Journal of the Brazilian Society of Mechanical Sciences and Engineering (2020) 42:6
https://doi.org/10.1007/s40430-019-2089-9

TECHNICAL PAPER

Accelerated cuckoo optimization algorithm for the multi-objective welding process

Mohamed Arezki Mellal[1] · **Abdellah Salhi**[2] · **Edward J. Williams**[3,4]

Received: 11 July 2019 / Accepted: 18 November 2019

Abstract

Welding is a well-known process in manufacturing industries due to its importance. Several process parameters should be tuned in order to perform a high-quality welding. Usually, the problem is described as an optimization one and the challenge is to reconcile conflicting objectives. This paper deals with a multi-objective welding process namely the submerged arc welding process, involving five objectives. The weighted sum approach is used to handle it. An accelerated cuckoo optimization algorithm is implemented for this process model and applied to a practical instance of it. On this practical example, the superiority of the proposed optimization technique has been demonstrated in terms of better solutions and fewer required generations of the cuckoos relative to the basic COA and four other optimization algorithms.

Keywords Welding process · Process parameters · Accelerated cuckoo optimization algorithm (ACCOA)

1 Introduction

Manufacturing processes are the baselines for any industrial firm to design and make a product. There is a wide range of processes involved such as turning, grinding, milling, ultrasonic machining, abrasive jet machining, and water jet machining, depending on the target product of the manufacturer. Manufacturers strive to optimize these processes individually or overall as the manufacturing circumstance may dictate. Consequently, in the literature, most of the processes are formulated as optimization problems. These often improve the process performances by providing the best values for the process parameters. Unfortunately, these problems are often intractable meaning that classical approaches are not effective.

In the last decade, it has been observed that soft computing methods (computational intelligence) are powerful enough to solve this kind of problems. In [1], a genetic algorithm has been applied to optimize the fiber-reinforced composite injection molding process. The heat-treatment process of an alloy of titanium has been optimized in [2] by using the Taguchi method, while the turning of the same alloy has been optimized in [3] by integrating the gray relational analysis with the Taguchi method. The production time of the multi-pass milling process has been optimized by using the artificial bee colony (ABC) approach, the particle swarm optimization (PSO), and simulated annealing (SA) in [4], whereas the cuckoo optimization algorithm (COA) appears in [5]. The unit production cost of the multi-pass turning process has been minimized by using the teaching–learning-based optimization algorithm (TLBO) [6] and the COA [7]. The machining parameters of other traditional and non-traditional processes have been investigated in [8, 9]. In engineering optimization, the objectives may vary and conflict at the same time, in which case, the problem becomes multi-objective. It can be converted into a single objective by combining the objectives or solving by a Pareto approach.

Technical Editor: Lincoln Cardoso Brandao.

✉ Mohamed Arezki Mellal
mellal.mohamed@gmail.com;
mellal.mohamed@univ-boumerdes.dz

1 LMSS, Faculty of Technology (FT), M'Hamed Bougara University, 35000 Boumerdes, Algeria

2 Department of Mathematical Sciences, University of Essex, Colchester CO4 3SQ, UK

3 Industrial and Manufacturing Systems Engineering Department, College of Engineering and Computer Science, University of Michigan, Dearborn 48126, USA

4 Decision Sciences, College of Business, University of Michigan, Dearborn 48126, USA

Published online: 25 November 2019

Some authors focused on the optimization of the important manufacturing process of welding [10–16]. The aim of the current work is to deal with the multi-objective optimization problem of the submerged arc welding process (SAW) [11, 16]. Our approach is to convert it into a single objective using the weighted sum method. This allows handling the different objectives by resorting to weights assigned to each objective function [17]. In [11], a regression model has been established by experimental means and the optimization problem has been solved using the teaching–learning-based optimization algorithm, whereas in [16], the Jaya algorithm has been improved for this purpose. An accelerated cuckoo optimization algorithm (ACCOA) is implemented for solving the problem in the current work.

The remainder of the paper is organized as follows: Section 2 defines the multi-objective problem of the SAW process expressed by the weighted sum method. Section 3 describes the steps of the implemented ACCOA. A discussion of the obtained results is given in Sect. 4. Finally, the conclusion summarizes the paper and outlines some further likely developments.

2 Multi-objective model representation of the submerged arc welding

The submerged arc welding process is defined by an arc maintained between a continuously fed bare wire electrode and the workpiece and a blanket of powdered flux which generates a protective gas shield. It is an economical method of metal joining [18].

The multi-objective optimization problem of the submerged arc welding of Cr–Mo–V steel investigated here is based on the empirical formulation developed by Rao and Kalyankar [11, 16]. The problem involves two minimization objectives: bead width (*BW*) in mm and weld reinforcement (*R*) in mm, and three maximization objectives: weld penetration (*P*) in mm, tensile strength (*TS*) in MPa, and weld hardness (*H*) in Rc. The control parameters of the considered SAW process are the welding current (*I*) in Amp, voltage (*V*) in volts, welding speed (*S*) in cm/min, and wire feed (*F*) in cm/min. It should be noted that the weld reinforcement must be greater than zero.

