Journal of Artificial Intelligence and Data Science (JAIDA) Vol. 1, No. 1, pp. 106-115, 2021
Research Article
https://jaida.ikcu.edu.tr/

Electron Beam Welding (EBW) of Aerospace Alloy (Inconel 825): Optimization and Modeling of Weld Bead Area

Gamze ÖZAKINCI [1], Levent AYDIN [2,*]

Department of Mechanical Engineering, İzmir Katip Çelebi University, Çiğli, İzmir, Turkey

Abstract
This study investigates the optimum weld area on a popular aerospace alloy (i.e., Inconel 825) made by the electron beam welding technique. Welding speed (S), beam current (I), accelerating voltage (V), and beam oscillation (O) are considered as process parameters to study the weld bead area (WA) of the weldments. An instructive study on multiple non-linear neural regression analyses has been done as a basic introduction to neuro regression modeling with artificial neural network (ANN) philosophy. To do this, the experimental prediction has been modeled with 14 predictive functional structures using fundamental regression modal types to test the accuracy of their predictions. To train the program with the chosen model $R^2_{training}$, test it $R^2_{testing}$, verify the accuracy $R^2_{validation}$ is used, and check whether the values are within the engineering limits. Optimization algorithms with three different scenarios have been applied. Only one of the 14 models gave realistic results. It has been seen that the scenario types, selection of different constraints, and different models for design variables affect the optimization results.
Keywords: Electron beam welding; neuro-regression modeling; optimization; weld bead area.

1. Introduction

Nickel and its alloys (Inconel), being popular aerospace alloys, have gained popularity in the manufacturing industry because of their excellent properties but processing them has concerns, so researchers had to utilize new methods such as electron beam welding (EBW). In addition, the influence of welding processes and their parameters on weld characteristics must be investigated.

The Ni-base alloys are used in a wide range of applications in engineering systems exposed to extreme conditions, such as highly corrosive and high-temperature environments; for this reason, they are often used as an aerospace alloy and marine, automobile nuclear, chemical processing industries. Most of these applications require the use of welding processes during manufacture. The weld must perform to a level similar to the base metal [1, 2]. However, the processing or fabrication of this form of nickel-based alloys is not easy and has shown some concern. Formation of deleterious phases like laves phase, segregation of alloying element, and the tendency of micro-fissuring are considered the significant concerns affecting welded joints' mechanical characteristics. Hence, the implementation of statistical and soft computing-based approaches is essential [1]. Researchers utilized different welding methods such as friction stir welding (FSW), gas metal arc welding (GMAW), gas tungsten arc welding (GTAW), higher energy-intensive welding techniques like electron beam welding (EBW), and laser beam welding (LBW), to weld nickel/titanium alloys [3].

Mechanical properties and bead geometry of weld play great importance in controlling weld quality. Cross-section of weld-bead geometry defines the distortion and residual stresses induced while the shape of weld bead geometry rules mechanical properties of the weld. In welding, weld quality must remain the same as the base metal. This can be achieved by the appropriate selection and controlling welding variables. Therefore, the implementation of statistical approaches is essential [1, 5].

Researchers have utilized multiple linear regression (MLR) and soft computing-based intelligent approaches like artificial neural network (ANN), adaptive neuro-fuzzy inference system (ANFIS), and fuzzy logic (FL) to predict different responses in different welding processes. *Soft computing* is an approach that brings together the human mind with an environment of uncertainty and imprecision [4]. These approaches can describe the complex and non-linear behavior of the responses concerning its process parameter with success.

The literature on predicting various weld bead characteristics can be discussed as follows.

Palanivel et al. [6] worked on a backpropagation (BP) based ANN approach and RSM for predicting ultimate tensile strength in a titanium tube. They combined the developed model with genetic algorithms to optimize the GTAW process parameters to obtain the best weld bead geometry. Zaharudin et al. [7] developed ANFIS and ANN models to predict the welding strength of resistance spot welded CR780 specimens. Prediction using the ANN model is found accurate than the RSM approach [8]. Gyasi et al. [9] predicted the structural integrity of GMAW welded UHSS welded joins using the ANN approach. Satpathy et al. [10] used regression, ANN, and ANFIS to predicting the joining strength of the aluminum-copper dissimilar welded joins. Narang et al. [11]

*Corresponding author e-mail address: levent.aydin@ikcu.edu.tr

Received: 24/07/2021; Accepted: 25/08/2021

established a fuzzy logic (FL) model to predict the weld bead geometry of TIG-welded structural steel weldments. Sivagurumanikandan et al. [12] applied RSM and ANN-based modeling methods to study the influence of welding variables on the optimal strength of weldments. Akbari et al. [13] developed a numerical model to estimate the temperature distribution and weld geometry of laser-welded titanium alloy weldments. Anand et al. [14] used different strategies for training the ANN model for predicting weld strength in FSW of Incoloy 800 weldments. Both models gave excellent results in the prediction of the welded specimens. ANN model trained with backpropagation (BP) algorithm was used to predict WA on stainless steel by Balasubramanian et al. [15].

After the literature studies in the relevant area, we have introduced a novel approach on a modeling design-optimization process to see the optimum weld area of Inconel 825, which is influenced by the welding speed (S), beam current (I), accelerating voltage (V) and beam oscillation (O):

- A detailed study on multiple non-linear neuro-regression analyses, including linear, quadratic, trigonometric, logarithmic, and their rational forms for the output of our process, has been performed.
- The boundedness of the candidate models has been checked to provide generating realistic values.
- The different direct search methods have been performed methodically, including stochastic optimization algorithms (modified versions of differential evaluation, Nelder-Mead, random search, and simulated annealing algorithms).

