

RESEARCH

Open Access



A comparison of two hybrid optimization techniques: the Taguchi-BBD-firefly and the Taguchi-regression-firefly methods on the IS 2062-E250 steel plates boring problem

Ebun Fasina¹, Babatunde Alade Sawyerr¹, Yakubu Umar Abdullahi² and Sunday Ayoola Oke^{2*}

*Correspondence:
sa_oke@yahoo.com

¹ Department of Computer
Science, University of Lagos,
Lagos, Nigeria

² Department of Mechanical
Engineering, University of Lagos,
Lagos, Nigeria

Abstract

The expanding proliferation of components for engineering applications requires greater optimisation of parameters, which consequently increases the need for more efficient boring practices. The Taguchi Pareto-Box Behnken design is an effective optimisation procedure for the process parametric optimisation of the IS 2062 E250 steel plates. However, the weakness of the Taguchi method in its inability to distinguish which parameters have greater effects on the boring process needs to be further suppressed. Consequently, this study investigates the coupling of the firefly algorithm to the Taguchi-Pareto-Box Behnken design method for the processing of the IS 2062 E250 steel plates during the boring operation. Linear programmes were developed for the problem formulation with two variants of the objective function definition. In the first variant, the Box Behnken design optimized parameters and the firefly-oriented optimisation procedure was addressed to attain optimal solutions. For the second variant, a regression equation was substituted as the objective function and the firefly procedure was implemented to obtain the optimal solutions. Based on a defined population for the problem, an initial test of convergence was actualized and 50 iterations were found as an effective convergence point for the iterations. Numerical simulation coupled with experimental data analysis was conducted to ascertain the effectiveness of the proposed method. Literature data on IS 2062 E250 steel plate processing on the CNC machine was used in the testing. The results revealed that the proposed method exhibits good performance for boring operations in machine shops. Using the Taguchi-Pareto-Box Behnken-firefly algorithm, the obtained results are promising. The application of this proposal would aid machining to better decisions that improve the quality of products and reduces the cost of production.

Keywords: Firefly algorithm, Boring operation python programming, Optimisation

Introduction

Optimization research in boring operations is presently an area of vast interest because it aids more efficiency by minimizing boring errors, material and manhour losses and costs, leading to profit improvement and the accomplishment in business competition [1–3]. Furthermore, the theoretical bases for optimization approaches are widespread with applications in several methods such as integrated harmony search and grey relational analysis [4], particle swarm optimization [5], Taguchi method [6] and differential evolution [7] response surface methodology coupled with firefly method [8]. However, of the available optimization methods in this article, preference is given to the Taguchi-Pareto and Box Behnken design method, which has been proposed in previous research by Abdullahi and Oke [9]. The Taguchi-Pareto method is subscribed to as it aids efficient and reliable processes. Then, the Box Behnken design is chosen as it permits the assessment of the influence of several boring parameters and the possible interfaces of these parameters on the outcomes such as the surface roughness. But the combination of these methods had been demonstrated in the literature to handle critical parameters that determine the success of the boring operation in overcoming resistance in the machining market competitive fight as the operations maintain efficiency [9].

Furthermore, a review of available literature on the further optimisation of process parameters while boring the IS 2062 E250 steel plates on CNC machines while an evolutionary algorithm is appended to the Taguchi-Pareto-Box Behnken design method reveals poor results when the objective function was formulated from the Box Behnken design results and introduced into the genetic algorithm procedure for optimisation using the python codes. This may be primarily due to the evolutionary method utilized for the optimisation procedure [10]. Thus, the type of evolutionary algorithm used may be changed. Interestingly, the substitution of the genetic algorithm with the firefly algorithm may enhance the optimisation performance of the surface roughness outcome and the corresponding process parameters of speed, feed, depth of cut and nose radius. The attributes of the firefly algorithm, which entails absorption, randomness and attractiveness were deployed to optimize the outcome and the process parameters of the IS 2062 E250 steel plates during the boring operation. In the literature on the firefly algorithm, these attributes were found to be favourable, yielding optimal results for the problems solved.

Furthermore, integrating the firefly algorithm with the emerging method of Taguchi-Pareto-Box Behnken design may be a productive investigation route to enhancing the performance of both the boring parameters of speed, feed, depth of cut and nose radius as well as the surface roughness outcome. Based on this motivation, the objective of this study is to introduce a new method called the Taguchi-Pareto-box Behnken design-firefly method and test it with experimental data from the literature using Patel and Deshpande's [11] study. This goal was pursued to see if any enhancement could be achieved with the replacement of the genetic algorithm component of the integrated method with the firefly algorithm using a previously announced method as the benchmark.

Literature review

General

There is a pressing need to improve manufacturing performance, particularly in growing and developing economies, so that industrial pollution and material and energy usage are reduced. Energy efficiency and product quality have become essential industry criteria [3]. Where optimization of machining process parameters is properly performed, there is the likelihood of significantly reducing industrial pollution, material usage and energy consumption in the manufacturing processes, making optimization of machining parameters critical for manufacturing industries. On the other hand, optimization of machining parameters is also required for the production of high-quality goods at reasonable prices. In this literature review, a summary of the literature is presented for an understanding of the progress made in the present area of study. This is shown in Table 1.

Research gap

In this article, a literature review was presented, which reveals an overview of the studies conducted by various engineers and scientists on firefly optimization and optimization regarding conventional machining systems. Such aspects as the machining types, materials and others were treated. Besides, some investigators attempted to optimize the boring parameters while machining different materials. Moreover, few investigators attempted to study the IS 2062 E250 steel plates such as the evaluation of the performance of the material using the Taguchi method by Patel and Deshpande [11]. More recently, very few investigators analysed the various optimization schemes in the boring of the IS 2062 E250 steel plates. Notwithstanding, very few investigators have attempted to use the combined Taguchi method and the Box Behnken design method as framework optimization schemes are deployed for further improvement of the method's performance. Thus, extremely limited information is available in the machining literature associated with the introduction of evolutionary algorithms in the boring process of IS 2062 E250 steel plates on the CNC machine. The unique introduction of the firefly algorithm to the established optimization scheme of the Taguchi/Taguchi-Pareto-Box Behnken design method for the IS 2062 steel plate is the principal novelty of the present study. Thus, the firefly algorithm is used as an additional optimizer while focusing on the surface roughness improvement and the parameters to be optimized are the speed, feed, depth of cut and nose radius in the processing of IS 2062 E250 steel plates for the boring operation on the CNC machine.

Furthermore, the firefly algorithm is introduced as an innovative metaheuristic into an objective function that accounts for flashing patterns and behaviour of fireflies at night. Fireflies produce sparks of light in the air. They are often found in gardens containing flowers. Experiences of people revealed that fireflies have been found in rice fields particularly when searching under the light of the moon. The objective function is introduced into an already developed amalgamated model of the Taguchi method

Table 1 Summary of related literature

S/No	Author(s)	Work material used	Machining operation/tool	Output	Inputs	Methodology	Findings	Conclusion
1	Prasanth et al. [12]	Stainless steel 304	Turning/CNC machine	Surface roughness and material removal rate	Speed, feed and depth of cut	Taguchi method and firefly algorithm	Roughly 2.57% reduction in specific cutting energy resulted from using nitrogen gas as a shielding gas in the cutting zone	The mathematical model and the experimental results were in agreement
2	Dutta et al. [2]	Commercial 1000 series aluminium	Milling/vertical milling machine	Surface roughness and material removal rate	Cutting speed, feed and tool diameter	ANOVA and firefly algorithm	Using the firefly optimization algorithm, an improvement of 32.42% and 37.25% were recorded for material removal rate and surface roughness respectively	The error between the experimental result and the predicted results
3	Singh & Shukla [13]	Copper Plates	Electrochemical micromachining and electrochemical discharge machining	Material removal rate, radial overcut and heat-affected zone	Pulse on/off ratio, machining voltage, electrolyte concentration, voltage frequency and tool vibration	Firefly algorithm	Single-objective optimization is better than the multi-objective optimization problem. The MRR increased while the ROC and HAZ reduced in the two machining processes	Firefly algorithm is very effective with a high convergence rate, simple, flexible and versatile
4	Hebbar et al. [14]	AZ31 (biodegradable magnesium alloy)	Face milling	Surface roughness	Speed, feed rate and depth of cut	Firefly algorithm, particle swarm optimization and artificial neural network	The optimal surface roughness from the firefly algorithm and particle swarm optimization are 0.1835 and 0.214 μm, respectively	The optimal surface roughness from both methods is in agreement

