equation
		α_0	Free term of regression equation
		X_i	Control variable term of regression equation
		β_i	Linear coefficient term of regression equation
		β_{ii}	Quadratic coefficient term of regression equation
		β_{ij}	Interacting coefficient term of regression equation
		D	Number of decision variables to be optimized
		N	Population size
		T_f	Teaching factor that can be set as either 1 or 2
		T_{f1}, T_{f2}	Teaching factors with the range of 1 to 2 generated from uniform distribution

C_n	Domination count to indicate the number of solutions dominate the n -th learner	γ	Counter of function evaluations
$Rank_n$	Non-domination rank value of the n -th learner	Γ	Maximum fitness evaluation numbers
$\Delta_{a,r}$	Crowding distance of the a -th member in the r -th front	d_a	Smallest Euclidean distance between the a -th and b -th Pareto optimal solutions
R	Upper limit of front counter	\bar{d}	Average value of all smallest Euclidean distance
$ F_r $	Number of members in the r -th front	S	Spacing measure
$X_{n,d}$	The d -th component of n -th candidate solution	SD	Standard deviation
$X^{teacher}$	Solution vector that represents the best solution known as teacher	R^2	Percentage of variation of data
X^{mean}	Solution vector that represents the average knowledge level of population	P	Significance of control variables
X_n^{new}	New solution vector produced by the n -th learner during the teacher or learner phases	c_1, c_2	Acceleration coefficients
X_d^U	Upper limit of the d -th dimensional component	α	Grid inflation rate
X_d^L	Lower limit of the d -th dimensional component	$nGrid$	Number of grid per dimension
$X_{a,d}^{Cand}$	Solution vector of the d -th dimensional component of a -th candidate teacher	$nGroup$	Number of group created for multiple group learning
$X^{preferred}$	Solution vector of most preferred Pareto optimal solution	ε	Parameter used for epsilon dominance method
$X_n^{teacher}$	Solution vector of teacher assigned to the n -th learner	$ A $	Archive size
\tilde{X}_n^{mean}	Weighted mean position vector assigned to the n -th learner		
$E_{n,a}$	Normalized Euclidean distance between the n -th learner and the a -th candidate teacher		
r_1, r_2, r_3, r_4	Random numbers with the range of 0 to 1 generated from uniform distribution		
r_5	Random numbers with the range of -1 to 1 generated from uniform distribution		
P_{cr}	Crossover rate		
P_{mut}	Mutation probability		
d_r	Randomly selected dimensional component for mutation		
Ψ_m^U	Utopia point of a multi-objective optimization problem in the m -th objective function		
Ψ_m^{SN}	Pseudo nadir point of a multi-objective optimization problem in the m -th objective function		
μ_a^m	Membership value of the a -th Pareto optimal solution in the m -th objective function		
μ_a	Total degree of optimality of each a -th Pareto optimal solution		
w_m	Relative importance of each m -th objective function		
w_1	Relative importance level of minimizing surface finish		
w_2	Relative importance level of maximizing material removal rate		

Introduction

Polytetrafluoroethylene (PTFE) is a fluoropolymer that has many excellent properties such as high hydrophobicity, high oleophobicity, high chemical resistance, high antifouling property, high sliding property, high thermal resistance, high weather resistance, low relative permittivity, and low dielectric loss tangent. It has been used in many applications including polymer bearing, gaskets, gears, valve seats, piston rings, seals, bushes, hose assemblies, high voltage switches, extension joints, cook wears, clinical, containers and pipework for reactive and corrosive chemicals. It has also been used in biomedical applications such as oxygenator membrane, vascular graft and catheter coating. The virgin PTFE rods or components are manufactured through sintering or compression molding. The specific grade of PTFE which has the greater dielectric strength is frequently used in wire and cable wrap, and to separate conductive surfaces in capacitors.

As an instance, thick walled close tolerance extruded PTFE can be machined into standoff insulators and can also be used in high voltage encapsulation devices for miniature components with high dimensional accuracy and integrity. These components may require further machining operations to have net shape, suitable for the specific application. The accuracy of the machining process is decided by the control parameters such as feed, depth of cut, speed, nose radius etc. The poorly set control parameters do increase the surface roughness and lead to the poor net shape. The current research is aimed to determine the optimum control parameters for obtaining the minimum surface roughness and maximum material removal rate of material during machining of PTFE. The minimization surface roughness

increases the quality of the products and the maximization of material remove rate increases the production rate (quantity) of the products.

In most existing literatures, the regression models were developed as nonlinear functions consisting of several input process parameters with boundary limits. Although the idea of utilizing optimization method to solve a given regression model is proven feasible to obtain the optimum machining parameters, the solutions produced by traditional optimization algorithms (e.g., nonlinear programming, quadratic programming, dynamic programming and etc.) tend to stagnate into local optima due to the difficulty of achieving good guess of initial solutions (Mukherjee and Ray 2006). The emergence of new metaheuristic search algorithms (Lim and Isa 2015; Ji et al. 2017; Al-Omouh et al. 2019; Hu et al. 2019; Fan and Yan 2016) has become the popular alternative to overcome these challenges. Because, these algorithms do not strongly rely on the quality of initial solutions and the gradient information of objective functions to solve an optimization problem. The substantial amounts of real-world optimization problems have been tackled using the metaheuristic search algorithms due to their promising search robustness. These include the energy management optimization (Yao et al. 2015, 2017; Mathew et al. 2018; Zhao et al. 2013; Zhao et al. 2012), robust control system design (Tang et al. 2013, 2014; Solihin et al. 2009, 2010), machine vision and image processing (Senthilnath et al. 2016; Li et al. 2011; Jiao and Pan 2019; Mistry et al. 2017; Van and Kang 2016), just to name a few.

Majority of machining problems are formulated as the multi-objective optimization problems (MOPs) due to the simultaneous consideration of multiple performance characteristics. There could be three to four control parameters involved in the optimization of response variables. There are generally two popular approaches used to tackle the MOPs, namely the priori method and the posterior method (Collette and Siarry 2003). In contrast to the priori method that only be able to produce one optimum solution at each trial of simulation by considering the predefined relative importance levels of objective functions, the posterior method is able to produce a complete set of trade-off optimal solutions with single run. Referring to the relative importance of objective functions that normally not known in advance, the process planner can choose a unique solution from the complete set of trade-off solutions obtained (Abbas et al. 2016). Due to its excellent capability to cater the frequent changes of customer requirements in real-world scenario of machining process, the posterior method is considered as a more promising approach than the priori method to address the machining problems with MOP characteristic. In addition, the performance selection index method for solving machining multi-criteria decision making problems was discussed by Petković et al. (2017).

Substantial research works have been reported to design various posterior version of multi-objective evolutionary algorithms (MOEAs), aiming to effectively optimize different multi-objective machining problems. These MOEAs can be generally categorized under two different frameworks known as the Pareto-dominance-based (Deb et al. 2005; Corne et al. 2001; Zitzler et al. 2001) and decomposition-based (Zhang and Li 2007; Deb and Jain 2014). The micro-Electric Discharge Machining (microEDM) of Stainless Steel (SS) 316L was modelled with response surface methodology (RSM). Three machining parameters known as discharge current, pulse-on and pulse-off times were optimized by a multi-objective genetic algorithm (Suresh et al. 2014) in order to achieve the maximization of material removal rate and the minimization of tool wear rate simultaneously. Three process parameters for the electrochemical machining (ECM) of die tool steel with nanofluid coolant (i.e., tool feed rate, applied voltage and electrolyte discharge rate) were optimized using a multi-objective genetic algorithm developed by Sathiyamoorthy et al. (2015a). Sathiyamoorthy et al. (2015b) also attempted to optimize the machining of AISI 202 Austenitic stainless steel by considering multiple objectives. A regression model was developed by Viswanathan et al. (2017) to correlate the response variable of cutting temperature for AZ91 Magnesium alloy with three process parameters known as the depth of rate, cutting speed and feed rate. Notably, the cutting speed was reported to play significant role in minimizing the cutting zone temperature. The multi-objective ultrasonic machining process was addressed by Teimouri et al. (2015) with an imperialist competitive algorithm (ICA), while the particle swarm optimization was employed by Mohanty et al. (2016) to address the multi-response optimization of electric discharging machining process. Recently, the cuckoo optimization algorithm and hoopoe heuristic were employed by Mellal and Williams (2016) to optimize the process parameters of two traditional machining processes (i.e., drilling process and grinding process) and four modern machining processes (i.e., abrasive jet machining, abrasive water jet machining, ultrasonic machining and water jet machining). A multi-objective version of Jaya algorithm has been proposed by Rao et al. (2017b) to optimize four non-traditional machining processes, i.e., focused ion beam (FIB) micro-milling, laser cutting, wire-electric discharge machining (WEDM) and ECM. Similar multi-objective Jaya algorithm was further applied by Rao et al. (2019) to optimize the process parameters of abrasive waterjet machining process and a multi-objective decision making method was incorporated to select a preferred solution from Pareto front based on the predefined requirements of process planner. The electrical discharge machining (EDM) of polycrystalline diamond (PCD) was modelled by Ong et al. (2018) using a radial basis function neural network. A multi-objective moth search algorithm was then

applied to determine the optimal machining parameters, aiming to maximize the material removal rate and minimize the electrode wear rate simultaneously. The teaching–learning-based optimization (TLBO) and most of its modified versions have emerged as the popular optimizers to solve various traditional and modern machining processes, because these approaches do not require the fine tuning of any algorithm-specific control parameters. For instance, the simultaneous minimization of operation time incurred and carbon emission produced from a turning process were achieved with the multi-objective TLBO (MOTLBO) proposed by Lin et al. (2015). A nonlinear regression was used by Abhishek et al. (2017) to formulate the machining model of carbon fibre reinforced polymer (CFRP) composite using single point high-speed steel cutting tool. A prior method developed using TLBO was then applied to optimize four machining parameters known as the spindle speed, feed rate, depth of cut and fibre orientation angle, aiming to achieve satisfactory machining performances in terms of material removal rate, surface roughness and cutting force. A non-dominated sorting TLBO (NSTLBO) was designed to find the optimum parameters settings of four non-traditional machining processes of FIB micro-milling, laser cutting, ECM and WEDM (Rao et al. 2018). An enhanced multi-objective TLBO (EMOTLBO) was proposed by Natarajan et al. (2018) to achieve the maximization of material removal rate and the minimization of surface roughness for the machining process of Delrin material. Apart from the machining process, other practical applications of MOTLBO and its variants can be found in existing literatures (Sultana and Roy 2014; Rao and Waghmare 2015; Yu et al. 2015; Li et al. 2016; Rao et al. 2016, 2017a).

This research aims to address the multi-response optimization of machining of PTFE. Given its overwhelming demands for industrial applications, substantial works related to investigation of mechanical and wear properties of unfilled PTFE and reinforced PTFE have been conducted. Nevertheless, the investigation on the machining characteristics of PTFE has yet to be conducted based on the best of the authors' knowledge. In order to bridge this research gap, the first major contribution of this paper, as presented in Sect. 2, focused on the experimental design, derivation of regression model of PTFE based on the experimental results. The second major contribution (in Sect. 3) focused on designing an enhanced version of multi-objective algorithm known as the non-dominated sorting modified teaching–learning-based optimization (NSMTLBO), aiming to

optimize the machining parameters of PTFE that respond to the minimum surface roughness and maximum material removal rate simultaneously. Some modifications were incorporated into the proposed NSMTBLO in order to solve the MOPs effectively. For instance, a teacher selection mechanism based on the nearest Euclidean distance and the derivation of weighted mean position for each learner were introduced into the modified teacher phase to enhance the exploitation search capability of algorithm. A self-learning mechanism was also integrated into the modified learner phase, aiming to enhance the exploration search of algorithm through a probabilistic mutation. Finally, a fuzzy decision maker (Aghaei et al. 2011) was also applied to determine the most suitable machining parameters from the Pareto front obtained, based on the relative importance of all objective functions that may change frequently. Comprehensive simulation and experimental studies were performed and discussed in Sect. 4, followed by the concluding remarks in Sect. 5.