The regression models of the objectives are given as follows:

Bead width (BW)

$$\begin{aligned}\text{Minimize BW} = 475.425 &- 0.9814I - 15.0015V + 2.4805S \\ &- 0.351F + 0.001179I^2 + 0.25575V^2 \\ &- 0.109781S^2 + 0.000773F^2 + \varepsilon_{\text{BW}}\end{aligned} \tag{1}$$

where ε_{BW} is the error term which takes value 0.656092.

Weld reinforcement (R)

$$\begin{aligned}\text{Minimize } R = 931.851 &- 2.45118I - 30.4892V - 2.44028S \\ &+ 0.111489F + 0.0778514IV + 0.00841464IS \\ &- 0.0171696VS + \varepsilon_R\end{aligned} \tag{2}$$

where ε_R is the error term which takes value 0.60023.

Weld penetration (P)

$$\begin{aligned}\text{Maximize } P = -668.516 &+ 0.094333I + 43.0883V \\ &+ 0.47667S + 0.064944F - 0.000092I^2 \\ &- 0.7175V^2 - 0.018515S^2 - 0.000134F^2 + \varepsilon_P\end{aligned} \tag{3}$$

where ε_P is the error term equal to 0.623273.

Tensile strength (TS)

$$\begin{aligned}\text{Maximize TS} = -1148.73 &- 0.1934I + 20.1667V + 9.5S \\ &+ 9.774F + 0.001467I^2 - 0.0834V^2 \\ &- 0.4037S^2 - 0.01885F^2 + \varepsilon_{\text{TS}}\end{aligned} \tag{4}$$

where ε_{TS} is the error term equal to 2.170478.

Weld hardness (H)

$$\begin{aligned}\text{Maximize } H = 772.444 &- 1.45667I - 30V - 0.04167S + 0.00556F \\ &+ 0.0018I^2 + 0.5V^2 + \varepsilon_H\end{aligned} \tag{5}$$

where ε_H is the error term equal to 0.078337.

The upper and lower bounds of each parameter are given as follows:

$$350 \text{ (Ampere)} \le I \le 450 \text{ (Ampere)} \tag{6}$$

$$28 \text{ (Volt)} \le V \le 32 \text{ (Volt)} \tag{7}$$

$$4 \text{ (cm/min)} \le S \le 20 \text{ (cm/min)} \tag{8}$$

$$190 \text{ (cm/min)} \le F \le 310 \text{ (cm/min)} \tag{9}$$

In the literature, two scenarios are considered for the above objectives, i.e., with and without error terms.

The result of combining objectives using the weighted sum method can be written as follows:

$$\begin{aligned}\text{Minimize } Z = w_1\left(\frac{BW}{BW*}\right) + w_2\left(\frac{R}{R*}\right) - w_3\left(\frac{P}{P*}\right) \\ - w_4\left(\frac{TS}{TS*}\right) - w_5\left(\frac{H}{H*}\right)\end{aligned} \tag{10}$$

where BW^*, R^*, P^*, TS^*, and H^* are the optimal values of the objectives when the problem is solved as a single-objective problem. Here, the values of the weights used in [16] are maintained, i.e., w_i=0.2 for i = 1,…,5.

3 Accelerated cuckoo optimization algorithm

The cuckoo optimization algorithm (COA) has been introduced by Rajabioun in [19]. It is a soft computing method inspired by the special lifestyle of the cuckoo. This bird has the trait of laying its eggs in other birds' nests of different species. The patterns of invaded birds eggshells are mimicked to evade recognition which may result in the destruction of the eggs. However, this is not always successful and some dissimilar eggs are indeed destroyed. It is also the case that, some cuckoo chicks will starve after hatching, as they eat more than the chicks of the invaded species. The algorithm is based on an egg laying radius (ELR) and the migration of mature cuckoos. Its effectiveness has been proved and it has been implemented for solving various engineering optimization problems, such as the PID controller [19, 20], pattern recognition [21], replacement of obsolete components [22, 23], data mining and clustering [24, 25], combined heat and economic power dispatch [26], and machining parameters [5, 7, 9, 27]. The procedure which determines ELR and that which sets the run parameters form crucial steps of COA. They may be the aspects of the algorithm which contribute to the loss of the best solution when dealing with combined objective functions.