Table 1 (continued)

S/No	Author(s)	Work material used	Machining operation/tool	Output	Inputs	Methodology	Findings	Conclusion
5	Raja et al. [15]		Electrochemical discharge machining	Surface roughness and machining time	Current and pulse time	Firefly algorithm	The result of the predicted and the actual machining time and surface roughness are in agreement	The firefly optimization algorithm solves the machining parameter optimization problems
6	Majumder et al. [16]	Indian RAFM steel	Wire electrical discharge machining	Cutting speed and surface roughness	Pulse current, pulse-on-time, pulse-off time and wire tension	Standard deviation method and firefly algorithm	The firefly optimization algorithm is better than differential evolution and particle swarm optimization algorithms	Firefly optimization algorithm is sufficient to obtain optimal machining parameters in the WEDM process
7	Senthilkumar et al. [8]	AISI 1045 (medium carbon steel)	Turning	Flank wear, surface roughness, and material removal rate (MRR)	Cutting speed, feed rate and depth of cut	RSM and firefly algorithm	The optimal parameter obtained by the application of the firefly algorithm is a cutting speed of 253.29 m/min, feed rate of 0.203 mm/rev, and depth of cut of 0.3 mm	The findings act as a technical database for the automobile industry to improve quality and productivity
8	Johari et al. [17]	Medium carbon steel (AISI 1045)	Turning operation	Surface Roughness	Feed rate, depth of cut and spindle speed	Firefly algorithm, PSO and ANOVA	Hybrid (firefly-PSO) produced the best result, the firefly algorithm and lastly PSO with surface roughness values of 0.1216, 0.1228 and 0.1376 respectively	Multiobjective optimization problems can be solved by modification of the hybrid algorithm (firefly-PSO)
9	Thamizhmani et al. [18]	SCM 440 alloy steel	Turning operation	Surface roughness	Cutting speed, feed and depth of cut	Taguchi method and ANOVA	Depth of cut is the most important parameter on surface roughness followed by feed rate and lastly the speed	The results are valuable in subsequent studies and may provide new insights into tool vibrations, cutting forces, and other topics

Table 1 (continued)

S/No	Author(s)	Work material used	Machining operation/tool	Output	Inputs	Methodology	Findings	Conclusion
10	Lee & Ponnambalam [5]		Multi-pass turning operation (rough machining, finish machining)	Unit production cost	Cutting speed, feed rate and depth of cut	Genetic algorithm and artificial immune system (GA-AIS)	Implementing three different GA-AIS algorithms the binary tournament selection turns out to be the best for minimization of unit production cost	GA-AIS is appropriate for use in problems similar to the minimization or maximization of an output parameter
11	Rudrapati & Patil [19]	D3 steel	Vertical milling centre five-axis machining	Surface roughness	Speed, feed, and depth of cut	RSM(BBD), hybrid RSM and Jaya optimization algorithm	ANOVA result shows that the factors significant to surface roughness are the direct effects of speed, feed, depth of cut and squared combination of speed and also all interaction effects	Using Jaya optimization algorithm, the optimized process parameters obtained are depth of cut, 1 mm; speed, 3000 rpm and feed, 1500 rev/min and a surface roughness value of 0.19 micron
12	Pujara et al. [20]	LM6 Aluminum MMC	WEDM	Surface roughness, material removal rate and dimensional accuracy	Pulse ON/OFF time, peak current, servo voltage and wire feed rate	TLBO	The gap current and flushing pressure were not significant as input parameters. Using the optimized parameter a surface roughness and material removal rate of 0.1222 µm and 0.3845 gm/min were respectively recorded	TLBO successfully reduced the surface roughness and increased the material removal rate and it yield better results at lower iteration numbers compared to other optimization methods

Table 1 (continued)

S/No	Author(s)	Work material used	Machining operation/tool	Output	Inputs	Methodology	Findings	Conclusion
13	Liu et al. [21]		Multi-pass grinding process	Grinding time and geometric error (cylindricity) and surface roughness	Initial, rough stage, semi-rough stage, semi-finish stage, finish stage, spark out stage	Non-dominated sorting firefly algorithm (NSFA)	Non-dominated sorting firefly algorithm (NSFA) has better convergence speed and less number of iterations in obtaining the optimal process parameters compared to the non-dominated genetic algorithm (NSGA-II) and non-dominated particle swarm optimization (NSPSO)	The NSFAs improved method may satisfy all process quality requirements at the same time while increasing grinding efficiency by 32.4%
14	Nee et al. [7]	Co28Cr6Mo medical alloy	CNC lathe	Surface roughness	Rotational speed, feed rate, depth of cut, and tool tip radius	Differential evolution	The implementation of DE as a solution strategy in process parameter optimization greatly improved the surface roughness (Ra), with Ra values lowered	Differential evolution can be applied to various kinds of objective functions in the machining process
15	Kumari et al. [4]	AISI D2 Steel	Dry turning	Material removal rate (MRR) and surface roughness (Ra)	Spindle speed (N), feed rate (f) and depth of cut (d)	Grey relational analysis, harmony search (HS)	The global optimal value is 0.7102 while the optimal combination is (N= 560 RPM, f= 2.5 mm/rev; d= 0.5 mm)	Grey-based harmony search algorithm is a successful strategy towards parametric assessment and simultaneous optimization of process output reactions during the turning of AISI D2 steel

Table 1 (continued)

S/No	Author(s)	Work material used	Machining operation/tool	Output	Inputs	Methodology	Findings	Conclusion
16	Zubair & Mansor [22]	Tool inserts	Turning	Unit production cost	Cutting speed, feed rate and depth of cut	Computer-aided process planning (CAPP) system,	Optimization of machining data using firefly optimization and CAPP system could be developed to select the best cutting tool insert automatically	Firefly optimization and CAPP system is effective
17	Dhandapani et al. [23]	AlSi4340 Steel	Turning	Flank wear, surface roughness and material removal rate	Cutting speed, feed rate and depth of cut	Non-dominated Sorting Genetic Algorithm (NSGA-II), Taguchi's DOE	Reductions in flank wear of 58.65%, surface roughness of 10.56%, and MRR of 58.06% were experienced	The NSGA-II could be used in obtaining optimal machining inputs and outputs
18	Palanikumar et al. [24]	Glass fiber-reinforced polymer	Turning	Surface roughness	Cutting speed, work-piece fiber orientation angle, depth of cut and feed rate	Response graph, response table, normal probability plot, interaction graphs and analysis of variance	In the interaction effect between parameters, work piece fiber orientation and depth of cut has greater effect on the surface roughness of the composite material	Feed rate has the most significant influence on surface roughness
19	Nalbant et al. [6]	AlSi 1030 steel bar	Turning	Surface roughness	Insert radius, feed rate, and depth of cut	Taguchi method and analysis of variance	Taguchi method is effective. The insert radius and feed rate are the most influential parameter of surface roughness	Surface roughness improved by 335%. Optimized process parameters came with reduced cost and time for industrial practitioners
20	Yang & Tarnq [25]	S45C steel	Turning/lathe	Surface roughness	Cutting speed, feed rate and depth of cut	Taguchi method and ANOVA	There is a significant improvement of 250% in tool life and surface roughness	Taguchi approach is more effective than most optimization strategies