Experimental design and modelling of PTFE

Experimental details

The virgin PTFE rod in 30 mm diameter, prepared by sintering process was used for experimental analysis. The virgin PTFE is a pure and higher grade material which does not have recycled or reclaimed materials. It is in white colour and preferred to use in industries for its hygienic, better frictional resistance, excellent chemical resistance, outstanding electrical properties and excellent flexural properties. Table 1 shows the physical and mechanical properties of the material. Three levels of cutting speed (V_c), feed rate (f), depth of cut (ap) and nose radius (N_r) were used in design of experiments (DOE) to design L_{27} orthogonal array (4×27). The nose radius of $N_r = 0.4, 0.8$ and 1.2 mm as per the guidance from the cutting tool manufacture was used. The machining parameters and levels used for DOE is shown in Table 2. The surface roughness (R_a) and material removal rate (MRR) were considered as response variables. The CNC turning centre (Model: Sprint 16TC FANUC 0i T Mate Model C) was used for machining. The experiments were carried out with servo super cut coolant 32t according to DOE. The time taken for each pass of the required length was recorded. Each sample's surface roughness was instantaneously measured

Table 1 Mechanical properties of PTFE

Tensile strength (MPa)	Tensile modulus (MPa)	Elongation (%)	T_g (°C)	T_m (°C)	Water absorption (%)	Water contact angle (°)
14–35	400	200–400	–10	327	0	110

Table 2 Three levels of input machining parameters

Machining parameters	Level		
	I	II	III
Cutting speed, V_c (m/min)	80	120	160
Feed rate, f (mm/revolution)	0.1	0.3	0.5
Depth of cut, ap (mm)	0.5	0.75	1.0
Nose radius, N_r (mm)	0.4	0.8	1.2

with Mitutoyo make surf tester. Four trials were done for each experiment and the average of the measurements was recorded. The MRR value was computed by referring to the rate of material's volume removed from the sample as specified in Eq. (1) shown below:

$$MRR = \frac{\frac{\pi}{4}(D_{initial} - D_{final})^2 \times L}{T} \quad (1)$$

where MRR refers to material removal rate (cm^3/min); $D_{initial}$ and D_{final} refer to the initial and final diameters of PTFE sample (cm) before and after machining processes, respectively; L refers to the length of cut (cm); T refers to the time taken to cut the sample (min). The experimental design involved three (3) center runs which vary from low to high (-1 is low, 0 is medium and $+1$ is high). The values -1 , 0 and 1 represent the first level, second level and third level of each control parameter respectively. Figure 1 shows the machined sample, while Table 3 presents the control parameters and the corresponding response variables for each experiment conducted.

Fig. 1 Machined PTFE sample

Response surface methodology (RSM)

Response surface methodology (RSM) is a common modeling technique used to investigate the effects of several input control variables and one or more output response variables. In this study, RSM was used to explore the relationship between four input machining parameters known as cutting speed (V_c), feed rate (f), depth of cut (ap) and nose ratio (N_r) on two output response variables of surface roughness (R_a) and material removal rate (MRR).

Without loss of generality, a second-degree polynomial RSM model is represented as follows:

$$Y = \alpha_0 + \sum_{i=1}^I \beta_i X_i + \sum_{i=j}^I \beta_{ij} X_i X_j + \sum_{i=1}^I \beta_{ii} X_i^2 \quad (2)$$

where Y represents the response term; α_0 represents the free term of regression equation; X_1, X_2, \dots, X_I represent the set of control variable terms; β_i , β_{ii} and β_{ij} refer to the linear coefficient, quadratic coefficient and interacting coefficient terms, respectively. In this research, the response variable Y is associated with the four control variables, viz. V_c, f, ap and N_r , hence the function denoted as $\psi(\cdot, \cdot, \cdot, \cdot)$ is,

$$Y = \psi(V_c, f, ap, N_r) \quad (3)$$

Let $\psi_1(\cdot, \cdot, \cdot, \cdot)$ and $\psi_2(\cdot, \cdot, \cdot, \cdot)$ be the operators used to return the output response variables of R_a and MRR , respectively, based on the four control variables of V_c, f, ap and N_r , where:

$$R_a = \psi(V_c, f, ap, N_r) \quad (4)$$

$$MRR = \psi(V_c, f, ap, N_r) \quad (5)$$

Table 3 Control parameters and experimental results of PTFE machining

Run no.	Coded parameters				Actual parameters				Responses				
	Cutting speed (V_c)	Feed rate (f)	Depth of cut (a_p)	Nose radius (N_r)	Cutting speed (V_c)	Feed rate (f)	Depth of cut (a_p)	Nose radius (N_r)	Feed rate (f)	Depth of cut (a_p)	Nose radius (N_r)	Surface roughness (R_a)	Material removal rate (MRR)
Units	mm/min	mm/rev	mm	mm	mm/min	mm/rev	mm	mm	mm/rev	mm	mm	μm	cm^3/min
1	0	0	1	-1	120	0.3	1	0.4	0.3	1	0.4	2.26	46.92
2	0	0	0	0	120	0.3	0	0.8	0.3	0.75	0.8	2.67	32.04
3	-1	1	0	0	80	0.5	0	0.8	0.5	0.75	0.8	3.19	48.27
4	1	0	0	1	160	0.3	0	1.2	0.3	0.75	1.2	2.69	36.83
5	1	0	0	0	160	0.3	0	0.8	0.3	0.75	0.8	2.85	35.27
6	1	0	0	-1	160	0.3	0	0.4	0.3	0.75	0.4	2.46	35.63
7	0	1	0	-1	120	0.5	0	0.4	0.5	0.75	0.4	2.19	59.39
8	-1	0	-1	0	80	0.3	-1	0.8	0.3	0.5	0.8	1.76	23.72
9	-1	0	0	1	80	0.3	0	1.2	0.3	0.75	1.2	2.14	19.21
10	1	1	0	0	160	0.5	0	0.8	0.5	0.75	0.8	2.32	55.01
11	1	-1	0	0	160	0.1	0	0.8	0.1	0.75	0.8	3.22	25.66
12	-1	0	0	-1	80	0.3	0	0.4	0.3	0.75	0.4	2	17.21
13	0	0	-1	-1	120	0.3	-1	0.4	0.3	0.5	0.4	1.89	22.26
14	0	1	0	1	120	0.1	0	1.2	0.1	0.75	1.2	1.9	30.36
15	0	1	-1	0	120	0.1	-1	0.8	0.1	0.5	0.8	2.48	47.48
16	0	0	0	0	120	0.3	0	0.8	0.3	0.75	0.8	2.69	32.04
17	0	1	1	0	120	0.5	1	0.8	0.5	1	0.8	2.96	96.46
18	0	1	-1	0	120	0.5	-1	0.8	0.5	0.5	0.8	2.23	37.14
19	1	0	1	0	160	0.3	1	0.8	0.3	1	0.8	2.29	55.1
20	-1	0	1	0	80	0.3	1	0.8	0.3	1	0.8	2.54	47.57
21	0	0	0	0	120	0.3	0	0.8	0.3	0.75	0.8	2.15	32.04
22	0	-1	1	0	120	0.1	1	0.8	0.1	1	0.8	1.77	26.87
23	0	1	0	1	120	0.5	0	1.2	0.5	0.75	1.2	2.84	48.34
24	0	0	1	1	120	0.3	1	1.2	0.3	1	1.2	2.1	42.81
25	0	0	-1	1	120	0.3	-1	1.2	0.3	0.5	1.2	2.43	38.15
26	0	-1	0	-1	120	0.1	0	0.4	0.1	0.75	0.4	2.18	14.13
27	-1	-1	0	0	80	0.1	0	0.8	0.1	0.75	0.8	1.54	13.37

The regression models represented by Eqs. (4) and (5) are the second-degree polynomial regression model which can be determined with RSM-Box-Behnken design-uncoded units in Design Expert software. The coefficients of $\alpha_0, \beta_i, \beta_{ii}$ and β_{ij} in Eq. (2) were determined from the measured experimental data, which further used for obtaining quadratic regression models (Mia 2018; Chabbi et al. 2017; Natarajan et al. 2018). The quadratic regression models of R_a and MRR are expressed as:

$$R_a = 5.95983 + 0.057355 \times V_c + 2.93750 \times f + 7.43219 \times ap + 2.99103 \times N_r - 0.079688 \times V_c \times f - 0.036660 \times V_c \times ap + 7.20000 \times f \times ap + 2.90625 \times f \times N_r - 1.75000 \times ap \times N_r - 2.49455 \times ap^2 - 1.44817 \times N_r^2 \tag{6}$$

$$MRR = 124.84157 + 0.16068 \times V_c - 298.32469 \times f - 333.61027 \times ap + 67.27500 \times N_r + 399.65000 \times f \times ap - 85.25000 \times f \times N_r - 50.00000 \times ap \times N_r + 240.99254 \times f^2 + 193.44869 \times ap^2 \tag{7}$$

Problem formulation of PTFE machining

The objective functions for solving the multiobjective optimization machining optimization of PTFE were developed based on the regression models stated in Eqs. (6) and (7). The four input parameters to be optimized in the multiobjective machining model of PTFE include the cutting speed, feed rate, depth of cut and nose ratio denoted as $V_c, f, ap,$ and $N_r,$ respectively. Meanwhile, the surface roughness and material removal rate, represented by R_a and $MRR,$ respectively, are two contradict objective functions to be optimized because these output responses imply the quality and quantity of products, respectively.

The mathematical formulation of multi-response optimization of PTFE machining process was presented in Eq. (8). The optimal parameters of $V_c, f, ap,$ and N_r were solved using the proposed NSMTLBO as explained in the following sections. For sake of convenience, the maximization problem of MRR was converted into the minimization of negative MRR so that both objective functions can be expressed as the minimization problems.

$$\begin{cases} \text{Minimize } R_a \\ \text{Maximize } MRR \end{cases}_{V_c, f, ap, N_r} \tag{8}$$

s.t. $80 \text{ mm/minute} \leq V_c \leq 200 \text{ mm/minute}$
 $0.1 \text{ mm/revolution} \leq f \leq 0.5 \text{ mm/revolution}$
 $0.5 \text{ mm} \leq ap \leq 1.0 \text{ mm}$
 $0.4 \text{ mm} \leq N_r \leq 1.2 \text{ mm}$

Proposed methodology

Teaching–learning-based optimization

The teaching–learning-based optimization (TLBO) algorithm was inspired by the classical teaching and learning processes in school (Rao et al. 2011). A group of learners with the population size of N are randomly initialized at the beginning stage of optimization. Each of the n -th learner, denoted as $X_n = [X_{n,1}, \dots, X_{n,d}, \dots, X_{n,D}]$, is assumed to be the candidate solution of a given optimization problem, where $d \in [1, D]$ and D represent the index and the total dimensional size of problem, respectively. Let $\Psi(X_n)$ be the objective function value of each n -th solution and it is used to represent the knowledge level of each n -th learner in a classroom that can be enhanced via the teacher or learner phases.

During the teacher phase, all learners can improve their knowledge by learning from the teacher denoted as $X^{teacher}$, i.e., an individual with the best fitness in population, by considering the average knowledge level of all learners in the population represented as X^{mean} , where:

$$X^{mean} = \frac{1}{N} \sum_{n=1}^N X_n \tag{9}$$

Let $r_1 \in [0, 1]$ be a random number obtained from uniform distribution; $T_f \in \{1, 2\}$ be a teaching factor used to emphasize the importance of learner’s average quality X^{mean} . During the teacher phase, a new solution of n -th learner, denoted as X_n^{new} , can be obtained as:

$$X_n^{new} = X_n + r_1 (X^{teacher} - T_f X^{mean}) \tag{10}$$

For learner phase, each of the n -th learner aims to improve its knowledge level (i.e., fitness) by interacting with its peers in the population. Let s be the index of a learner randomly selected by the n -th learner from population for peer learning, where $s \in [1, N]$ and $s \neq n$; $r_2 \in [0, 1]$ is a random number generated from the uniform distribution. If the learner X_s that is randomly selected from the population has better fitness than X_n , the latter learner is attracted towards the former one as shown in Eq. (11). Otherwise, a repel mechanism is introduced in Eq. (12) to prevent the convergence of a learner X_n towards the peer learner X_s with inferior fitness.

$$X_n^{new} = X_n + r_2 (X_s - X_n) \tag{11}$$

$$X_n^{new} = X_n + r_2 (X_n - X_s) \tag{12}$$

If the new solution X_n^{new} generated from the teacher or learner phases has better fitness than the n -th learner, the latter solution will be replaced by the former one. Both of the

teacher and learner phases are repeated in until the termination criteria of TLBO are met. The teacher solution $X^{teacher}$ is considered as best solution obtained from the optimization process.

