Optimization of Bead Geometry Parameters of Electron Beam Welding Using Advanced Optimization Techniques

Abhinaw Kumar[1] , Ugale S. N[2], Solanki A[3], Mohit Kumar[4], Kamble A. G.[5]
[1, 2, 3, 4, 5] Department of Mechanical Engineering
[1, 2, 3, 4, 5] Sinhgad Institute of Technology, Kusgaon (Bk), Lonavala,

Abstract- *The welding processes are used to get the welded joint of desired weld bead geometry, excellent mechanical strength and minimum distortion. The input parameters of welding process play very substantial role to determine the quality of a weld joint. This paper deals with the optimization aspects of the electron beam welding parameters. The objective is to minimize the weldment area (bead height and bead width) simultaneously satisfying the condition of maximum bead penetration of austenitic stainless steel plates of grade ASS-304 of thickness 5.0mm. The various input parameters such as welding voltage, welding current, welding speed are considered to optimize the bead geometry. For this constrained problem, the optimization is carried out using selected advanced optimization techniques namely teaching and learning based optimization, artificial bee colony, jaya algorithm, particle swarm optimization. The results of the presented algorithms are compared with genetic algorithm and it is found that the better minimum weldment area and maximum bead penetration is obtained by using these selected algorithms.*

Keywords- Advance optimization techniques, Bead geometry parameters, Electron beam welding.

I. INTRODUCTION

In today's world, manufacturing quality plays a vital role. Welding is a multi-input and multi output process. In the welding process the quality of the weld is determined by the weld bead geometry (bead width, bead height, bead penetration) and the grain size of the weldment zone, which are directly influenced by proper combination of input parameters. Electron beam welding is suitable for joining steel plates of thickness varying from 0.2mm to 300mm in a single run. The high energy density of about 10^9 W/cm^2 is emitted in this process which melts the work piece and leads formation of key holes. This process is more widely used over other welding processes to get very high aspect ratio of the weld. A human process planner selects the process parameter and input parameters using his own experience or from the handbooks but these parameters do not give the optimal result. Therefore, the optimization techniques act as an important tool for selecting specific process parameters.

The various optimization methods have been implemented for solving different engineering application which have been reported in the following literatures. Artificial bee colony algorithm (ABC) is used for optimizing new design method based on artificial bee colony algorithm for digital IIR filters [1], Modelling and optimization of process parameters of wire electrical discharge machining [2], Scheduling a single batch processing with non-identical job sizes [3], Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms [4]. Particle swarm optimization (PSO) algorithm is used for Optimization for ice-storage air-conditioning system [5], Power systems operation using particle swarm optimization technique [6], Particle Swarm Optimization: Technique, System and Challenges [7], Tuning of neural networks using particle swarm optimization to model MIG [8], Optimization of flux cored arc welding process parameters using particle swarm optimization technique [9]. Teaching and learning based optimization (TLBO) is used for Parameter optimization of machining processes using teaching–learning-based optimization algorithm [10], Selection of laser bending process parameters for maximal deformation angle through neural network and teaching and learning based optimization algorithm [11],Design of planar steel frames using Teaching–Learning Based Optimization[12], Multi-objective optimization of heat exchangers using a modified teaching-learning-based optimization algorithm[13].The optimization of different process parameters of electron beam welding to get the weldment of minimum cross-sectional area as well as maximum penetration by the use of selected optimization techniques has been reported in this paper.

II. IMPLIMENTATION OF ADVANCE OPTIMIZATION TECHNIQUES

A. ABC ALGORITHM

This algorithm is originally introduced by Karaboga in 2005, on the intelligent foraging behaviour of honey bees for numerical optimization problems.

In ABC algorithm, the population consists of a total number of possible solutions x_i represented by the positions of food sources, whose nectar amount corresponds to the quality of the associated solution. The colony of artificial bees contains three types of bees: employed bees, onlooker bees and scout bees. A bee that is currently exploiting a food source is called employed bee, the bee waiting in the hive for making decision to choose a food source is named as onlooker bee and the bee which carries out random searches for new food sources named scout bees [14]. There has been growing interest, recently, in applications of ABC algorithm to many complex problems of the real world due to only one control parameter 'limit' and for its more optimized results.