2. Materials and Methods

2.1. Modeling

In the modeling phase, the predictions are tested with the regression analysis. Regression models generally estimate the degree of correlation between input and output variables and determine their relationship form. Linear regression is mainly fitted by the least-squares method, but it can also be fitted by other methods, such as minimizing the "underfitting" in some other specifications or minimizing the penalty version of the least square loss function (such as Ridge regression). Linear regression is divided into two categories: simple and multiple linear regression [5].

With this method, the data was split into three parts; 84%, 10%, and 6% of the chosen data calculated for $R^2_{training}$, $R^2_{testing}$, and $R^2_{validation}$, respectively. This process reduces the error between predicted and the experimental results by changing the models in *Table 1* for regression and splitting the data into the correct sections. However, the resulting R^2 values are not sufficient to confirm the accuracy of the results; therefore, apart from the results found from the candidates, maximum and minimum predicted values have been found and checked if the results are within the engineering limits. In addition, optimization algorithms with three different scenarios have been tried, and the optimum number of inputs investigated [16, 17].

2.2. Optimization

In essence, the optimization of the structure can be described as obtaining the best design by minimizing or maximizing a single specified goal or multiple goals corresponding to all constraints. Optimization techniques can be divided into traditional and non-traditional. Traditional optimization techniques are only suitable for continuous and differentiable functions, such as constraint changes and Lagrangian multipliers. In engineering design problems, traditional optimization techniques cannot be used due to particularities. In these cases, stochastic optimization methods such as genetic algorithm (GA), particle swarm (PS), and simulated annealing (SA) are advantageous. However, due to the nature of stochastic methods, accurate solutions cannot be obtained, and using more than one method with different phenomenological foundations for the same optimization problem will increase the solution's reliability.

The mathematical optimization problem handled in this article has the following issues:

- Multiple non-linear objective functions,
- Objective functions having many local extremum points,
- Mixed-integer (discrete) - continuous nature of the design variables
- Non-linear constraints.

The optimization scenarios discussed in this study include the condition given in the first three items. In addition, four different optimization algorithms (MDE, MNM, MSA, MRS) have been selected to solve optimization problems.

Table 1: *Multiple regression model types including linear, quadratic, trigonometric, logarithmic, their rational forms and hybrid forms*