In the current work, the ELR is replaced by a binary procedure to improve COA when solving the problem with the combined objective function of the SAW process. This led to the so-called accelerated cuckoo optimization algorithm (ACCOA). It is implemented as follows:

3.1 ACCOA: the accelerated cuckoo optimization algorithm

Begin ACCOA

Step 1: Generate a random number of solutions which represents a set of candidate habitats.

$$\begin{aligned} &\text{Habitat}_1 = [I, V, S, F] \\ &\text{Habitat}_2 = [I, V, S, F] \\ &\vdots \\ &\text{Habitat}_N = [I, V, S, F] \end{aligned} \tag{11}$$

where N is the number of total habitats.
Step 2: Dedicate some eggs to each cuckoo.
Step 3: Binary egg laying.
Some of the dedicated eggs hatch and those remaining are detected and destroyed by the invaded birds. A binary value is randomly generated for each egg.

$$Egg = \begin{cases} 0 & \text{if the egg is not recognized} \\ \text{else} & \\ 1 & \end{cases} \tag{12}$$

Equation (12) is used for the intensification of the algorithm.
Step 4: Limit the total number of surviving cuckoos.
Step 5: Evaluate fitness.
Step 6: Find the best habitat.
Step 7: Migrate the cuckoo to the best habitat.
Step 8: If the number of cuckoo iterations is reached, stop; otherwise, go to **Step 2**.

End ACCOA

It should be noted that the number of habitats is constant at each iteration, and the best habitat is introduced in the next iteration. Figure 1 shows the flowchart.

4 Results and discussion

ACCOA has been coded in MATLAB 2015 and run on a personal computer with a processor G620 (2.60 GHz, Sandy Bridge, 4 GB Memory, Windows 7, 32 bits). The algorithm has been applied to the five objectives as single-objective problems [see Eq. (1)–(5)]. It has then been applied to the problem involving the combination of all five objective functions in a single objective [see Eq. (10)].

Table 1 summarizes the optimal results obtained by the teaching–learning-based algorithm (TLBO) [11], Jaya algorithm (Jaya) [16], quasi-oppositional-based Jaya algorithm (Q-O Jaya) [16], the plant propagation algorithm (PPA) [28–30], the simple cuckoo optimization algorithm (COA) [19], and the accelerated cuckoo optimization algorithm (ACCOA) for the single regression models without the error terms, where the bold type represents the best value. The number of habitats fixed for the COA and the ACCOA is 20 in order to compare the different performances. As reported in [11, 16], the population size was fixed to 20 for Jaya and Q-O Jaya in order to be able to compare the required number of generations to reach the optimal solutions. The optimal BW is similar for ACCOA, COA, Jaya algorithm and Q-O Jaya, i.e., 17.062. However, the ACCOA required the smallest number of iterations (eight iterations). PPA provided better results (17.0748) than TLBO (17.110). For the R objective, ACCOA decreased its value to 1.3312E − 05 and required only six iterations, compared to COA (0.0011 with 14 iterations), Jaya algorithm (0.00355 with 15 iterations), PPA (9.3467E − 04 with 23 iterations), and Q-O Jaya (0.0027 with 12 iterations). For the P objective, ACCOA

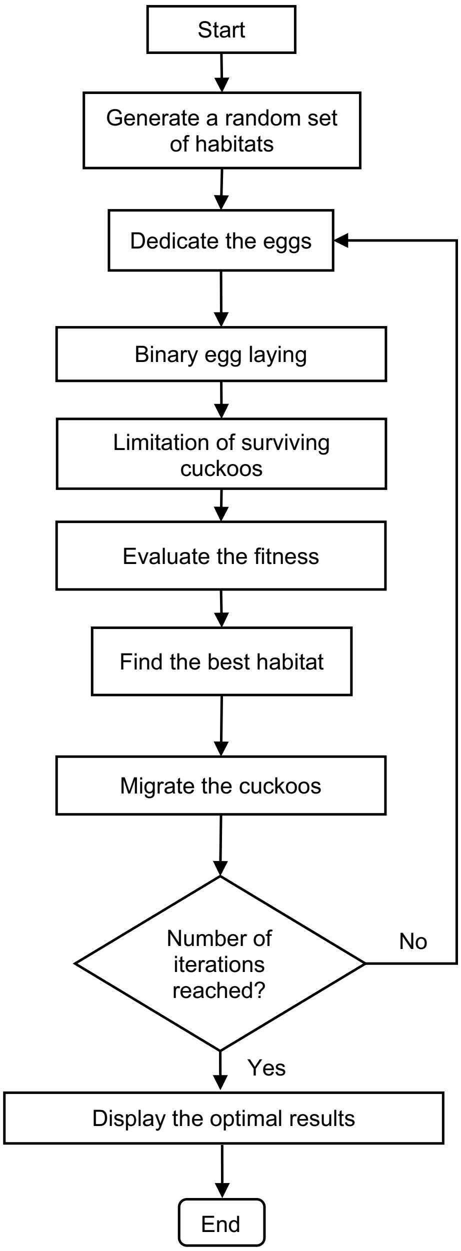

Fig. 1 Flowchart of the accelerated cuckoo optimization algorithm

(equal to 13.1402) has also outperformed the other algorithms: COA (13.1402), TLBO (11.16), Jaya algorithm (11.50), PPA (13.0652), and Q-O Jaya (11.50). The values provided by Jaya and Q-O Jaya for the *TS* objective are better than those of COA and ACCOA. However, the value achieved by ACCOA is better than that of COA and required fewer generations. The *H* objective value is similar for ACCOA, ACO, Jaya algorithm, PPA, and Q-O Jaya (36.66). PPA required only one generation. It seems that this objective has reached the maximum possible value.