and Box Behnken Design (BBD) plan for the minimization of surface roughness of the IS 2062 E250 steel plates undergoing the boring process [26–28]. An alternative computational procedure is set forth such that the regression model replaces the BBD plan earlier stated to make the Taguchi method-regression model. In the past years, surface roughness has been optimized in the machining process and boring process, in particular, using the Taguchi method [11], integrated artificial neural network and bat algorithm [28], artificial neural network and fuzzy inference system [27], fuzzy analytic hierarchy process and mark or chain WSM, WPM/WASPAS approaches [26]. However, a drastic change in the literature on surface roughness in machining took place in recent years when researchers introduced the integrated Taguchi-Pareto Box Behnken design method. Also, unlike Patel and Deshpande [11] that concentrated on the experimental plans of the Taguchi method to improve the surface integrity of the mentioned steel, Abdullahi and Oke [10] directed attention to the optimization and selection of parameters using the combined Taguchi-Pareto Box Behnken-genetic algorithm method. This article introduces the firefly algorithm that leverages the flashing characteristics of the fireflies. This utilizes how a firefly flashes another using the flashing light to attract mates in courtships and also to establish predators [22, 29]. This handles the surface roughness behaviour by the introduction of attractive constant, absorption coefficient and the cooling factor into the boring operational interpretation of the firefly algorithm in the Taguchi-BBD and Taguchi-regression model interfaces. Both the BBD and regression models were employed, offering a wide solution to the minimization of the surface roughness phenomenon for the boring operation. The approaches offer a flexible yet effective solution to boring operators to minimize the surface roughness of IS 2062 E250 steel plates. Eventually, the effectiveness of the approaches is established using experimental data derived from Patel and Deshpande [11] thus offering understanding for machining stakeholders and practitioners to design and install cost-effective and sustainable machining operations.

Methods

This section focuses on the methodology of research introduced to achieve the stated objectives earlier mentioned in the article. It also solves the problem formulated through a research gap analysis of the present literature on the boring of IS 2062 E250 steel plates. The analysis of the parameters for the boring operation involving speed, feed, depth of cut and nose radius was done in this article while keeping in focus the optimisation of the surface integrity of the IS 2062 E250 steel plate through the reduction of surface roughness of the bored material. To attain a detailed understanding of the method, the steps followed to satisfy the objectives and the outcomes of the study are elaborated here.

Procedure for Taguchi-Pareto-Box Behnken design-firefly approach

-
- Step 1a Implement step 1a to step 3a in the Procedure for Taguchi-Pareto-Box Behnken Design-TLBO Approach of the work of Abdullahi and Oke [30] to establish the objective function, the constraints, population size, number of iterations, firefly population based on the chosen population size and computed fitness values of fireflies in the population
 Take the objective function as

$$F(x) = 20S + 0.002F + 0.01DC + 0.003NR \quad (1)$$
 And the constraints as $5 \leq S \leq 50$, $0.001 \leq f \leq 0.005$, $0.01 \leq dc \leq 0.07$, and $0.1 \leq S \leq 0.005$
 Also take the population size and number of iterations as 4 and 3, respectively
 Furthermore, take an instance of a firefly in the population as [15, 0.004, 0.05, 0.01], there should be four of such instances of a firefly in the population as the chosen population size is 4
 Lastly assuming the computed fitness value of fireflies in the population are 300.0014, 280.0022, 503.0450 and 412.7902
- Step 1b Compare the first firefly to all other fireflies in the population using the fitness value as the basis of comparison for instance, the fitness value of the first firefly in the population is 300.0014 comparing it with the next firefly's fitness value in the population to minimize the objective function that is (minimization problem). Observe that the next firefly fitness value which is 280.0022 is less than the first firefly fitness value in the population, with this the first firefly would tend to be attracted to the next or second firefly thereby moving toward it the aid of the mathematical function

$$X_i^{t+1} = X_i^t + \beta_0 e^{-\gamma r_{ij}^2} (X_j^t - X_i^t) + \alpha_t \epsilon_i^t \quad (2)$$
 where
 X_i^{t+1} is the new position of a parameter component of a firefly
 X_i^t is the old or former position
 X_j^t is the position a firefly is attracted to
 α_t is the randomized parameter, which is $\alpha_0 \delta^t$
 where α_0 is the initial randomness given by $(0.01 \times U-L)$, U and L are the upper and lower bounds of a factor
 δ is the cooling factor, which is taken between 0.95 and 0.97 though generally in application α_t ranges between 0 and 1
 ϵ_i^t is a random number drawn from normal or other distribution
 β_0 is called the attractive constant, it controls attractiveness and is usually taken as 1 in most application
 γ is the absorption coefficient usually taken as 0.01
 r_{ij}^2 is the distance between the current firefly and the firefly that the current firefly moves towards
- Step 1c Assuming the first firefly position in the population is as [15, 0.004, 0.05, 0.01], and the next firefly position in the population is [12, 0.0042, 0.03, 0.012] using the above mathematic function for firefly movement, we compute a new position for the first firefly in the population by substituting values as thus; the term r_{ij}^2 is given by

$$r_{ij}^2 = \sqrt{(15 - 12)^2 + (0.004 - 0.0042)^2 + (0.05 - 0.03)^2 + 0.01 - 0.012^2} = 3.0000$$
 The term $\beta_0 e^{-\gamma r_{ij}^2}$ is computed as $1 \times e^{-0.01 \times 3.000} = 0.9704$
 The term $\alpha_t \epsilon_i^t$ is computed as thus and it should be recomputed as each factor in the new firefly position is computed; $\alpha_t \epsilon_i^t = \alpha_0 \delta \times rand - 0.5$
 Take $rand = 0.32$
 $0.45 \times 0.97 \times 0.32 - 0.5 = -0.3603$, with all necessary terms values in the movement function gotten, the new firefly position is computed as thus;
- Step 1d The first parameter for the new firefly position is
 $x_1 = 15 + 0.9704(12 - 15) + (-0.3603) = 11.7285$
 Recompute $\alpha_t \epsilon_i^t = \alpha_0 \delta \times rand - 0.5$ for the next factor in the new firefly position using a different random number, take $rand = 0.67$
 $\alpha_t \epsilon_i^t = 0.45 \times 0.97 \times 0.67 - 0.5 = -0.2075$
 Therefore,
 $x_2 = 0.004 + 0.9704(0.0042 - 0.004) + (-0.2075) = -0.2033$
 Again recompute $\alpha_t \epsilon_i^t = \alpha_0 \delta \times rand - 0.5$ for the 3rd factor in the new firefly position using a different random number, take $rand = 0.12$
 $\alpha_t \epsilon_i^t = 0.45 \times 0.97 \times 0.12 - 0.5 = -0.44762$
 Therefore,
 $x_3 = 0.005 + 0.9704(0.03 - 0.05) + (-0.2075) = -0.1769$
 Once more recompute $\alpha_t \epsilon_i^t = \alpha_0 \delta \times rand - 0.5$ for the 3rd factor in the new firefly position using a different random number, take $rand = 0.55$
 $\alpha_t \epsilon_i^t = 0.45 \times 0.97 \times 0.55 - 0.5 = -0.2599$
 Then
 $x_4 = 0.01 + 0.9704(0.012 - 0.01) + (-0.2599) = -0.2479$
 Therefore, the new firefly position is $X_{new} = [11.7285, -0.2033, -0.1769, -0.2599]$
-