Model Name	Nomenclature	Formula
Multiple linear	L	a[1] + a[2] x1 + a[3] x2 + a[4] x3 + a[5] x4
Multiple linear rational	LR	(a[1] + x1 a[2] + x2 a[3] + x3 a[4]) / (b[1] + x1 b[2] + x2 b[3] + x3 b[4])
Second order multiple nonlinear	SON	a[1] + a[2] x1 + a[3] x2 + a[4] x3 + a[5] x4 + a[6] x1^2 + a[7] x2^2 + a[8] x3^2 + a[9] x4^2 + a[10] x1 x2 + a[11] x1 x3 + a[12] x1 x4 + a[13] x2 x3 + a[14] x2 x4 + a[15] x3 x4
Second order multiple nonlinear rational	SONR	(a[1] + 2 x1 a[2] + x1^2 a[3] + 2 x2 a[4] + 2 x1 x2 a[5] + x2^2 a[6] + 2 x3 a[7] + 2 x1 x3 a[8] + 2 x2 x3 a[9] + x3^2 a[10] + 2 x4 a[11] + 2 x1 x4 a[12] + 2 x2 x4 a[13] + 2 x3 x4 a[14] + x4^2 a[15]) / (b[1] + 2 x1 b[2] + x1^2 b[3] + 2 x2 b[4] + 2 x1 x2 b[5] + x2^{2} b[6] + 2 x3 b[7] + 2 x1 x3 b[8] + 2 x2 x3 b[9] + x3^2 b[10] + 2 x4 b[11] + 2 x1 x4 b[12] + 2 x2 x4 b[13] + 2 x3 x4 b[14] + x4^2 b[15])
Third order multiple nonlinear	TON	a[1] + 3 x1 a[2] + 3 x1^2 a[3] + x1^3 a[4] + 3 x2 a[5] + 6 x1 x2 a[6] + 3 x1^2 x2 a[7] + 3 x2^2 a[8] + 3 x1 x2^2 a[9] + x2^3 a[10] + 3 x3 a[11] + 6 x1 x3 a[12] + 3 x1^2 x3 a[13] + 6 x2 x3 a[14] + 6 x1 x2 x3 a[15] + 3 x2^2 x3 a[16] + 3 x3^2 a[17] + 3 x1 x3^2 a[18] + 3 x2 x3^2 a[19] + x3^3 a[20] + 3 x4 a[21] + 6 x1 x4 a[22] + 3 x1^2 x4 a[23] + 6 x2 x4 a[24] + 6 x1 x2 x4 a[25] + 3 x2^2 x4 a[26] + 6 x3 x4 a[27] + 6 x1 x3 x4 a[28] + 6 x2 x3 x4 a[29] + 3 x3^2 x4 a[30] + 3 x4^2 a[31] + 3 x1 x4^2 a[32] + 3 x2 x4^2 a[33] + 3 x3 x4^2 a[34] + x4^3 a[35]
First order trigonometric multiple nonlinear	FOTN	a[1] + a[2] Sin[x1] + a[3] Sin[x2] + a[4] Sin[x3] + a[5] Sin[x4] + a[6] Cos[x1] + a[7] Cos[x2] + a[8] Cos[x3] + a[9] Cos[x4]
First order trigonometric multiple nonlinear rational	FOTNR	(a[1] + a[2] Sin[x1] + a[3] Sin[x2] + a[4] Sin[x3] + a[5] Sin[x4] + a[6] Cos[x1] + a[7] Cos[x2] + a[8] Cos[x3] + a[9] Cos[x4]) / (b[1] + b[2] Sin[x1] + b[3] Sin[x2] + b[4] Sin[x3] + b[5] Sin[x4] + b[6] Cos[x1] + b[7] Cos[x2] + b[8] Cos[x3] + b[9] Cos[x4])
Second order trigonometric multiple nonlinear	SOTN	a[1] + a[2] Sin[x1] + a[3] Sin[x2] + a[4] Sin[x3] + a[5] Sin[x4] + a[6] Cos[x1] + a[7] Cos[x2] + a[8] Cos[x3] + a[9] Cos[x4] + a[10] Sin[x1]^2 + a[11] Sin[x2]^2 + a[12] Sin[x3]^2 + a[13] Sin[x4]^2 + a[14] Cos[x1]^2 + a[15] Cos[x2]^2 + a[16] Cos[x3]^2 + a[17] Cos[x4]^2
Second order trigonometric multiple nonlinear rational	SOTNR	(a[1] + a[2] Sin[x1] + a[3] Sin[x2] + a[4] Sin[x3] + a[5] Sin[x4] + a[6] Cos[x1] + a[7] Cos[x2] + a[8] Cos[x3] + a[9] Cos[x4] + a[10] Sin[x1]^2 + a[11] Sin[x2]^2 + a[12] Sin[x3]^2 + a[13] Sin[x4]^2 + a[14] Cos[x1]^2 + a[15] Cos[x2]^2 + a[16] Cos[x3]^2 + a[17] Cos[x4]^2) / (b[1] + b[2] Sin[x1] + b[3] Sin[x2] + b[4] Sin[x3] + b[5] Sin[x4] + b[6] Cos[x1] + b[7] Cos[x2] + b[8] Cos[x3] + b[9] Cos[x4] + b[10] Sin[x1]^2 + b[11] Sin[x2]^2 + b[12] Sin[x3]^2 + b[13] Sin[x4]^2 + b[14] Cos[x1]^2 + b[15] Cos[x2]^2 + b[16] Cos[x3]^2 + b[17] Cos[x4]^2)
First order logarithmic multiple nonlinear	FOLN	a[1] + a[2] Log[x1] + a[3] Log[x2] + a[4] Log[x3] + a[5] Log[x4]
First order logarithmic multiple nonlinear rational	FOLNR	(a[1] + a[2] Log[x1] + a[3] Log[x2] + a[4] Log[x3] + a[5] Log[x4]) / (b[1] + b[2] Log[x1] + b[3] Log[x2] + b[4] Log[x3] + b[5] Log[x4])
Second order logarithmic multiple nonlinear	SOLN	a[1] + a[2] Log[x1] + a[3] Log[x2] + a[4] Log[x3] + a[5] Log[x4] + a[6] Log[x1]^2 + a[7] Log[x2]^2 + a[8] Log[x3]^2 + a[9] Log[x4]^2 + a[10] Log[x1 x2] + a[11] Log[x1 x3] + a[12] Log[x1 x4] + a[13] Log[x2 x3] + a[14] Log[x2 x4] + a[15] Log[x3 x4]
Second order logarithmic multiple nonlinear rational	SOLNR	(a[1] + a[2] Log[x1] + a[3] Log[x2] + a[4] Log[x3] + a[5] Log[x4] + a[6] Log[x1]^2 + a[7] Log[x2]^2 + a[8] Log[x3]^2 + a[9] Log[x4]^2 + a[10] Log[x1 x2] + a[11] Log[x1 x3] + a[12] Log[x1 x4] + a[13] Log[x2 x3] + a[14] Log[x2 x4] + a[15] Log[x3 x4]) / (b[1] + b[2] Log[x1] + b[3] Log[x2] + b[4] Log[x3] + b[5] Log[x4] + b[6] Log[x1]^2 + b[7] Log[x2]^2 + b[8] Log[x3]^2 + b[9] Log[x4]^2 + b[10] Log[x1 x2] + b[11] Log[x1 x3] + b[12] Log[x1 x4] + b[13] Log[x2 x3] + b[14] Log[x2 x4] + b[15] Log[x3 x4])
Hybrid	H	a[1] + 3 a[2] Sin[x1] + 3 a[3] Sin[x1]^2 + a[4] Sin[x1]^3 + 3 a[5] Sin[x2] + 6 a[6] Sin[x1] Sin[x2] + 3 a[7] Sin[x1]^2 Sin[x2] + 3 a[8] Sin[x2]^2 + 3 a[9] Sin[x1] Sin[x2]^2 + a[10] Sin[x2]^3 + 3 a[11] Sin[x3] + 6 a[12] Sin[x1] Sin[x3] + 3 a[13] Sin[x1]^2 Sin[x3] + 6 a[14] Sin[x2] Sin[x3] + 6 a[15] Sin[x1] Sin[x2] Sin[x3] + 3 a[16] Sin[x2]^2 Sin[x3] + 3 a[17] Sin[x3]^2 + 3 a[18] Sin[x1] Sin[x3]^2 + 3 a[19] Sin[x2] Sin[x3]^2 + a[20] Sin[x3]^3 + 3 a[21] Sin[x4] + 6 a[22] Sin[x1] Sin[x4] +3 a[23] Sin[x1]^2 Sin[x4] + 6 a[24] Sin[x2] Sin[x4] + 6 a[25] Sin[x1] Sin[x2] Sin[x4] + 3 a[26] Sin[x2]^2 Sin[x4] + 6 a[27] Sin[x3] Sin[x4] + 6 a[28] Sin[x1] Sin[x3] Sin[x4] + 6 a[29] Sin[x2] Sin[x3] Sin[x4] + 3 a[30] Sin[x3]^2 Sin[x4] + 3 a[31] Sin[x4]^2 + 3 a[32] Sin[x1] Sin[x4]^2 + 3 a[33] Sin[x2] Sin[x4]^2 + 3 a[34] Sin[x3] Sin[x4]^2 + a[35] Sin[x4]^3 + Sin[x1] a[36] + Sin[x2] a[37] + Sin[x3] a[38] + Sin[x4] a[39] + Sin[x1]^2 a[40] + Sin[x2]^2 a[41] + Sin[x3]^2 a[42] + Sin[x4]^2 a[43] + Sin[x1]^4 a[44] + Sin[x2]^4 a[45] + Sin[x3]^4 a[46] + Sin[x4]^4 a[47]