Table 2 summarizes the optimal results obtained for the combined objective. It clearly shows that the objective value provided by ACCOA (−0.1065) is better than those of COA (−0.0108), PPA (−0.0469), TLBO (19.00), Jaya algorithm (0.5644), and Q-O Jaya (0.1933). Moreover, ACCOA required fewer iterations (six iterations) than the rest of algorithms. PPA came second, overall.

The optimal results for the combined objective, considering the error terms, are reported in Table 3. Here again, ACCOA outperforms the other methods. The optimal value of ACCOA is (−0.3215), whereas that of COA is (−0.2907), PPA (−0.0720), and those of the Jaya algorithm and the Q-O Jaya are similar (−0.0064). Furthermore, ACCOA required fewer generations (three iterations) compared to COA (15 iterations), PPA (78 iterations), Jaya algorithm (11 iterations), and Q-O Jaya (seven iterations).

It should be noted that the required CPU times in the combined objective are as follows: without errors (ACOA: 17.25 s; ACCOA: 3.55 s) and with errors (ACOA: 17.67 s; ACCOA: 1.89 s). Figures 2, 3, 4, 5, 6, and 7 summarize the performance of each algorithm for the combined objective. The optimal value by TLBO in the case without errors has not been illustrated.

5 Conclusions

The goal of this paper was to evaluate the efficiency and robustness of a number of relatively new heuristics on a well-known multi-objective problem that arises in manufacturing. One of these algorithms, namely ACCOA which we introduce here for the first time, is a modification (acceleration) of the well-known cuckoo optimization algorithm. The test problem is the multi-objective optimization model of the submerged arc welding process expressed with the weighed sum method. In the literature, the problem has five objectives: the bead width, the weld reinforcement, the weld penetration, the tensile strength, and the weld hardness. ACCOA implements a binary decision to avoid the disadvantage due to the egg laying radius of the original cuckoo optimization algorithm. The results reveal the effectiveness of the current approach in terms of better results (robustness) and lower numbers of required iterations (efficiency) for reaching the optimum results. The disadvantage of the current work is related to the decision on the number of eggs. Further work on this issue is underway. Moreover, work on an application to a welding process involving more than five objectives and four decision variables will be reported in the future. On the other hand, availability of adequate equipment will experimentally investigate the results.

Table 1 Optimal results for a single-objective problem (without error terms)

Objective	Method	*I* (Ampere)	*V* (Volt)	*S* (cm/min)	*F* (cm/min)	Optimum result	Required no. of iterations
BW*	**ACCOA**	416.357	29.337	20.000	226.852	**17.062**	**8**
	COA	416.200	29.328	20.000	227.037	**17.062**	20
	PPA	415.5685	29.1592	20	229.3993	17.0748	179
	TLBO [11]	412.000	29.000	20.000	228.000	17.110	–
	Jaya [16]	416.200	29.327	20.000	227.043	**17.062**	25
	Q-O Jaya [16]	416.500	29.342	20.000	226.940	**17.062**	17
R*	**ACCOA**	426.0490	31.9591	9.3009	191.0474	**1.3312E – 05**	**6**
	COA	450.0000	32.0000	4.0000	202.6015	0.0011	14
	PPA	450	32	5.7628	190	9.3467E – 04	23
	TLBO [11]	378.0000	31.0000	18.0000	214.0000	0.0086	–
	Jaya [16]	375.8213	30.9250	7.1382	233.7626	0.00355	15
	Q-O Jaya [16]	350.0000	30.8981	4.9221	236.2409	0.0027	12
P*	**ACCOA**	449.9999	30.0266	12.8725	242.3283	**13.1402**	**7**
	COA	450.0000	30.0303	12.9001	242.7443	13.1401	19
	PPA	450.0000	29.8164	12.1109	226.7450	13.0652	15
	TLBO [11]	444.0000	29.0000	5.0000	241.0000	11.16	–
	Jaya [16]	450.0000	30.1887	4.0000	277.1496	11.50	24
	Q-O Jaya [16]	368.4300	30.3395	11.9418	241.3833	11.50	15
TS*	**ACCOA**	450.0000	32.0000	11.7662	259.2621	944.0975	**5**
	COA	450.0000	32.0000	11.7946	259.3121	944.0971	18
	PPA	450.0000	32.0000	4.0000	259.1989	881.7714	95
	TLBO [11]	448.0000	32.0000	11.0000	253.0000	940.90	–
	Jaya [16]	450.0000	32.0000	11.7660	259.2569	**944.12**	20
	Q-O Jaya [16]	450.0000	32.0000	11.7660	259.2569	**944.12**	9
H*	**ACCOA**	350.0000	28.0000	4.0000	310.0000	**36.66**	2
	COA	350.0000	28.0000	4.0000	310.0000	**36.66**	3
	PPA	350.0000	28.0000	4.0000	310.0000	**36.66**	**1**
	TLBO [11]	350.0000	28.0000	4.0000	307.0000	36.65	–
	Jaya [16]	350.0000	28.0000	4.0000	310.0000	**36.66**	3
	Q-O Jaya [16]	350.0000	28.0000	4.0000	310.0000	**36.66**	2