Non-dominated sorting modified teaching–learning-based optimization

In this section, the non-dominated sorting modified teaching–learning-based optimization (NSMTLBO) is proposed as an improved posterior approach to solve the multi-objective optimization problems. The modified versions of teacher phase and learner phase are introduced into NSMTLBO to enhance its exploration and exploitation capabilities while searching for the optimal solutions in the parameter spaces. Two core concept known as the fast non-dominated sorting and crowding distance (Deb et al. 2002) are also incorporated into NSMTLBO to ensure that the proposed work can solve multi-objective optimization problems effectively and efficiently. Without loss of generality, the minimization problems are considered in the remaining subsections, while explaining the search mechanisms of NSMTLBO in detail here.

Pareto dominance concept

Suppose that a total M objective functions are considered in a given multi-objective optimization problem. Let $\Psi_m(X_n)$ be the objective function value of the n -th learner X_n for the m -th objective function. For a single objective optimization, it is intuitive to compare the quality of different solutions based on its objective function values. Nevertheless, it is nontrivial to rank all solutions from the best to worst for multi-objective optimization problems due to the presence of multiple conflicting objectives.

Pareto dominance concept is a commonly used approach to overcome the challenges of multi-objective optimization problems (Deb et al. 2002). For the self-contained purpose, the definitions of Pareto dominance are presented as follows:

Definition 1 A solution X_n is considered to dominate another solution X_s if X_n shows better or equal objective values on all objectives and produce a better result in at least one objective function. In other words, $X_n > X_s$ if and only if $\Psi_i(X_n) \leq \Psi_i(X_s)$ for $\forall i \in \{1, 2, \dots, M\}$ and $\Psi_j(X_n) < \Psi_j(X_s)$ for $\exists j \in \{1, 2, \dots, M\}$, or vice versa.

Definition 2 A solution denoted as X^* is considered as a Pareto optimal solution if there is no existence of another solution X such that $X^* > X$. A Pareto optimal set can be determined from these Pareto optimal solutions.

Definition 3 Pareto front can be obtained by mapping the Pareto optimal set in the objective space.

Fast non-dominated sorting and crowding distance

An initial population of NSMTLBO denoted as $\mathbf{P} = [X_1, \dots, X_n, \dots, X_N]$ consisting of N learners is randomly generated. Similar with majority posteriori version of multi-objective algorithms, NSMTLBO aims to generate an approximated Pareto front consisting of a set of non-dominated solutions that are evenly distributed. In order to solve the multi-objective optimization problems effectively and efficiently, both of the fast non-dominated sorting approach and the crowding distance mechanism are incorporated into NSMTLBO to determine the ranking of all learners.

The fast non-dominated sorting approach is employed in NSMTLBO to ensure that only non-dominated solutions are selected to guide the learners during the search process in order to ensure the population can converge towards the true Pareto front. During the fast non-dominated sorting process, two entities are computed for each n -th learner X_n , i.e., (i) the domination count denoted as C_n to indicate the number of solutions that dominate the learner X_n and (ii) a set denoted as S_n to store the solution sets that are dominated by the learner X_n . All non-dominated solutions with $C_n = 0$ found in the first round of sorting process are allocated in the first front denoted as F_1 . For every n -th learner with $C_n = 0$ stored in the first front F_1 , each of the s -th learner X_s stored in the set S_n is visited to reduce its domination count by one. If the domination count C_s of any s -th learner is reduced to zero, it is stored into a new list denoted as Q to form the second front F_2 . The same process is repeated for all members stored in list Q to identify the third front F_3 . From Fig. 2, the fast non-dominated sorting process continues until the non-domination rank $\mathbf{Rank} = [Rank_1, \dots, Rank_n, \dots, Rank_N]$ and fronts $\mathbf{F} = [F_1, \dots, F_r, \dots, F_R]$ of all N learners are found, where r and R represent the index and upper limit of front counters, respectively.

In order to prevent premature convergence, it is crucial for NSMTLBO to maintain the diversity of its Pareto optimal solution sets by choosing the non-dominated solutions from sparse regions as teacher in guiding the search process of remaining learners. The crowding distance (Deb et al. 2002) denoted as Δ_n is a metric used to measure the density of solution X_n around the n -th learner in population by calculating the average distance of two nearest solutions on both sides of X_n along each of the M objectives. Suppose that $|F_r|$ is to the number of members allocated to the r -th front. The initial crowding distance of each a -th member is set as $\Delta_{a,r} = 0$, where $a = 1, \dots, |F_r|$. For every m -th objective function, a sorting process is performed on all members allocated to the r -th front in ascending manner by referring to their respective objective values and a list represented as $L_{m,r}$

Algorithm 1: [F, R]= Fast Non Dominated Sorting (P)	
1: for $n = 1$ to N do	
2: $C_n = 0$ and $S_n = \emptyset$;	
3: end for	
4: Reset the first front as $F_1 = \emptyset$;	
5: for $n = 1$ to N do	
6: for $s = n + 1$ to N do	
7: if $X_n \succ X_s$ then	<i>/*X_n dominates X_s*/</i>
8: $C_s = C_s + 1$;	<i>/*Increase the domination count of X_s*/</i>
9: $S_n = S_n \cup \{X_s\}$;	<i>/*Add X_s to the solution set S_n*/</i>
10: else	<i>/*X_s dominates X_n*/</i>
11: $C_n = C_n + 1$;	<i>/*Increase the domination count of X_s*/</i>
12: $S_s = S_s \cup \{X_n\}$;	<i>/*Add X_n to the solution set S_s*/</i>
13: end if	
14: end for	
15: if C_n is equal to 0 then	<i>/*X_n is not dominated by any learners*/</i>
16: $Rank_n = 1$;	<i>/*Assign the rank of X_n*/</i>
17: $F_1 = F_1 \cup \{X_n\}$;	<i>/*X_n is stored into the first front*/</i>
18: end if	
19: end for	
20: $r = 1$;	<i>/*Initialize the front counter*/</i>
21: while $F_r \neq \emptyset$ do	
22: $Q = \emptyset$;	<i>/*Clear the list Q*/</i>
23: for $n = 1$ to $ F_r $ do	<i>/*F_r = number of members in F_r*/</i>
24: for $s = 1$ to $ S_n $ do	<i>/*S_n = number of members in S_n*/</i>
25: $C_s = C_s - 1$;	<i>/*Reduce the domination count of X_s*/</i>
26: if C_s is equal to 0 then	
27: $Rank_s = r + 1$;	<i>/*Assign the rank of X_s*/</i>
28: $Q = Q \cup \{X_s\}$;	<i>/*Stored X_s into the next front*/</i>
29: end if	
30: end for	
31: end for	
32: $r = r + 1$;	<i>/*Update the front counter*/</i>
33: $F_r = Q$;	<i>/*Assign members of list Q into the next front*/</i>
34: end while	

Fig. 2 The pseudo-code of fast non-dominated sorting approach

is used to store all sorted members. Suppose that for the r -th front, the a -th member is sorted to become the j -th element denoted as $L_{m,r}[j]$ for the m -th objective, where

$j = 1, \dots, |F_r|$. Then, the crowding distance for every j -th sorted member with the objective function value denoted as $\Psi_m(X_{L_{m,r}[j]})$ can be calculated as:

$$\Delta_{L_{m,r}[j]} = \begin{cases} \infty, & \text{if } j = 1 \text{ or } j = |F_r| \\ \Delta_{L_{m,r}[j]} + \frac{\Psi_m(X_{L_{m,r}[j+1]}) - \Psi_m(X_{L_{m,r}[j-1]})}{\Psi_m(X_{L_{m,r}[|F_r|]}) - \Psi_m(X_{L_{m,r}[1]})}, & \\ \infty, & \text{if } j = 2, \dots, (|F_r| - 1) \end{cases} \quad (13)$$

From Eq. (13), the front members with larger crowding distances are located on the more sparse regions of objective space and vice versa. For every m -th objective function, the boundary solutions stored in the sorted list $L_{m,r}$ are assigned with infinite crowding distance values. Similar procedures are repeated for all M objective functions and all R fronts as shown in Fig. 3. A set denoted as $\Delta = [\Delta_1, \dots, \Delta_n, \dots, \Delta_N]$ used to store the overall crowding-distance value of each n -th learner is then computed as the sum of individual distance values corresponding to each m -th objective function value.

Referring to the non-domination rank $Rank_n$ and the crowding distance Δ_n of each n -th learner, a crowding-comparison operator denoted as $<_{cco}$ is used to compare the superiority of two solutions during the selection process so that a uniformly spread-out optimal Pareto front can be generated to solve the given multi-objective optimization problem. From Eq. (14), if both learners of X_n and X_s have different non-domination ranks, X_n is preferred over X_s if the former solution has better (i.e., lower) rank than the latter one. If both learners belong to the same rank, the solution

located in the less crowded regions with larger Δ_n value is preferred as follow:

$$X_n <_{cco} X_s, \text{ if } (Rank_n < Rank_s) \text{ or } (Rank_n = Rank_s) \text{ and } (\Delta_n > \Delta_s) \quad (14)$$

Modified teacher phase

During the teacher phase of TLBO, each of the n -th learner adjusts its position X_n based on the differences between the teacher $X^{teacher}$ and the mainstream knowledge of population denoted as a mean value X^{mean} as shown in Eq. (10). For single objective optimization problem, the selection of teacher is intuitive and can be achieved by identifying the learner with best objective function value from population. On the other hand, the existence of mutually contradictory objectives in the multi-objective optimization problems tends to produce multiple non-dominated learners that are able to guide other learners as teachers in searching for the Pareto front.

A modified teacher selection scheme is introduced for teacher phase of NSMTLBO, aiming to leverage the expertise of multiple teachers in solving the multi-objective optimization problems. In contrast to the conventional TLBO where a same teacher is assigned to guide the search processes of all learners, the nearest non-dominated learner stored in the Pareto front F_1 is assigned as the teacher for every n -th learner during the teacher phase of NSMTLBO. Let $|F_1|$ be the number of candidate teachers contained in the Pareto front F_1 and D represents total numbers of design

Algorithm 2: $\Delta =$ Crowding Distance Calculation (F)	
1:	Calculate total number of fronts from F as R ;
2:	for $r = 1$ to R do
3:	for $m = 1$ to M do
4:	Perform sorting for all members in r -th front in ascending manner by referring to the values of m -th objective function and then store these sorted members into the list of $L_{m,r}$;
5:	for $j = 1$ to $ F_r $ do
6:	Determine the crowding distance value of every j -th sorted member in the list $L_{m,r}$ using Eq. (13);
7:	end for
8:	end for
9:	for $a = 1$ to $ F_r $ do
10:	Calculate the overall crowding distance value $\Delta_{a,r}$ of every a -th member by summing the individual distance values;
11:	Identify the n -th learner that is corresponding to the a -th member stored in the r -th front and assign its crowding distance Δ_n as $\Delta_{a,r}$;
12:	end for
13:	end for

Fig. 3 The pseudo-code of computing the crowding distance value of each learner

variables that need to be optimized. Let X_d^U and X_d^L be the d -th dimensional component of the upper and lower limits, respectively, where $d = 1, \dots, D$. Denote $X_{a,d}^{Cand}$ and $X_{n,d}$ as the d -th dimension of the a -th candidate teacher and n -th learner, where $a = 1, \dots, |F_1|$ and $n = 1, \dots, N$, respectively. Then, the normalized Euclidean distance between the n -th learner and the a -th candidate teacher represented with $E_{n,a}$ are computed as follow:

$$E_{n,a} = \sqrt{\sum_{d=1}^D \left(\frac{X_{a,d}^{Cand} - X_{n,d}}{X_d^U - X_d^L} \right)^2} \tag{15}$$

As shown in Fig. 4, the computation of normalized Euclidean distances between the n -th learner and all $|F_1|$ candidate teachers are repeated using Eq. (15). Based on the $E_{n,a}$ values, the closest candidate teacher denoted as $X_n^{teacher}$ is identified as the teacher for the n -th learner during the teacher phase. The adopted teacher selection mechanism is anticipated to enhance the exploitation capability of NSMTLBO because each learner is guided by its closest teacher. The diversity of Pareto front can also be preserved due to the employment of different teachers in guiding different learners during the teacher phase of NSMTLBO.

For conventional TLBO, the contributions of all learners are equal in determining the mean value of population X^{mean} . It is anticipated that the conventional TLBO is susceptible to premature convergence because the same values of $X^{teacher}$ and X^{mean} are used to update all of the learners in each generation. If the most knowledgeable teacher is trapped in a local optimum, it tends to misguide the rest of learners searching towards the local optimum and produce the same mean position X^{mean} . When both of the $X^{teacher}$ and X^{mean} becomes undistinguishable, the position of learners cannot be updated and stuck at local optimum because the second term of Eq. (10) tends to approach zero value when teaching factor obtained by all learners are $T_f = 1$.