This algorithm involves following steps [15]:

1: Generate the initial population Xi, i =1,2……SN
2: Evaluate the initial population
3: Set cycle to 1
4: repeat
5: for each employed bee {Produce new solution V_i using Eq.(2) Calculate the value fit_i Apply greedy selection process}
6:Calculate the probability values *pi for* the solutions (X_i) by Eq.(3)
7:for each onlooker bee {Select a solution X_i depending on P_i produces new solution. calculate the value fit_i *and* Apply greedy selection process}
8: if there is an abandoned solution then replace it with a new solution which will be randomly produced by a scout using Eq.(1)
9: Memorize the best solution so far
10: cycle = cycle + 1
11: until cycle = MCN

For the initialization of the algorithm, a set of solutions (food source positions) are randomly generated by the scout bees, let Xi = {xi1,xi2, . . . xin} represent the ith food source in the population, and then each solution is generated by Eq.(1).

$$X_{ij}=X^j_{min}+ rand(0,1)(x^j_{max}-X^j_m \qquad (1)$$

where x^j_{min} and x^j_{max} are the lower and upper bounds of the jth parameters of the solution i.

After the initialization, the population is evaluated and then, it is repeated with search processes through the employed bees. Each employed bee is associated with a particular individual food source and in each iteration, she searches a new food source in the neighbourhood of the food source in her memory by using Eq. (2).

$$V_{ij} = X_{ij} + \Psi_{ij}(X_{ij}-X_{kj}) \qquad (2)$$

Where Ψ_{ij} is a random number between [-1, 1], $V_i$is the candidate food source position, X_i is the current food source position, $X_k$is a neighbour food source position, and j€2 {1, 2,..., D} is randomly chosen index which represents a component of each food source position and D is the dimension of the problem. Once V_i is obtained and evaluated, it is compared to X_i. If V_i is better than X_i, it will replace with X_i and become a new member of the population or apply greedy selection process; when the nectar amount of the new source is higher than that of the previous one, the bee memorizes the new food position and forgets the old one; otherwise, she keeps the position of the previous one in her memory. When all employed bees complete the search process; they come into the hive and share the nectar information of their sources with onlooker bees by performing particular dance. Then, each onlooker prefers a food source area depending on the nectar information distributed by the employed bees. The onlooker bees set their preference probabilistically using formula [15],

$$P_i= [(0.9*fit_i)/ (fit_{best}) + 0.1] \qquad (3)$$

Where fit_{best} is the quality of the best solution among the current solutions and fit_i the quality of the solution i which is proportional to the nectar amount of the food source i, given as,

$$fit_i= 1/(1+fi_i) \quad \text{if } fi_i>=0 \qquad (4a)$$

or,

$$fit_i= 1+abs(fi_i) \quad \text{if } fi_i<0 \qquad (4b)$$

In order to select the best solution (food source) roulette wheel selection method is being used by Onlooker bees. When the nectar of a food source is exhausted by the employed and onlooker bees, the employed bees of that source become the scout bees and randomly determine a new food source by Eq. (1) and replaces it with the abandoned (exhausted) one. In order to decide whether a food source is abandoned or not, a control parameter called 'limit' is used. At the end of all iterations, limit values are compared with the number of unimproved tries of each solution. There is a counter for each solution which is incremented by one in each fail or is set to zero in each successful try carried out by either an employed bee or an onlooker bee.

Fig1. Flow chart of ABC Algorithm [17]

B. PSOALGORITHM

PSO algorithm is a heuristic global optimization method developed by Doctor Kennedy and Eberhart in 1995. This is an evolutionary computation technique developed from swarm intelligence and is based on the behaviour of bird flocking, or fish schooling where they find randomly searching food together in a specific area. In PSO algorithm, solution swam is compared with the bird swarm, the birds moving from one place to another is equal to the development of the solution swarm. Here good information is equal to the most optimist solution, and the food resource is equal to the most optimist solution during the whole course.

In the basic PSO algorithm, particle swarm consists of “n” particles. Each potential solution is referred to as a particle position and each particle is initialized by a random position in three-dimensional space [18]. The particles change its position and velocity according to the following conditions:

- to keep its inertia
- to change the condition according to its most optimist position
- to change the condition according to the swarm’s most optimist position.