2.2.1 Modified Nelder-Mead Algorithm

One of the search methods is the Nelder–Mead (NM) optimization algorithm. Therefore, it does not require any derivative information and starts with simplex to minimize the function. As a result, the iteration maintained up to the simplex becomes flat. It means that the resulting value of the function is almost the same at all the vertices. The iteration steps of the Nelder-Mead algorithm are ordering, centroid, and transformation. In the present version of the algorithm, a penalty function is added to the flow to solve the prescribed constrained minimization problem. The construction of the initial working simplex S is the first step. Second, minimizing the function moves the search course away from the peak, which is the worst function value [16].

2.2.2 Modified Differential Evolution Algorithm

Differential evolution (DE) is one of the suitable ways of the stochastic optimization method. It can be used in complex structured engineering design problems to find the optimum result. The productive parameters of the DE algorithm are population size, crossover, and scaling factor. It handles a population of solutions instead of iterating over solutions. The DE algorithm is proposed to be robust and efficient in the literature; it does not satisfy the global optimum points for all optimization problems [17].

2.2.3 Modified Simulated Annealing Algorithm

Another search method based on the physical annealing process of metal is simulated annealing (SA). The material moves to a lower energy state throughout the melting process and becomes tougher. Because of the intrinsic structure of the algorithm, it is better at finding the global optimum. Moreover, it can solve continuous, mixed-integer, or discrete optimization problems [17].

2.2.4 Modified Random Search Algorithm

The first step in the traditional random search algorithm is to produce a population of random starting points. Then, it uses a local optimization method from each starting point to get closer to a local extremum point at this stage. In the proposed algorithm version, there are some booster subroutines such as the conjugate gradient, principal axis, Levenberg Marquardt, Newton, QuasiNewton, and non-linear interior-point method the localization of the values of all variables for the objective function. In this step, the fitness function is evaluated with the variables being symbolic, and then the process continues again and again. It is also possible to solve mixed-integer-continuous global optimization problems [18].

2.3. Problem Description

1. The data given in Table 2 were selected from the reference study [1]. It should be noted that the parameters x1-x4 included in the models given in Table 1 correspond in terms of engineering to accelerating voltage, beam current, welding speed, and beam oscillation parameters, respectively.
2. 14 candidate functional constructs have been suggested to model the experimental data have been tested for the proper ones in terms of $R^2_{training}$, $R^2_{testing}$, and $R^2_{validation}$ values, and then boundedness of the functions is also checked.
3. Three different optimization scenarios have been introduced using the appropriate model obtained, and these problems are solved through four different direct search methods: MDE, MSA, MRS, and MNM

2.4.Optimization Scenarios

Scenario 1

In this optimization problem, the objective functions define the weld bead area (mm^2) of a welded Inconel 825 alloy. All the design variables are considered to be real numbers, and the search space is continuous. In this case, the limit values for the system inputs are 48 < accelerating voltage (kV) < 60, 38 < beam current (mA) < 4, 900 < welding speed (mm/min) < 1200 and 200 < beam oscillation (Hz) < 600. The main objective is to maximize the weld bead area. In this way, the theoretical limits of the objective function can also be seen mathematically.

Intervals are considered to be real numbers: 48 < accelerating voltage (kV) < 60, 38 < beam current (mA) < 4, 900 < welding speed (mm/min) < 1200 and 200 < beam oscillation (Hz) < 600. Additionally, to examine the constraints of design variables {accelerating voltage, beam current, welding speed, beam oscillation} ∈ integers are proper.

Scenario 2

Besides knowledge-based in Scenario 1, more applicable problem cases for the weld bead area need to be added. For this purpose, a new optimization problem has been described that assumes the maximization of the weld bead area. All the design variables in the intervals are considered to be real numbers: 48 < accelerating voltage (kV) < 60, 38 < beam current (mA) < 4, 900 < welding speed (mm/min) < 1200 and 200 < beam oscillation (Hz) < 600. Additionally, to examine the constraints of design variables {accelerating voltage, beam current, welding speed, beam oscillation} ∈ integers are proper.

Scenario 3

Based on only the prescribed experimental setup, the more specific optimization problem can also be defined as involving (I) maximization of the beam oscillation, (II) minimization of accelerating voltage, (III) optimizing the welding speed, (IV) minimizing the beam current, (V) all the design variables are assumed to be real numbers and (VI) the constraints are accelerating voltage ∈ {48, 54, 60}; beam current ∈ {38, 42, 46}; welding speed ∈ {900, 1050, 1200}; and beam oscillation ∈ {200, 400, 600}.