In order to address this issue, an alternate approach in obtaining the mean knowledge of population is proposed for the teacher phase of NSMTLBO. Unlike TLBO, the

proposed NSMTLBO only leverages the benefits of useful information carried by the non-dominated learners stored in Pareto front F_1 as the source of influences in guiding the search process of other learners. In order to prevent the diversity loss in second component of Eq. (10), each a -th non-dominated learner from F_1 is assigned with different weightage to reflect its unique contribution in constructing the weighted mean position of the n -th learner. Denote $r_a \in [0, 1]$ as the random number obtained from a uniform distribution for each a -th non-dominated learner in F_1 with the position value of X_a . Suppose that \tilde{X}_n^{mean} represents the weighted mean position of n -th learner, then

$$\tilde{X}_n^{mean} = \frac{\sum_{a=1}^{|F_1|} r_a X_a}{\sum_{a=1}^{|F_1|} r_a} \tag{16}$$

Given the closest teacher $X_n^{teacher}$ and weighted mean \tilde{X}_n^{mean} , the new position of each n -th NSMTLBO learner is updated through the modified teacher phase, where

$$X_n^{new} = X_n + r_3 (X_n^{teacher} - T_{f1} X_n) + r_4 (\tilde{X}_n^{mean} - T_{f2} X_n) \tag{17}$$

where r_3 and r_4 are the random number with the range of 0 to 1 generated from uniform distribution; T_{f1} and T_{f2} are the teaching factors with the range of 1 to 2 generated from uniform distribution, where $T_{f1}, T_{f2} \in [1, 2]$ (Rao and Patel 2013; Lin et al. 2015). As shown in Fig. 5 and Eq. (17), each of the n -th NSMTLBO learner can learn directly from multiple teachers by updating its knowledge based on the differences between its current knowledge and (i) the closest teacher’s knowledge and (ii) weighted mean knowledge of other teachers in classroom. These uniform distributions of T_{f1} and T_{f2} are used to emulate a more realistic classroom paradigm by generating the learners with different levels of learning tendency from the teachers. Particularly, the NSMTLBO learner is assumed to learn everything from the teachers successfully when T_{f1} and T_{f2} are equal to 2, while the learner fails to learn anything when $T_{f1} = T_{f2} = 1$ (Rao and Patel 2013; Lin et al. 2015).

<p>Algorithm 3: $X_n^{teacher} = \text{Find_Closest_Teacher} (F_1, X_n, n, X^U, X^L)$</p> <ol style="list-style-type: none"> 1: Calculate total number of candidate teachers stored in F_1 as F_1; 2: for $a = 1$ to F_1 do <li style="padding-left: 20px;">3: Calculate the distance of $E_{n,a}$ between the n-th learner and a-th candidate teacher using Eq. (15); 4: end for 5: Identify the candidate teacher that is closet (i.e., smallest $E_{n,a}$) to the n-th learner and return it as $X_n^{teacher}$;
--

Fig. 4 The pseudo-code of finding the closest teacher for each learner in the teacher phase

Fig. 5 The pseudo-code of modified teacher phase in NSMTLBO

Algorithm 4: $X_n^{new} = \text{Modified_Teacher_Phase}(F_1, X_n, n, X^U, X^L)$

- 1: $X_n^{teacher} = \text{Find_Closest_Teacher}(F_1, X_n, n, X^U, X^L);$ /*Algorithm 3*/
- 2: Calculate the weighted mean \tilde{X}_n^{mean} of each n -th learner using Eq. (16);
- 3: Randomly generate T_{f1} and T_{f2} where $T_{f1}, T_{f2} \in [1, 2]$;
- 4: Update the new position X_n^{new} of n -th learner using Eq. (17);

Modified learner phase

The learner phase of TLBO promotes the exploration search of algorithm by introducing the repelling mechanism in Eq. (12) when the n -th learner is randomly assigned to an inferior peer. As the iterative generation of TLBO becomes larger, the changes in population tends to be stabilized because all learners tend to converge towards a particular region of search space. Without the notable changes of population diversity, it becomes challenging for a randomly selected learner to assist the particular learner to break out from sub-optimal region, especially if the optimization problem consists of a complicated Pareto front. It is also more difficult for the learner in a multi-objective algorithm to randomly choose another peer that is completely dominated by it, hence the likelihood of triggering repelling mechanism in Eq. (12) is reduced further.

In order to address these challenges, a modified learner phase is proposed for NSMTLBO. A probabilistic mutation scheme is first designed as an alternative learning strategy in the learner phase of NSBTLBO, aiming to perturb the learners with a mutation probability represented as $P_{mut} = 1/D$. From the perspective of classroom teaching–learning paradigm, different types of learners exist in the same classroom and some learners might prefer to improve their knowledge level via self-learning rather than interacting with their peers. The inclusion of this mutation operator is to facilitate the self-learning of some NSMTLBO learners during the learner phase after they complete the teacher phase. When the self-learning mechanism is triggered for the n -th learner with probability P_{mut} ,

a random dimensional component of n -th learner, denoted as $d_r \in [1, D]$, is selected for perturbation, where

$$X_{n,d_r}^{new} = X_{n,d_r} + r_5(X_{n,d_r}^U - X_{n,d_r}^L) \tag{18}$$

where $r_5 \in [-1, 1]$ is a random number obtained from uniform distribution; X_{n,d_r}^{new} , X_{n,d_r}^U and X_{n,d_r}^L are the d_r -th dimensional component of the n -th learner, the upper and lower boundary limits of decision variables, respectively. The pseudocode of self-learning mechanism is described in Fig. 6.

For the n -th learner that prefer to improve its knowledge by interacting with other peers, a random peer learner denoted as X_s is chosen from population to update X_n using Eq. (11) or Eq. (12), where $n, s \in [1, N]$ and $s \neq n$. If the randomly chosen peer learner X_s dominates the learner X_n , the latter learner is attracted towards the former one as shown in Eq. (11). On the other hand, the learner X_n tends to be driven away from the inferior peer learner X_s using Eq. (12) to prevent the premature convergence of population in sub-optimal regions of fitness landscapes. If both of the learner X_n and peer X_s are non-dominated with each other, one of the learning strategy expressed using Eq. (11) or Eq. (12) will be randomly selected with an equal chance for updating the n -th learner at the modified learner phase of NSMTLBO as shown in Fig. 7.

Fuzzy decision maker

After obtaining the Pareto front F_1 of a given multi-objective optimization problem, a fuzzy decision maker is then used by NSMTLBO to determine the most suitable solution

Fig. 6 The pseudo-code of self-learning mechanism in the modified learner phase

Algorithm 5: $X_n^{new} = \text{Self_Learning}(X_n, D, X^U, X^L)$

- 1: Produce a random dimension index denoted as $d_r \in [1, D]$;
- 2: Extract the d_r -th component of X_n , X^U , and X^L ;
- 3: Perform perturbation on X_{n,d_r} to produce X_{n,d_r}^{new} using Eq. (18);
- 4: Return X_n^{new} as the perturbed solution;

Fig. 7 The pseudo-code of modified learner phase in NSMTLBO

```

Algorithm 6:  $X_n^{new} = \text{Modified\_Learner\_Phase}(X_n, D, X^U, X^L, N, P_{mut})$ 
1: Randomly generate a number  $Rand \in [0, 1]$ ;
2: if  $Rand \leq P_{mut}$  then           /*the  $n$ -th learner prefers self-learning*/
3:    $X_n^{new} = \text{Self\_Learning}(X_n, D, X^U, X^L)$ ;           /*Algorithm 5*/
4: else           /*the  $n$ -th learner prefers interacting with other learners*/
5:   while  $n$  is equal to  $s$  do
6:     Randomly generate the peer learner index  $s \in [1, N]$  ;
7:     Compare the indices  $n$  and  $s$ ;
8:   end while
9:   Obtain  $X_s$  as the peer learner of  $X_n$  where  $s \neq n$  ;
10:  if  $X_s$  dominates  $X_n$  then
11:    Calculate  $X_n^{new}$  using Eq. (11);
12:  else if  $X_n$  dominates  $X_s$  then
13:    Calculate  $X_n^{new}$  using Eq. (12);
14:  else           /*  $X_n$  and  $X_s$  are non-dominated solutions*/
15:    Produce a random number denoted as  $Rand \in [0, 1]$  ;
16:    if  $Rand \leq 0.5$  then
17:      Calculate  $X_n^{new}$  using Eq. (11);
18:    else
19:      Calculate  $X_n^{new}$  using Eq. (12);
20:    end if
21:  end if
22: end if

```

from the Pareto front obtained by referring to the relative importance levels of all objective functions stated by the stakeholders.

Denote $\Psi^U = [\Psi_1^U, \dots, \Psi_m^U, \dots, \Psi_M^U]$ as the utopia point of a given multi-objective optimization problem consisting of M objective functions. Utopia point is a specific point located in objective space with all objective function values demonstrate the best possible values. On the other hand, the pseudo nadir point denoted as $\Psi^{SN} = [\Psi_1^{SN}, \dots, \Psi_m^{SN}, \dots, \Psi_M^{SN}]$ is another extreme point in objective space that consisting of the worst values for all objective functions. In order to select the most preferred optimal solution from Pareto front F_1 , the membership value of every a -th Pareto front member with $a = 1, \dots, |F_1|$ for the m -th objective function is computed by calculating the relative distances between the objective function value of a -th Pareto front member, denoted as $\Psi_m(X_a)$, and both of the utopia and pseudo nadir points in each m -th objective function. For any m -th objective function, if the a -th Pareto optimal solution is closer to the utopia point, the higher membership value denoted as μ_a^m is assigned to indicate the higher degree of optimality for the a -th Pareto front member in the m -th objective function and vice versa.

For minimization problem, the fuzzification process used to determine the membership value μ_a^m of each a -th Pareto front member with the objective function value $\Psi_m(X_a)$ in the m -th objective function is shown as follow:

$$\mu_a^m = \begin{cases} 1, & \Psi_m(X_a) < \Psi_m^U \\ \frac{\Psi_m^{SN} - \Psi_m(X_a)}{\Psi_m^{SN} - \Psi_m^U}, & \Psi_m^U \leq \Psi_m(X_a) \leq \Psi_m^{SN} \\ 0, & \Psi_m(X_a) > \Psi_m^{SN} \end{cases} \quad (19)$$

The value of μ_a^m for maximization problem is computed as:

$$\mu_a^m = \begin{cases} 0, & \Psi_m(X_a) < \Psi_m^{SN} \\ \frac{\Psi_m(X_a) - \Psi_m^{SN}}{\Psi_m^U - \Psi_m^{SN}}, & \Psi_m^{SN} \leq \Psi_m(X_a) \leq \Psi_m^U \\ 1, & \Psi_m(X_a) > \Psi_m^U \end{cases} \quad (20)$$

Assume that w_m represents the relative importance level of every m -th objective function given by stakeholders, while μ_a refers to the total degree of optimality for every a -th Pareto front member. Then,

$$\mu_a = \sum_{m=1}^M w_m \mu_a^m \tag{21}$$

From Fig. 8, the a -th Pareto front member with the largest value of μ_a is selected from F_1 as the most preferred Pareto optimal solution denoted as $X^{preferred}$ because it produces better optimization on all objective functions of a given multi-objective optimization problem as compared with other Pareto front members by referring to the relative importance of all objective functions.

The complete NSMTLBO algorithm

The complete procedures of NSMTLBO are presented in Fig. 9, where γ and Γ denote the counter of function evaluations and maximum fitness evaluation numbers, respectively. During the initialization stage of NSMTLBO, a population \mathbf{P} consists of N learners is randomly generated, followed by the function evaluation of these N learners in all M objective functions. The front number, non-domination rank and crowding distance of all learners are then computed, followed by the sorting of members of population \mathbf{P} from the best to worst by referring to these metrics.

During the search process, the new solution X_n^{new} of each n -th learner is generated via the modified teacher or learner phases and stored in an offspring population denoted as \mathbf{P}^{off} with the population size of N . A combined population

denoted as $\mathbf{P}^{comb} = \mathbf{P} \cup \mathbf{P}^{off}$ with the population size of $2N$ is constructed and all members of \mathbf{P}^{comb} are sorted from the best to worst based on their non-domination ranks and crowding distances. An operator denoted as $Trunc(\cdot, \cdot)$ is then applied to truncate the best N members with the lowest $Rank_n$ and largest Δ_n in \mathbf{P}^{comb} , aiming to create a new population of \mathbf{P} for the next phase or next generation. These procedures are repeated until the termination condition is satisfied, i.e., when the values of fitness evaluation counter exceed the maximum fitness evaluation numbers, i.e., $\gamma > \Gamma$. When the proposed NSMTLBO is terminated, the preferred optimal solution denoted as $X^{preferred}$ is determined from the Pareto front of F_1 with the fuzzy decision maker based on the relative importance level specified for each objective function.

