BIO-INSPIRED METAHEURISTICS SEARCH ALGORITHMS FOR OPTIMIZING MANUFACTURING PROCESS

[1]YANG LIE, [2]SHETA ALAA

[1,2]Department of Computing Sciences, Texas A&M University-Corpus Christi, Corpus Christi, TX 78412, USA
E-mail: [2]alaa.sheta@tamucc.edu

Abstract - Traditional search methods used to solve manufacture engineering optimization problems fail to find the global optimum when dealing with a complex objective function and a large number of decision variables. Bio-inspired metaheuristics search algorithms shows significant performance in handling multi-model function optimization with many constraints. They can efficiently explore the search space and produce results that are more global in nature. Compared with other metaheuristic algorithm, the CSA has the advantage of less parameter setting and easy implementation. Two cutting tools Cases study of engineering problems are used to demonstrate the CSA. The results of case studies show the CSA perform a promising result in engineering optimization problem. A performance comparison with genetic algorithms, particle swarm optimization and cuckoo search is also presented in the paper and shows the ascendancy with better convergence rate.

Keywords - Crow Search Algorithm, Engineering Optimization Problem, Parameter Optimization, Cutting Tools

I. INTRODUCTION

Bio-inspired metaheuristics algorithms were used successfully to handle many problems in manufacture. For example, bat algorithm (BA), firefly algorithm (FA), and artificial bee colony (ABC) algorithm were used to repeatedly update the synaptic weights of an Artificial Neural Network (ANN) to model the dynamics of the Tennessee Eastman (TE) chemical reactor [1]. These algorithms were used to enhance the ability of ANN to model manufacturing systems dynamic and avoid deceiving into a local optimum. Metaheuristic methods are well-developed and applied to solve optimization problems in recent decades. Researchers and scholars proposed these new algorithms by observing and studying intelligent behaviors from natural phenomena. Those nature-inspired algorithms are named metaheuristic algorithm [2]. The randomization factor applied to these algorithms help to explore the search space and tend to find the global optimum. The algorithms will keep a balance between the global search and local search based on the setting parameter [3]. With a population of seekers used in the algorithms, it reduces the time to explore the search space. Metaheuristic algorithms perform a promising result in finding the local optimum as well as the global optimum during the optimization problem [4].

Some well-known bio-inspired algorithms are genetic algorithms (GA) based on the natural selection [5], bat algorithm (BA) based on echolocation behavior of microbats [6], particle swarm optimization (PSO) [7], cuckoo search (CS) [8], [9] and harmony search (HS) [10], etc. A comparison between a number of Metaheuristic algorithms including Genetic Algorithms (GA), Particle Swarm Optimization (PSO) and the Accelerated PSO, Cuckoo Search, Krill Herd, and Firefly algorithm was provided in [11] to solve the parameter estimation problem for a Winding process in the industry. Researchers are still exploring the development new algorithm for users to easily understand and apply in problem-solving as well as improving the performance during computing. Similar to other metaheuristic algorithms, CSA is based on the intelligent behavior of crows [12]. CSA is a nature-inspired algorithm proposed by Askarzadeh in 2016. With top bird IQ scale, crows are considered one of the most intelligent animals in the world according to their ability to memorize faces, use tools, communicate in sophisticated ways and hide and retrieve food [13], [14]. The intelligent behaviors of a flock of crows can be implemented in the optimization processing. The proposed CSA algorithm by Askarzadeh [12] in 2016 showed promising results in solving constrained and engineering design optimization problems.

In engineering industries, it is common to deal with engineering optimization problems in order to find out a cost-efficient way in manufacturing. Sometimes the problems involve a variety of parameters and constraints as well as complex objective functions. Numerous of decision-making needs to be considered in order to satisfy the required specifications of products or to save money with effective methods [12]. To find an optimal solution for problems, all available feasible solutions should be taken into considerations. However, some problems cannot be solved directly with formulas. For instance, in industrial manufacturing, the cutting tool is one of the essential tools which is widely used to manufacture workpieces. During the metal-cutting process, the heat generated from the contact point between cutting tools and workpieces results in temperature increases. This temperature affects not only the tools of life but also the quality of the products. Although

experienced workers know the parameters setting of machine and materials affect the temperature, there is no specific or clear theoretical formula can directly calculate the temperature like area or power. In [15], authors show that the selection of optimal machining conditions is a key to achieve higher efficiency and lower cost. In this Case, empirical models and an optimization technique are introduced to solve this problem and achieve the goal. This optimization problem is named parameter optimization, and it aims to minimize the difference between real data and calculated value from the models. Optimization techniques are used to estimate the optimal coefficients for developing the best fit models, which subsequently implemented in actual manufacturing processes [16].