-
- Step 2a Check the if new firefly position is within the bounds of the search space of each parameter. If a parameter in the new firefly position is less than the lower bound of that particular parameter, then the value of the lower bound of that parameter replaces the computed value of that particular parameter in the newly computed firefly position and if a parameter in the new firefly position is greater than the upper bound of that particular parameter, then the value of the upper bound of that parameter replaces the computed value of that particular parameter in the newly computed firefly position
Therefore, applying the boundary check to X_{new} , gives $X_{new} = [11.7285, 0.001, 0.01, 0.1]$
 - Step 2b Compute the fitness value of X_{new}
 $F(x) = 20 \times 11.7285 + 0.002 \times 0.001 + 0.01 \times 0.01 + 0.03 \times 0.1 = 234.5704$
 - Step 3 Perform greedy selection by checking if the new firefly position fitness value is better i.e. lower than the fitness value of the first firefly in the population being considered since our objective is minimization, if X_{new} fitness value i.e. the new firefly position fitness value is less than X_i the current firefly position fitness value then X_i is discarded and replaced by X_{new} including their fitness values. On the other hand, if X_{new} fitness value is greater than X_i fitness value, then X_{new} and its corresponding fitness value are discarded thereby retaining the X_i and its fitness value. This concludes the movement and processes of the first firefly toward the more attractive second firefly
 - Step 4 Next, similarly, the first firefly fitness value is compared to the third (3rd) and 4th firefly fitness value in the population and step 1b to step 3 is repeated each time to complete the comparison of the first firefly with the rest of the firefly in the population
 - Step 5 Again the second firefly in the population goes through step 1b to step 3; likewise, the 3rd and the 4th firefly would go through the same procedure of step 1b to step 3 accordingly to complete the first iteration
 - Step 6 After the first iteration the fitness value of the new population is computed and the procedure from step 1b to step 3 is then repeated for the set or fixed numbers of iterations, which is set as 3, so the whole firefly algorithm would run 3 times, at the end of each iteration the best firefly fitness value and the corresponding firefly position are capture and stored
 - Step 7 The firefly optimization approach explained above is then coded using the python programming language
-

Extending the TP-BBD method to the TP-BBD-FF method

The details of the TP-BBD method for the optimization of the boring process while considering the IS 2062 E250 steel plates on the CNC machine were considered in Abdullahi and Oke [9]. As explained in the work, this kind of problem relies on the signal-to-noise ratio, which evaluates the desirable values of the signal and the undesirable values of noise. The signal-to-noise ratios are then used to produce higher-level surface roughness values with few essential runs. Thus, by applying the firefly in the integration of the Taguchi-Pareto and Box Behnken design problems, the attributes of the new method can be discerned from the experimental runs of the IS 2062 E250 steel plates on the CNC machine. The optimized method is set with the maximum iterations of 50, a firefly population of 200, initial randomness (alpha) of 1, a cooling factor (delta) of 0.97, an attractive constant (beta) of 1 and an absorption coefficient (gamma) of 0.01.

The new TP-BBD-FF method is similar to a previously developed method of TP-BBD-GA, which was discussed in detail in Abdullahi and Oke [10]. The TP-BBD-FF method has been validated using literature results from Patel and Deshpande [11]. The contribution of the present article shows the results of adjustment in the evolutionary algorithm from a replacement of the genetic algorithm to the firefly algorithm. Tables 1 and 2 show the results of the optimal solutions as the objective function produced depends on the Box Behnken design optimized parameters and regression equation optimized parameters, respectively.

The experimental plan started with the use of literature data from Patel and Deshpande [11] which we analyzed to obtain the optimal parametric setting using the various steps in the Taguchi method. We now expanded by using the Box Behnken method to

Table 2 Summary of values of variates *s*, *f*, *doc* and *nr*

Random values	<i>s</i> (rpm)	<i>f</i> (mm/rev)	<i>doc</i> (mm)	<i>nr</i> (mm)	<i>f</i> (<i>s</i> , <i>f</i> , <i>doc</i> , <i>nr</i>)
1st	842	0.084	1.37	0.84	918,547.7
2nd	824	0.784	1.32	0.82	898,911.3
3rd	878	0.084	1.41	0.88	957,820.5
4th	923	0.0692	1.13	0.92	1,006,911
5th	1166	0.0703	1.25	1.17	1,272,002

get the optimal solution of the parameters, including spindle speed, feed rate, depth of cut and nose radius. But still, we were not satisfied with the results of the Box Behnken method. So we further used the basic idea of linear programming to generate the objective function, which we used in the firefly optimization algorithm. But meanwhile, there is another scenario where we used the regression equation during the process of solving the problem with the Box Behnken method. We also solved the regression equation that was created with the firefly algorithm. In the end, the two scenarios were compared. It happened that the solution for the linear programming generated as the objective function was not as good as the regression equation generated during the process of solving using the Box Behnken Design method.

Results and discussion

Simulation results

The starting point of analysis is to set the population size, number of iterations and β_0 at some values. But population size often describes the number of individuals in a population. In the integrated TBB-FF method considered in this work, population size directly affects the ability of the firefly algorithm to search for an optimum solution within the search space. However, the number of iterations is the requirement to achieve optimal boring process parameters when the neutral population is chosen through random search. Moreover, β_0 is the attractive coefficient. Notice that these tiny-winged beetles called fireflies produce light (i.e. cold light as it has little or no heat generated when the light is flashed). On producing the light flashes, mates are attracted to this light generator and the value of the attractiveness is represented by a coefficient termed the attractiveness coefficient. So, in the work, the population size was set at 200. Although the population size in this implementation does not change throughout the program implementation, however, it is known that if convergence is not obtained satisfactorily, a resizing of the population size would be embarked upon to correct this unsatisfactory result. Next, the number of iterations is set as 50. Also, β_0 , referred to as beta or the attractiveness coefficient, is set at 1. The gamma symbol γ is set at 0.01 as it is common in the literature. Then, the α_0 is set at 1. This could be interpreted from the randomness strength. Also, the delta value is 0.97. However, in the literature, the delta value is always set to be any number between 0.95 and 0.97. Furthermore, in the firefly algorithm, the goal is to calculate and update values, whose calculation platform is laid as two matrices termed solution and $F(x)$, respectively. The solution contained values based on the variables considered in the objective function. In the present instance, consider the objective function given in Eq. (3):

$$\text{Objective_function } Z = 1090.9091*s + 0.06*f + 1.250*doc + 0.6061*nr \quad (3)$$

where s is the speed, f is the feed rate, doc is the depth of cut and nr is the nose radius.

The coefficients of (s , f , doc , nr) in Eq. (3) is the optimized parameters from the Box Behnken part of the research.

Equation (3), which is a linear programming-based objective function, was obtained from the optimal parameters of the BBD method. During the implementation of BBD, we optimized parameters using the Minitab software (2020). These optimized parameters are then used to generate the objective function using the basic principles of linear programming whereby we used the optimized parameters as the coefficients of each of the decision variables.

Looking closely at Eq. (3), where $f(s, f, doc, nr)$ has four variables which should be accounted for in the solution matrix, equivalent to four columns of variables with the starting variable being s , the next is f , this is followed by doc and the last variable is the nr . However, the values generated under each of these variables will be 200 items. But this is tedious for manual computation while the Python programming language is then used to do the computation. Besides, for an explanation, a population of 5 is used, indicating that there will be five rows of values for each variable under solutions and also for the function $f(s, f, doc, nr)$. Now, the first variable, s , is considered. By considering the experimental data on the cutting parameters with the associated levels indicated in Patel and Deshpande [11], the variable, speeds are of interest to us at this point. It is observed that there are four levels of speed, notably 800 rpm, 1000 rpm, 1200 rpm and 1400 rpm, respectively, for levels 1 to 4, it then implies that random members should be generated 200 times and values of speed between 800 and 1200 rpm are predicted. But for the simplicity of computations, we limit this manual explanation to the generation of five random speeds between 800rpm and 1200rpm. Therefore, to proceed, random numbers are generated through the calculator, in reality, codes were written in Python to generate the 200 random numbers needed at this time. But in this manual computation, only five random numbers are generated. Alternatively, a random number table, obtainable using the search word random number table from the Google search engine was used. For the table, depending on the version used, 13 columns of numbers running into 24 rows for each of the second to the thirteenth column was observed. In reading these numbers, the first number of the first column is read through to the last number in that column before moving to the first number in the second column. Thereafter, the movement continues along the column-wise then wisely until the numbers are exhausted, and then, you start again in that manner until will the necessary random numbers are exhausted. In this instance of obtaining randomly generated speeds, the first random number is 36,518. But comparing these numbers running with our range of numbers running from 80 to 120 rpm, it is outside it as the number must start with "80" and end with a number less than "1200". Next, 46,132 is outside the range and ignored. We move down to the fourth number in the first column of the random number table to have 84,180. This number can be modified as 842 rpm where the first three digits are recognized. But we need four more numbers which are then 8214 rpm from the eighth random