Table 2 Optimal results for combined objective (without error terms)

Method	*I* (Ampere)	*V* (Volt)	*S* (cm/min)	*F* (cm/min)	*BW*	*R*	*P*	*TS*	*H*	Min *Z*	Required no. of iterations
ACCOA	382.5168	32.0000	19.6968	214.9693	20.9075	7.5969E – 06	8.1865	812.3678	30.9921	**– 0.1065**	6
COA	404.5336	28.0591	17.6312	206.8119	21.8616	0.0010	9.0596	770.4275	30.0361	– 0.0108	15
PPA	450.0000	32.0000	5.7625	190.0000	26.2462	6.9561E – 04	9.0434	784.4098	34.2587	– 0.0469	78
TLBO [11]	445.0000	32.0000	7.0000	193.0000	27.05	0.826	9.32	846.6	33.45	19.00	–
Jaya [16]	423.1719	29.8221	4.0000	267.0907	20.89	0.0152	11.19	856.75	29.69	0.5644	13
Q-O Jaya [16]	382.41	29.416	20.0000	190.0000	19.47	0.0062	10.36	717.99	29.02	0.1933	18

Table 3 Optimal results for combined objective (with error terms)

Method	*I* (Ampere)	*V* (Volt)	*S* (cm/min)	*F* (cm/min)	BW	R	P	TS	H	Min Z	Required no. of iterations
ACCOA	449.6237	30.6568	4.0000	251.0311	22.4001	0.0011	12.0065	900.1521	32.9043	**−0.3215**	3
COA	450.0000	30.7105	4.0000	248.9756	22.3942	7.1693E−04	11.9645	900.6669	32.9908	−0.2907	15
PPA	450.0000	32.0000	5.0089	190.0000	25.9241	3.9623E−04	9.4577	789.8573	34.3685	−0.0720	138
Jaya [16]	350.0000	28.0000	4.0000	190.0000	26.865	5.636	6.9176	672.855	36.077	−0.0064	11
Q-O Jaya [16]	350.0000	28.0000	4.0000	190.0000	26.865	5.636	6.9176	672.855	36.077	−0.0064	7

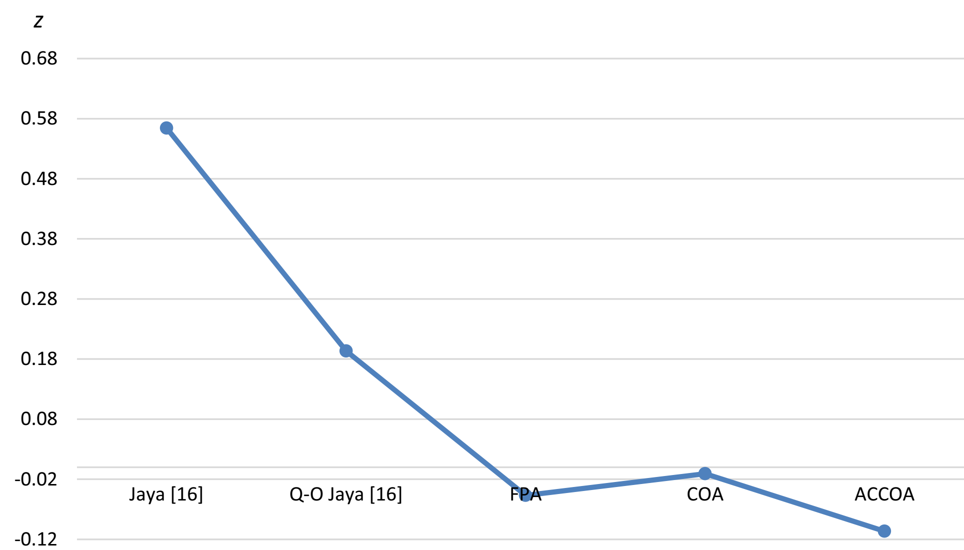

Fig. 2 Optimal value of z for combined objective (without error terms)