The parameter optimization problem for modeling manufacturing processed is solved by traditional methods, such as statistical least square methods (LSE) [17]. Traditional methods for parameter optimization show some promising performance while solving this problem. However, LSE has shortcomings when dealing with some complicated problems with a variety of parameters taken into consideration. However, they have many drawbacks 1) they tend to find local optimum rather than the global optimum, 2) they have a hard time to solve problems with numerous of decision variable and complex objective functions, such as nonlinear and multimodal [12], 3) they even fail to solve the problem when constraints are involved. In order to find the global optimum and improve performance while solving the engineering optimization problem, an efficient approach is essential and imperative.

In this paper, the proposed CSA methods will be applied to the parameter optimization problems for cutting tools. The remainder of this paper is organized as follows. Section II will describe CSA algorithms in details as well as the CSA implementation for optimization. Engineering optimization problems, cutting tool temperature modeling, will be presented in Section III as simulation examples applied with CSA algorithm and the experimental result will be discussed in Section IV The conclusion will be drawn in the Section V as well as the future work will be mentioned.

II. CROW SEARCH ALGORITHMS (CSA)

The inspiration of CSA is according to the crows hiding food behavior as well as the search mechanism for others' food hiding place. Crows will hide their excess food in some places they memorized and retrieve when they need. As greedy birds, crows will try to find others' hiding place to look for better food by following others and do thievery. Meanwhile, crows also prevent their food from being stolen. The anti-tailing ability enables the crows to notice if another follows it by a probability. If the crows noticed it is followed, they would fool the following crow to random places in the environment rather than taking them to their hiding places. From the optimization point of view, a flock of crows can be treated as a population of seekers in the search space. The number of crows is the population size, and the environment is the search space. Each position in the environment, represents a feasible solution, to the problem at hand; the position of the hiding place is the best solution for the corresponding crows observing so far. The objective (fitness) function is the standard to evaluate the quality of food. In this Case, crows are searching for the best food in the environment, where the best food is precisely the global optimum solution for the optimization problem. Based on this simulation on the intelligent of crows, the CSA algorithm is aimed to find the optimal solution for the optimization problem.

The key idea of simulating the intelligent behavior of crows is the food search mechanism and the way that crow prevent their food from being stolen from the hiding places by fool another crow to a random position. The experience of thievery also teachers themselves to take prevention to store their food in safe places to protect their caches from being pilfered [18]. Leading the followed crows to random positions provides the randomization in the CSA algorithm. Based on the intelligent behaviors of crow mentioned above, the principles of CSA can be concluded as follow:

- Crows live as a form of the flock.
- Crows memorize their food, hiding places and retrieve when they need.
- Crows steal others' food by following other crows.
- When Crows noticed it is followed, fool the followed crows to a random position.

Simulating the principles into mathematical model for optimization problem. Assuming the d-dimensional environment is the search space and each position in the environment represents the feasible solution for the problem. The number N of crows in the flock is the population size and d is the number of decision variable. $x^{i,t}$ represents the current position of a crow i at iteration t in search space, where i = 1, 2, ..., N; t = 1, 2, ..., t_{max} and t_{max} is the maximum number of iterations. The position of crow i at iteration t is defined as:

$x^{i,t} = [x_1^{i,t}, x_2^{i,t}, \dots, x_d^{i,t}]$,

$x^{i,t}$is a set of solution to the problem. Each crow has its corresponding memory m to store the (hiding) position, where the crow i memorizes the position with the best food obtained so far at iteration t and stores in $m^{i,t}$. During the optimization processing, the crow i is following the crow j to the hiding place $m^{j,t}$

of crow j at iteration t. There are two situations and either one will occur:

A. Situation 1

Crow j doesn't know it is followed by crow i and crow isuccessfully find the hiding place $m^{j,t}$, then crow i will updated its current position:

$$x^{i,t+1} = x^{i,t} + r_i \times l^{i,t} \times (m^{j,t} - x^{i,t}) \quad (1)$$

where r_i is a uniform distributed random number $\in [0,1]$, which effect the randomization during searching. The flight length $l^{i,t}$ represent the unit step of movement. Figure 1 shows the schematic of flight length in Case 1. The flight length fl is one of the parameters setting in CSA algorithm and it makes significant impact on search capability. Small fl value will result in local search, the neighborhood search space close to its current position $x^{i,t}$. As Figure 1 shows, the next position $x^{i,t+1}$ of crow i will be located on the dash line between $x^{i,t}$ and $m^{j,t}$ if the fl is set to less than 1. On the other hand, large value will tend to global search, the search space far away from current position $x^{i,t}$as Figure 1 shows. When the fl is set to large than one, the next position of crow i, $x^{i,t+1}$, will be located on any spot on dash line. It may exceed the position $m^{j,t}$ depends on the r_i.

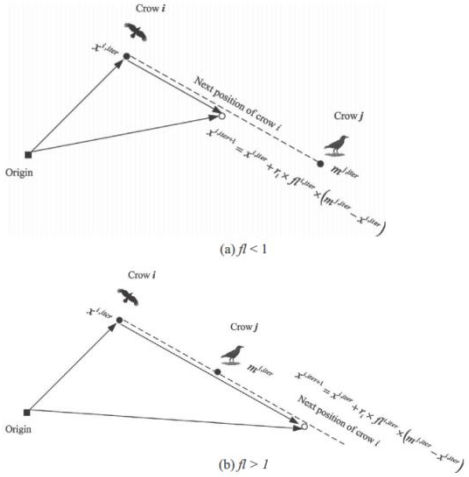


Fig. 1: Flight length of Case1 (a) fl< 1 and (b) fl> 1 inspired from [12]

B. Situation 2

Crow j noticed it is followed by crow i and fool it to a somewhere else in the environment. In the Case, crow i will be led to a random search space to stay away from crow j's hiding place, $m^{j,t}$.

By combining the two Cases together, the mathematical equation can be defined as follow:

$$x^{i,t+1} = \begin{cases} x^{i,t} + r_i \times l^{i,t} \times (m^{j,t} - x^{i,t}), & r_j \geq AP^{j,t} \\ \text{a random position}, & \text{Otherwise} \end{cases} \quad (2)$$

where r_j is a uniform distributed random number $\in [0,1]$ contributing the randomization during searching with the awareness probability AP. $AP^{j,t}$ is the awareness probability of crow j at iteration t. Awareness probability (AP) is another setting parameter in the CSA algorithm to provide the tradeoff of randomization and local search during the searching. A good tradeoff of between diversification and intensification is essential in metaheuristic algorithm [4]. In CSA algorithm, (AP) is used to directly controlled the balance between diversification and intensification. Small value of AP lead to find the good solution in the neighborhood region, which will increase intensification. On the other hand, large awareness probability will have high chance to explore the search space and tend to global search, which will increase diversification. The value of awareness probability should be set in between 0 and 1.

After the crow generates a new position by Eq.2, the feasibility of the new position is checked. The crow will not move nor update its position unless the new position is feasible. The update of the memory position of the crow depends on the comparison between the fitness function value of the new position and the fitness function value of the current memory position. If the fitness function value of the new position is better, the crow will dynamically update its memory. The mathematical equation of the memory update can be defined as follows:

$$m^{i,t+1} = \begin{cases} x^{i,t+1}, & f(x^{i,t+1}) \text{ is better than } f(m^{i,t}) \\ m^{i,t}, & \text{Otherwise} \end{cases} \quad (3)$$

where f() represent the fitness function value. The process will repeat until the number of iterations reached maximum. The best position among the memory of the crows at maximum iteration is the optimal solution for the problem.

