

Article

A Cuckoo Search Algorithm Using Improved Beta Distributing and Its Application in the Process of EDM

Dili Shen ¹, Wuyi Ming ², Xingguo Ren ^{3,*}, Zhuobin Xie ², Yong Zhang ⁴ and Xuewen Liu ³

¹ School of Mechanical-Electronic and Automobile Engineering, Zhengzhou Institute of Technology, Zhengzhou 450052, China; shendili@163.com

² Mechanical and Electrical Engineering Institute, Zhengzhou University of Light Industry, Zhengzhou 450002, China; mingwuyi@gmail.com (W.M.); xzb18336947581@163.com (Z.X.)

³ School of Vehicle and Automation, Guangzhou Huaxia Vocational College, Guangzhou 510900, China; liuxuewen@gzhxtc.edu.cn

⁴ School of Advanced Manufacturing Technology, Guangdong Mechanical & Electrical College, Guangzhou 510550, China; 2018010044@gdmec.edu.cn

* Correspondence: Renxinggui@gzhxtc.edu.cn

Abstract: Lévy flights random walk is one of key parts in the cuckoo search (CS) algorithm to update individuals. The standard CS algorithm adopts the constant scale factor for this random walk. This paper proposed an improved beta distribution cuckoo search (IBCS) for this factor in the CS algorithm. In terms of local characteristics, the proposed algorithm makes the scale factor of the step size in Lévy flights showing beta distribution in the evolutionary process. In terms of the overall situation, the scale factor shows the exponential decay trend in the process. The proposed algorithm makes full use of the advantages of the two improvement strategies. The test results show that the proposed strategy is better than the standard CS algorithm or others improved by a single improvement strategy, such as improved CS (ICS) and beta distribution CS (BCS). For the six benchmark test functions of 30 dimensions, the average rankings of the CS, ICS, BCS, and IBCS algorithms are 3.67, 2.67, 1.5, and 1.17, respectively. For the six benchmark test functions of 50 dimensions, moreover, the average rankings of the CS, ICS, BCS, and IBCS algorithms are 2.83, 2.5, 1.67, and 1.0, respectively. Confirmed by our case study, the performance of the ABCS algorithm was better than that of standard CS, ICS or BCS algorithms in the process of EDM. For example, under the single-objective optimization convergence of *MRR*, the iteration number (13 iterations) of the CS algorithm for the input process parameters, such as discharge current, pulse-on time, pulse-off time, and servo voltage, was twice that (6 iterations) of the IBCS algorithm. Similar, the iteration number (17 iterations) of BCS algorithm for these parameters was twice that (8 iterations) of the IBCS algorithm under the single-objective optimization convergence of *Ra*. Therefore, it strengthens the CS algorithm's accuracy and convergence speed.

Keywords: cuckoo search algorithm; self-adaption; beta distribution; dynamic step-size control factor; EDM



Citation: Shen, D.; Ming, W.; Ren, X.; Xie, Z.; Zhang, Y.; Liu, X. A Cuckoo Search Algorithm Using Improved Beta Distributing and Its Application in the Process of EDM. *Crystals* **2021**, *11*, 916. <https://doi.org/10.3390/cryst11080916>

Academic Editors: Youmin Rong, Zhen Zhang and Zhi Chen

Received: 11 July 2021

Accepted: 2 August 2021

Published: 6 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Evolutionary computing algorithms, such as the particle swarm optimization (PSO) algorithm [1,2], the artificial bee colony (ABC) algorithm [3,4], the glowworm swarm optimization (GSO) algorithm [5,6], and the wolf colony (WC) algorithm [7,8], have the advantages of reliable performance and global search when solving continuous function optimization problems, which have attracted the interest many scholars in applying them to related practical parameter optimization problems. These algorithms are mainly inspired by the intelligent phenomenon of biological groups in the natural world, and are a kind of random optimization algorithm proposed by imitating the behavior of social animals. They are heuristic search algorithms that are optimized based on a given goal, the core premise of

which is that a group composed of many simple individuals can achieve complex functions through simple cooperation with each other. These algorithms can solve problems such as data mining, network routing optimization, robot path planning, logistics and distribution vehicle scheduling, and wireless sensor networks. However, these algorithms have their own shortcomings. For example, the disadvantage of the PSO algorithm is the lack of dynamic adjustment of speed, which makes it easy to fall into local optimization, resulting in low convergence accuracy and difficult convergence; The GSO algorithm must require excellent individuals within the perceptual range to provide information, otherwise the individuals will stop searching, thus reducing the convergence speed. Therefore, Yang and Deb [9,10] proposed a cuckoo search (CS) algorithm, in which this evolutionary calculation simulates the behavior of cuckoos looking for nests and laying eggs and letting the host birds hatch them on their behalf. The CS algorithm uses Lévy flights to move randomly instead of simple isotropic random search, which enhances the search performances of the algorithm and shows certain advantages in solving function optimization problems [11,12]. The main advantages of the CS algorithm are fewer parameters, simple operation, easy implementation, strong random search path optimization and optimization capabilities, and the ability to converge to the global optimal. The application of cuckoo search to engineering optimization problems has shown its high efficiency, such as the solution of flow shop scheduling problems.

Thereafter, some scholars improve the search ability of the CS algorithm by mixing other algorithms with the cuckoo search algorithm. Wang et al. [13] proposed a cuckoo search hybridized with the particle swarm optimization algorithm, named the CSPSO algorithm. Hu et al. [14] introduced cooperative coevolution framework technology into the cuckoo search algorithm. On the other hand, some scholars have studied and improved the key search components in the cuckoo search algorithm. Wang et al. [15,16] used the dimension by dimension updating search mode and orthogonal crossover operation to enhance the search efficiency of bias random walk. Valian et al. [17] dynamically updated the foreign egg discovery probability and the Lévy flights random walk step factor in the standard algorithm. The rule was to decrease with the increase in the iteration step size, and the authors proposed an improved cuckoo search algorithm (ICS). Wang et al. [18] proposed using uniformly distributed random numbers to dynamically set the Lévy flights random walk step factor. Because uniform distribution is a simple probability distribution, Lin et al. [19] proposed a cuckoo search algorithm with beta distribution search (BCS) by replacing beta distribution with uniform distribution, and the simulation results confirmed the effectiveness of the improved strategy.

As a non-traditional processing method, electrical discharge machining (EDM) has been widely used in the aerospace, molding, automobile and other industries due to its ability to process difficult-to-machine workpieces. EDM is a violent thermal processing process, requiring a very short time through the gap between the workpiece and the tool for thousands of discharges to remove a certain volume of metal materials. The material removal rate (MRR) and surface roughness (SR) are important indicators to measure efficiency and quality. Many input process parameters such as peak current, pulse-on time, pulse-off time and servo voltage will affect the output performance; therefore, proper input parameters must be selected to obtain good results. Due to the many factors that affect the effectiveness of EDM, such as the complexity and randomness of the machining process, even a skilled engineer, it is difficult to achieve the best results using advanced EDM technology. On the contrary, improper selection of parameters may also lead to serious consequences, such as an abnormal discharge state, surface cracks, a thick white layer and poor material removal, thereby reducing productivity.