Performance metrics

Two evaluation metrics are adopted in this study to compare the optimization performances of all algorithms in tackling the multi-response optimization of PTFE machining problem. The first metric used for performance comparison of two Pareto fronts is a coverage operator of two sets by computing the percentage of members of one Pareto front to be dominated by those of another Pareto front (Zitzler et al. 2000). Denote $C(\cdot, \cdot)$ as the coverage operator used to compare two Pareto fronts of F_1^A and F_1^B , then

Algorithm 7: $X^{preferred} = \text{Fuzzy_Decision_Maker} (F_1, M, w_m)$

- 1: Determine the number of non-dominated solutions stored in the Pareto front F_1 as $|F_1|$;
- 2: **for** $a = 1$ to $|F_1|$ **do**
- 3: **for** $m = 1$ to M **do**
- 4: Perform fitness evaluation of X_a using the m -th objective function;
- 5: **end for**
- 6: **end for**
- 7: Obtain the utopia point $\Psi^U = [\Psi_1^U, \dots, \Psi_m^U, \dots, \Psi_M^U]$ and the pseudo nadir point $\Psi^{SN} = [\Psi_1^{SN}, \dots, \Psi_m^{SN}, \dots, \Psi_M^{SN}]$ by referring to the best and worst values in all objective functions, respectively;
- 8: **for** $a = 1$ to $|F_1|$ **do**
- 9: **for** $m = 1$ to M **do**
- 10: Obtain the membership function value μ_a^m with Eq. (19) or Eq. (20);
- 11: **end for**
- 12: Obtain the total degree of optimality μ_a with Eq. (21);
- 13: **end for**
- 14: Identify X_a with the largest μ_a as the most preferred Pareto front member denoted as $X^{preferred}$;

Fig. 8 The pseudo-code of fuzzy decision maker

Algorithm 8: NSMTLBO

```

1: Initialize the fitness evaluation counter as  $\gamma = 0$  ;
2: for  $n = 1$  to  $N$  do
3:   Randomly generate the initial position of each  $n$ -th learner;
4:   Perform fitness evaluation for each  $n$ -th learner with  $M$  objective functions;
5:    $\gamma \leftarrow \gamma + M$  ;
6: end for
7: Store all learners into the current population  $\mathbf{P}$ ;
8:  $[\mathbf{F}, \mathbf{R}] = \text{Fast\_Non\_Dominated\_Sorting}(\mathbf{P})$ ; /*Algorithm 1*/
9:  $\Delta = \text{Crowding\_Distance\_Calculation}(\mathbf{F})$ ; /*Algorithm 2*/
10: Sort all members of  $\mathbf{P}$  from the best to worst based on  $\mathbf{R}$  and  $\Delta$  values;
11: while  $\gamma \leq \Gamma$  do
    /*Execute modified teacher phase using Algorithm 4*/
12:   for  $n = 1$  to  $N$  do
13:      $X_n^{new} = \text{Modified\_Teacher\_Phase}(F_1, X_n, n, X^U, X^L)$ ;
14:     Perform fitness evaluation on the  $n$ -th new solution with  $M$  objective functions;
15:      $\gamma \leftarrow \gamma + M$  ;
16:   end for
17: Store all new solutions into the offspring population  $\mathbf{P}^{off}$  ;
18:  $\mathbf{P}^{comb} = \mathbf{P} \cup \mathbf{P}^{off}$  ;
19:  $[\mathbf{F}, \mathbf{R}] = \text{Fast\_Non\_Dominated\_Sorting}(\mathbf{P}^{comb})$ ; /*Algorithm 1*/
20:  $\Delta = \text{Crowding\_Distance\_Calculation}(\mathbf{F})$ ; /*Algorithm 2*/
21: Sort all members of  $\mathbf{P}^{comb}$  from the best to worst based on  $\mathbf{R}$  and  $\Delta$  values;
22:  $\mathbf{P} \leftarrow \text{Trunc}(\mathbf{P}^{comb}, N)$ ; /*Truncate the  $N$  best members*/
23:   /*Execute modified learner phase using Algorithm 6*/
24:   for  $n = 1$  to  $N$  do
25:      $X_n^{new} = \text{Modified\_Learner\_Phase}(X_n, D, X^U, X^L, N, P_{mut})$ ;
26:     Perform fitness evaluation on the  $n$ -th new solution with  $M$  objective functions;
27:      $\gamma \leftarrow \gamma + M$  ;
28:   end for
29: Store all new solutions into the offspring population  $\mathbf{P}^{off}$  ;
30:  $\mathbf{P}^{comb} = \mathbf{P} \cup \mathbf{P}^{off}$  ;
31:  $[\mathbf{F}, \mathbf{R}] = \text{Fast\_Non\_Dominated\_Sorting}(\mathbf{P}^{comb})$ ; /*Algorithm 1*/
32:  $\Delta = \text{Crowding\_Distance\_Calculation}(\mathbf{F})$ ; /*Algorithm 2*/
33: Sort all members of  $\mathbf{P}^{comb}$  from the best to worst based on  $\mathbf{R}$  and  $\Delta$  values;
34:  $\mathbf{P} \leftarrow \text{Trunc}(\mathbf{P}^{comb}, N)$ ; /*Truncate the  $N$  best members*/
35: end while
36: Extract the first front  $F_1$  as the Pareto front;
37:  $X^{preferred} = \text{Fuzzy\_Decision\_Maker}(F_1, M, w_m)$ ; /*Algorithm 7*/

```

Fig. 9 The pseudo-code of complete NSMTLBO

$$C(F_1^A, F_1^B) = \frac{|\{b \in F_1^B; \exists a \in F_1^A : a \leq b\}|}{|F_1^B|} \quad (22)$$

If $C(F_1^A, F_1^B) = 1$, it implies that all members of the Pareto front F_1^B are dominated or perform equally to all members from F_1^A . In contrary, $C(F_1^A, F_1^B) = 0$ means that none of the solution in F_1^B are covered by those of F_1^A .

Notably, it is not guarantee that $C(F_1^A, F_1^B)$ must be equal to $1 - C(F_1^B, F_1^A)$, therefore it is crucial to obtain both results during performance comparisons.

Spacing metric (Coello et al. 2004) is the second metric used for performance comparison by measuring the uniformity of Pareto fronts distribution obtained by different algorithms. Suppose that M refers to the numbers of objective functions to be optimized, while $|F_1|$ represents the number of non-dominated solutions obtained from Pareto front. Then, the smallest Euclidean distance between the a -th and b -th Pareto optimal solutions for each m -th objective function in the objective space is:

$$d_a = \min_{a, a \neq b} \sum_{m=1}^M |\Psi_m(X_a) - \Psi_m(X_b)|, \quad a, b = 1, \dots, |F_1| \quad (23)$$

The average value of all d_a is then computed as:

$$\bar{d} = \frac{\sum_{a=1}^{|F_1|} d_a}{|F_1|} \quad (24)$$

Define S as the spacing measure to quantify the distance variance of neighbouring Pareto optimal solutions, then

$$S = \sqrt{\frac{1}{|F_1| - 1} \sum_{a=1}^{|F_1|} (\bar{d} - d_a)^2} \quad (25)$$

The spacing measure of $S = 0$ is the best possible value produced by a compared algorithm because it implies that all of the Pareto front members are equidistant from each other.

Experimental studies

In this section, the capability of the proposed PTFE regression model to fit in the given experimental data is first investigated. The quantitative and qualitative analyses of Pareto fronts produced by NSMTLBO and another six well-established algorithms are then conducted and compared through simulation studies. Based on the optimized machining parameters obtained from simulations, the performance deviations between the actual and simulated results were then investigated through further validation experiments in order to confirm the practicability NSMTLBO in solving the PTFE machining problem.

Evaluation of PTFE modelling using analysis of variances (ANOVA)

The goodness-of-fit of the measured data and model was done using a statistical tool known as ANOVA. SD , R^2 and P are the output parameters of ANOVA that represent the standard deviation, percentage of variation of data and significance of the control variable (adequacy), respectively. The parameter SD represents how far the data values fall from the fitted values, which seems to be an indicator to show how good of a model in describing the response. The percentage of data variation in the response is qualified using the parameter R^2 with the range of 0–100%, where greater values of R^2 implies for better fitting in modelling. The probability P is used to check if the tested parameters are essential in determining the responses of model. If $P < 5\%$, it indicates that the parameters are significant on responses of an adequate model. On the other hand, the parameter is concluded to be insignificant on the responses of an adequate model for $P > 5\%$.

The Box-Behnken uncoded units were used in this study and the 95% confidence interval was set to find the variance among input machining parameters (V_c , f , ap and N_r) and output responses R_a and MRR . The contribution of each process parameter on the output responses were calculated as shown in Tables 4 and 5. Table 6 shows the parameter R^2 which is used to assess the percentage of data variance to the response. It is seen that 97% variability of data of MRR and 95% variability data of surface roughness are around the mean. In other words, the higher R^2 shown above, reveals the strength of the relationship between the model and the dependent variable. It is the evidence of the fitness of the models.

Performance analysis of NSMTLBO in PTFE machining problem

The parameter settings of all involved multi-objective optimization algorithms considered in solving the proposed PTFE machining optimization problem are first presented. It is followed by the presentation and discussion of simulation results obtained by all algorithms. Both of the qualitative and quantitative analyses are used to compare the optimization performance demonstrated by each compared methods. Finally, the performance deviations between the simulated and experiment results obtained for the surface roughness and material removal rate are compared based on the selected optimized machining parameters.

Parameter settings for simulation and experimental studies

Extensive performance evaluation of NSMTLO was conducted by comparing it with another six well-established

Table 4 ANOVA results for modelling the surface roughness of PTFE

Source	Sum of squares	Degree of freedom	Mean square	F-value	P value	% Contribution	Remarks
Model	4.61	11	0.419	28.16	<0.0001		Significant
V_c	0.6365	1	0.6365	42.79	<0.0001	13.18	
f	0.5808	1	0.5808	39.04	<0.0001	12.02	
ap	0.0017	1	0.0017	0.1131	0.7413	0.04	
N_r	0.1045	1	0.1045	7.03	0.0182	2.16	
$V_c \times f$	1.63	1	1.63	109.28	<0.0001	33.75	
$V_c \times ap$	0.3567	1	0.3567	23.98	0.0002	7.39	
f and ap	0.5184	1	0.5184	34.85	<0.0001	10.73	
f and N_r	0.2162	1	0.2162	14.53	0.0017	4.48	
ap and N_r	0.1225	1	0.1225	8.23	0.0117	2.54	
ap^2	0.1485	1	0.1485	9.99	0.0065	3.07	
N_r^2	0.3477	1	0.3477	23.37	0.0002	7.20	
Residual	0.2231	15	0.0149				
LACK OF FIT	0.0357	13	0.0027	0.0293	1		
Pure error	0.1875	2	0.0937				
Cor total	4.83	26					

Table 5 ANOVA results for modelling the material removal rate of PTFE

Source	Sum of squares	Degree of freedom	Mean square	F-value	P-value	% Contribution	Remarks
Model	7608.96	9	845.44	67.66	<0.0001		Significant
V_c	484.8	1	484.8	38.8	<0.0001	6.20	
f	2905.99	1	2905.99	232.57	<0.0001	37.15	
ap	896.34	1	896.34	71.73	<0.0001	11.46	
N_r	33.87	1	33.87	2.71	0.118	0.43	
$f \times ap$	1597.2	1	1597.2	127.83	<0.0001	20.42	
f and N_r	186.05	1	186.05	14.89	0.0013	2.38	
ap and N_r	100	1	100	8	0.0116	1.28	
f^2	607.02	1	607.02	48.58	<0.0001	7.76	
ap^2	918.25	1	918.25	73.49	<0.0001	11.74	
Residual	212.42	17	12.5				
Lack of fit	212.42	15	14.16				
Pure error	0	2	0				
Cor total	7821.38	26					

Table 6 Fit statistics (R^2) values for material removal rate and surface roughness

Material removal rate, MRR	SD	3.53	R^2	0.9728
	Mean	37.75	Adjusted R^2	0.9585
	C.V. %	9.36	Predicted R^2	0.9269
Surface roughness, R_a	SD	0.122	R^2	0.9538
	Mean	2.36	Adjusted R^2	0.9200
	C.V. %	5.17	Predicted R^2	0.9261

algorithms. This included the multi-objective particle swarm optimization (MOPSO) (Coello et al. 2004), non-dominated sorting genetic algorithm II (NSGA-II) (Deb et al. 2002), multi-objective grey wolf optimizer (MOGWO) (Mirjalili et al. 2016), multi-objective teaching–learning based optimization (MOTLBO) (Lin et al. 2015), multi-objective improved teaching–learning based optimization (MO-ITLBO) (Patel and Savsani 2016) and non-dominated sorting teaching–learning based optimization (NSTLBO) (Rao et al. 2018). The concept of external archive is used by MOPSO, MOGOW and MO-ITLBO to store the non-dominated solution found during the optimization. Meanwhile,

both mechanisms of fast non-dominated sorting and crowding distance are incorporated into NSGA-II, MOTLBO and NSTLBO for determining the quality of solutions obtained in solving the multi-objective optimization problems. The excellent performances of these six selected algorithms in handling different multi-objective optimization problems were verified in their literatures, hence the performance comparison between NSMTLBO and these peer algorithms are expected to be convincing.