C. Implementation of CSA

The implementation of CSA will be immediate step by step as follow:

- Step 1: The problem and parameters need to be initialized, including decision variables, constraints, given objective function, the flock size (N), maximum number of iterations (t_{max}), flight length (fl) and awareness probability (AP).
- Step 2: Positions of each crow will be randomly generated in a d-dimensional search space, whered is the number of decision variable. The initial memory position is also randomly generated since there are no memories at the beginning.
- Step 3: The quality of the initial positions and memories are evaluated by the fitness function.
- Step 4: Each crow will randomly select one crow to follow and explore its food hidden position (memory position). If the crow successfully discovers the food hidden position, the new position will be generated according to Eq.1. If fail, the new position will be a random position in the search space.
- Step 5: The feasibility of each new position will be checked based on the constraints of the problem before updated. If the new position is

not feasible, the crow will stay at the current position.

- Step 6: The quality of the new positions will be evaluated by the fitness function for comparison in the next step.
- Step 7: The crows will update their memories if the quality of the new position is better than the quality of the memorized position. If not, the crows will keep their current memories.
- Step 8: Steps 4-7 are repeated until the number of iterations reaches the maximum. The best position among the memory at maximum iteration will be present at the optimal solution for the problem.

III. CASE STUDY: CUTTING TOOLS

The Case study is about finding the best fit empirical model of cutting tool for calculating the temperature by determining the coefficients in the model. In this Case study, a P05 horny steel cutting tool and an EN-31 steel alloy chip tool are used to examine the CSA in the experiment, respectively. Two corresponding datasets are used, and the root means square error (RMSE) is used as the objective function to evaluate the quality of the solutions. The goal is to minimize the RMSE value between the experimental temperature T and the calculated temperature of T_r. The RMSE function is shown as Eq.4. The result of the problem with CSA will be compared with CS, GA, and PSO, and discussed in the next section.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(T - T_r)^2} \quad (4)$$

where Tis the real actual value and $T_r$is the estimated target value. n is the total number of measurements.

A. Case 1: A P05 Horny Alloy Steel Cutting Tool

Yi-jian, Jian-ming, and Shuqing [19] illustrated the P05 horny alloy steel cutting machine and adopted the PSO algorithm to estimate the values of the coefficient of the empirical model. The experiment metal cutting system with cutting tool P05 horny alloy steel is shown as Figure 2. The empirical model to calculate the temperature T_r for the P05 horny alloy steel cutting system is shown as follow:

$$T_r = k a_p^x f^y v^z \quad (5)$$

where $T_r$is the output of temperature model, k is a coefficient depending on the machined material, a_p is the depth of cut (mm), fis the cutting feed rate (mm/rev), v is the cutting speed (m/min), x, y and z are coefficients dependent on the cutting tool type of material.

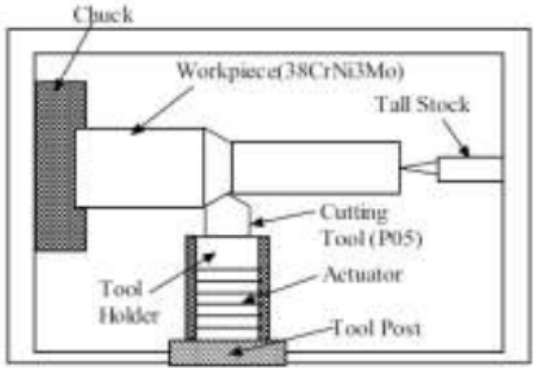


Fig.2: P05 Metal cutting system [20]

No.	a_p (mm)	f (mm/rev)	v (m/min)	T (°C)
1	1	0.1	60	740.24
2	1.41	0.2	60	793.8
3	2	0.4	60	854.8
4	1.41	0.1	94.8	819.42
5	2	0.2	94.8	870
6	1	0.4	94.8	904.8
7	2	0.1	150	879.6
8	1	0.2	150	911.1
9	1.41	0.4	150	974.1

TABLE I: Experimental data for P05 horny alloy steel Cutting Tool.