In PSO, each particle keeps the track of its coordinates in hyperspace which are associated with the best solution (fitness) it has achieved so far. The value of that fitness is also stored. This stored value is called “*pbest*”. Another “best” value is also tracked. The “global” version of the particle swarm optimizer keeps track of the overall best value and its location obtained so far by any particle in the population. This stored value is called “*gbest*”. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward *pbest* and *gbest* locations [19].

The basic rules of this algorithm can be explained in three main stages [20]:

- Evaluating the fitness value of each particle.
- Updating local and global best fitness and positions.
- Updating the velocity and the position of each particle.

This algorithm involves following steps:

1: For each particle $i = 1$,

2: Initialize the particle's position: $x_i \sim U(b_{lo}, b_{up})$, where b_{lo} and b_{up} are the lower and upper boundaries of the search-space.

3: Initialize the particle's best known position to its initial position: $p_i \leftarrow x_i$

4: If $(f(p_i) < f(g))$ update the swarm's best known position: $g \leftarrow p_i$

5: Initialize the particle's velocity: $v_i \sim U(-|b_{up}-b_{lo}|, |b_{up}-b_{lo}|)$.

6: repeat: For each particle $i = 1, ..., S$ do: For each dimension $d = 1, ..., n$ do: Pick random numbers: r_p, $r_g \sim U(0,1)$

7: Update the particle's velocity:

$v_{id}^{k+1} = \omega\, v_{id}^{k} + c_p r_p^{k} (pbest_{id}^{k} - x_{id}^{k}) + c_g r_g^{k} (gbest_{id}^{k} - x_{id}^{k})$

Update the particle's position:

$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1}$

8: If $(f(x_i) < f(p_i))$ do:

Update the particle's best known position: $p_i \leftarrow x_i$

If $(f(p_i) < f(g))$ update the swarm's best known position: $g \leftarrow p_i$

9: Now g holds the best found solution.

$$v_{id}^{k+1} = \omega\, v_{id}^{k} + c_p r_p^{k} (pbest_{id}^{k} - x_{id}^{k}) + c_g r_g^{k} (gbest_{id}^{k} - x_{id}^{k}) \quad (5)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (6)$$

Where,

v_i^k	Velocity of particle i at iteration k
v_i^{k+1}	Velocity of particle i at iteration k+1
ω	Inertia weight
C_j	Acceleration coefficients; j=1,2
r_i^k	Random number between 0-1; i=1,2
x_i^k	Current position of particle i at iteration k
x_i^{k+1}	Position of the particle i at iteration k+1
$pbest_i^k$	Best position of particle i at iteration k
$gbest_i^k$	Position of best particle in a population.