varieties of 82,448. The third, fourth and fifth random variates are 878 rpm, 923 rpm and 1166 rpm obtainable from the random numbers 87,789, 92,320 and 11,666, respectively. Remember that you are to generate 200 variates. But only five are used for illustration here. Next, we move to generate feed variate in the range 0.06mm/rev. Here, we start over again from the first random number in the table. The random numbers used are 84,180, 78,435, 82,448, 69,226 and 0.0703 number for the depth of cut, represented by “doc”, the range of values from the factor-level table is 1 mm, 1.24 mm, 1.4 mm and 1.5 mm. The generated random variates are 1.37 mm obtainable from the random numbers, approximated to 37 is put in front of 1 to make 1.37 mm and 1.25 mm. For the nose radius, the range of values is 0.08 mm and 1.2 mm. The random variates generated are 0.084 mm, 0.82 mm, 0.88 mm, 0.92 mm and 1.17 mm.

To apply the firefly algorithm, we have to compare each of the fireflies i.e. function 1 with the second function 2. Notice that for the first firefly, the $f(s, f, doc, nr)$ obtained is 918,547.7. The second, third, fourth and fifth fireflies have the following values: 898,911.3, 957,820.5, 1,006,911, and 1,272,002, respectively. Since $f_1 > f_2$, $918,547.7 > 898,911.3$ notice that the taste of the researcher is to minimize the surface roughness.

Notice that $f_1 = 918,547.7$ and $f_2 = 898,911.3$

The task is to minimize when the first firefly is considered at 918,547.7

Since $f_1 > f_2$, we move firefly 1 towards 2. But note that the condition for minimization is that $f_1 > f_2$. Furthermore, notice that firefly 1 means x_i while firefly 2 is x_j but $x_i = [8420.7841.320.82]$ while

$$x_j = [8240.841.320.82]$$

Now, r_{ij}^2 is to be calculated as follows:

$$\text{We have } (x_1, x_2) \text{ and } (y_1, y_2), \text{ then } r_{ij} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

But in this specific case, $x_1 = 842$, $x_2 = 0.084$, $x_3 = 1.37$ and $x_4 = 0.84$. Also $y_1 = 824$, $y_2 = 0.784$, $y_3 = 1.32$ and $y_4 = 0.82$. Then, r_{ij} is calculated as follows:

$$r_{ij} = \sqrt{(842 - 824)^2 + (0.084 - 0.784)^2 + (1.37 - 1.32)^2 + (0.84 - 0.82)^2}$$

$r_{ij} = 18.01$ while $r_{ij}^2 = 324.49$. We substitute this into Eq. (2) but with the interpretation of $\alpha_t \epsilon_i^\dagger$ as $\alpha_0 \delta (\text{rand} - \frac{1}{2})^*$ scale.

By looking through Eq. (2), we now know x_i , x_j , β_0 , and r_{ij}^2 . Then, we calculate $\beta_0 e^{-\gamma r_{ij}^2}$ as we know as 0.01 α_0 as 1, $\beta_0 = 1$. This is obtained as $1(e^{-0.01(324.49)}) = 0.0390$. Furthermore, the term $\alpha_0 \delta (\text{rand} - \frac{1}{2})^*$ scale is calculated where $\delta = 0.97$ and a random number newly generated is 0.3651 as read from the random number table. But concerning scaling, the scaling is the lower bound subtracted from the upper bound. However, four different parameters, namely, f , doc and nr , are involved and each parameter has its scale. For instance, reading from the experimental data provided by Patel and Deshpande [11] in their Table 4 of the article, s has a range of 1400–800, which is 600. But then, f has a range of 0.12–0.06, which is 0.06, doc has a range of 1.5–1, which is 0.5 while nr has a range of 1.2–0.8, which is 0.4. Now, since L for s is 600, α_0 is calculated as $0.01L$ initially, yielding 0.01 (600) or 6. But $\alpha_0 \delta (\text{rand} - \frac{1}{2})^*$ scale gives 6 (0.9) $(0.3651 - \frac{1}{2})^* 600$, which means -471.071 . Thus, $x_1 = 842 + 0.390 (824 - 842) - 4.1071$, which gives 1306.051. However, this is outside the 1200 upper

boundary for s , which means that the value 1306.051 is not valid. Thus, a recomputation is sought such that it will be within the boundaries of 800 and 1200. To achieve this, the random number is changed until this achieved. Once the random number has been changed to 0.46132, the term $\alpha_0\delta (\text{rand} - \frac{1}{2})^*$ scale gives -135.07 while x_1 gives 976.38, which is between 800 and 1400. We then move further to calculate x_2 along the following lives. First, the range of f from the Table 4 of Patel and Deshpande [11] is 0.06. Then, $0.01L$ is 0.0006. But $\alpha_0\delta (\text{rand} - \frac{1}{2})^*$ scale gives 0.0006 (0.97) $(0.46132 - \frac{1}{2})^*0.06 = -1.4E-0.6$. But x_2 is obtained as $0.084 + 0.039 (0.784 - 0.084) - (-1.4E-0.6) = 0.111301$. The value obtained for x_2 is within the range of 0.06 to 0.12, which is acceptable. Furthermore, for doc as a parameter, the range is 0.5 and $0.01L$ is 0.005. But $\alpha_0\delta (\text{rand} - \frac{1}{2})^*$ scale is (0.005) (0.9) $(0.46132 - \frac{1}{2})^*0.005 = -9.4E - 0.5$. The value calculated for x_3 is $1.37 + 0.039 (1.32 - 1.37) - (-9.4E - 0.5) = 1.38144$ which is within the range of 1 and 1.5. Besides, for nr as a parameter, the range is 0.4 and the values are from 0.8 to 1.2. But $0.01L$ is 0.04. Next, $\alpha_0\delta (\text{rand} - \frac{1}{2})^*$ scale is 0.004 (0.97) $(0.46132 - \frac{1}{2})^* 0.4 = -6E - 0.5$. The value calculated for x_4 is $0.84 + 0.039 (0.82 - 0.84) - (-6E - 0.5) = 0.83928$, which is a number falling in the range of nr of 0.8 to 1.2.

From the foregoing, the values of X_{new} is displayed in a matrix as follows:

$$X_{\text{new}} = [976.386 \ 0.111301 \ 1.368144 \ 0.83928] .$$

By substituting the value of X_{new} into the objective function, $f(X_{\text{new}})$ is obtained as $(1090.9091 \times 976.3686) + (0.06 \times 0.111301) + (1.25 \times 1.368144) + (0.6061 \times 0.83928) = 1,065,132$. But $f(X_{\text{new}}) = 1,065,132$ compared with $f = 918,547.7$. The new value is still higher than f_1 , and therefore, it will be rejected since minimum value is expected for updating. Here, the random number is changed and computation commences. After a success at obtaining a lower value for $f(X_{\text{new}})$, the first firefly is done with. Then, we move to the next firefly. The procedure is followed until the finalized results in Table 3 are obtained for all iterations.