Patel Gowdru Chandrashekarappa et al. [20] optimized the process parameters while electrical discharge machining HcHcr steel based on the Taguchi hybrid principal component analysis method. Prakash et al. [21] studied the surface modification of Ti-6Al-4V alloy partial sintered Ti-Nb electrode discharge coating. Moreover, our team studied [22,23] the process optimization of the magnetic field-assisted electric discharge machining method,

including not only the electrical parameters, but also the magnetic field strength. In addition, with the development of advanced ceramic materials, EDM also needs to consider its economics and needs to optimize its machining process [24]. Therefore, using appropriate modeling and optimization techniques, such as PSO, GA, and CS, to determine the relationship between machining performance (*MRR* and *SR*) and its key input parameters is an effective method to solve this problem.

In order to improve the user experience of online process optimization, the process optimization algorithm not only needs to converge, but also requires a small number of iterations. Inspired by Valian [17] and Wang [18], this study integrates the two improvement strategies: an improved cuckoo search algorithm and a cuckoo search algorithm with beta distribution. Then, the scale factor shows an exponential decay trend and a cuckoo search algorithm, using improved beta distributing search (IBCS), is proposed. Therefore, this study improves the standard CS algorithm and compares it with several other common variants of the CS algorithm, and has been verified in EDM.

The remaining sections of this study are as follows. Section 2 describes the necessary information about the cuckoo search algorithm. Then, the CS algorithm using an improved beta distribution strategy, proposed in this study, is detailed in Section 3. Subsequently, Section 4 depicts the simulation of the proposed ICBS algorithm, and analyzes the simulation results, which is compared to CS, ICS and BCS. It is important that a case study in the process of EDM is investigated in Section 5 to demonstrate the merits of the proposed ICBS algorithm. Lastly, a concise conclusion for the ICBS algorithm and its application in the process of EDM is drawn from Section 6.

2. Cuckoo Search Algorithm

The CS algorithm is an evolutionary algorithm based on biomimetics, which is optimized for D-dimensional search space $[x_{j,min}, x_{j,max}]$ ($j = 1, 2, \dots, D$) question. The CS algorithm first initializes the population, listed in Equation (1) [9,10],

$$x_{i,j,0} = x_{i,j,min} + r(x_{i,j,max} - x_{i,j,min}), \quad i = 1, 2, \dots, NP \quad (1)$$

where, $r \in [0, 1]$ is the scaling factor and NP is the size of the population. After initialization, the CS algorithm updates the population iteratively. In this process, the CS algorithm first uses Lévy flights to randomly walk and update the next generation of new individuals. Then, the individuals with good fitness will replace previous ones with poor fitness, and otherwise they will remain in place. Furthermore, the CS algorithm uses biased random walk to search for new individuals, and also uses the above greedy strategy to update individuals. After completing this round of iterative search, the algorithm updates the current optimal solution in the entire population. The updated strategy of Lévy flights randomly walking for the current individual is expressed as Equation (2) [9,10],

$$X_{i,G+1} = X_{i,G} + \alpha_0 \frac{\varnothing \times \mu}{|v|^{\frac{1}{\beta}}} (X_{i,G} - X_{best}) \quad (2)$$

where, $X_{i,G}$, $X_{i,G+1}$ and X_{best} represent the i th ($1, 2, \dots, NP$) current individuals in the G th generation, the new i th individuals of the population in the $(G + 1)$ th generation, and the best individuals in the G th generation, respectively; α_0 is the step size scale factor (generally is 0.01); μ and v obey the normal distribution, $\beta = 1.5$, and the \varnothing function corresponds to Equation (3) [9,10],

$$\varnothing = \left(\frac{\Gamma(1 + \beta) \times \sin\left(\frac{\pi \times \beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\frac{\beta-1}{2}}}\right)^{1/\beta} \quad (3)$$

In the biased random walk, each dimension of the individual updates the step size according to a certain probability, as shown in Equation (4) [9,10],

$$x_{i,j,G+1} = \begin{cases} x_{i,j,G} + r(x_{m,j,G} - x_{n,j,G}) & \text{if } rand > p_a \\ x_{i,j,G} & \text{otherwise} \end{cases} \quad (4)$$

where $x_{m,j,G}$ and $x_{n,j,G}$ represent the m th and n th individuals in the G th generation population ($m \neq n \neq i$); r is the scaling factor, $p_a \in [0, 1]$ represents the probability of foreign egg discovery (generally 0.25), and $rand$ is a uniformly distributed random number that obeys $[0, 1]$.

3. CS Algorithm Using Improved Beta Distribution Strategy

3.1. Improvement Strategies

Beta distribution is a kind of continuous probability distribution, which is defined between the intervals $(0, 1)$. It is widely used in mathematical statistics and machine learning. In the beta distribution, the random variable x obeys the probability density function of parameters a and b , as in Equation (5) [19],

$$f(x; a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} x^{a-1} (1-x)^{b-1} \quad (5)$$

According to the research results by Wang et al. [18], a larger step-size scale factor (α_0) is detrimental to the performance of the algorithm when solving most optimization problems. On the other hand, in order to compare the effectiveness of the algorithms in this paper, the values of the parameters a and b in the beta distribution in this study are the same as those in [19]. The condition is that $a \leq b$, $b_{\max} = 15$, $b_{\min} = 5$, and $a_{\min} = 1$. Therefore, the updated strategy for parameters a and b is as follows in Equation (6).

$$\begin{cases} b = b_{\min} + rand \times (b_{\max} - b_{\min}) \\ a = a_{\min} + rand \times (b - a_{\min}) \end{cases} \quad (6)$$

In Equation (6), the values of the parameters a and b are random. Then, Lévy flights move randomly, and the probability density function of the random variable x changes every time. Therefore, the random number generated by the beta distribution also changes, as for the step-size scale factor (α_0), which has a certain disturbing effect on the search evolution process of the CS algorithm. However, from the perspective of the overall evolution trend, the mean value of the step-size scale factor (α_0) does not decrease, because in Equation (6), the mean values of a and b gradually approach a constant with the increase in algebra in the evolution process. Hence, the mean value of the step scale factor (α_0) also gradually approaches a constant.

Therefore, in order to further increase the influence of disturbance in the evolution process, this paper adaptively reduces the mean value of α_0 according to evolution iterations. The strategy is to introduce a scaling factor ($\alpha_{0,abcs}$) to update α_0 . Therefore, the scaling update of the proposed IBCS algorithm α_0 is calculated as in Equation (7),

$$\alpha_{0,abcs} = r_{a,G} \times \alpha_{0,bcs} \quad (7)$$

In Equation (7), $\alpha_{0,abcs}$ is the beta distribution random value as the step-size scale factor, which is determined by the beta distribution, and $r_{a,G}$ is the current evolution iterations scaling factor, calculated by Equation (8),

$$r_{a,G} = r_{\max} \times \exp\left(\frac{1}{G} \ln\left(\frac{r_{\min}}{r_{\max}}\right)\right) \quad (8)$$

where G is the current iteration algebra, while r_{\min} and r_{\max} are 0.1 and 1, respectively. With the increase in evolution iterations, $r_{a,G}$ gradually becomes smaller, which means that

the disturbance of Lévy flights random walk becomes smaller. In the early CS algorithm, the disturbance of Lévy flights random walk is larger, which can avoid premature maturity.