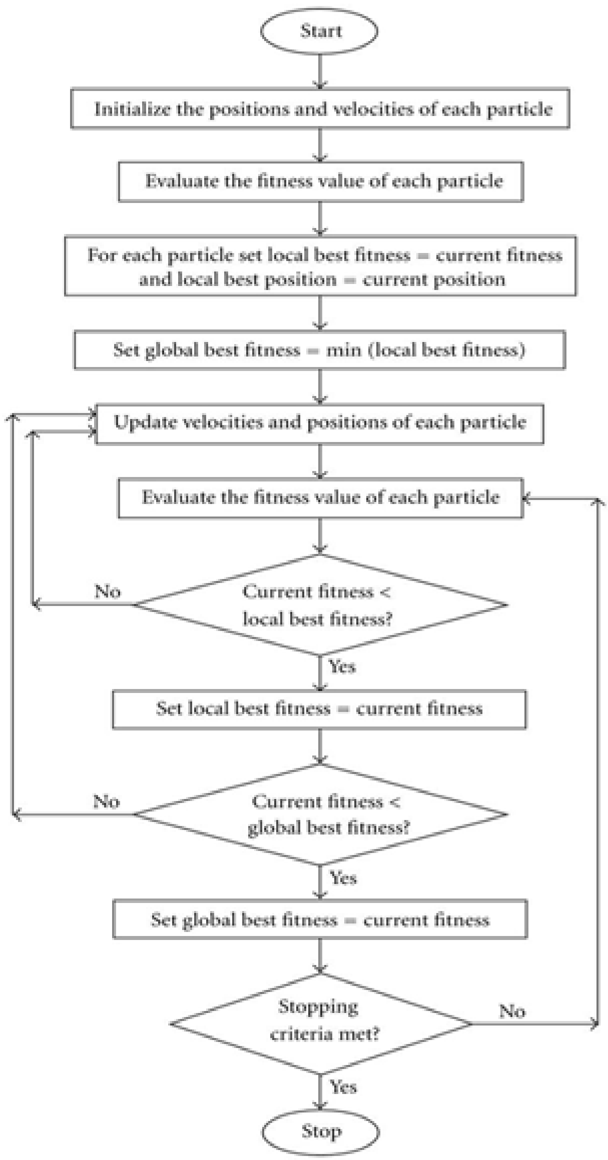


Fig 2. Flow chart of PSO Algorithm

C. TLBO

TLBO is a teaching-learning process inspired algorithm proposed by Rao et al. [21].It is based on the effect of influence of a teacher on the output of learners in a class. The algorithm describes two basic modes of the learning: (i) through teacher (known as teacher phase) and (ii) interacting with the other learners (known as learner phase). In this optimization algorithm a group of learners is considered as population and the design variables are actually the parameters involved in the objective function of the given optimization problem and the best solution is the best value of the objective function.

Teacher phase:

During this phase a teacher tries to increase the mean result of the class in the subject taught by him/her depending on his or her capability. At any iteration i, assume that there are 'm' number of subjects (i.e. design variables), 'n' number of learners (i.e. population size, k=1,2,…,n) and M j,i be the mean result of the learners in a particular subject 'j' (j=1,2,…,m). The best overall result X total-k_{best}, i considering all the subjects together obtained in the entire population of learners can be considered as the result of best learner k_{best}. However, as the teacher is usually considered as a highly learned person who trains learners so that they can have better results, the best learner identified is considered by the algorithm as the teacher. The difference between the existing mean result of each subject and the corresponding result of the teacher for each subject is given by,

$$\text{Difference Mean}_{j,k,i} = r_i\,(X_{j,kbest,i} - T_F M_{j,i}) \quad (7)$$

where, $X_{j,kbest,i}$is the result of the best learner (i.e. teacher) in subject j. T_F is the teaching factor which decides the value of mean to be changed, and r i is the random number in the range [0, 1]. Value of T_F can be either 1 or 2. The value of T_F is decided randomly with equal probability as,

$$T_F = \text{round}\,[1+\text{rand}(0,1)\{2-1\}] \quad (8)$$

After conducting a number of experiments on many benchmark functions it is concluded that the algorithm performs better if the value of T_F is between 1 and 2. However, the algorithm is found to perform much better if the value of T_F is either 1 or 2 and hence to simplify the algorithm, the teaching factor is suggested to take either 1 or 2 depending on the rounding up criteria given by Eq. (8). Based on the Difference_Mean$_{j,k,i}$, the existing solution is updated in the teacher phase according to the following expression.

$$X'_{j,k,i} = X_{j,k,i} + \text{Difference\_Mean}_{j,k,I} \quad (9)$$

where $X'_{j,k,i}$is the updated value of $X_{j,k,i}$. Accept $X'_{j,k,i}$ if it gives better function value. All the accepted function values at the end of the teacher phase are maintained and these values become the input to the learner phase. The learner phase depends upon the teacher phase.

Learner phase:

Learners increase their knowledge by interaction among themselves. A learner interacts randomly with other learners for enhancing his or her knowledge. A learner learns new things if the other learner has more knowledge than him or her. Considering a population size of 'n', the learning phenomenon of this phase is expressed below. Randomly select two learners P and Q such that $X'_{total-P,i} \neq X'_{total-Q,i}$(where, $X'_{total-P,i}$ and $X'_{total-Q,i}$ are the updated values of $X_{total-P,i}$ and $X_{total-Q,i}$ respectively at the end of teacher phase).

$$X''_{j,P,i}=X'_{j,P,i} + r_i (X'_{j,P,i}- X'_{j,Q,i}) \text{ If } X'_{total-P,i}<X'_{total-Q,I} \quad (10a)$$
$$X''_{j,P,i} =X'_{j,P,i} + r_i (X'_{j,Q,i}- X'_{j,P,i}), \text{ If } X'_{total-Q,I}<X'_{total-P,I} \quad (10b)$$

Accept X'' j,P,i if it gives a better function value.. All the accepted function values at the end of the learner phase are maintained and these values become the input to the teacher phase of the next iteration. The values of r_i used in Eqs. (7), (10a) and (10b) can be different. Repeat the procedure of teacher phase and learner phase till the termination criterion is met.