The parameter settings of all compared algorithms were tuned as shown in Table 7 based on the recommendations of their respective literatures. The inertia weight ω of MOPSO was set to be reduced from 0.9 to 0.4 in linear manner, while the parameter settings of both acceleration coefficients were given as $c_1 = c_2 = 2.05$. For MOPSO and MOGWO, the parameters of grid inflation and number of grid per dimension that are crucial in constructing the external archive were set as $\alpha = 0.1$ and $nGrid = 10$, respectively. The teaching factor T_f of NSTLBO was randomly generated as 1 or 2, while the NSGA-II has a crossover rate of $P_{cr} = 0.9$. A multiple group learning approach with the group number of $nGroup = 4$ was adopted by MO-ITLBO during the teaching phase and its archive was managed by epsilon-dominance method with $\epsilon = 0.007$ (Patel and Savsani 2016). For NSMTLBO,

the mutation probability were set similar as MOPSO and NSGA-II, i.e., $P_{mut} = 1/D$ (Deb et al. 2002; Coello et al. 2004). The teaching factors of MOTLBO, MO-ITLBO and NSMTLBO were generated as the values between 1 and 2 to show the different capabilities of learners to improve their knowledge level from the teachers.

Different population sizes of $N = 20, 30$ and 40 were set to investigate the effect of population sizes on the optimization performances of all compared algorithms based on the recommendation of selected previous works (Natarajan et al. 2018; Rao et al. 2017b). For MOPSO, MOGWO and MO-ITLBO, the archive size $|A|$ was set equal to N . Based on most official documents that specify the rules of competition among optimization algorithms the same maximum fitness evaluation number needs to be used as the termination condition of all compared algorithms for the sake of fair performance comparison (Suganthan et al. 2005; K. Tang et al. 2010; Liang et al. 2013). In this study, the maximum fitness evaluation number is set as $F = 20,000$. All algorithms were implemented with a Matlab 2017a software on the workstation consisting of Intel®Core i7-7500 CPU @ 2.70 GHz. Each compared algorithm was run 20 times independently to obtain their average results.

Table 7 The parameter settings of all involved multi-objective optimization algorithms

Algorithms	Parameter settings
MOPSO	$N, A \in \{20, 30, 40\}, \omega \in 0.9 \rightarrow 0.4, c_1 = c_2 = 2.05, P_{mut} = 1/D, \alpha = 0.1, nGrid = 10$
NSGA-II	$N \in \{20, 30, 40\}, P_{cr} = 0.9, P_{mut} = 1/D$
MOGWO	$N, A \in \{20, 30, 40\}, \alpha = 0.1, nGrid = 10$
MOTLBO	$N \in \{20, 30, 40\}, T_f \in [1, 2]$
MO-ITLBO	$N, A \in \{20, 30, 40\}, nGroup = 4, T_f \in [1, 2], \epsilon = 0.007$
NSTLBO	$N \in \{20, 30, 40\}, T_f \in [1, 2]$
NSMTLBO	$N \in \{20, 30, 40\}, P_{mut} = 1/D, T_{f1}, T_{f2} \in [1, 2]$

Table 8 Performance comparison of coverage metrics of all algorithms for different population sizes

Compared sets	N=20		N=30		N=40	
	Mean	SD	Mean	SD	Mean	SD
$C(R_p, S_p)$	0.175	0.250	0.098	0.195	0.091	0.195
$C(S_p, R_p)$	0.000	0.000	0.003	0.015	0.004	0.012
$C(R_p, T_p)$	0.220	0.357	0.207	0.354	0.329	0.418
$C(T_p, R_p)$	0.015	0.037	0.000	0.000	0.008	0.018
$C(R_p, U_p)$	0.065	0.104	0.035	0.054	0.026	0.038
$C(U_p, R_p)$	0.000	0.000	0.000	0.000	0.000	0.000
$C(R_p, V_p)$	0.050	0.087	0.103	0.098	0.169	0.067
$C(V_p, R_p)$	0.005	0.022	0.010	0.024	0.000	0.000
$C(R_p, W_p)$	0.108	0.069	0.116	0.093	0.076	0.041
$C(W_p, R_p)$	0.065	0.056	0.040	0.021	0.041	0.026
$C(R_p, X_p)$	0.025	0.057	0.020	0.038	0.023	0.037
$C(X_p, R_p)$	0.005	0.022	0.000	0.000	0.000	0.000

Results and discussions

Table 8 presents the mean and standard deviation (*SD*) values of coverage metric obtained by all compared algorithms for $N=20, 30$ and 40 . The proposed NSMTLBO is reported to have the most competitive optimization performance because of its capability to produce the Pareto fronts with larger percentages of non-dominated solutions as compared with the remaining six peer algorithms in all population sizes. For instance, it is observed that 32.9% of the members of Pareto front obtained by NSGA-II are inferior to those of NSMTLBO for $N=40$, whereas only 0.8% of the Pareto front members of NSMTLBO are inferior to those of NSGA-II. There are 17.5% and 6.5% of Pareto front members of MOPSO and MOGWO, respectively, in $N=20$ are worse than those of NSMTLBO. In contrary, all Pareto optimal solutions obtained by NSMTLBO are non-dominated by those of MOPSO and MOGWO. For all three TLBO variants considered in benchmarking, MOTLBO demonstrates the worst results because the percentages of its Pareto optimal solutions dominated by NSMTLBO increase with population sizes, i.e., 5.0%, 10.3% and 16.9% for $N=20, 30$ and 40 , respectively. In contrary, the Pareto optimal solutions of NSMTLBO that are dominated by those of MOTLBO are at least ten times lesser for all population sizes. It is also observed that the performance differences between NSTLBO and NSMTLBO are relatively small because not more than 2.5% of NSTLBO’s Pareto optimal solutions are dominated by NSMTLBO, while there are not more than 0.5% of the Pareto front members produced by NSMTLBO are worse than those of NSTLBO for the population sizes of $N=20, 30$ and 40 .

Remark The Pareto fronts generated by NSMTLBO, MOPSO, NSGA-II, MOGWO, MOTLBO, MO-ITLBO and NSTLBO algorithms are represented with $R_p, S_p, T_p, U_p, V_p, W_p$ and X_p respectively.

The mean and *SD* values of spacing metric obtained by MOPSO, NSGA-II, MOGWO, MOTLBO, MO-ITLBO, NSTLBO and NSMTLBO for the 20 independent runs in

all population sizes are summarized in Table 9. It is reported that the quality of Pareto fronts generated by NSMTLBO are the best for having the non-dominated solution sets that are diversified and yet uniformly distributed as indicated by the lowest spacing values obtained for $N=20, 30$ and 40 . Among all compared algorithms, the Pareto fronts obtained by NSGA-II in solving the PTFE machining problem has the worst quality because not only majority of the Pareto front members are dominated by those of NSMTLBO, but they also have the poorest distributions. While the MOPSO, MOGWO and MOTLBO can produce lesser Pareto optimal solutions dominated by NSMTLBO, the uniformity and diversity of their Pareto fronts are at least two times worse than those of NSMTLBO according to the spacing metric. It is noteworthy that the Pareto fronts generated by MO-ITLBO and NSTLBO have shown different observations. For MO-ITLBO, the Pareto fronts generated have second best spacing values but they consist of more inferior solutions dominated by NSMTLBO. In contrary, the Pareto fronts produced by NSTLBO have much higher spacing values although it can produce more non-dominated solutions. The poor quality of Pareto fronts obtained by NSTLBO reveals the existence of duplicated non-dominated solutions and this explains the better coverage and yet inferior spacing values. As compared with MO-ITLBO and NSTLBO, the proposed NSMTLBO has more competitive performance in producing the Pareto fronts with better quality and well distributed solutions.

In order to evaluate the computational efficiency of each compared algorithm, the mean computation time incurred by MOPSO, NSGA-II, MOGWO, MOTLBO, MOITLBO, NSTLBO and NSMTLBO in 20 independent runs in all population sizes are presented in Table 10. The proposed NSMTLBO has demonstrated promising computation efficiency because the computational time incurred to optimize the PTFE multi-objective machining problems is lesser than those of NSGA-II, MOGWO, MOTLBO and NSTLBO for all population sizes of $N=20, 30$ and 40 . Although the mean computational time incurred by MOPSO and MO-ITLBO are lower than NSMTLBO in most population sizes, the performance differences of computational efficiency between these three algorithms are not significant. In addition, the

Table 9 Performance comparison of spacing metric of all algorithms for different population sizes

Algorithms	$N=20$		$N=30$		$N=40$	
	Mean	<i>SD</i>	Mean	<i>SD</i>	Mean	<i>SD</i>
MOPSO	2.916	1.702	4.420	2.962	4.816	3.178
NSGA-II	4.129	1.545	5.621	2.340	7.792	3.010
MOGWO	3.266	1.799	5.697	2.867	7.635	3.967
MOTLBO	3.061	1.015	4.782	1.568	6.388	2.056
MO-ITLBO	2.227	1.461	2.509	1.666	2.745	1.437
NSTLBO	3.929	1.854	5.312	2.394	7.508	3.757
NSMTLBO	1.213	0.841	1.837	1.102	2.186	1.014

Table 10 Performance comparison of mean computation times of all algorithms for different population sizes

Algorithms	Mean computation time (s)		
	$N=20$	$N=30$	$N=40$
MOPSO	3.500	4.358	5.445
NSGA-II	4.681	6.548	7.102
MOGWO	7.564	9.545	11.133
MOTLBO	4.653	5.114	6.003
MO-ITLBO	2.542	3.294	3.986
NSTLBO	4.065	6.345	7.033
NSMTLBO	3.473	4.651	5.886

proposed NSMTLBO has better performance than MOPSO and MO-ITLBO for being able to produce the Pareto fronts consisting of higher percentages of non-dominated solutions and more uniformly distributed of solution sets. Based on the simulation results shown in Table 10, it is suggested that the proposed modifications incorporated into NSMTLBO can achieve better tradeoff between the extra computational overheads consumed and performance enhancement achieved as compared with majority of peer algorithms.