Table 2: *The data used for optimization operations in the article* [1]

SI. No.	Accelerating voltage(kV), V	Beam Current (mA), I	Welding speed (mm/min), S	Beam Oscillation (Hz), O	Weld bead area (mm^2), WA
1	48	38	900	200	4.86
2	60	46	1200	400	5.99
3	54	42	1050	600	5.39
4	48	38	1200	600	3.93
5	48	38	1200	200	3.24
6	60	38	1200	600	5.06
7	60	38	1200	200	5.12
8	48	38	900	600	4.81
9	48	46	900	600	6.06
10	48	46	1200	600	4.92
11	54	42	1050	400	5.33
12	60	38	900	600	5.69
13	60	46	900	600	6.07
14	48	46	900	200	6.15
15	54	42	1050	400	5.54
16	54	42	1050	400	5.38
17	48	46	1200	200	4.52
18	60	38	900	200	5.89
19	60	46	1200	200	5.42
20	60	46	900	200	5.53
21	54	46	1050	400	6.02
22	54	42	1200	400	4.93
23	54	42	1050	400	5.38
24	54	42	1050	600	5.32
25	54	38	1050	400	5.33
26	54	42	900	400	5.64
27	54	42	1050	200	5.56
28	48	42	1050	400	4.89
29	54	42	1050	400	5.63
30	60	42	1050	400	5.67

3.Results and Discussion

In this study, 14 different regression models with four parameters were tested for one output (see *Table 1*), and the results are listed in *Tables 3* to understand how the model successfully explained the process to estimate the $R^2_{training}$, $R^2_{testing}$ and $R^2_{validation}$ values of various regression models and determine the functional limitation (boundedness) of the model by estimating the maximum and minimum values generated by the corresponding model.

In order to understand that we have trained the program correctly, we should pay attention that the training value is <0.90, the testing value is <0.90, and the validation value is <0.85. After completing these processes, we check whether the maximum and minimum values we find for the inputs are within the engineering limits, the model that meets all the conditions can be used.

In *Table 3*, the suitability of the candidate models in terms of training, testing, validation coefficients, and boundedness, the following inferences were made:

- Training coefficients of all models are pretty high (>0.99) while the test coefficients are high for only LR, SON, FOTNR, SOTNR, FOLNR, and H. Therefore, the number of usable models in the testing phase decreases from 14 to 6.
- In the next stage, the compatibility of the validation value is examined. At this stage, we only have 1 model that meets this requirement, and that is SON.
- Besides, as mentioned in the previous section, it is also expected to meet the boundedness criterion for use in optimizing the model. When viewed from this angle, only one model is suitable, which is SON. As a result, only the SON model can meet all the desired criteria and be considered a realistic model.

Table 3: *Results of the Neuro-regression models for fitting performance and boundedness.*

Models	$R^2_{Training}$	$R^2_{Testing}$	$R^2_{Validation}$	Max (mm^2)	Min (mm^2)
L	0.99	0.74	0.80	6.50	4.19
LR	0.99	0.97	0.43	6.13	3.56
SON	0.99	0.97	0.87	6.20	3.56
SONR	0.99	0.50	0.87	6.12	4.70
TON	0.99	0.87	0.32	6.76	2.71
FOTN	0.99	0.79	0.89	6.50	-5.18
FOTNR	0.99	0.98	-22.2	$4.48496 * 10^{7}$	$-2.01089 * 10^{13}$
SOTN	0.99	0.79	0.89	6.45	-3.44
SOTNR	0.99	0.97	0.85	13.40	$-5.0818 * 10^{11}$
FOLN	0.99	0.75	0.81	6.50	4.17
FOLNR	0.99	0.96	0.32	6.12	3.58
SOLN	0.99	0.79	0.89	6.46	4.09
SOLNR	0.99	0.86	0.42	$5.58908 * 10^{10}$	3.87
H	0.99	0.95	0.93	11.63	-8.80

In *Table 4*, the model of SON is taken as the objective function, and the results are listed for three different optimization scenarios. This table uses MDE, MSA, MRS, and MNM algorithms for each scenario, and the results were compared. According to all algorithms for the first scenario, the maximum input values were calculated, and that gave us four different alternative input parameter triplets to achieve the weld beam area. The problem definition is the same in the second scenario, but the input parameters are forced to be integers. In this case, the input values have changed but still were close to numbers from scenario 1. In terms of reliability, achieving the very close results for the four direct search methods used in scenarios 1 and 2 increases the possibility that obtained values will be the global optimum. When we look at the results of the third scenario, we can say that we have reached the optimum parameter values similar to scenario 2. However, the best solution proposal of each algorithm is quite different from each other. This shows that it is essential to use more than one different phenomenological-based algorithm when there is a need to produce alternative optimum solutions for this problem and similar studies.

Table 4: *Optimization problem results*

Objective function	Scenario no	Constrains	Optimization Algorithm	Suggested Design (max)
$-2.20064 + 0.728057\ x_1 - 0.00512518\ x_1^2 - 0.663085\ x_2 - 0.00883957\ x_1 x_2 + 0.0131558\ x_2^2 + 0.0019017\ x_3 + 0.000244055\ x_1 x_3 + 0.0000806661\ x_2 x_3 - 0.0000105548\ x_3^2 - 0.0060513\ x_4 + 3.41694{*}10^{-6}\ x_1\ x_4 + 0.0000941879\ x_2\ x_4 + 2.57166{*}10^{-6}\ x_3 x_4 - 6.12662{*}10^{-7}\ x_4^2$	1	$48<x_1<60$; $38<x_2<46$; $900<x_3<1200$; $200<x_4<600$;	MDE	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.143$ $x_4 = 600$
			MSA	$x_1 = 54.6776$ $x_2 = 46$ $x_3 = 971.116$ $x_4 = 600$
			MRS	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.142$ $x_4 = 600$
			MNM	$x_1 = 54.6809$ $x_2 = 46$ $x_3 = 971.142$ $x_4 = 600$
	2	$48<x_1<60$; $38<x_2<46$; $900<x_3<1200$; $200<x_4<600$; $\{x_1, x_2, x_3, x_4\} \in$ Integers	MDE	$x_1 = 56$ $x_2 = 45$ $x_3 = 982$ $x_4 = 599$
			MSA	$x_1 = 55$ $x_2 = 45$ $x_3 = 937$ $x_4 = 503$
			MRS	$x_1 = 52$ $x_2 = 45$ $x_3 = 1015$ $x_4 = 599$
			MNM	$x_1 =56$ $x_2 =45$ $x_3 =982$ $x_4 =599$
	3	$x_1 \in \{48, 54, 60\}$, $x_2 \in \{38, 42, 46\}$, $x_3 \in \{900, 1050, 1200\}$, $x_4 \in \{200, 400, 600\}$	MDE	$x_1 =54$ $x_2 =46$ $x_3 =1006$ $x_4 =550$
			MSA	$x_1 =55$ $x_2 =45$ $x_3 =1010$ $x_4 =555$
			MRS	$x_1 =55$ $x_2 =49$ $x_3 =1006$ $x_4 =550$
			MNM	$x_1 =53$ $x_2 =45$ $x_3 =1001$ $x_4 =550$