Next, the whole procedure above is used in Eq. (4) to obtain the results summarized in Table 3.

$$\begin{aligned} \text{Objective_function(regressionequation)}Z = & -69.3 + 0.0233 * \text{Speed} - 3 * \text{Feed} \\ & + 2.5 * \text{Depth_of_cut} + 10.92 * \text{Nose_radius} \\ & - 0.000010 * \text{Speed} * \text{Speed} + 35 * \text{Feed} * \text{Feed} \\ & - 1.01 * \text{Depth_of_cut} * \text{Depth_of_cut} \\ & - 7.72 * \text{Nose_radius} * \text{Nose_radius} \\ & - 0.0000 * \text{Speed} * \text{Feed} \\ & + 0.00000 * \text{Speed} * \text{Depth_of_cut} \\ & - 0.00092 * \text{Speed} * \text{Nose_radius} \\ & - 0.0 * \text{Feed} * \text{Depth_of_cut} \\ & - 9.2 * \text{Feed} * \text{Nose_radius} \\ & + 0.00 * \text{Depth_of_cut} * \text{Nose_radius} \end{aligned} \tag{4}$$

We then combined the outcome in addition to form the objective function. Equation (4) was generated using the Minitab software version 2020 in obtaining those optimal parameters we used as the second scenario objective function. The function activated in the software is the regression model.

Table 3 Optimal solutions—Box Behnken design optimized parameters

Iteration	Optimal solutions
1	[805.1189387573768, 0.12, 1.0638638452004148, 0.07543349743436445]
2	[804.2262996003564, 0.06, 1, 0.07134658940265949]
3	[802.2754678354629, 0.10169228545563641, 1.0615785266576656, 1.0712213177246597]
4	[800, 0.06, 1, 0]
5	[800, 0.06, 1, 0]
6	[800, 0.06, 1, 0]
7	[800, 0.06, 1, 0]
8	[800, 0.06, 1, 0]
9	[800, 0.06, 1, 0]
10	[800, 0.06, 1, 0]
	The best firefly at the end of 10th iteration is 872,728.534
11	[800, 0.06, 1, 0]
12	[800, 0.06, 1, 0]
13	[800, 0.06, 1, 0]
14	[800, 0.06, 1, 0]
15	[800, 0.06, 1, 0]
16	[800, 0.06, 1, 0]
17	[800, 0.06, 1, 0]
18	[800, 0.06, 1, 0]
19	[800, 0.06, 1, 0]
20	[800, 0.06, 1, 0]
	The best firefly at the end of 20th iteration is 872,728.534
21	[800, 0.06, 1, 0]
22	[800, 0.06, 1, 0]
23	[800, 0.06, 1, 0]
24	[800, 0.06, 1, 0]
25	[800, 0.06, 1, 0]
26	[800, 0.06, 1, 0]
27	[800, 0.06, 1, 0]
28	[800, 0.06, 1, 0]
29	[800, 0.06, 1, 0]
30	[800, 0.06, 1, 0]
	The best firefly at the end of 30th iteration is 872,728.534
31	[800, 0.06, 1, 0]
32	[800, 0.06, 1, 0]
33	[800, 0.06, 1, 0]
34	[800, 0.06, 1, 0]
35	[800, 0.06, 1, 0]
36	[800, 0.06, 1, 0]
37	[800, 0.06, 1, 0]
38	[800, 0.06, 1, 0]
39	[800, 0.06, 1, 0]
40	[800, 0.06, 1, 0]
	The best firefly at the end of 40th iteration is 872,728.534
41	[800, 0.06, 1, 0]
42	[800, 0.06, 1, 0]
43	[800, 0.06, 1, 0]
44	[800, 0.06, 1, 0]
45	[800, 0.06, 1, 0]

Table 3 (continued)

Iteration	Optimal solutions
46	[800, 0.06, 1, 0]
47	[800, 0.06, 1, 0]
48	[800, 0.06, 1, 0]
49	[800, 0.06, 1, 0]
50	[800, 0.06, 1, 0]
	The best firefly at the end of 50th iteration is 872,728.5336000001
Optimal solution	[800, 0.06, 1, 0]

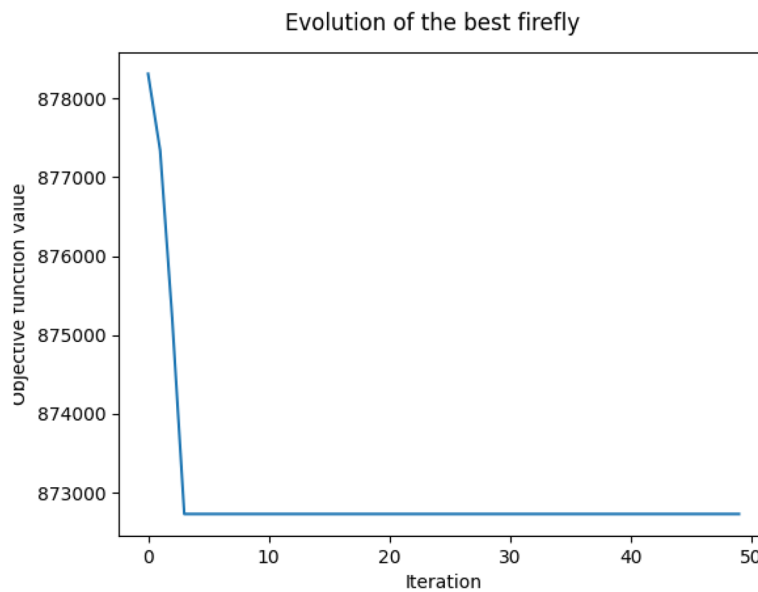


Fig. 1 Plot when objective function is generated using optimized bbd parameters

Firefly optimization data when the objective function is generated using the Box Behnken design optimized parameters

The following are the characteristics of the optimization model and these attributes are produced in Table 3 and the pictorial description in Fig. 1.

- Maximum iteration = 50
- Firefly Population = 200
- Initial randomness (alpha) = 1
- Cooling factor (delta) = 0.97
- Attractive constant (beta) = 1
- Absorption coefficient (gamma) = 0.01

The purpose of Table 3 is to summarize the results of the simulation, providing information on the performance of each of the parameters of speed, feed, depth of cut and nose radius regarding the best and worst parameters. In a viewing activity, optimal values for each parameter after each iteration are displayed. To enrich our understanding, the best firefly after ten iterations are shown to facilitate comparison. For the optimized BBD parameters, Table 3 displays the best firefly based on Eq. (3) and instantly shows

Table 4 Optimal solutions—regression equation

Iterations	Optimal solution
1	[1140.1158331648326, 0.06, 1.2840913289854767, 0.5797149545201336]
2	[1117.222748864728, 0.06, 1.234945703097946, 0.5897388383153699]
3	[1117.222748864728, 0.06, 1.234945703097946, 0.5897388383153699]
4	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
5	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
6	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
7	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
8	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
9	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
10	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
The best firefly at the end of 50th iteration is – 51.430	
11	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
12	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
13	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
14	[1127.8996008001955, 0.06, 1.2635041040436774, 0.5999608499125828]
15	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
16	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
17	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
18	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
19	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
20	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
The best firefly at the end of 20th iteration is – 51.429	
21	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
22	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
23	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
24	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
25	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
26	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
27	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
28	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
29	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
30	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
The best firefly at the end of 30th iteration is – 51.429	
31	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
32	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
33	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
34	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
35	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
36	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
37	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
38	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
39	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
40	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
The best firefly at the end of 40th iteration is – 51.429	
41	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
42	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
43	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
44	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
45	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]

Table 4 (continued)