3.2. IBCS Algorithm

Step 1: For the objective function $f(x)$, $X = (x_1, x_2, \dots, x_d)^T$, n variables (population size) as the initial position of the host bird's nest x_i ($i = 1, 2, \dots, n$) are randomly generated by Equation (1). The relevant parameters of the IBCS algorithm are initialized, such as population size $n = 30$, foreign egg discovery probability $p_a = 0.25$, the value of $b_{\max} = 15$, the value of $b_{\min} = 5$, the value of $a_{\min} = 1$, the value of $r_{\max} = 1$, and the value of $r_{\min} = 0.1$ in scaling the boundary of the D-dimensional search space $[x_{j,\min}, x_{j,\max}]$.

Step 2: Generate a random IBCS population, and calculating the objective function value for each host bird's nest, and recording the current iteration number as $N_{\text{iter}} = 1$.

Step 3: Use Equations (5)–(8) to calculate the scale factor (α_0) of Lévy flights random walk, and executing the Lévy flights random walk for all individuals in the population according to Equations (2) and (3), and judging whether to update current individual.

Step 4: Use Equations (4) to perform biased random walk on all individuals in the population, and determining whether to update the current individual.

Step 5: If the end condition is met, output the current optimal position and the corresponding optimal value of the objective function, and the procedure ends; otherwise, go to step 6.

Step 6: After the new population is generated, the objective function value for each host bird's nest is calculated again, updating the current iteration number to $N_{\text{iter}} = N_{\text{iter}} + 1$. Then, go to step 3.

According to the description of the above IBCS algorithm, the flow chart of the algorithm is shown in Figure 1.

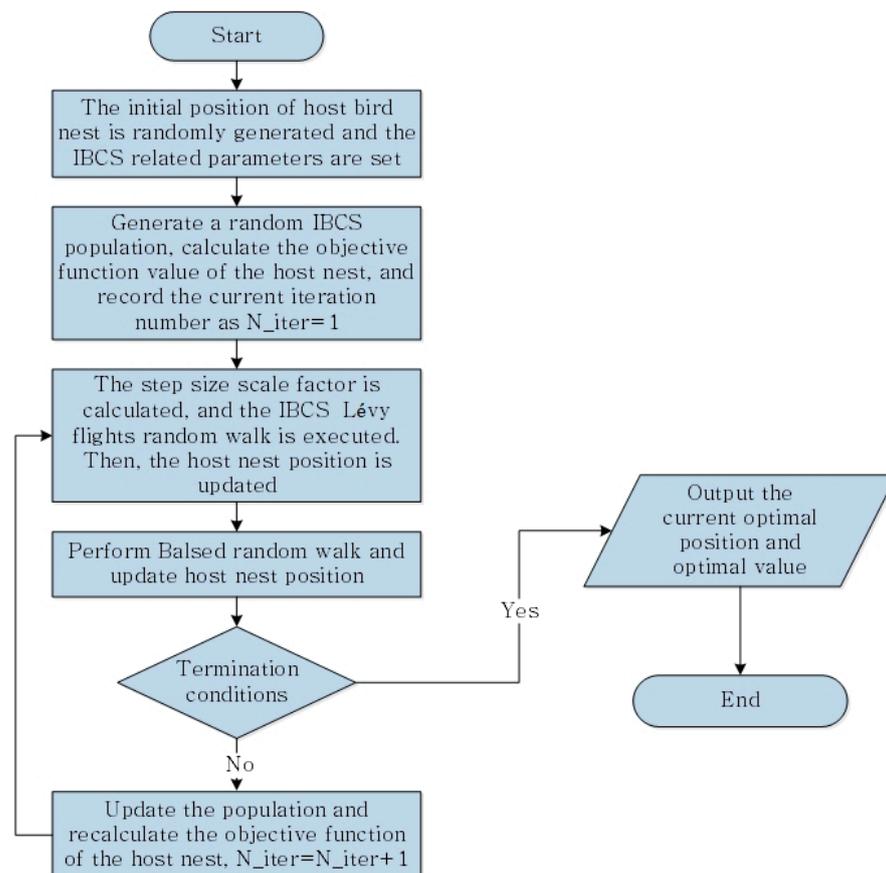


Figure 1. Flow chart of IBCS algorithm.

4. Simulation and Analysis

4.1. Benchmark Functions

In order to estimate the performance of the IBCS algorithm, six typical test functions are used for comparative analysis. The dimension, variable value range and target value of the test function are shown in Table 1.

Table 1. The dimensionality, initial value range and target value of the benchmark function.

No.	Function Name	Dimension	Initial Value Range	Target Value
f_1	Shpere	30	[−5, 5]	0
		50	[−10, 10]	0
f_2	Ackley	30	[−5, 5]	0
		50	[−10, 10]	0
f_3	Rastrigin	30	[−5, 5]	0
		50	[−10, 10]	0
f_4	Rosenbrock	30	[−5, 5]	0
		50	[−10, 10]	0
f_5	Griewank	30	[−5, 5]	0
		50	[−10, 10]	0
f_6	Schwefel	30	[−5, 5]	0
		50	[−10, 10]	0

4.2. Parameter Settings

In order to facilitate the comparison of the effectiveness of the IBCS algorithm, the main parameter settings of the IBCS algorithm are the same as those in BCS and ICS. For the IBCS parameter settings, they are set in Section 3. For the benchmark test function, the evaluation times are $D \times 6000$. In order to evaluate the performance of the algorithm fairly, this paper uses the average error to evaluate, and each benchmark function runs 30 times independently.

4.3. Results and Analysis

It can be seen from the calculation results in Table 2 that for the six benchmark test functions of 30 dimensions, the average rankings of the four algorithms are 3.67, 2.67, 1.5, and 1.17, respectively. Except for the f_6 function, the IBCS algorithm is slightly worse than the BCS algorithm, but better than ICS and standard CS. In other cases, the IBCS algorithm is better than the other three algorithms. From the calculation results in Table 3, it can be seen that for the six benchmark test functions of 50 dimensions, the average rankings of the calculation performance of the four algorithms are 2.83, 2.5, 1.67, and 1.0, respectively. For the high-dimensional f_6 function, the IBCS algorithm has obvious advantages over the BCS algorithm this time, and is better than ICS and standard CS. Therefore, combining the 30-dimensional and 50-dimensional calculation results, the performance of IBCS is better than the other three algorithms, and the hybrid improvement strategy is more effective than the standard algorithm or a single improvement strategy.

The IBCS algorithm is obviously better than the other three algorithms. This is because the IBCS algorithm is improved on the basis of the ICS algorithm and the BCS algorithm. The change in the scale length factor (α_0) for the Lévy flights random walk step has a greater impact on the performance of the cuckoo search algorithm. Figures 2–5 show the historical record values of α_0 during the evolution of the standard CS algorithm, the ICS algorithm, the BCS algorithm, and the IBCS algorithm in 30 dimensions.

Table 2. Test results of four optimization algorithms on 30-dimensional test functions.