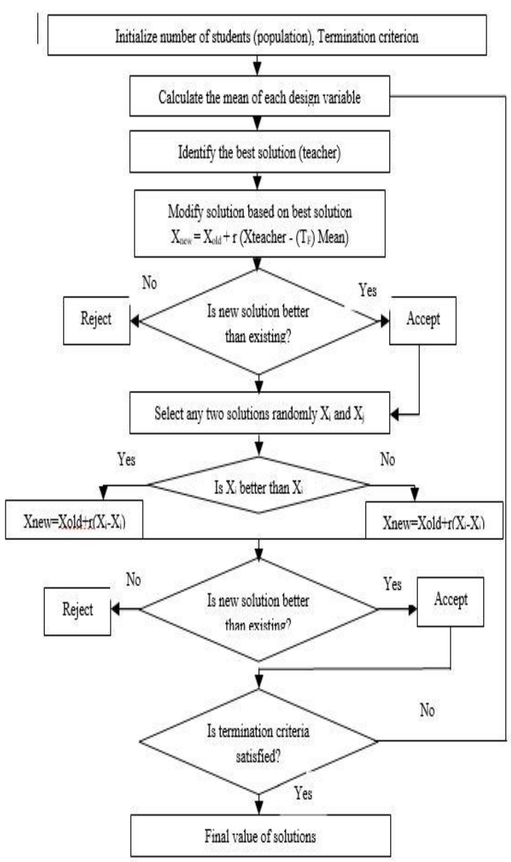


Fig 3. Flow chart of TLBO Algorithm[21]

D. JAYA ALGORITHM

A new, simple and very powerful optimization algorithm is proposed by R. V. Rao [22], which is capable of solving both constrained and unconstrained optimization problems. Like, TLBO [21] this algorithm, also works on common controlling parameters like population size and number of generations, and not on algorithm-specific parameters. The simplicity of Jaya algorithm is evident from the fact that it uses one phase unlike, two phases (Teacher and learner phase) in TLBO algorithm. The basic concept of the algorithm pushes the obtained solution towards the best solution and restricts movement towards worst solution, because this algorithm pushes solution towards the best value it is named as Jaya (a Sanskrit word meaning victory). The working of Jaya algorithm is described below with the following steps*:*

1:Initialize the population size, number of generations, number of design variables, limits of design variables, and define the optimization problem as: Minimize/Maximize f(x) where x is the input parameters.

2:Generate a random population according to the population size and number of design variables.

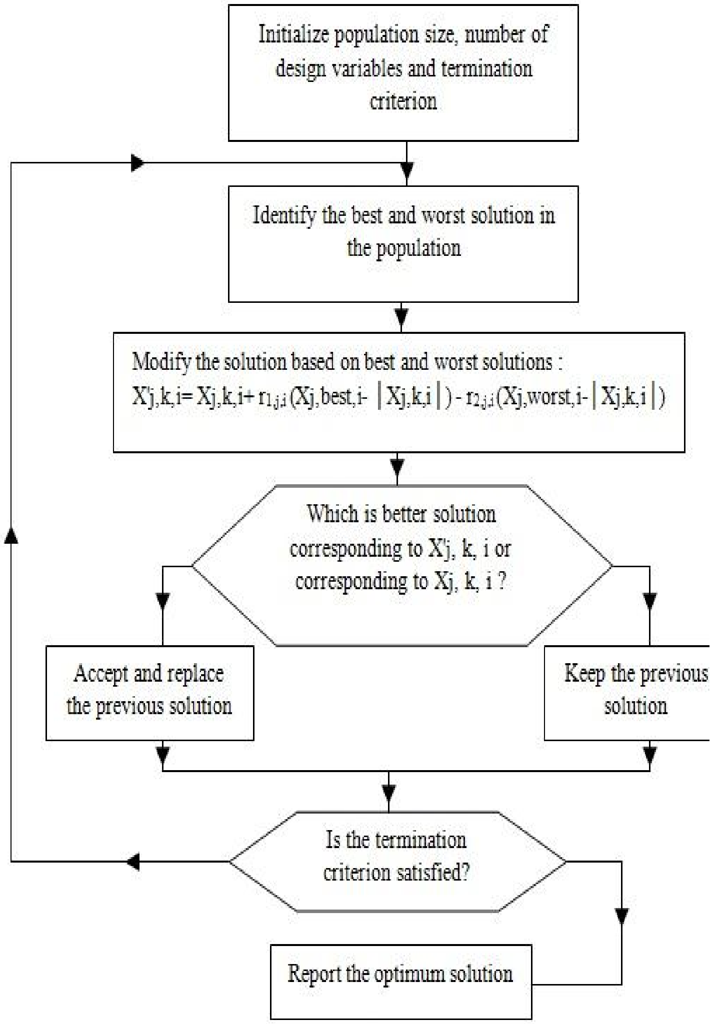
3:Select the best and worst value of the function f(x) as $f(x)_{best}$ and $f(x)_{worst}$ respectively, from the entire population size.