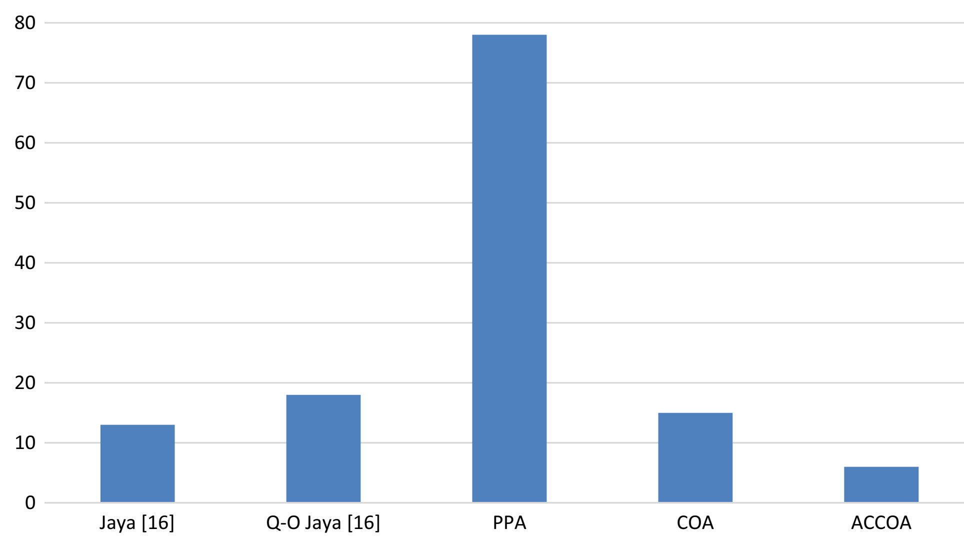

Fig. 3 Required number of iterations for combined objective (without error terms)

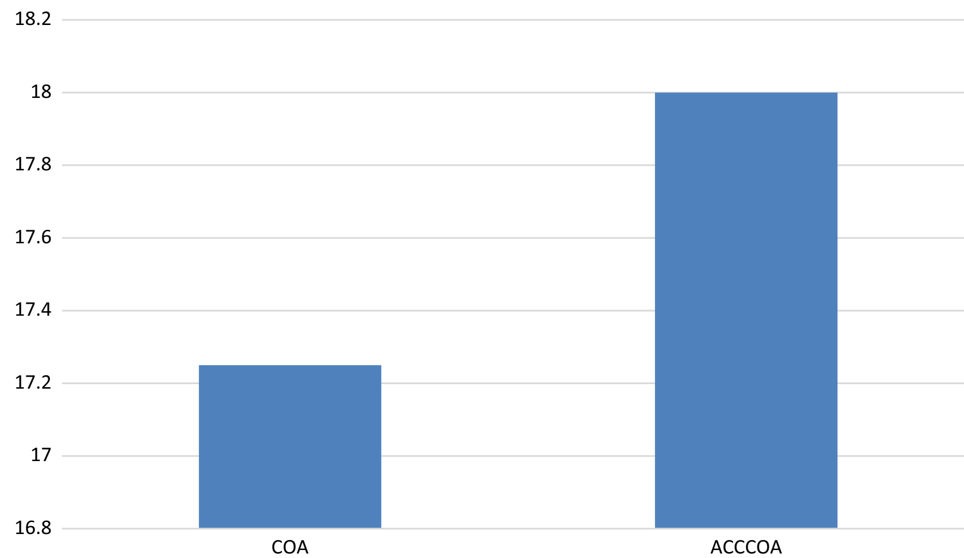

Fig. 4 Required CPU time (s) for combined objective (without error terms)

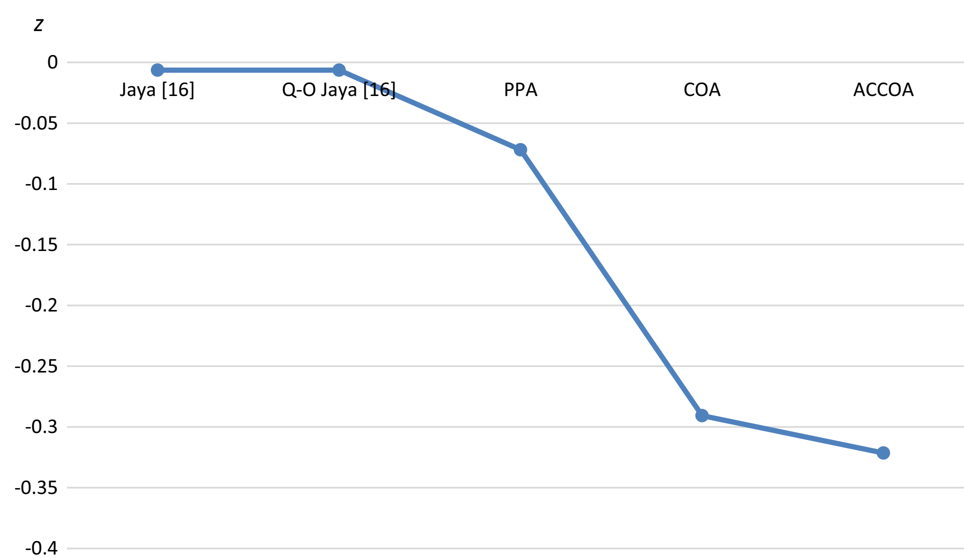

Fig. 5 Optimal value of z for combined objective (with error terms)