Iterations	Optimal solution
46	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
47	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
48	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
49	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
50	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]
	The best firefly at the end of 50th iteration is – 51.42868856154767
Optimal solution	[1134.0310579393426, 0.06, 1.245121420058297, 0.6065968439671544]

whether the surface roughness has been improved or not. Viewing each iteration, a matrix containing four elements is shown, which shows the results of the speed, feed, depth of cut and nose radius. For instance, after iteration 1, the speed parameter was 805.12 rpm, the feed is 0.12 mm/rev, the depth of cut is 1.06 mm and the nose radius is 0.075 mm. Then, the surface roughness, measured as roughness average, Ra is computed by substituting the values of s, f, doc and nr in Eq. (3). The roughness average value obtained after the first iteration is 878,314.2. Although this is not shown in Table 3, it is stored in a space within the python programming language. However, an improvement on this roughness average value is desired in iteration two and subsequent iterations. Now, moving to iteration two, the roughness average, Ra, is obtained by substituting the value of speed as 804.23 rpm, feed, depth of cut and nose radius as 0.06 mm/rev 1 mm and 0.07 mm, respectively (Table 3). By substituting Eq. (3), we obtained 877,342.9 as the roughness average. This is the result after iteration two. But we need further computations to confirm whether there are improvements or not and convergence is shown when there is no further improvement in the roughness average value over many iterations. In this work, a smaller roughness average indicates an improvement result of the roughness average after iterations one and two, which yielded 87,831.2 and 877,342.9, respectively. The effectiveness of the firefly algorithm is shown in the further reduction of the roughness average, which indicates the surface roughness at each iteration until it stabilizes. Therefore, after the tenth iteration, Ra was obtained as 872,728.534. This remained the value of the Ra after the twentieth, thirtieth and fortieth iterations. Furthermore, when additional iterations were run, to obtain value after the fiftieth iteration, only a very marginal decrease of 0.00639999 was observed. But considering the computer resources, it is not worthwhile to keep the simulation running with very minimal charges. Thus, the simulation was terminated after the fiftieth iteration. Therefore, using the optimized BBD parameters, the optimal output obtained (Table 3) is 872,728.533600001. Besides, by following the procedure analysed here, which was deployed to understand Table 4 are obtained for the optimized procedure when as the objective function.

However, Figs. 1 and 2 show the progress made after every ten iterations for stability and convergence. Take Fig. 1 as an instance, when substituting the values of the simulated parameters into the objective function (Eq. 3) to obtain the roughness average, convergence is quickly obtained roughly after the third iteration. This is shown by a

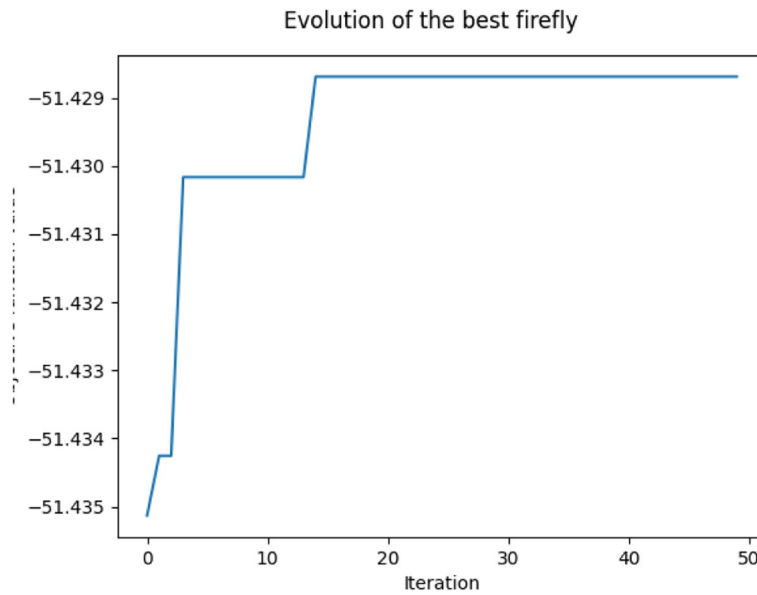


Fig. 2 Plot when the regression equation is taken as objective function

sharp drop in the objective function value from iteration zero when the simulation has not started to iteration three). It then moves on constantly at roughly 872,728.53 until termination of the procedure was initiated as a result of perceived waste of computer resources if simulation continues beyond this iteration fifty.

Firefly optimization data when regression equation is used as objective function

The following are the characteristics of the optimization model, and these attributes are produced in Table 4 and the pictorial description in Fig. 2.

- Maximum iteration = 50.
- Firefly population = 200.
- Initial randomness (alpha) = 1.
- Cooling factor (delta) = 0.97.
- Attractive constant (beta) = 1.
- Absorption coefficient (gamma) = 0.01.

Now, the characteristics of Fig. 2 are different from Fig. 1 as there are multiple points shown as stability. Consider Fig. 2 and the changes in the objective function value along the Y-axis. Because the simulation produces negative Ra, a decrease is shown by a rewards movement of the line. For this Fig. 2, at iterations one to three, there seems to be stability but it is short-lived. There is then a sharp drop in value between iterations there and four to iteration thirteen. But surprisingly, a further reduction in the objective function values was experienced at iteration thirteen which remained stable until iteration fifty. Then, the procedure was terminated at iteration fifty and the roughness average value, obtained from Table 4 as - 51.428688 is taken as the desired.

Conclusions

In this work, the problem solved concerns the determination of optimal surface roughness of the IS 2062 E250 steel plates during the boring process. The firefly evolutionary algorithm-based TP-BBD method used for the boring operation of the IS 2062 E250 steel plates permits solving optimization problems for surface roughness optimization. It was concluded that the use of the TBB-FF method was feasible with the experimental data considered; the new development in the present work is the introduction of the firefly algorithm into an existing integrated method of the Taguchi method and Box Behnken Design method. The method's validation process reveals that it is feasible to achieve modifications comparable to those achieved in a previous work where the genetic algorithm replaces the firefly algorithm. This paper contributed to the boring optimization literature by providing a robust TBBS-FF method for the optimization of boring operations parameters. The method presented exhibits unusual characteristics of good efficiency for particular problems. Besides, since the cost of implementation of the procedure is often of concern, the proposed method also assures that only a small number of iterations are needed to obtain optimal solutions. Precisely, this paper differs from previous Taguchi-BBD optimization studies by providing a robust TBBD-FF to handle efficiency problems while reducing the computational cost of the method. This paper also differs by introducing a nonlinear optimization function, with changing directions each time it is implemented. Furthermore, we discovered that better surface roughness was achieved than previously developed methods on the integrated Taguchi method-Box Behnken design-Teaching learning-based optimization (TBB-TLBO) method. Besides, the computational convergence results even exceed the alternative TBB-TLBO method because of the nonlinear function embedded in the firefly algorithm. Also, the optimal solutions of the TBB-FF algorithm method exceed those obtained for the TBB-TLBO method. This particular work was treated as a single objective function problem where the parameters were related to a single goal of the boring process such as the improvement of the surface integrity of the steel plates. However, the goal of the boring process may be many, including optimizing the material removal process in addition to the surface roughness reduction effort. By looking into the material removal process optimization incorporated with the reduction in the surface roughness of the steel plates, multiple objective functions could be introduced as a future research aspect. Here, the firefly algorithm component can be edified to solve the multiple objective function optimization problems. At the end of the day, the machining workshop can utilize the optimal parameters obtained from the TBB-FF method to be more efficient. The implementation of the optimal solutions from the TBB-FF method in the machining shop makes the job easier. Instead of being confronted with several sub-optimal solutions, the system can utilize an optimal solution to get the job done fast. It also provides a benchmark with which the current performance could be judged. The output is also generated at a very high quality. Notwithstanding, the limitation of this work is the dependence on published work which limits the flexibility in analysis. This could be corrected in a future work that should focus on collecting experimental data from the workshop.