4:Modify the design variables as per the relation below:

$$X'_{j,k,i}=X_{j,k,i}+r_{1,j,i}(X_{j,best,i}-|X_{j,k,i}|)-r_{2,j,i}(X_{j,worst,i}-|X_{j,k,i}|)$$

5:X'j, k, i is accepted if gives better functional value.

6:All the accepted function values at the end of iteration are maintained and these values become the input to the next iteration



Where,

$X_{j,k,i}$	represent the value of i^{th} variable for j^{th} population during k^{th} iteration.
$X'_{j,k,i}$	modified value of $X_{j,k,i}$
$X_{j,best,i}$	value of the variable $j^{th}$for the bestcandidate.
$X_{j,worst,i}$	the value of the variable $j^{th}$for the worst candidate.
$r_{1,j,I}$&$r_{2,j,i}$	two random numbers for the j^{th} variable during the i^{th} iteration , in the range [0, 1]
$r_{1,j,i}$(Xj,best,i-|Xj,k,i|)	takes the solution towards best value.
- $r_{2,j,i}$(Xj,worst,i-|Xj,k,i|)	stops the movement of solution toward worst solution.

III. PROBLEM STATEMENT

The experiment on welding was conducted by Deyet al. [23] to show how optimization of different process parameters of welding results into minimum weldment area, with maximum bead penetration. Electron beam welding was used for welding of 5.00 mm thick austenitic stainless steel plates of grade ASS-304, in the selected case study. The bead geometry has significant influence on the welding. From its geometry cooling rate of welding, residual stresses in weldment structure, reasons of weld cracks, mechanical properties of the weld, etc. can easily be known. The bead geometry is profoundly influenced by different input process parameters used in the welding processes. The principal objective of the present study is to obtain better weldment area, by minimizing bead height and width and maximizing bead penetration using advanced optimization techniques. The working ranges of the input parameters i.e. accelerating voltage (V), beam current (I) and welding speed(S) were decided from the experience of the welder and these were selected as 60-90 kV, 7-9 mA and 60-90 cm/min respectively. The results of the experiments and response surface methodology were used to generate the response equation of bead height, width, and penetration. The un-coded response equation of BH, BW and BP were found to be as follows:

$$BH_{un\text{-}coded} = 0.2691 - 0.0162V + 0.1033I + 0.0008S + 0.0002V^2 + 0.0012I^2 + 0.00002S^2 - 0.0006VI - 0.00006VS - 0.0004IS \quad (11)$$

$$BW_{un\text{-}coded} = 6.6778 + 0.0103V - 0.8200I - 0.0437S - 0.00019V^2 + 0.0539I^2 + 0.0004S^2 + 0.0027VI - 0.00007VS - 0.0023IS \quad (12)$$

$$BP_{un\text{-}coded} = -11.2643 - 0.3047V + 4.9734I + 0.0697S + 0.0028V^2 - 0.2201I^2 - 0.0004S^2 - 0.0089VI - 0.00006VS - 0.0021IS \quad (13)$$

Minimization and maximization of above response equations were done using Jaya algorithm, Artificial Bee Colony algorithm, Teacher-Learner Based Optimization algorithm and Particle Swamp Optimization. MATLAB R2013a was used to write the codes of mentioned algorithms. The minimum value of BH and BW and maximum value of BP obtained were used in the weldment area (P) response equation and is given by:

$$P = C*[\,(BH/BH_{min})^a + (BW/BW_{min})^b + (BP_{max}/BP)^c\,] \quad (14)$$

Where,
a, b, c and C are non-negative numbers, there values are 2, 2, 22 and 10 respectively.