4.Conclusion

In this research, electron beam welding of Inconel 825 is performed to investigate the influence of welding process variables WA. Thirty welding experiment results have been taken from Ref [1] on four weld factors. First, predictive modeling for WA is developed using non-linear multiple regression analysis employing the philosophy of the popular method ANN, and the effectiveness is investigated. Then, the obtained results were checked whether the selected models are also bounded or not in the engineering parameter intervals. Finally, modified versions of four direct search methods (Differential Evolution, Simulated Annealing, Random Search, and Nelder-Mead) were used during the optimization process. From this investigation, the followings are some of the critical conclusions obtained.

The training values selected to train the program should be as many as possible so that the testing and validation values are as accurate as possible. Nevertheless, it is not enough to just have a large amount of data in the training part; the input values of the selected data must also include the maximum and minimum input values because the program has difficulty estimating a value, not in the range. Another factor affecting the consistency of the estimated values is that the outputs are very variable; the model may give closer results with outputs close to each other. During model selection, estimates are used because there is no information about which model will give better results, different results are obtained by trial and error, and which model gives more accurate results is continued. If we had bad results from the direct search methods, we could use the particular options that apply to Nelder-Mead: ReflectRatio, Tolerance, ShrinkageRatio, Randomseed, and more not needed.

Declaration of Interest

The authors declare that there is no conflict of interest.

References

[1] B. Choudhury, and M. Chandrasekaran, “Electron beam welding of aerospace alloy (inconel 825): a comparative study of RSM and ANN modeling to predict weld bead area,” Optik - International Journal for Light and Electron Optics, Elsevier GmbH, vol. 219, 2020

[2] J. W. Sowards, and J. Caron, “Weldability of nickel-base alloys,” Comprehensive Materials Processing, Elsevier, Oxford, vol. -1, no. 6, pp. 151-179, 2014.

[3] B. Choudhury, and M. Chandrasekaran, “Investigation on welding characteristics of aerospace materials –a review,” Selection and Peer-review under responsibility of the Committee Members of International Conference on Advancements in Aeromechanical Materials for Manufacturing, Elsevier, vol. 4, pp. 7519–7526, 2017.

[4] M. Chandrasekaran, M. Muralidhar, C. M. Krishna, and U. S. Dixit, “Application of soft computing techniques in machining performance prediction and optimization: a literature review,” Int J Adv Manuf Technol, Springer-Verlag London Limited, vol. 46, pp. 445–464, 2009.

[5] F. Khademi, S. M. Jamal, N. Deshpande, and S. Londhe, “Predicting strength of recycled aggregate concrete using artificial neural network, adaptive neuro fuzzy inference system and multiple linear regression,” International Journal of Sustainable Built Environment, Elsevier BV, vol. 5, no. 2, pp. 355-369, 2016.

[6] R. Palanivel, I. Dinaharan, and R. F. Laubscher, “Application of an artificial neural network model to predict the ultimate tensile strength of friction welded titanium tubes,” J. Braz. Soc. Mech. Sci. Eng., Springer, vol. 41, no. 111, 2019.

[7] M. F. A. Zaharuddin, D. Kim, and S. Rhee, “An ANFIS based approach for predicting the weld strength of resistance spot welding in artificial intelligence development,” J. Mech. Sci. Technol., Springer, vol. 31, no. 11, pp. 5467–5476, 2017.

[8] K. Anand, R. Shrivastava, K. Tamilmannan, and P. Sathiya, “A comparative study of artificial neural network and response surface methodology for optimization of friction welding of incoloy 800 H.” Acta Metall. Sin. (Engl. Lett.), Springer, vol. 28, no. 7, pp. 892–902, 2015.

[9] E. A. Gyasi, P. Kah, H. Wu, and M. A. Kesse, “Modeling of an artificial intelligence system to predict structural integrity in robotic GMAW of UHSS fillet welded joints,” Int. J. Adv. Manuf. Technol., Springer, vol. 93, pp. 1139–1155, 2017.

[10] M. P. Satpathya, S. B. Mishraa, and S. K. Sahoob, “Ultrasonic spot welding of aluminum-copper dissimilar metals: a study on joint strength by experimentation and machine learning techniques,” J. Manuf. Process., Elsevier, vol. 33, pp. 96–110, 2018.

[11] H. K. Narang, U. P. Singh, M. M. Mahapatra, and P. K. Jha, “Prediction of the weld pool geometry of TIG arc welding by using fuzzy logic controller,” Int. J. Eng. Sci. Technol., MultiCraft, vol. 3, no. 9, pp. 77–85, 2011.

[12] N. Sivagurumanikandan, S. Saravanan, G. S. Kumar, S. Raju, and K. Raghukandan, “Prediction and optimization of process parameters to enhance the tensile strength of Nd: YAG laser welded super duplex stainless steel,” Optik, Elsevier Gmbh., vol. 157, pp. 833–840, 2018.

[13] M. Akbari, S. Saedodin, A. Panjehpour, M. Hassani, M. Afrand, and M. J. Torkamany, “Numerical simulation and designing artificial neural network for estimating melt pool geometry and temperature distribution in laser welding of Ti6Al4V alloy,” Opt. – Int. J. Light Electron. Opt., 2016.