No.	Algorithm	Mean	Variance	Sort
f_1	Standard CS [9,10]	2.059×10^{-19}	4.473×10^{-38}	4
	ICS [17]	1.643×10^{-33}	8.102×10^{-65}	3
	BCS [19]	8.217×10^{-34}	9.779×10^{-66}	2
	IBCS	4.109×10^{-34}	5.064×10^{-66}	1
f_2	Standard CS [9,10]	2.603×10^{-09}	3.669×10^{-17}	4
	ICS [17]	3.553×10^{-15}	0	3
	BCS [19]	5.092×10^{-15}	3.206×10^{-30}	2
	IBCS	3.789×10^{-15}	8.124×10^{-31}	1
f_3	Standard CS [9,10]	4.069×10^1	3.625×10^1	3
	ICS [17]	4.138×10^1	4.775×10^1	3
	BCS [19]	3.448×10^1	1.564×10^2	2
	IBCS	3.184×10^1	5.774×10^1	1
f_4	Standard CS [9,10]	1.633×10^1	2.391×10^0	3
	ICS [17]	1.598×10^1	2.253×10^1	3
	BCS [19]	1.056×10^1	5.114×10^{-1}	1
	IBCS	1.108×10^1	5.927×10^{-1}	1
f_5	Standard CS [9,10]	5.070×10^{-16}	7.711×10^{-30}	4
	ICS [17]	0	0	1
	BCS [19]	0	0	1
	IBCS	0	0	1
f_6	Standard CS [9,10]	3.655×10^{-17}	3.172×10^{-33}	4
	ICS [17]	3.159×10^{-32}	1.221×10^{-63}	3
	BCS [19]	2.327×10^{-33}	4.768×10^{-66}	1
	IBCS	1.679×10^{-32}	1.501×10^{-64}	2

Table 3. Test results of four optimization algorithms on 50-dimensional test functions.

No.	Algorithm	Mean	Variance	Sort
f_1	Standard CS [9,10]	1.953×10^{-20}	2.935×10^{-40}	4
	ICS [17]	1.252×10^{-32}	2.005×10^{-65}	3
	BCS [19]	8.432×10^{-34}	3.898×10^{-68}	2
	IBCS	5.699×10^{-36}	3.701×10^{-72}	1
f_2	Standard CS [9,10]	4.757×10^{-10}	2.0625×10^{-19}	4
	ICS [17]	6.394×10^{-15}	2.244×10^{-30}	1
	BCS [19]	6.751×10^{-15}	1.262×10^{-30}	3
	IBCS	6.394×10^{-15}	2.244×10^{-30}	1
f_3	Standard CS [9,10]	6.946×10^1	8.525×10^1	1
	ICS [17]	1.023×10^2	3.295×10^2	4
	BCS [19]	6.338×10^1	4.604×10^2	1
	IBCS	6.192×10^1	2.147×10^2	1
f_4	Standard CS [9,10]	3.535×10^1	3.095×10^0	3
	ICS [17]	3.853×10^1	3.104×10^0	3
	BCS [19]	2.817×10^1	3.292×10^{-1}	1
	IBCS	2.731×10^1	8.941×10^{-1}	1
f_5	Standard CS [9,10]	0	0	1
	ICS [17]	0	0	1
	BCS [19]	0	0	1
	IBCS	0	0	1
f_6	Standard CS [9,10]	2.131×10^{-17}	2.567×10^{-34}	4
	ICS [17]	2.558×10^{-30}	4.766×10^{-60}	3
	BCS [19]	6.789×10^{-31}	5.379×10^{-61}	2
	IBCS	6.648×10^{-33}	5.753×10^{-65}	1

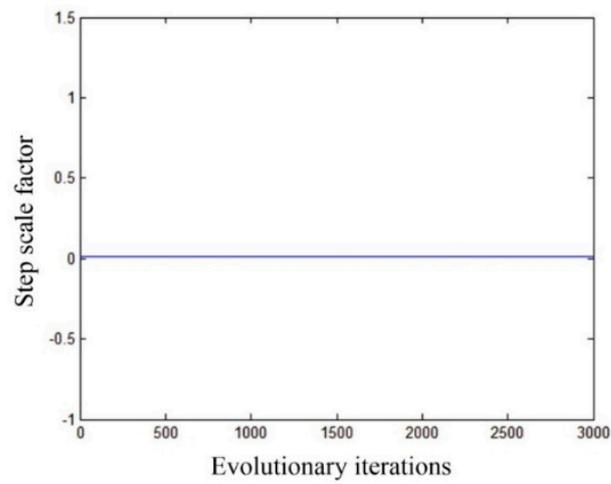


Figure 2. α_0 (30 dimensions) in the evolution of the standard CS algorithm.

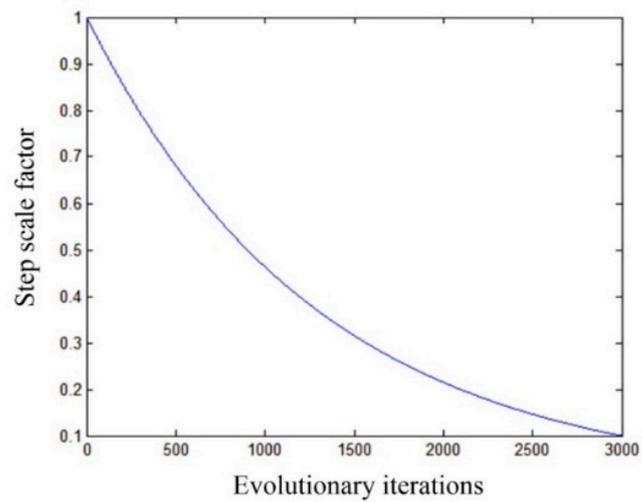


Figure 3. α_0 (30 dimensions) in the evolution of ICS algorithm.

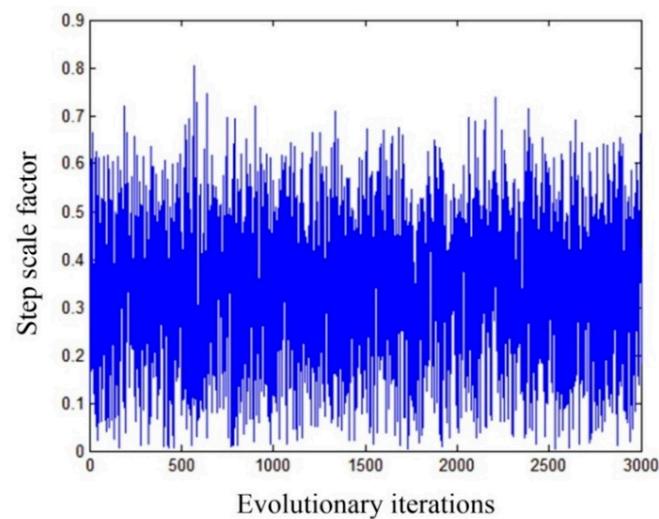


Figure 4. α_0 (30 dimensions) in the evolution of the BCS algorithm.