IV. RESULTS AND DISCUSSION

In this study, different computing techniques like ABC, PSO, TLBO, JAYA Algorithms are selected to optimise the parameters of EBW process for optimum weld quality of austenitic stainless steel plates of grade ASS-304 having thickness 5.0mm. A parametric study of population size and number generation are carried out based on the several runs (20 runs), the parameters are selected and optimisation are carried out until the termination criteria is satisfied. The optimisation of the parameters is done by using MATLAB 2013a [RAM 4GB, Intel Core i7 CPU, 2.8GHz, Win10 (64-bit) OS,] environment.

The 51 number of experiment as the population size, 3 input process parameters as the design variables and 100 number of generation are considered for all the bead geometry parameters. The lower and upper bond for design variables in section 3. The BH and BW are to be minimised (Eq.11 & 12) whereas, BP is to be maximised (Eq. 11). An attempt is made initially to determine the minimum value of BH and BW and maximum value of BP when single objective optimisation problem is considered and solved for the constraints within the ranges.

Table 1Optimum values for Single Objective Function

Methods	ABC	PSO	TLBO	Jaya Algorithm
BH (mm)	0.2050	0.2010	0.2010	0.2010
BW (mm)	1.9878	1.9367	1.9364	1.9874
BP (mm)	5.0002	5.0002	5.0002	5.0002

The combined objective function is formulated using these three outputs for minimum weldment area corresponding to maximum bead penetration(Eq.14). The results are compared with the Genetic Algorithm.

Table4Optimum values of bead geometries parameters

Methods	ABC	PSO	TLBO	Jaya Algorithm	GA
V (kV)	90.0000	89.9970	90.0000	90.0000	90.00
I (mA)	9.0000	8.7458	8.8248	8.8229	8.78
S (cm/min)	60.0000	74.6921	75.2719	75.3003	75.99
BH (mm)	0.5520	0.4641	**0.4609**	**0.4607**	0.48
BW (mm)	2.5015	**2.1441**	2.1464	2.2092	1.87
BP (mm)	**5.0002**	4.8566	4.8512	4.8505	5.02

V. CONCLUSION

The conclusions drawn from the stated work is carried out on ASS-304 austenitic stainless steel of thickness 5mm using EBW process are given below:

a) In this experimental approach the objective function is aimed at minimizing bead height and bead width, and maximizing bead penetration.
b) The optimization of process parameters by using TLBO, ABC, PSO and JAYA algorithms are implemented and the optimal parameters conditions are found out.
c) It is found that better value of bead height is obtained from Jaya Algorithm and TLBO, bead width from PSO and bead penetration from ABC.
d) A conclusion can easily be made from the result table that selected optimization techniques are equally good for the referred case study.

As a compliment this work elucidated that the searching time for the optimal solution can be made faster by using the proposed algorithms.

REFERENCES

[1] N. Karaboga ―A new design method based on ABC algorithm for digital IIR filters, Journal of Franklin Institute, vol. 346(4), 2009, pp. 328-338.

[2] R. V Rao, P. I. Pawar ―Modelling and optimization of process parameters of wire electrical discharge machining, Journal of Manufacture Engineering, 2009.

[3] LI. Dun-Ming, Ba-Yi. Cheng ―Artificial bee colony algorithm for scheduling a single batch processing machine with non-identical job sizes, Journal of Sichuan University (Natural Science Edition), 2009.

[4] R. V. Rao, P. J. Pawar ―Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms, Applied Soft Computing Journal, vol. 10(2), 2010, pp. 445-456.

[5] W. Lee, Y. Chen, T. Wu ―Optimization for ice-storage air-conditioning system using particle swarm algorithm, Applied Energy, vol. 86, 2009, pp. 1589-1595.