The Pareto-fronts of PTFE machining problem obtained by MOPSO, NSGA-II, MOGWO, MOTLBO, MO-ITLBO, NSTLBO and NSMTLBO for $N=30$ are illustrated in Fig. 10. In general, the qualitative results of Fig. 10 are aligned with the quantitative performance analyses reported in Tables 2 and 3. NSGA-II shows the worst optimization

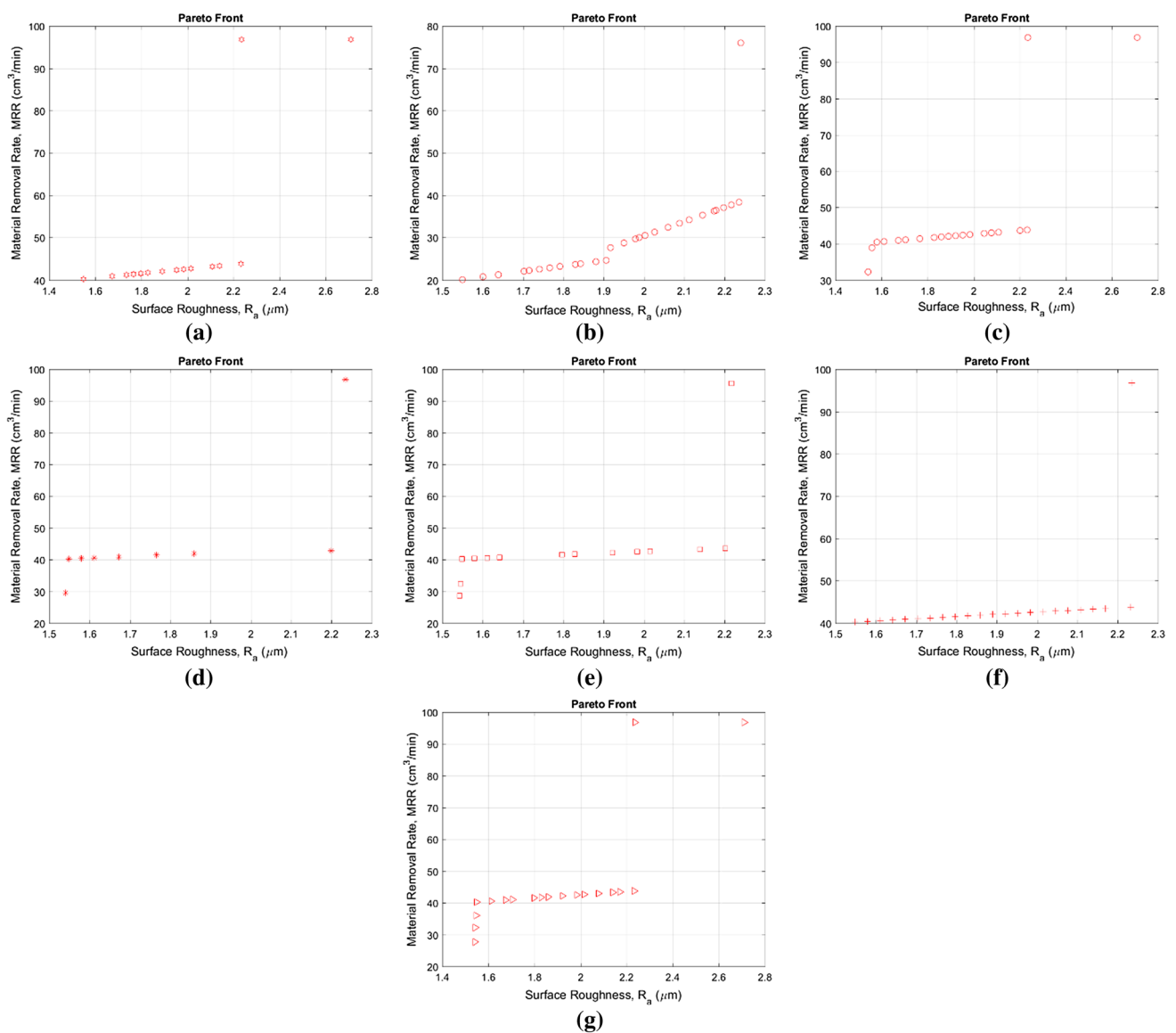


Fig. 10 The Pareto fronts of (a) MOPSO, b NSGA-II, c MOGWO, d MOTLBO, e MO-ITLBO, f NSTLBO and g NSMTLBO

performance not only because it has Pareto-front with uneven distribution of solutions, but some solutions obtained are also dominated by the other compared algorithms. For instance, the material removal rate of solution produced by NSGA-II is around $20 \text{ cm}^3/\text{mm}$ when the surface roughness value is $1.6 \text{ }\mu\text{m}$, while other peer algorithms such as MOGWO, MOTLBO, MO-ITLBO, NSTLBO and NSMTLBO can produce the material removal rates higher than $40 \text{ cm}^3/\text{mm}$ with same surface roughness. In addition, it is observed that the Pareto-fronts generated by MOPSO, MOGWO, MOTLBO, MO-ITLBO and NSTLBO are inferior than that of NSMTLBO because the proposed approach can produce higher percentages of non-dominated solutions, especially those at the boundary regions of objective space. For instance, the proposed NSMTLBO is able to obtain five Pareto solutions when surface roughness changes from 1.4 to $1.6 \text{ }\mu\text{m}$, while the MOPSO, MOGWO, MOTLBO and NSTLBO can only produce one, four, four and three Pareto solutions, respectively, for the same range of surface roughness. In contrast to NSMTLBO, the Pareto fronts of MOTLBO, MO-ITLBO and NSTLBO fail to detect a Pareto front member with the surface roughness of $2.7 \text{ }\mu\text{m}$ and the material removal rate of $98 \text{ cm}^3/\text{mm}$. The capability of NSMTLBO in identifying the Pareto optimal solutions located in the boundary regions of objective space is attributed to the promising exploitation strength offered by the modified teacher phase. The mechanism of assigning the nearest non-dominated solution as teacher of each learner allows the refinement of search process around a promising solution. Meanwhile, the unique mean positions generated for each learner by considering different weight contributions of all non-dominated solutions allow the learners to search the regions slightly beyond the promising solutions in order to reach other unvisited non-dominated solutions. Finally, the proposed NSMTLBO also outperforms the MOPSO, MOGWO, MOTLBO and MO-ITLBO for being able to produce the Pareto front with better uniformity. The extra momentum offered by the probabilistic self-learning mechanism in the modified learner phase during the search process can prevent the stagnation of NSMTLBO at the local Pareto-front as well as maintaining the diversity of solutions in Pareto-front.

After completing the quantitative and qualitative analyses of Pareto-fronts obtained by NSMTLBO, it is equally crucial to validate the optimal process parameters obtained by considering relative importance levels of all objective functions with those actual values of surface roughness (R_a) and material removal rate (MRR) obtained from experiments. Define w_1 and w_2 as the weightage to imply the importance levels of objectives in minimizing R_a and maximizing MRR , respectively, where $w_1 + w_2 = 1$. In this section, the weightage setting of $w_1 = w_2 = 0.5$ are considered on both contradict objectives. This implies that the equal emphasis levels

were considered in maximizing the quality and quantity of PTFE product simultaneously during the machining process. The proposed NSMTLBO was first executed with a population size of $N=30$ to produce a Pareto-front similar to that of in Fig. 10g. A fuzzy decision maker was then applied to determine the preferred optimum machining parameters based on the predefined weights of w_1 and w_2 using Eqs. (19) to (21). The predicted and experimental values of R_a and MRR obtained based on the optimum machining parameters were presented in Table 11 together with their error values.

Given the equal weightage values of $w_1 = w_2 = 0.5$, the optimum machining parameters obtained by the proposed NSMTLBO are $V_c = 160 \text{ m/min}$, $f = 0.50 \text{ mm/rev}$, $ap = 0.98 \text{ mm}$ and $N_r = 0.8 \text{ mm}$. The predicted values of surface roughness and material removal rate based on the simulation with these machining parameter settings are $R_a = 2.2347 \text{ }\mu\text{m}$ and $MRR = 96.8347 \text{ cm}^3/\text{mm}$, respectively. By referring to the validation results presented in Table 11, it is observed that the deviations between the predicted and experimental values of R_a and MRR are 3.26% and 3.7%, respectively. Referring to these negligibly small performance deviations, it can be concluded that the good consistency between the simulated and actual experimental results is achieved.

Conclusions

The experimental design was done and the L27 orthogonal array consisting of three-level of cutting speed (V_c), feed (f), depth of cut (ap) and nose radius (N_r) was formulated. The experiments were performed using a CNC turning machine with cemented carbide tool at an insert angle of 80° . The surface roughness was measured and material removal rate was computed instantly in each experiment. The response surface model (RSM) was rendered from the experimental results and the minimization function of surface roughness and maximization function of material removal rate were

Table 11 Comparison between the predicted and experimental values

Machining condition		$w_1 = w_2 = 0.5$
Machining parameters	Cutting speed, V_c	160
	Feed rate, f	0.5
	Depth of cut, ap	0.98
	Nose radius, N_r	0.8
Predicted values	Surface roughness, R_a	2.2347
	Material removal rate, MRR	96.8347
Experimental values	Surface roughness, R_a	2.31
	Material removal rate, MRR	93.25
Error rates	ΔR_a (%)	3.36
	ΔMRR (%)	3.70

derived. ANOVA was applied to determine the relationship between the dependent variable and independent variables.

Apart from developing the regression model of PTFE, an enhanced version of multi-objective optimization algorithm abbreviated as NSMTLBO was designed to tackle the multi-objective machining problem. Some notable modifications were introduced into the proposed NSMTLBO, including: (i) a modified teacher phase consisting of the teacher selection mechanism based on the nearest Euclidean distance and the derivation of weighted mean position for each learner, (ii) a modified learner phase consisting of the probabilistic mutation operator to emulate the self-learning mechanism of learner and (iii) a fuzzy decision maker to determine the preferred non-dominated solution from Pareto front based on the predefined importance levels of objective functions. The optimization performance of NSMTLBO was evaluated through extensive simulation studies and it was proven that the proposed approach can outperform the other six peer algorithms due to its excellent capability in generating the Pareto fronts with better uniformity and higher numbers of non-dominated solutions. Finally, the simulation results obtained by NSMTLBO were validated and it was reported that the performance deviations between the simulated and experimental results of both surface roughness and material removal rates for PTFE are less than 3.70%, implying the applicability of proposed work in machining of PTFE.

Acknowledgements This work is partially supported by UCSI University Pioneer Scientist Incentive Fund (PSIF) with Project Code of Proj-In-FETBE-34 and Proj-In-FETBE-50.

References

- Abbas, A. T., Aly, M., & Hamza, K. (2016). Multiobjective optimization under uncertainty in advanced abrasive machining processes via a fuzzy-evolutionary approach. *Journal of Manufacturing Science and Engineering*, 138(7), 071003–071009. <https://doi.org/10.1115/1.4032567>.
- Abhishek, K., Rakesh Kumar, V., Datta, S., & Mahapatra, S. S. (2017). Parametric appraisal and optimization in machining of CFRP composites by using TLBO (teaching–learning based optimization algorithm). *Journal of Intelligent Manufacturing*, 28(8), 1769–1785. <https://doi.org/10.1007/s10845-015-1050-8>.
- Aghaei, J., Amjady, N., & Shayanfar, H. A. (2011). Multi-objective electricity market clearing considering dynamic security by lexicographic optimization and augmented epsilon constraint method. *Applied Soft Computing*, 11(4), 3846–3858. <https://doi.org/10.1016/j.asoc.2011.02.022>.
- Al-Omoush, A. A., Alsewari, A. A., Alamri, H. S., & Zamli, K. Z. (2019). Comprehensive review of the development of the harmony search algorithm and its applications. *IEEE Access*, 7, 14233–14245. <https://doi.org/10.1109/access.2019.2893662>.
- Chabbi, A., Yallese, M. A., Nouioua, M., Meddour, I., Mabrouki, T., & Girardin, F. (2017). Modeling and optimization of turning process parameters during the cutting of polymer (POM C) based on RSM, ANN, and DF methods. *The International Journal of Advanced Manufacturing Technology*, 91(5), 2267–2290. <https://doi.org/10.1007/s00170-016-9858-8>.
- Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transactions on Evolutionary Computation*, 8(3), 256–279. <https://doi.org/10.1109/tevc.2004.826067>.
- Collette, Y., & Siarry, P. (2003). *Multiobjective optimization: Principles and case studies*. Berlin: Springer.
- Corne, D. W., Jerram, N. R., Knowles, J. D., & Oates, M. J. (2001). PESA-II: Region-based selection in evolutionary multiobjective optimization. In *Paper presented at the proceedings of the 3rd annual conference on genetic and evolutionary computation*, San Francisco, California.
- Deb, K., & Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 18(4), 577–601. <https://doi.org/10.1109/tevc.2013.2281535>.
- Deb, K., Mohan, M., & Mishra, S. (2005). Evaluating the ϵ -domination based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions. *Evolutionary Computation*, 13(4), 501–525. <https://doi.org/10.1162/106365605774666895>.
- Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017>.
- Fan, Q., & Yan, X. (2016). Self-adaptive differential evolution algorithm with zoning evolution of control parameters and adaptive mutation strategies. *IEEE Transactions on Cybernetics*, 46(1), 219–232. <https://doi.org/10.1109/tycyb.2015.2399478>.
- Hu, P., Chen, S., Huang, H., Zhang, G., & Liu, L. (2019). Improved alpha-guided grey Wolf optimizer. *IEEE Access*, 7, 5421–5437. <https://doi.org/10.1109/access.2018.2889816>.
- Ji, J., Gao, S., Wang, S., Tang, Y., Yu, H., & Todo, Y. (2017). Self-adaptive gravitational search algorithm with a modified chaotic local search. *IEEE Access*, 5, 17881–17895. <https://doi.org/10.1109/access.2017.2748957>.
- Jiao, K., & Pan, Z. (2019). A novel method for image segmentation based on simplified pulse coupled neural network and gbest led gravitational search algorithm. *IEEE Access*, 7, 21310–21330. <https://doi.org/10.1109/access.2019.2894301>.
- Li, Y., Gong, H., Feng, D., & Zhang, Y. (2011). An adaptive method of speckle reduction and feature enhancement for SAR images based on curvelet transform and particle swarm optimization. *IEEE Transactions on Geoscience and Remote Sensing*, 49(8), 3105–3116. <https://doi.org/10.1109/tgrs.2011.2121072>.
- Li, D., Zhang, C., Shao, X., & Lin, W. (2016). A multi-objective TLBO algorithm for balancing two-sided assembly line with multiple constraints. *Journal of Intelligent Manufacturing*, 27(4), 725–739. <https://doi.org/10.1007/s10845-014-0919-2>.
- Liang, J., Qu, B.-Y., Suganthan, P., & Hernandez-Diaz, A. (2013). Problem definitions and evaluation criteria for the CEC 2013 special session and competition on real-parameter optimization. In *Tech. Rep. Zhengzhou, China: Computational Intelligence Laboratory, Zhengzhou University*.
- Lim, W. H., & Isa, N. A. M. (2015). Particle swarm optimization with dual-level task allocation. *Engineering Applications of Artificial Intelligence*, 38, 88–110. <https://doi.org/10.1016/j.engappai.2014.10.022>.
- Lin, W., Yu, D. Y., Wang, S., Zhang, C., Zhang, S., Tian, H., et al. (2015). Multi-objective teaching–learning-based optimization algorithm for reducing carbon emissions and operation time in turning operations. *Engineering Optimization*, 47(7), 994–1007. <https://doi.org/10.1080/0305215x.2014.928818>.
- Mathew, D., Rani, C., Kumar, M. R., Wang, Y., Binns, R., & Busawon, K. (2018). Wind-driven optimization technique for estimation of