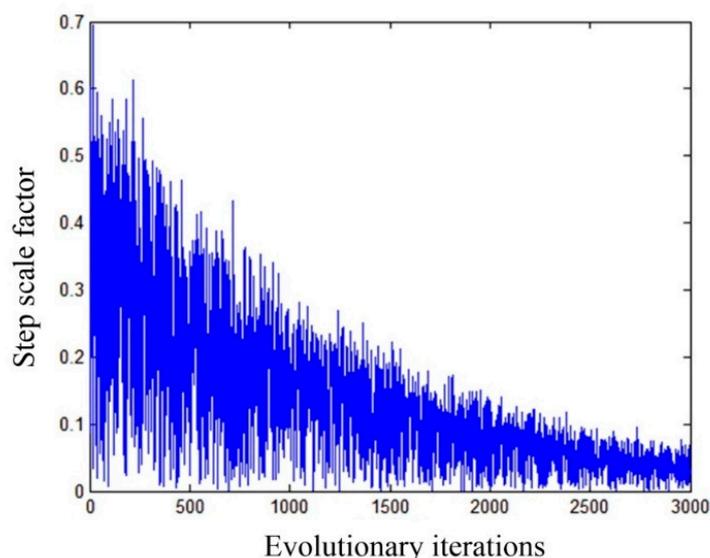


Figure 5. α_0 (30 dimensions) in the evolution of ABCS algorithm.

For the standard CS algorithm [9,10], the α_0 curve in Figure 2 remains constant, and the step scale factor in Lévy flights random walk remains unchanged. For the ICS algorithm [17], the α_0 curve exhibits exponential decay. In the early stage of evolution, a relatively large α_0 is needed to facilitate global search. In the later stage, a smaller α_0 is beneficial to local optimization. Therefore, the ICS algorithm [17] is better than the standard CS algorithm [9,10] for these six benchmark functions both in 30-dimensional and 50-dimensional. However, during the evolution of the BCS algorithm [19], the α_0 curve, depicted in Figure 4, does not show exponential decay, its average value is between 0.45 and 0.55, and α_0 varies greatly between adjacent iterations and obeys the beta distribution in the process of evolution. This helps to prevent the cuckoo search algorithm from falling into a local optimal solution, and α_0 varies between 0 and 0.9. This is conducive to the diversity of step scale factor in the Lévy flights random walk. Therefore, the BCS algorithm [19] is better than the standard CS algorithm [9,10], which can also be verified from the data in Tables 2 and 3.

For the IBCS algorithm proposed in this study (shown in Figure 5), the step scale factor in Lévy flights random walk presents a beta random distribution in the evolution process. At the same time, it makes the evolution process show an exponential decay trend on the whole. Therefore, the trend of the step scale factor curve in the IBCS combines the effects of ICS [17] and BCS [19]. It can be seen from the above analysis that such a step scale factor should have a better optimization effect. From the data in Tables 2 and 3, we can see that the IBCS algorithm, proposed in this study, is not only better than the standard CS algorithm [9,10], but also better than the ICS algorithm [17] and the BCS algorithm [19] of a single optimization problem.

Furthermore, this study presents the convergence curves of test function f_1 , test function f_3 and test function f_5 under the 30-dimensional function and 50-dimensional function, respectively, as shown in Figures 6–8. The other three test functions are limited by the length of the paper and are not given in this study. It can also be directly observed from the graph that the convergence speed of the IBCS algorithm, proposed in this paper, is significantly better than that of the standard CS algorithm [9,10]. Compared with the ICS algorithm and the BCS algorithm [17,19], it also has certain advantages, and the effect is more obvious for high dimensions (50 dimensions).

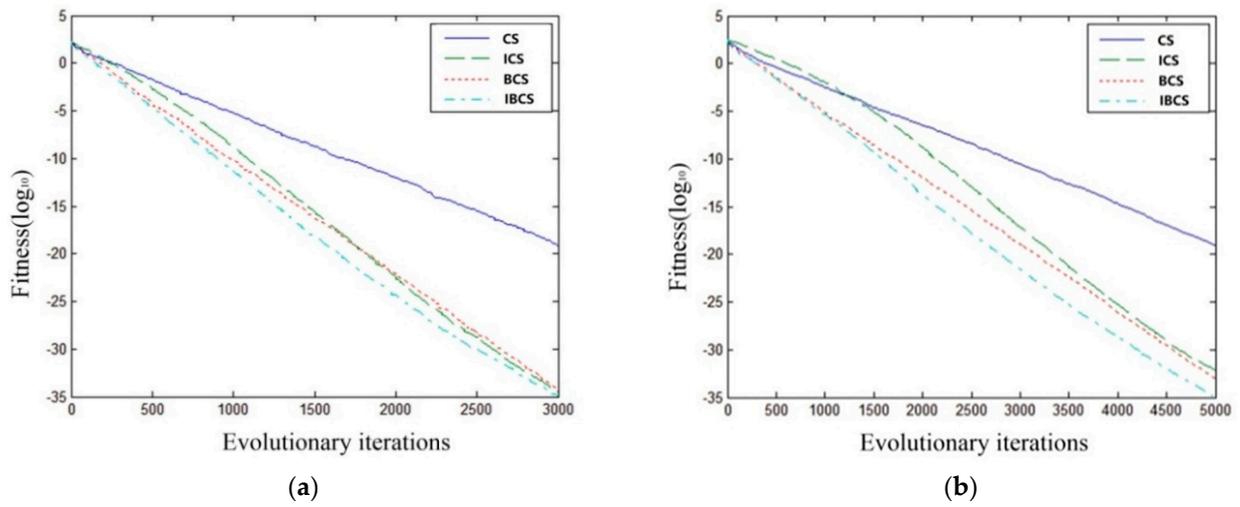


Figure 6. Comparison of the convergence results of the Shpere function; (a) 30–dimensional; (b) 50–dimensional.

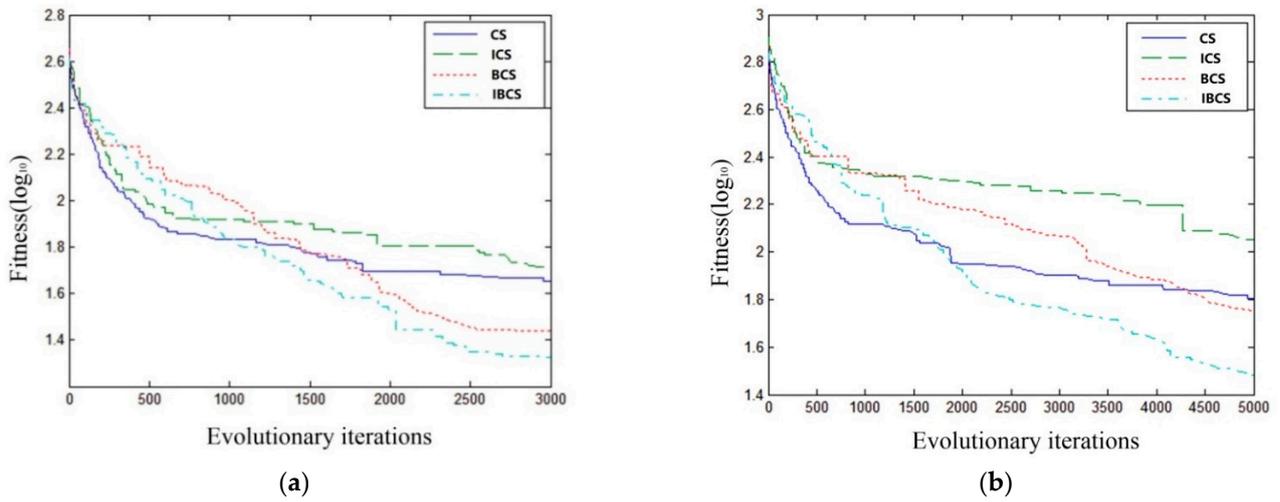


Figure 7. Comparison of convergence results of the Rastrigin function; (a) 30-dimensional; (b) 50-dimensional.