[14] K. Anand, B. K. Barik, K. Tamilmannan, and P. Sathiya, "Artificial neural network modeling studies to predict the friction welding process parameters of incoloy 800H joints," Eng. Sci. Technol. Int. J., Elsevier BV, vol. 18, pp. 394–407, 2015.

[15] K. R. Balasubramanian, G. Buvanashekaran, and K. Sankaranarayanasamy, “Modeling of laser beam welding of stainless steel sheet butt joint using neural networks,” CIRP J. Manuf. Sci. Technol., Elsevier, vol. 3, pp. 80–84, 2010.

[16] I. Polatoglu, and L. Aydin, “A new design strategy with stochastic optimization on the preparation of magnetite cross-linked tyrosinase aggregates (MCLTA),” Process Biochemistry, vol. 99, pp. 131–138, 2020.

[17] I. Polatoglu, L. Aydin, B. Ç. Nevruz, and S. Ozer, “A novel approach for the optimal design of a biosensor,” Anal. Lett., pp. 1–18, 2020. doi.org/10.1080/ 00032719.2019.1709075.

[18] A. B. Ceylan, L. Aydin, and M. Nil, H. Mamur, İ. Polatoğlu and H. Sözen. “A new hybrid approach in selection of optimum establishment location of the biogas energy production plant,” *Biomass Conv. Bioref,* 2021. https://doi.org/10.1007/s13399-021-01532-8

APPENDICES

Nomenclature	Models
L	1.05302 + 0.0606329 x1 + 0.0866439 x2 - 0.00260376 x3 + 0.00029121 x4
LR	(-1.42443*10^6 + 42920.6 x1 - 4984.72 x2 - 302.626 x3 + 68.1846 x4) / (-212520. + 7358.63 x1 - 1833.02 x2 - 38.4445 x3 + 11.5423 x4)
SON	-2.20064 + 0.728057 x1 - 0.00512518 x1^2 - 0.663085 x2 - 0.00883957 x1 x2 + 0.0131558 x2^2 + 0.0019017 x3 + 0.000244055 x1 x3 + 0.0000806661 x2 x3 - 0.0000105548 x3^2 - 0.0060513 x4 + 3.41694*10^-6 x1 x4 + 0.0000941879 x2 x4 + 2.57166*10^-6 x3 x4 - 6.12662*10^-7 x4^2
SONR	(0.999275 + 1.92477 x1 + 1.0596 x1^2 + 1.9466 x2 + 2.29488 x1 x2 + 1.17203 x2^2 + 0.536888 x3 + 24.0295 x1 x3 + 14.7801 x2 x3 + 1.31359 x3^2 + 1.32937 x4 + 10.1391 x1 x4 + 6.95023 x2 x4 - 5.89455 x3 x4 + 0.545452 x4^2) / (1.00445 + 2.57484 x1 + 3.65083 x1^2 + 2.39654 x2 + 8.45453 x1 x2 + 1.38953 x2^2 + 11.0051 x3 + 1.26633 x1 x3 - 0.25641 x2 x3 + 0.523906 x3^2 + 6.0619 x4 + 4.87717 x1 x4 + 3.59808 x2 x4 - 1.51806 x3 x4 + 0.289919 x4^2)
TON	-11.8606 + 0.224046 x1 + 0.00488066 x1^2 - 0.0000194012 x1^3 - 0.138815 x2 - 0.00203656 x1 x2 - 0.000152265 x1^2 x2 + 0.00150761 x2^2 - 0.000147576 x1 x2^2 + 0.00017423 x2^3 + 0.00711543 x3 + 0.000204546 x1 x3 + 4.53442*10^-7 x1^2 x3 + 0.0000324893 x2 x3 + 7.21174*10^-6 x1 x2 x3 + 1.55683*10^-6 x2^2 x3 + 4.53854*10^-6 x3^2 - 7.37788*10^-9 x1 x3^2 - 8.21767*10^-8 x2 x3^2 - 1.36093*10^-8 x3^3 + 0.0670044 x4 - 0.0010011 x1 x4 - 1.36903*10^-7 x1^2 x4 - 0.00130697 x2 x4 + 0.0000359974 x1 x2 x4 - 4.92474*10^-6 x2^2 x4 - 0.0000768777 x3 x4 - 7.3771*10^-7 x1 x3 x4 - 7.41981*10^-7 x2 x3 x4 + 7.01739*10^-8 x3^2 x4 + 0.000070046 x4^2 + 3.50838*10^-7 x1 x4^2 + 8.10061*10^-7 x2 x4^2 + 3.92122*10^-9 x3 x4^2 - 1.05786*10^-7 x4^3
FOTN	0.665315 - 4.20932 Cos[x1] - 0.366313 Cos[x2] + 0.172438 Cos[x3] - 0.0653933 Cos[x4] - 1.26012 Sin[x1] + 0.310245 Sin[x2] + 0.882573 Sin[x3] + 0.0241337 Sin[x4]
FOTNR	(-950.462 - 772.317 Cos[x1] - 9.46081 Cos[x2] - 45.0644 Cos[x3] + 12.786 Cos[x4] - 661.373 Sin[x1] - 20.6108 Sin[x2] - 44.7748 Sin[x3] + 16.1305 Sin[x4]) / (-171.53 - 138.883 Cos[x1] - 2.17954 Cos[x2] - 7.43295 Cos[x3] + 2.30761 Cos[x4] - 117.893 Sin[x1] - 3.41943 Sin[x2] - 6.79714 Sin[x3] + 2.97475 Sin[x4])
SOTN	0.407069 - 1.18265 Cos[x1] + 1.28031 Cos[x1]^2 - 0.241127 Cos[x2] + 0.393791 Cos[x2]^2 + 0.413412 Cos[x3] + 0.329156 Cos[x3]^2 - 0.0575362 Cos[x4] + 0.745032 Cos[x4]^2 - 2.15281 Sin[x1] - 1.4521 Sin[x1]^2 + 0.315919 Sin[x2] + 0.628466 Sin[x2]^2 + 0.750764 Sin[x3] + 0.700909 Sin[x3]^2 - 0.280635 Sin[x4] + 0.362098 Sin[x4]^2
SOTNR	(2.31883 - 6.25099 Cos[x1] + 11.8668 Cos[x1]^2 + 0.711155 Cos[x2] + 1.33971 Cos[x2]^2 + 1.39785 Cos[x3] + 1.1556 Cos[x3]^2 + 0.901391 Cos[x4] + 2.47489 Cos[x4]^2 + 9.87237 Sin[x1] - 8.54795 Sin[x1]^2 - 0.15083 Sin[x2] + 1.97912 Sin[x2]^2 + 2.46102 Sin[x3] + 2.16323 Sin[x3]^2 + 1.28901 Sin[x4] + 0.843944 Sin[x4]^2) / (0.175012 - 0.382517 Cos[x1] + 2.69676 Cos[x1]^2 + 0.452111 Cos[x2] + 0.354456 Cos[x2]^2 + 0.200628 Cos[x3] + 0.498388 Cos[x3]^2 + 0.1765 Cos[x4] + 0.260945 Cos[x4]^2 + 1.38414 Sin[x1] - 1.52174 Sin[x1]^2 - 0.115724 Sin[x2] + 0.820556 Sin[x2]^2 + 0.219855 Sin[x3] + 0.676624 Sin[x3]^2 + 1.02702 Sin[x4] + 0.914067 Sin[x4]^2)
FOLN	-3.5355 + 3.29747 Log[x1] + 3.64491 Log[x2] - 2.67695 Log[x3] + 0.12492 Log[x4]
FOLNR	(-5061.54 + 2010.94 Log[x1] - 200.597 Log[x2] - 283.919 Log[x3] + 20.2834 Log[x4]) / (-810.22 + 344.877 Log[x1] - 71.2596 Log[x2] - 36.5198 Log[x3] + 3.36826 Log[x4])
SOLN	-378.91 + 67.2296 Log[x1] - 11.8256 Log[x1]^2 - 164.035 Log[x2] + 26.8554 Log[x2]^2 - 20.4811 Log[x1 x2] + 81.1211 Log[x3] - 11.69 Log[x3]^2 + 41.6929 Log[x1 x3] + 14.2396 Log[x2 x3] - 5.26173 Log[x4] + 0.0303843 Log[x4]^2 + 9.02977 Log[x1 x4] - 26.5893 Log[x2 x4] + 22.5746 Log[x3 x4]
SOLNR	(0.941802 + 6.08834 Log[x1] + 40.7971 Log[x1]^2 - 1.65513 Log[x2] - 16.8924 Log[x2]^2 + 3.43321 Log[x1 x2] + 0.178359 Log[x3] - 9.24925 Log[x3]^2 + 5.2667 Log[x1 x3] - 2.47677 Log[x2 x3] + 0.82184 Log[x4] - 0.568154 Log[x4]^2 + 5.91018 Log[x1 x4] - 1.83329 Log[x2 x4] + 0.00019938 Log[x3 x4]) / (1.13334 + 0.939816 Log[x1] + 7.07759 Log[x1]^2 + 1.28176 Log[x2] - 4.89594 Log[x2]^2 + 1.22157 Log[x1 x2] + 0.616195 Log[x3] - 1.6354 Log[x3]^2 + 0.55601 Log[x1 x3] + 0.897953 Log[x2 x3] + 0.457665 Log[x4] - 0.174584 Log[x4]^2 + 0.397481 Log[x1 x4] + 0.739424 Log[x2 x4] + 0.0738598 Log[x3 x4])