- solar photovoltaic parameters. *IEEE Journal of Photovoltaics*, 8(1), 248–256. <https://doi.org/10.1109/jphotov.2017.2769000>.
- Mellal, M. A., & Williams, E. J. (2016). Parameter optimization of advanced machining processes using cuckoo optimization algorithm and hoopoe heuristic. *Journal of Intelligent Manufacturing*, 27(5), 927–942. <https://doi.org/10.1007/s10845-014-0925-4>.
- Mia, M. (2018). Mathematical modeling and optimization of MQL assisted end milling characteristics based on RSM and Taguchi method. *Measurement*, 121, 249–260.
- Mirjalili, S., Saremi, S., Mirjalili, S. M., & Coelho, L. D. S. (2016). Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization. *Expert Systems with Applications*, 47, 106–119. <https://doi.org/10.1016/j.eswa.2015.10.039>.
- Mistry, K., Zhang, L., Neoh, S. C., Lim, C. P., & Fielding, B. (2017). A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition. *IEEE Transactions on Cybernetics*, 47(6), 1496–1509. <https://doi.org/10.1109/tycb.2016.2549639>.
- Mohanty, C. P., Mahapatra, S. S., & Singh, M. R. (2016). A particle swarm approach for multi-objective optimization of electrical discharge machining process. *Journal of Intelligent Manufacturing*, 27(6), 1171–1190. <https://doi.org/10.1007/s10845-014-0942-3>.
- Mukherjee, I., & Ray, P. K. (2006). A review of optimization techniques in metal cutting processes. *Computers and Industrial Engineering*, 50(1), 15–34. <https://doi.org/10.1016/j.cie.2005.10.001>.
- Natarajan, E., Kaviarasan, V., Lim, W. H., Tiang, S. S., & Tan, T. H. (2018). Enhanced multi-objective teaching-learning-based optimization for machining of delrin. *IEEE Access*, 6, 51528–51546. <https://doi.org/10.1109/access.2018.2869040>.
- Ong, P., Chong, C. H., Bin Rahim, M. Z., Lee, W. K., Sia, C. K., & Bin Ahmad, M. A. H. (2018). Intelligent approach for process modelling and optimization on electrical discharge machining of polycrystalline diamond. *Journal of Intelligent Manufacturing*. <https://doi.org/10.1007/s10845-018-1443-6>.
- Patel, V. K., & Savsani, V. J. (2016). A multi-objective improved teaching-learning based optimization algorithm (MO-ITLBO). *Information Sciences*, 357, 182–200. <https://doi.org/10.1016/j.ins.2014.05.049>.
- Petković, D., Madić, M., Radovanović, M., & Gečevska, V. (2017). Application of the performance selection index method for solving machining MCDM problems. *Petković*, 15(1), 97–106.
- Rao, R. V., More, K. C., Taler, J., & Ocion, P. (2016). Optimal design of stirling heat engine using an advanced optimization algorithm. *Sadhana*, 41(11), 1321–1331.
- Rao, R. V., More, K. C., Taler, J., & Ochoń, P. (2017a). Multi-objective optimization of thermo-acoustic devices using teaching-learning-based optimization algorithm. *Science and Technology for the Built Environment*, 23(8), 1244–1252. <https://doi.org/10.1080/23744731.2017.1296319>.
- Rao, R. V., & Patel, V. (2013). Multi-objective optimization of heat exchangers using a modified teaching-learning-based optimization algorithm. *Applied Mathematical Modelling*, 37(3), 1147–1162. <https://doi.org/10.1016/j.apm.2012.03.043>.
- Rao, R. V., Rai, D. P., & Balic, J. (2017b). A multi-objective algorithm for optimization of modern machining processes. *Engineering Applications of Artificial Intelligence*, 61, 103–125. <https://doi.org/10.1016/j.engappai.2017.03.001>.
- Rao, R. V., Rai, D. P., & Balic, J. (2018). Multi-objective optimization of machining and micro-machining processes using non-dominated sorting teaching-learning-based optimization algorithm. *Journal of Intelligent Manufacturing*, 29(8), 1715–1737. <https://doi.org/10.1007/s10845-016-1210-5>.
- Rao, R. V., Rai, D. P., & Balic, J. (2019). Multi-objective optimization of abrasive waterjet machining process using Jaya algorithm and PROMETHEE method. *Journal of Intelligent Manufacturing*, 30(5), 2101–2127. <https://doi.org/10.1007/s10845-017-1373-8>.
- Rao, R. V., Savsani, V. J., & Vakharia, D. P. (2011). Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, 43(3), 303–315. <https://doi.org/10.1016/j.cad.2010.12.015>.
- Rao, R. V., & Waghmare, G. G. (2015). Multi-objective design optimization of a plate-fin heat sink using a teaching-learning-based optimization algorithm. *Applied Thermal Engineering*, 76, 521–529. <https://doi.org/10.1016/j.applthermaleng.2014.11.052>.
- Sathiyamoorthy, V., Sekar, T., & Elango, N. (2015a). Optimization of processing parameters in ECM of die tool steel using nanofluid by multiobjective genetic algorithm. *The Scientific World Journal*, 2015, 6. <https://doi.org/10.1155/2015/895696>.
- Sathiyamoorthy, V., Sekar, T., Suresh, P., Vijayan, R., & Elango, N. (2015b). Optimization of processing parameters in electrochemical machining of AISI 202 using response surface methodology. *Journal of Engineering Science and Technology*, 10(6), 780–789.
- Senthilnath, J., Kulkarni, S., Benediktsson, J. A., & Yang, X. S. (2016). A novel approach for multispectral satellite image classification based on the bat algorithm. *IEEE Geoscience and Remote Sensing Letters*, 13(4), 599–603. <https://doi.org/10.1109/lgrs.2016.2530724>.
- Solihin, M. I., Akmeliawati, R., Muhida, R., & Legowo, A. Guaranteed robust state feedback controller via constrained optimization using Differential Evolution. In *2010 6th International colloquium on signal processing and its applications*, 2010 (pp. 1–6). <https://doi.org/10.1109/cspa.2010.5545279>.
- Solihin, M. I., Wahyudi, & Akmeliawati, R. PSO-based optimization of state feedback tracking controller for a flexible link manipulator. In *2009 International conference of soft computing and pattern recognition*, 2009 (pp. 72–76). <https://doi.org/10.1109/socpar.2009.26>.
- Suganthan, P., Hansen, N., Liang, J., Deb, K., Chen, Y.-P., Auger, A., et al. (2005). Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. In *Tech. Rep.* Singapore: Nanyang Technological University.
- Sultana, S., & Roy, P. K. (2014). Multi-objective quasi-oppositional teaching learning based optimization for optimal location of distributed generator in radial distribution systems. *International Journal of Electrical Power and Energy Systems*, 63, 534–545. <https://doi.org/10.1016/j.ijepes.2014.06.031>.
- Suresh, P., Venkatesan, R., Sekar, T., Elango, N., & Sathiyamoorthy, V. (2014). Optimization of intervening variables in MicroEDM of SS 316L using a genetic algorithm and response-surface methodology. *Strojnikski Vestnik/International Journal of Mechanical Engineering*, 60(10), 9–10. <https://doi.org/10.5545/sv-jme.2014.1665>.
- Tang, S. H., Ang, C. K., Ariffin, M. K. A. B. M., & Mashohor, S. B. (2014). Predicting the motion of a robot manipulator with unknown trajectories based on an artificial neural network. *International Journal of Advanced Robotic Systems*, 11(10), 176. <https://doi.org/10.5772/59278>.
- Tang, S. H., Ang, C. K., Nia, D. N., Ariffin, M. K. A. M., & Khaksar, W. (2013). Planning for redundant manipulator based on back-propagation neural network. *Advanced Science Letters*, 19(11), 3307–3310. <https://doi.org/10.1166/asl.2013.5151>.
- Tang, K., Li, X., Suganthan, P., Yan, Z., & Wiese, T. (2010). Benchmark Functions for the CEC 2010 Special Session and Competition on Large-Scale Global Optimization. In *Tech. Rep.* Hefei, China: School of Computer Science and Technology, University of Science and Technology of China.
- Teimouri, R., Baseri, H., & Moharami, R. (2015). Multi-responses optimization of ultrasonic machining process. *Journal of Intelligent Manufacturing*, 26(4), 745–753. <https://doi.org/10.1007/s10845-013-0831-1>.
- Van, M., & Kang, H. (2016). Bearing defect classification based on individual wavelet local fisher discriminant analysis with particle

- swarm optimization. *IEEE Transactions on Industrial Informatics*, 12(1), 124–135. <https://doi.org/10.1109/tii.2015.2500098>.
- Viswanathan, R., Ramesh, S., Elango, N., & Kamesh Kumar, D. (2017). Temperature measurement and optimisation in machining magnesium alloy using RSM and ANOVA. *Pertanika Journal of Science and Technology*, 25(1), 255–262.
- Yao, L., Damiran, Z., & Lim, W. H. Energy management optimization scheme for smart home considering different types of appliances. In *2017 IEEE international conference on environment and electrical engineering and 2017 IEEE industrial and commercial power systems Europe (EEEIC/I&CPS Europe)*, 2017 (pp. 1–6). <https://doi.org/10.1109/eeeic.2017.7977565>.
- Yao, L., Lai, C., & Lim, W. H. Home energy management system based on photovoltaic system. In *2015 IEEE international conference on data science and data intensive systems*, 2015 (pp. 644–650). <https://doi.org/10.1109/dsdis.2015.42>.
- Yu, K., Wang, X., & Wang, Z. (2015). Self-adaptive multi-objective teaching-learning-based optimization and its application in ethylene cracking furnace operation optimization. *Chemometrics and Intelligent Laboratory Systems*, 146, 198–210. <https://doi.org/10.1016/j.chemolab.2015.05.015>.
- Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6), 712–731. <https://doi.org/10.1109/tevc.2007.892759>.
- Zhao, Z., Lee, W. C., Shin, Y., & Song, K. (2013). An optimal power scheduling method for demand response in home energy management system. *IEEE Transactions on Smart Grid*, 4(3), 1391–1400. <https://doi.org/10.1109/tsg.2013.2251018>.
- Zhao, J., Wen, F., Dong, Z. Y., Xue, Y., & Wong, K. P. (2012). Optimal dispatch of electric vehicles and wind power using enhanced particle swarm optimization. *IEEE Transactions on Industrial Informatics*, 8(4), 889–899. <https://doi.org/10.1109/tii.2012.2205398>.
- Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2), 173–195. <https://doi.org/10.1162/106365600568202>.
- Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Proceedings of the evolutionary methods for design, optimization and control with applications to industrial problems*, Athens, Greece.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.