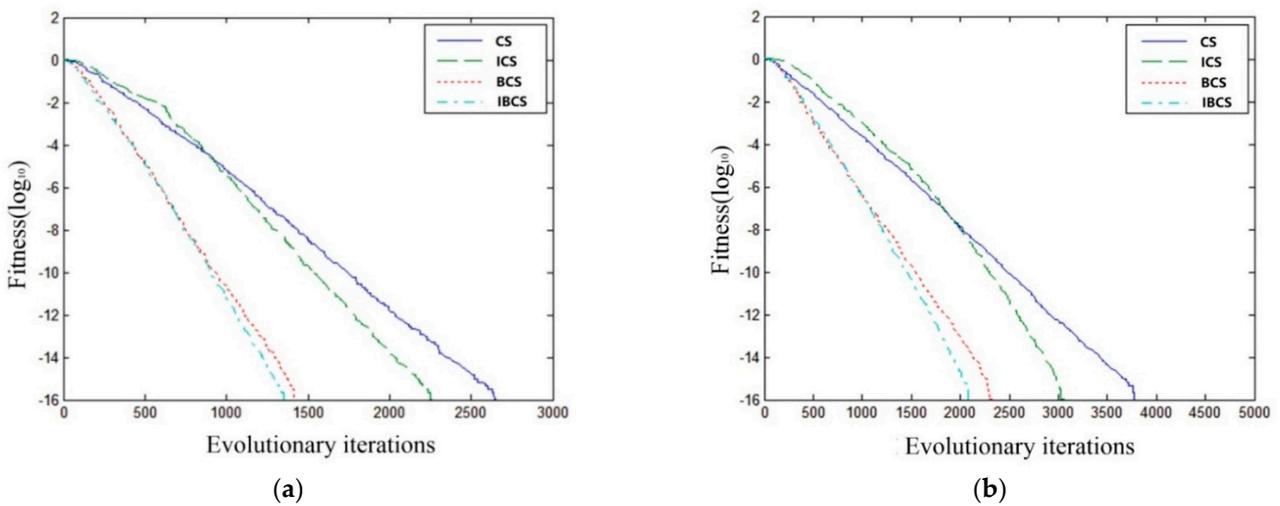


Figure 8. Comparison of the convergence results of the Griewank function; (a) 30–dimensional; (b) 50–dimensional.

5. Case Study

As a non-traditional processing method, the process of electric discharge machining (EDM) has been widely used in the aerospace, molding, automotive and other industries. The EDM aims to remove a certain volume of metal material in a very short time by relying on the heat generated by electric discharge [25,26]. Two evaluation indexes of important indexes in EDM are the material removal rate (*MRR*) and surface roughness (*Ra*) [26–28]. Due to many factors affecting the machining effects, such as peak current, pulse turning time, pulse shutdown time, and servo voltages affect the performance of the output. Even a skilled engineer, due to the complexity and randomness of the processing process, will find it difficult to achieve the best results through advanced processing technology. In addition, improper parameters may also result in serious consequences, such as abnormal discharge state, surface cracking, etc., thereby reducing processing quality. Therefore, the relationship between the processing performance (*MRR* and *Ra*) and its main input parameters is determined, and the optimal process parameter combination is obtained, which is an effective method for solving this problem.

As a stable compound of C and Si, the lattice structure of SiC is composed of two closely arranged sublattices. Each Si (or C) atom is bound to the surrounding C (Si) atom by a directional strong tetrahedral sp³ bond. Although the tetrahedral bond of SiC is very strong, the energy of stacking fault formation is very low, which determines the polytype phenomenon of SiC. The most common multi-vectors are cubic-mounted 3C/SiC and hexagonal macro 4H, 6H/SiC. Fouly et al. made Al/SiC composites in high-frequency sintering methods and studied their mechanics and tribological properties [29]. Chen et al. adopted a vacuum pressure impregnation method to prepare a high-volume fraction of high thermal conductivity SiC/Al composites [30].

Thereafter, Ming et al. [28] proposed soft computing models, of an intelligent optimization system, to optimize the process of electro-discharge machining of SiC/Al composites. In their study [28], a series of experiments on a block of SiC/Al composites (8 × 7 × 0.2 mm) had been conducted on Topend ED-50 EDM machine (depicted in Figure 9), which was manufactured by Kingone Co. Ltd. The SiC/Al composite machined was a high-volume fraction material, and its particle size, containing 45 % SiC particles, was 5 μm. Its yield pressure and elastic modulus were 329.9 MPa and 150.6 GPa, respectively. In the machining process, the type of dielectric was Mold DY-1, which was a specialized dielectric with high stability for the EDM process [28]. Additionally, the material of the tool electrode was copper, and its shape was 8 × 7 × 2 mm. Additionally, other detailed information was depicted in the literature [28].

Based on the machining data and regression method, the models of MEE and surface roughness (*Ra*) are shown in Equations 9 and 10 (*I_p*: discharge current, *T_{on}*: pulse-on time, *T_{off}*: pulse-off time, and *S_v*: servo voltage), respectively [28].

$$\begin{aligned} MRR = & -4.76 + 0.2201I_p + 0.620T_{on} + 0.352T_{off} + 0.0720S_v \\ & + 0.01976I_p^2 - 0.01240I_p \times T_{off} + 0.0182T_{on} \times T_{off} \\ & - 0.00976T_{on} \times S_v - 0.00586T_{off} \times S_v \end{aligned} \quad (9)$$

$$\begin{aligned} Ra = & -59.4 + 3.001I_p - 0.1053T_{on} - 1.441T_{off} + 1.73S_v \\ & + 0.1026T_{off}^2 - 0.1145S_v^2 + 0.1032I_p \times T_{off} - 0.0396I_p \times S_v \end{aligned} \quad (10)$$

In order to verify the effectiveness of the improved cuckoo algorithm in this paper, we perform a single-objective optimization based on Equations (9) and (10). Through the optimization algorithm (CS, ICS, BCS, and IBCS), the discharge parameter combination at the maximum *MRR* can be obtained. Similarly, the discharge parameter combination at the minimum value of *Ra* can be calculated. The optimization of the CS, ICS, BCS, and IBCS algorithms was performed 12 times, and the number of iterations of *MRR* and *Ra* under the optimal value (less than 0.5% error) was obtained by statistics, respectively. Table 4 lists the test results of four optimization algorithms on the *MRR* single-objective optimization. As depicted in Table 4, the performance of IBCS, proposed in this study, is better than that

of standard CS, ICS and BCS. Similarly, the same conclusion can be obtained from Table 5, and the success ratio of 10 iterations in the IBCS algorithm is the best. For the success ratio of 50 iterations, all four of the optimization algorithms could achieve the desired results for both the *MRR* and *Ra* single-objective optimization problem.



Figure 9. Experimental setup and workpiece; (a) Topend ED–50 EDM machine; (b) workbench; (c) workpiece. Reprinted with permission from ref. [28]. Copyright 2021 Copyright Springer Nature.

Table 4. Test results of four optimization algorithms on the *MRR* single-objective optimization.