H	3.72692 - 2.83153 Sin[x1] - 1.94904 Sin[x1]^2 + 1.82501 Sin[x1]^3 - 1.10419 Sin[x1]^4 + 2.11394 Sin[x2] - 0.249112 Sin[x1] Sin[x2] - 2.77911 Sin[x1]^2 Sin[x2] + 0.266018 Sin[x2]^2 - 2.79166 Sin[x1] Sin[x2]^2 - 0.976365 Sin[x2]^3 - 0.649314 Sin[x2]^4 + 0.258553 Sin[x3] + 0.317103 Sin[x1] Sin[x3] + 0.772664 Sin[x1]^2 Sin[x3] - 2.01385 Sin[x2] Sin[x3] - 3.47612 Sin[x1] Sin[x2] Sin[x3] - 0.152631 Sin[x2]^2 Sin[x3] + 0.973056 Sin[x3]^2 + 1.12739 Sin[x1] Sin[x3]^2 - 0.0749866 Sin[x2] Sin[x3]^2 + 0.143497 Sin[x3]^3 - 0.0653614 Sin[x3]^4 + 0.995053 Sin[x4] - 1.34491 Sin[x1] Sin[x4]- 0.383878 Sin[x1]^2 Sin[x4] + 0.575175 Sin[x2] Sin[x4] + 2.06164 Sin[x1] Sin[x2] Sin[x4] + 0.220295 Sin[x2]^2 Sin[x4] - 0.0890705 Sin[x3] Sin[x4] + 0.456294 Sin[x1] Sin[x3] Sin[x4] - 0.230877 Sin[x2] Sin[x3] Sin[x4] + 0.759084 Sin[x3]^2 Sin[x4] + 0.539582 Sin[x4]^2 + 0.586279 Sin[x1] Sin[x4]^2 - 1.08923 Sin[x2] Sin[x4]^2 + 0.644453 Sin[x3] Sin[x4]^2 - 0.772504 Sin[x4]^3 + 1.45937 Sin[x4]^4