[6] A.A. Abou El-Ela, T. Fetouh, M.A. Bishr, R.A.F. Saleh―Power systems operation using particle swarm optimization technique, Electric Power Systems Research, vol. 78, 2008, pp. 1906–1913.

[7] P. R. Dian, M. S. Siti, S. Y. Siti ―Particle Swarm Optimization: Technique, System and Challenges, International Journal of Computer Applications, vol. 14 (1), 2011, pp. 19-27.

[8] R. Malviya, D. K. Pratihar―Tuning of neural networks using particle swarm optimization to model MIG,Swarm and Evolutionary Computation, vol. 1, 2011, pp. 223–235.

[9] T. Kannan, N. Murugan ―Optimization of flux cored arc welding process parameters using particle swarm

optimization technique, Journal for Manufacturing Science and Production, vol. 9, 2008, pp. 229-236.

[10] P. J. Pawar, R. V. Rao ―Parameter optimization of machining processes using teaching–learning-based optimization algorithm, International Journal of advanced Manufacturing Technology, vol.6, 2012,pp.995-1006.

[11] M. Omidvar, R. K. Fard, H. Sohrabpoor, R. Teimouri ―Selection of laser bending process parameters for maximal deformation angle through neural network and teaching and learning based optimization algorithm, Methodologies and application, vol. 19, 2015,pp.609-620.

[12] V. Togan ―Design of planar steel frames using Teaching–Learning Based Optimization, Engineering structures, vol. 34,2012 pp.225–232.

[13] R. V. Rao, V. Patel―Multi-objective optimization of heat exchangers using a modified teaching-learning-based optimization algorithm, Applied Mathematical Modelling, vol. 37,2013, pp.1147–1162.

[14] D. Karaboga, B. Basturk ―On the performance of artificial bee colony (ABC) algorithm, Computer Engineering Department, vol8, 2007, pp: 687-697.

[15] D. Karaboga, C. Ozturk, N. Karaboga, B. Gorkemli, ―Artificial bee colony programming for symbolic regression, Intelligent Systems Research Group, 209, 2012, pp: 1-15.

[16] Karaboga, ―An idea based on honey bee swarm for numerical optimization, Department of Computer Engineering, vol. 06, pp: 1-7.

[17] B**.** Akay, D. Karaboga, ―A modified artificial bee colony algorithm for real-parameter optimization, Department of Computer Engineering, 192, 2010, pp: 120-142.

[18] B. Qinghai ―Analysis of Particle Swarm Optimization Algorithm, Computer and Information Science, vol. 3, 2010, pp. 1.

[19] V. Hosseinnezhad, M. Rafiee, M. Ahmadian, & M. T. Ameli ―Species-based Quantum Particle Swarm Optimization for economic load dispatch, Electrical Power and Energy Systems, vol. 63, 2014, pp. 311-322.

[20] W. Al-Saedi, S. W. Lachowicz, D. Habibi, & O. Bass ―Power flow control in grid-connected microgrid operation using particle swarm optimization under variable load conditions. Electrical Power and Energy Systems, vol. 49, 2013, pp. 76-85.

[21] R. V. Rao, V. J. Savsni, D. P. Vakharia ―Teaching and learning based optimization; N optimization method for continues non-liner large scale problem, Information Sciences, vol. 183, 2012, pp. 1-15.

[22] R. V. Rao Jaya―Simple and new optimization algorithm for solving constrained and unconstrained optimization problems, International Journal of Industrial Engineering Computations, vol. 7, 2015, pp. 19-31.

[23] V. Dey, D. K. Prathihar, G. L. Datta, M. N. Jha, T. K. Saha, A. V. Bapat―Optimization of bead geometry in electron beamwelding using a Genetic Algorithm, Journal of Materials Processing Technology, vol. 209, pp. 1151-1157.

[24] R. V. Rao R ―Teaching–Learning-Based Optimization: An optimization method for continuous non-linear large scale problems, Computer Aided Design, vol. 43, 2012, pp. 303-315.

[25] A. M. Zain, H. Haron, S. Sharif ―Application of GA to optimize cutting conditions for minimizing surface roughness in end milling machining process, Expert Systems with Applications, vol. 37, 2010, pp. 4650–4659.