Algorithm	Success Ratio (%)			Sort
	10 Iterations	50 Iterations	Average	
Standard CS [9,10]	58.33	100	79.17	4
ICS [17]	66.66	100	83.33	3
BCS [19]	83.33	100	91.67	2
IBCS	91.67	100	95.83	1

Table 5. Test results of four optimization algorithms on the *Ra* single-objective optimization.

Algorithm	Success Ratio (%)			Sort
	10 Iterations	50 Iterations	Average	
Standard CS [9,10]	50	100	75	4
ICS [17]	75	100	87.5	2
BCS [19]	66.66	100	83.33	3
IBCS	83.33	100	91.67	1

Figure 10 demonstrates the comparison of the convergence results of the single objective optimization for *MRR* and *Ra* in the process of EDM, respectively. As depicted in Figure 10a, the IBCS algorithm obtains the desired convergence results of *MRR* by

6 iterations. However, the standard CS algorithm acquires the desired convergence results of MRR by 13 iterations. Additionally, the lowest iteration numbers of the ICS and BCS algorithms, satisfying the desired convergence, are both 9. Under the single-objective optimization convergence of MRR , the iteration number of CS algorithm is twice that of IBCS algorithm. After the first iteration, moreover, the optimization results of IBCS algorithm are better than the other three algorithms (CS, ICS or BCS). It also can be concluded from Figure 10b that the IBCS algorithm obtains the desired convergence results of Ra by 8 iterations, and the CS, ICS and BCS algorithms acquire the desired convergence results of Ra by 13, 15, and 17 iterations, respectively. The results demonstrate that the iteration number of BCS algorithm is twice that of IBCS algorithm under the single-objective optimization convergence of Ra . Therefore, the IBCS algorithm is the best for the single-objective optimization in the process of EDM.

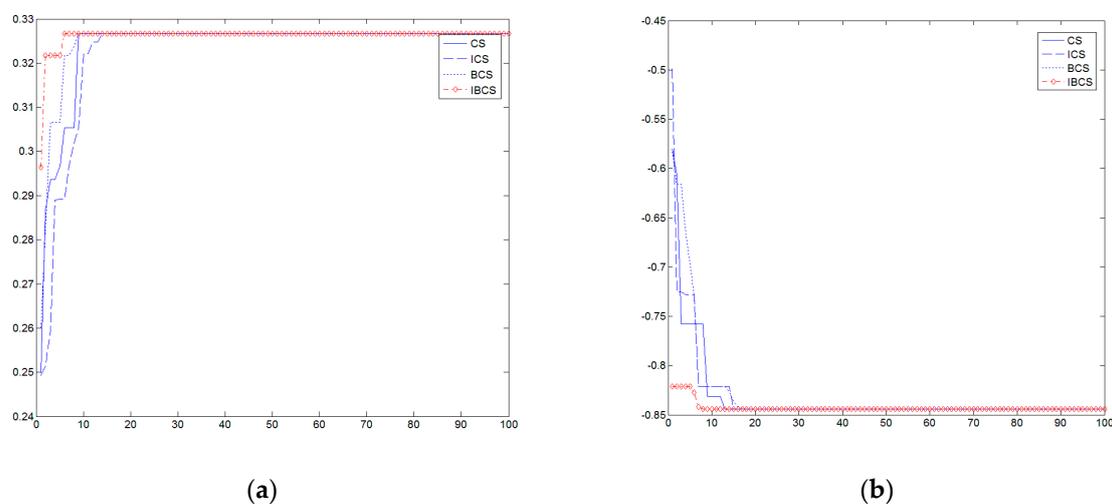


Figure 10. Comparison of the convergence results of the single objective optimization in the process of EDM; (a) for MRR ; (b) for Ra .

6. Conclusions

The CS algorithm is a new bionic evolutionary algorithm that has emerged in recent years. This paper studies a cuckoo search algorithm using improved beta distribution and its application in the process of EDM. The following conclusions can be obtained.

(1) This paper proposed a new cuckoo search algorithm that uses an improved beta distribution strategy, which causes the Lévy flights random walk step scale factor to present a beta random number distribution in the evolutionary process. At the same time, the step-size scale factor of the evolution process had an exponential decay trend.

(2) The simulation test results show that the proposed strategy in the IBCS algorithm is feasible and can effectively improve the convergence speed and solution performance, which is compared to the standard CS algorithm. For the six benchmark test functions of 30 dimensions, the average rankings of the CS, ICS, BCS, and IBCS algorithms are 3.67, 2.67, 1.5, and 1.17, respectively. For the six benchmark test functions of 50 dimensions, moreover, the average rankings of the CS, ICS, BCS, and IBCS algorithms are 2.83, 2.5, 1.67, and 1.0, respectively.

(3) The IBCS algorithm combines the advantages of the ICS algorithm and the BCS algorithm, and its performance is better than these two algorithms. At the same time, as the scale of the solution becomes larger, the performance of the IBCS algorithm remains stable. Compared with the standard CS algorithm, ICS algorithm and BCS algorithm, the IBCS algorithm can better reflect its superiority.

(4) As confirmed by our case study, the performance of ABCS algorithm was better than that of the standard CS, ICS or BCS algorithms in the process of EDM. For example, under the single-objective optimization convergence of MRR , the iteration number

(13 iterations) of the CS algorithm was twice that (6 iterations) of the IBCS algorithm. Similar, the iteration number (17 iterations) of the BCS algorithm was twice that (8 iterations) of the IBCS algorithm under the single-objective optimization convergence of Ra.

Nomenclature and Abbreviations

ABC	artificial bee colony
BCS	beta distribution search
CS	cuckoo search
EDM	electrical discharge machining
GSO	glowworm swarm optimization
IBCS	improved beta distributing search
ICS	improved cuckoo search
PSO	particle swarm optimization
WC	wolf colony
I_p	the discharge current
T_{on}	the pulse-on time
T_{off}	the pulse-off time
S_v	the servo voltage
$X_{i,G}$	the i th ($1, 2, \dots, NP$) current individuals in the G th generation
$X_{i,G+1}$	the new i th individuals of the population in the ($G + 1$)th generation
X_{best}	the best individuals in the G th generation
α_0	the step size scale factor
p_a	the probability of foreign egg discovery
r	the scaling factor