Abbreviations

EMM	Electrochemical micromachining
ECDM	Electrochemical discharge machining
ROC	Radial overcut (ROC)
Ton	Pulse-on time
Toff	Pulse-off time
WT	Wire tension
I	Pulse current
PSO	Particle swarm optimisation
GA-AIS	Genetic Algorithm and Artificial Immune System
VMC	Vertical milling center
MMC	Metal matrix composites
WEDM	Wire electrical discharge machining
NSPSO	Non-dominated particle swarm optimisation
NSFA	Non-dominated sorting firefly algorithm
RSM	Response surface methodology
Ra	Surface roughness
HS	Harmony search
CS	Cutting speed
f	Feed rate
d	Depth of cut
CAPP	Computer-aided process planning
UPC	Unit production cost
NSGA-II	Non-dominated sorting genetic algorithm
Taguchi's DOE	Taguchi's design of the experiment
MRR	Material removal rate
ANOVA	Analysis of variance
GFRP	Glass fibre-reinforced polymer
TP-BBD	Taguchi-Pareto-Box Behnken Design
TP-BBD-FF	Taguchi-Pareto-Box Behnken design-firefly
TLBO	Teaching learning-based optimisation

Acknowledgements

Not applicable

Authors' contributions

EF contributed to the writing of the manuscript; BAS contributed to the writing of the manuscript; YUA conceived, analysed and interpreted the data; KAA contributed to the writing of the manuscript; JR contributed to the writing of the manuscript; SAO conceived and interpreted the data and also supervised computational experimentation and contributed to writing the manuscript, and EOO contributed to the writing of the manuscript. The authors read and approved the final manuscript.

Funding

No funding was received for the work.

Availability of data and materials

Data for the analysis was extracted from Abdullahi and Oke [9] and Patel and Deshpande's [11] study, and they are in the open-access domain.

Declarations**Competing interests**

The authors declare that they have no competing interests.

Received: 1 March 2023 Accepted: 3 May 2023

Published online: 18 May 2023

References

1. Dostál, P. (2013). The use of optimization methods in business and public services. In: Zelinka, I., Snášel, V., Abraham, A. (eds) Handbook of Optimization. Intelligent Systems Reference Library, vol 38. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-30504-7_29
2. Dutta A, Das A, Joshi SN (2017) Optimum process parameters for efficient and quality thin wall machining using firefly algorithm. *Int J Additive Subtract Mat Manufact* 1:3–22
3. Kant G, Sangwan KS (2014) Prediction and optimization of machining parameters for minimizing power consumption and surface roughness in machining. *J Clean Prod* 83:151–164
4. Kumari S, Kumar A, Yadav RK, Vivekananda K (2018) Optimisation of machining parameters using grey relation analysis integrated with harmony search for turning of AISI D2 steel. *Materials Today: Proceedings* 5:12750–12756

5. Lee YZ, Ponnambalam SG (2012) Optimisation of multipass turning operations using PSO and GA-AIS algorithms. *Int J Prod Res* 50:6499–6518
6. Nalbant M, Gökaya H, Sur G (2007) Application of Taguchi method in the optimization of cutting parameters for surface roughness in turning. *Mater Des* 28:1379–1385
7. Nee CY, Saad MS, Mohd Nor A, Zakaria MZ, Baharudin ME (2018) Optimal process parameters for minimizing the surface roughness in CNC lathe machining of Co28Cr6Mo medical alloy using differential evolution. *Int J Adv Manufact Technol* 97:1541–1555
8. Senthilkumar N, Tamizharasan T, Gobikannan S (2014) Application of response surface methodology and firefly algorithm for optimizing multiple responses in turning AISI 1045 steel. *Arab J Sci Eng* 39:8015–8030
9. Abdullahi YU, Oke SA (2022) Optimizing the boring parameters on CNC machine using IS 2062 E250 steel plates: Taguchi-Pareto-Box Behnken design and Taguchi-ABC-Box Behnken design perspectives. *Engineering Access* 8(2):219–241
10. Abdullahi YU, Oke SA (2022) Optimization and selection of boring process parameters for IS 2062 E250 steel plates using hybrid Taguchi-Pareto Box Behnken-genetic algorithm method. *Indones J Ind Eng Manag* 3(2):131–150
11. Patel, M., & Deshpande, V. 2014. Application of taguchi approach for optimization roughness for boring operation of E 250 B0 for standard IS: 2062 on CNC TC 2(2) :2528–2537.
12. Prasanth, P., Sekar, T., & Sivapragash, M. (2021). Investigations on the effects of nitrogen gas in CNC machining of SS304 using Taguchi and Firefly Algorithm. *Bull Polish Acad Sci Tech Sci* 69(1):e136211. <https://doi.org/10.24425/bpasts.2020.136211>
13. Singh D, Shukla RS (2016) Optimisation of electrochemical micromachining and electrochemical discharge machining process parameters using firefly algorithm. *Int J Mechatron Manuf Syst* 9:137–159
14. Hebbbar G, D'Mello G, Pai PS (2018) Surface roughness optimization in machining of biodegradable magnesium alloys. *Materials Today: Proceedings* 5:11787–11793
15. Raja SB, Pramod CVS, Krishna KV, Ragunathan A, Vinesh S (2015) Optimization of electrical discharge machining parameters on hardened die steel using Firefly Algorithm. *Eng Comput* 31:1–9
16. Majumder A, Das A, Das PK (2018) A standard deviation based firefly algorithm for multi-objective optimization of WEDM process during machining of Indian RAFM steel. *Neural Comput Appl* 29:665–677
17. Johari NF, Zain AM, Mustaffa NH, Udin A (2017) Machining parameters optimization using hybrid firefly algorithm and particle swarm optimization. *J Phys Conf Ser* 892:012005
18. Thamizhmanii S, Sapparudin S, Hasan S (2007) Analyses of surface roughness by turning process using Taguchi method. *J Ach Materials Manufact Eng* 20:503–506
19. Rudrapati R, Patil A (2019) Optimization of cutting conditions for surface roughness in VMC 5-Axis. *Mater Sci Forum* 969:631–636
20. Pujara JM, Kothari KD, Gohil AV (2017) Process parameter optimization for MRR and surface roughness during machining LM6 aluminum MMC on WEDM. *Adv Eng Forum* 20:42–50
21. Liu Z, Li X, Wu D, Qian Z, Feng P, Rong Y (2019) The development of a hybrid firefly algorithm for multi-pass grinding process optimization. *J Intell Manuf* 30:2457–2472
22. Zubair AF, Mansor MS (2019) Embedding firefly algorithm in optimization of CAPP turning machining parameters for cutting tool selections. *Comput Ind Eng* 135:317–325
23. Dhandapani K, Vasanthkumar P, Nagarajan S (2014) A meta-heuristic evolutionary algorithm to optimize machining parameters in turning AISI 4340 steel. *J Adv Eng Res* 1:105–113
24. Palanikumar K, Karunamoorthy L, Karthikeyan R (2006) Assessment of factors influencing surface roughness on the machining of glass fiber-reinforced polymer composites. *Mater Des* 27:862–871
25. Yang WH, Tarnq YS (1998) Design optimization of cutting parameters for turning operations based on the Taguchi method. *J Mater Process Technol* 84(1–3):122–129
26. Abiola IT, Oke SA (2022) Fuzzy analytic hierarchy process and markov chain-WSM/WPM/WASPAS approaches to solving the surface roughness problem in the boring of carbon steel IS 2062 GR E250 plates on CNC. *Indones J Indus Eng Manag* 3(1):47–71
27. Ighravwe DE, Oke SA (2015) The application of artificial neural network and fuzzy inference system for machine process with respect to material removal rate and surface roughness. *Kenya J Mech Eng* 8(1):9–23
28. Ighravwe DE, Oke SA (2022) Surface roughness prediction and optimization using novel joints artificial neural network and bat algorithm. *Int J Integrat Eng* 14(4):20–34
29. Yang X-S (2010) Firefly algorithm, stochastic test functions and design optimisation. *Int J Bio Inspired Comput* 2:78–84
30. Abdullahi YU, Oke SA (2022) Optimising the machining process of IS 2062 E250 steel plates with the boring operation using a hybrid Taguchi-Pareto Box Behnken-teaching learning-based algorithm. *Indones J Ind Eng Manag* 4(2):49–63