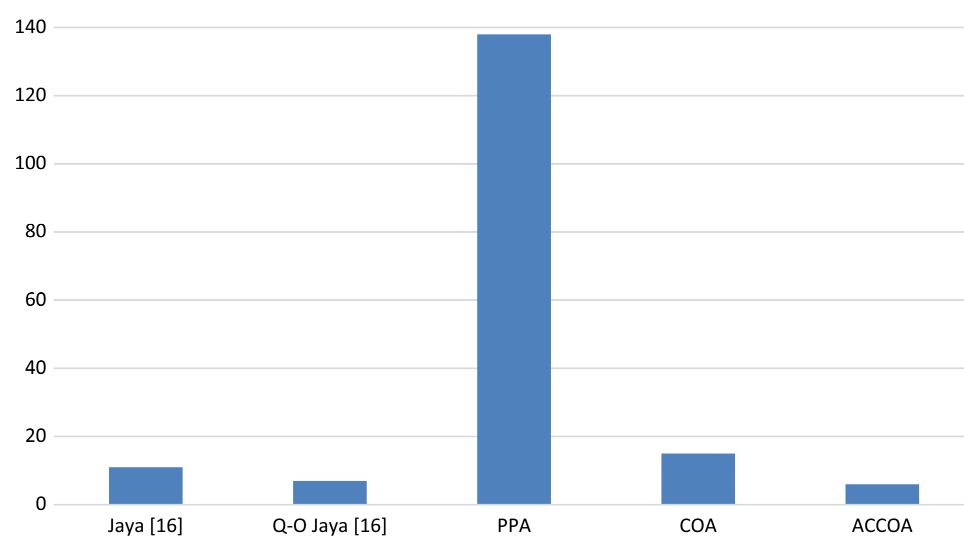

Fig. 6 Required number of iterations for combined objective (with error terms)

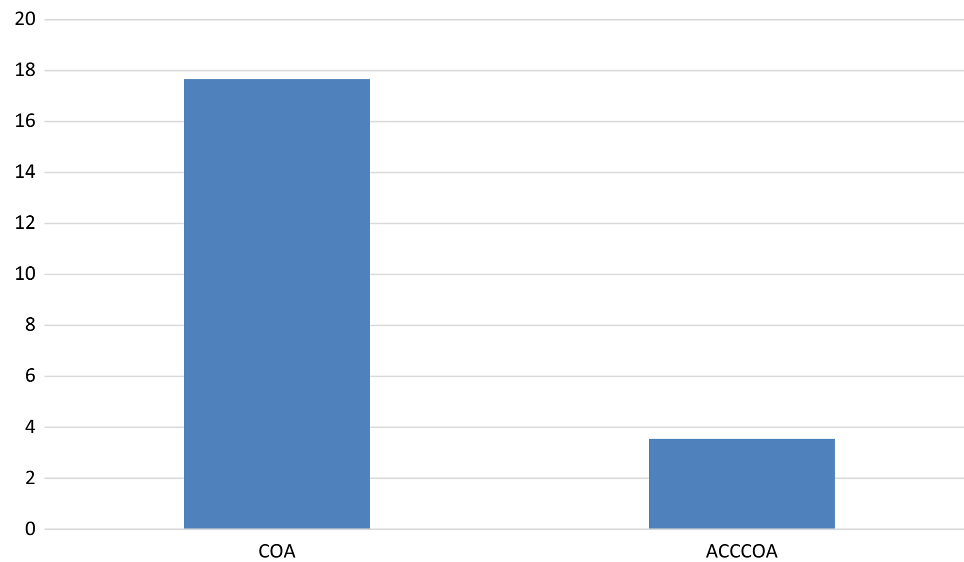

Fig. 7 Required CPU time (s) for combined objective (with error terms)

Acknowledgements We are grateful to ESRC for funding this research under grant number ES/L011859/1.