Author Contributions: Conceptualization, D.S. and W.M.; methodology, Y.Z.; software, W.M. and Y.Z.; validation, X.L., and Z.X.; formal analysis, X.R.; investigation, D.S.; resources, W.M.; writing—original draft preparation, D.S.; writing—review and editing, X.R.; visualization, W.M.; supervision, X.R.; funding acquisition, X.R., W.M. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the Local Innovative and Research Teams Project of Guangdong Pearl River Talents Program (2017BT01G167). Additionally, it is supported by the Guangdong Youth Talent Innovation Project (2019GKQNCX092), and by the Guangdong Basic and Applied Basic Research Foundation, China (2019A1515011349).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Kennedy, J.; Eberhart, R.C. A discrete binary version of the particle swarm algorithm. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Orlando, FL, USA, 12–15 October 1997; Volume 5, pp. 4104–4108.
- Zhang, G.; Liu, M.; Li, J.; Ming, W.; Shao, X.; Huang, Y. Multi-objective optimization for surface grinding process using a hybrid particle swarm optimization algorithm. *Int. J. Adv. Manuf. Technol.* **2014**, *71*, 1861–1872. [[CrossRef](#)]
- Karaboga, D.; Basturk, B.A. powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **2007**, *39*, 459–471. [[CrossRef](#)]
- Kuo, M.T. Application of the Artificial Bee Colony Algorithm to Scheduling Strategies for Energy Storage Systems of a Microgrid with Self-healing Functions. *IEEE Trans. Ind. Appl.* **2021**, *57*, 2156–2167. [[CrossRef](#)]
- Krishnanand, K.N.; Ghose, D. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intell.* **2009**, *3*, 87–124. [[CrossRef](#)]
- Xia, P.; Ni, Z.; Zhu, X.; Zhang, J.; Jin, Y. A Novel Prediction Method Based on Improved Binary Glowworm Swarm Optimization and Multi-Fractal Dimension for P2P Lending Investment Risk. *IEEE Access* **2020**, *8*, 23232–23245. [[CrossRef](#)]
- Qiang, Z.; Zhou, Y.; Xin, C.A. Wolf Colony Search Algorithm Based on the Complex Method for Uninhabited Combat Air Vehicle Path Planning. *Int. J. Hybrid Inf. Technol.* **2014**, *7*, 183–200.

8. Ming, W.; Hou, J.; Zhang, Z.; Huang, H.; Xu, Z.; Zhang, G.; Huang, Y. Integrated ANN-LWPA for cutting parameter optimization in WEDM. *Int. J. Adv. Manuf. Technol.* **2016**, *84*, 1277–1294. [[CrossRef](#)]
9. Yang, X.S.; Deb, S. Cuckoo search via Levy flights. In Proceedings of the Nature & Biologically Inspired Computing, (NaBIC 2009), Coimbatore, India, 9–11 December 2009; IEEE Publications: Piscataway, NJ, USA, 2009; pp. 210–214.
10. Yang, X.S.; Deb, S. Engineering optimisation by cuckoo search. *Int. J. Math. Model. Numer. Optim.* **2010**, *1*, 330–343. [[CrossRef](#)]
11. Walton, S.; Hassan, O.; Morgan, K.; Brown, M.R. Modified cuckoo search: A new gradient free optimisation algorithm. *Chaos Solitons Fractals* **2011**, *9*, 710–718. [[CrossRef](#)]
12. Yang, X.S.; Deb, S. Cuckoo Search: Recent Advances and Applications. *Neural Comput. Appl.* **2014**, *24*, 169–174. [[CrossRef](#)]
13. Wang, F.; Luo, L.; He, X.; Yan, W. Hybrid optimization algorithm of PSO and cuckoo search. In Proceedings of the International Conference on Artificial Intelligence, Management Science and Electronic Commerce, Deng Feng, China, 8–10 August 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1172–1175.
14. Hu, X.; Yin, Y. Cooperative Co-Evolutionary Cuckoo Search Algorithm for Continuous Function Optimization Problems. *Pattern Recognit. Artif. Intell.* **2013**, *26*, 1041–1049.
15. Wang, L.; Yin, Y.; Zhong, Y. Cuckoo Search Algorithm with Dimension by Dimension Improvement. *Ruan Jian Xue Bao/J. Softw.* **2013**, *24*, 2687–2698. (In Chinese) [[CrossRef](#)]
16. Wang, L.; Zhong, Y.; Yin, Y. Orthogonal crossover cuckoo search algorithm with external archive. *J. Comput. Res. Dev.* **2015**, *52*, 2496–2507. (In Chinese)
17. Valian, E.; Mohanna, S.; Tavakoli, S. Improved Cuckoo search algorithm for global optimization. *Int. J. Commun. Inf. Technol.* **2011**, *1*, 31–44.
18. Wang, L.; Yin, Y.; Zhong, Y. Cuckoo search with varied scaling factor. *Front. Comput. Sci.* **2015**, *9*, 623–635. [[CrossRef](#)]
19. Lin, Y.; Liang, Z.; Hu, H. Cuckoo search algorithm with beta distribution. *J. Nanjing Univ. (Nat. Sci.)* **2016**, *52*, 638–646. (In Chinese)
20. Prakash, C.; Singh, S.; Pruncu, C. Surface Modification of Ti-6Al-4V Alloy by Electrical Discharge Coating Process Using Partially Sintered Ti-Nb Electrode. *Materials* **2019**, *12*, 1006. [[CrossRef](#)]
21. Handrashekarappa, M.P.G.; Kumar, S.; Jagadish; Pimenov, D.Y. Experimental Analysis and Optimization of EDM Parameters on HcHcr Steel in Context with Different Electrodes and Dielectric Fluids Using Hybrid Taguchi-Based PCA-Utility and CRITIC-Utility Approaches. *Metals* **2021**, *11*, 419. [[CrossRef](#)]
22. Ming, W.; Shen, F.; Zhang, Z. A comparative investigation on magnetic field-assisted EDM of magnetic and non-magnetic materials. *Int. J. Adv. Manuf. Technol.* **2020**, *109*, 1103–1116. [[CrossRef](#)]
23. Ming, W.; Jia, H.; Zhang, H. A comprehensive review of electric discharge machining of advanced ceramics. *Ceram. Int.* **2020**, *46*, 21813–21838. [[CrossRef](#)]
24. Zhang, Z.; Zhang, Y.; Ming, W. A review on magnetic field assisted electrical discharge machining. *J. Manuf. Process.* **2021**, *64*, 694–722. [[CrossRef](#)]
25. Kunieda, M.; Lauwers, B.; Rajurkar, K.P.; Rajurkar, B.M. Advancing EDM through Fundamental Insight into the Process. *CIRP Ann.* **2005**, *54*, 64–87. [[CrossRef](#)]
26. Wang, Y.; Liu, Z.; Shi, J.; Dong, Y.; Yang, S.; Zhang, X.; Lin, B. Analysis of material removal and surface generation mechanism of ultrasonic vibration-assisted EDM. *Int. J. Adv. Manuf. Technol.* **2020**, *110*, 177–189. [[CrossRef](#)]
27. Ming, W.; Zhang, G.; Li, H.; Guo, J.; Zhang, Z.; Huang, Y.; Chen, Z. A hybrid process model for EDM based on finite-element method and Gaussian process regression. *Int. J. Adv. Manuf. Technol.* **2014**, *74*, 1197–1211. [[CrossRef](#)]
28. Ming, W.; Ma, J.; Zhang, Z.; Huang, H.; Shen, D.; Zhang, G.; Huang, Y. Soft computing models and intelligent optimization system in electro-discharge machining of SiC/Al composites. *Int. J. Adv. Manuf. Technol.* **2016**, *87*, 201–217. [[CrossRef](#)]
29. Fouly, A.; Almotairy, S.M.; Aijaz, M.O.; Alharbi, H.F.; Abdo, H.S. Balanced Mechanical and Tribological Performance of High-Frequency-Sintered Al-SiC Achieved via Innovative Milling Route—Experimental and Theoretical Study. *Crystals* **2021**, *11*, 700. [[CrossRef](#)]
30. Chen, M.; Bai, Y.; Zhang, Z.; Zhao, H. The Preparation of High-Volume Fraction SiC/Al Composites with High Thermal Conductivity by Vacuum Pressure Infiltration. *Crystals* **2021**, *11*, 515. [[CrossRef](#)]