Compliance with ethical standards

Conflict of interest The authors declare no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Li K, Yan S, Pan W, Zhao G (2017) Warpage optimization of fiber-reinforced composite injection molding by combining back propagation neural network and genetic algorithm. Int J Adv Manuf Technol 90:963–970
2. Lee KH, Yang SY, Yang JG (2017) Optimization of heat-treatment parameters in hardening of titanium alloy Ti–6Al–4 V by using the Taguchi method. Int J Adv Manuf Technol 90:753–761
3. Mia M, Khan MA, Rahman SS, Dhar NR (2017) Mono-objective and multi-objective optimization of performance parameters in high pressure coolant assisted turning of Ti–6Al–4V. Int J Adv Manuf Technol 90:109–118
4. Rao RV, Pawar PJ (2010) Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms. Appl Soft Comput 10:445–456
5. Mellal MA, Williams EJ (2016) Total production time minimization of a multi-pass milling process via cuckoo optimization algorithm. Int J Adv Manuf Technol 87:747–754. https://doi.org/10.1007/s00170-016-8498-3
6. Rao RV, Kalyankar VD (2013) Multi-pass turning process parameter optimization using teaching-learning-based optimization algorithm. Sci Iran 20:967–974. https://doi.org/10.1016/j.scient.2013.01.002
7. Mellal MA, Williams EJ (2015) Cuckoo optimization algorithm for unit production cost in multi-pass turning operations. Int J Adv Manuf Technol 76:647–656. https://doi.org/10.1007/s00170-014-6309-2
8. Pawar PJ, Rao RV (2013) Parameter optimization of machining processes using teaching–learning-based optimization algorithm. Int J Adv Manuf Technol 67:995–1006
9. Mellal MA, Williams EJ (2016) Parameter optimization of advanced machining processes using cuckoo optimization algorithm and hoopoe heuristic. J Intell Manuf 27:927–942
10. Rambabu G, Balaji Naik D, Venkata Rao CH, Srinivasa Rao K, Madhusudan Reddy G (2015) Optimization of friction stir welding parameters for improved corrosion resistance of AA2219 aluminum alloy joints. Def Technol 11:330–337. https://doi.org/10.1016/j.dt.2015.05.003
11. Rao RV, Kalyankar VD (2013) Experimental investigation on submerged arc welding of Cr–Mo–V steel. Int J Adv Manuf Technol 69:93–106
12. Elangovan S, Anand K, Prakasan K (2012) Parametric optimization of ultrasonic metal welding using response surface methodology and genetic algorithm. Int J Adv Manuf Technol 63:561–572. https://doi.org/10.1007/s00170-012-3920-y
13. Kanigalpula PKC, Pratihar DK, Jha MN, Derose J, Bapat AV, Pal AR (2016) Experimental investigations, input–output modeling and optimization for electron beam welding of Cu–Cr–Zr alloy plates. Int J Adv Manuf Technol 85:711–726. https://doi.org/10.1007/s00170-015-7964-7
14. Chen F, Tong GQ, Yue XK, Ma XL, Gao XP (2017) Multi-performance optimization of small-scale resistance spot welding process parameters for joining of Ti–1Al–1Mn thin foils using hybrid approach. Int J Adv Manuf Technol 89:3641–3650
15. Resendiz-Flores EO, Lopez-Quintero ME (2017) Optimal identification of impact variables in a welding process for automobile seats mechanism by MTS-GBPSO approach. Int J Adv Manuf Technol 90:437–443
16. Rao RV, Rai DP (2017) Optimisation of welding processes using quasi-oppositional-based Jaya algorithm. J Exp Theor Artif Intell. https://doi.org/10.1080/0952813X.2017.1309692

17. Konak A, Coit DW, Smith AE (2006) Multi-objective optimization using genetic algorithms: a tutorial. Reliab Eng Syst Saf 91:992–1007. https://doi.org/10.1016/j.ress.2005.11.018
18. Mitra U (1984) Kinetics of slag metal reactions during submerged arc welding of steel. Massachusetts Institute of Technology, USA
19. Rajabioun R (2011) Cuckoo optimization algorithm. Appl Soft Comput 11:5508–5518
20. Fard AN, Shahbazian M, Hadian M (2016) Adaptive fuzzy controller based on cuckoo optimization algorithm for a distillation column. In: ICCIA international conference computational intelligence and application
21. Khormali A, Addeh J (2016) A novel approach for recognition of control chart patterns: type-2 fuzzy clustering optimized support vector machine. ISA Trans. https://doi.org/10.1016/j.isatra.2016.03.004
22. Mellal MA, Adjerid S, Williams EJ, Benazzouz D (2012) Optimal replacement policy for obsolete components using cuckoo optimization algorithm based-approach: dependability context. J Sci Ind Res 71:715–721
23. Mellal MA, Adjerid S, Williams EJ (2013) Optimal selection of obsolete tools in manufacturing systems using cuckoo optimization algorithm. Chem Eng Trans 33:355–360. https://doi.org/10.3303/CET1333060
24. Afshari MH, Dehkordi MN, Akbari M (2016) Association rule hiding using cuckoo optimization algorithm. Expert Syst Appl 64:340–351. https://doi.org/10.1016/j.eswa.2016.08.005
25. Amiri E, Mahmoudi S (2016) Efficient protocol for data clustering by fuzzy cuckoo optimization algorithm. Appl Soft Comput 41:15–21
26. Mellal MA, Williams EJ (2015) Cuckoo optimization algorithm with penalty function for combined heat and power economic dispatch problem. Energy 93:1711–1718. https://doi.org/10.1016/j.energy.2015.10.006
27. Mellal MA, Williams EJ (2017) The cuckoo optimization algorithm and its applications. In: Handbook of neural computation. Elsevier, Amsterdam, pp 269–277
28. Salhi A, Fraga ES (2011) Nature-inspired optimisation approaches and the new plant propagation algorithm. In: International conference on numerical analysis optimization
29. Sulaiman M, Salhi A, Selamoglu BI, Kirikchi OB (2014) A plant propagation algorithm for constrained engineering optimisation problems. Math Probl Eng. https://doi.org/10.1155/2014/627416
30. Sulaiman M, Salhi A, Fraga ES, Mashwani WK, Rashidi MM (2015) A novel plant propagation algorithm: modifications and implementation. Sci Int 28:201–209