The experimental data used to examine the algorithm is shown as Table I. This empirical model has been studied with other approaches with the same dataset, including PSO [19], GA [21] and CS [22]. The result models with PSO, GA and CS shown in Table IV will be evaluated and compared with the result obtained by CSA method in the next section.

No.	v (m/min)	f (mm/rev)	a_p (mm)	r (mm)	T (° C)
1	39	0.06	0.2	0.4	186
2	39	0.06	0.2	1.2	149
3	39	0.06	0.6	0.4	290
4	39	0.06	0.6	1.2	224
5	39	0.15	0.2	0.4	249
6	39	0.15	0.2	1.2	238
7	39	0.15	0.6	0.4	327
8	39	0.15	0.6	1.2	310
9	189	0.06	0.2	0.4	364
10	189	0.06	0.2	1.2	350
11	189	0.06	0.6	0.4	403
12	189	0.06	0.6	1.2	389
13	189	0.15	0.2	0.4	458
14	189	0.15	0.2	1.2	439
15	189	0.15	0.6	0.4	180
16	189	0.15	0.6	1.2	467
17	112	0.1	0.4	0.8	413
18	112	0.1	0.4	0.8	399
19	112	0.1	0.4	0.8	387
20	112	0.1	0.4	0.8	408
21	112	0.1	0.4	0.8	395
22	112	0.1	0.4	0.8	427
23	112	0.1	0.4	0.8	390
24	112	0.1	0.4	0.8	383

TABLE II: Experimental data for EN-31 steel alloy Chip Tool.

B. Case 2: An En-31 Steel Alloy Chip Tool

The second study Case is to determine another empirical model of a tool-chip interface temperature during turning of EN-31 steel alloy with tungsten

carbide. With a tool-work thermocouple technique, an experimental dataset recorded by Abhand and Hameedullah [23] is shown in Table II. They measured the tool-chip interface temperature under different levels of cutting conditions and formulated an empirical model, as shown in Eq.6 based on the experimental data.

$$T_r = k a_p^a f^b v^c r^d \quad (6)$$

where $T_r$is the output of temperature model, k is a coefficient depending on the machined material, a_p is the depth of cut (mm), fis the cutting feed rate (mm/rev), v is the cutting speed (m/min), r is the tool noise radius, a, b, c and d are coefficients dependent on the cutting tool type of material. A study by Faris and Sheta [22] also applied GA, PSO and CS methods to determine this empirical model with same dataset. The result of the best optimized models is shown in the Table V. The performance of these models is also compared with our proposed model developed by CSA in the next section.

IV. EXPERIMENTAL SETUP

In this section, we show the encoding method for the solution-based CSA metaheuristic algorithms to solve the parameter optimization problem for the cutting tools process.We formulate the problem as a minimization problem. Thus, the goal of the optimization problem is to minimize the f by tuning the process parameters as follows:

$$\text{Min. } f(k, x, y, z)$$

Candidate solutions are defined as four and five dimensional vectors of parameters for Case 1 and Case 2 as follows:

k	x	y	z

k	a	b	c	d

The parameters are presented in a vector notation format. For each vector of parameter, there is an associated function value serves as its fitness, with lower values preferred for minimization problems. We run the experiment for four times with various flock size of 50, 100, 250 and 500 respectively. The different flock size setting is used to compare the population impact on computation results.

Parameter	Value
Maximum iterations	1000
Flock size	50,100,250,500
Flight length (*fl*)	2
Awareness probability (*AP*)	0.1

TABLE III: Parameter setting for CSA in cutting tool Case study.

A. Evaluation Metrics

In order to examine the performances of the CSA approach with the study Cases, the measurements used for evaluation are the Variance Accounted For (VAF), the Euclidian distance (ED), the mean squares error (MSE), Mean Absolute Error (MAE) and mean magnitude of relative error (MMRE) [24]. These measurements approaches are also used as the criteria for performance comparison among the CSA, PSO, GA, and CS methods. The VAF, ED, MSE, MAE, and MMRE measurements are computed as:

$$VAF = \left[1 - \frac{var(T - T_r)}{varT}\right] \times 100\% \quad (7)$$

$$ED = \sqrt{\sum_{i=1}^{n} |T - T_r|^2} \quad (8)$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (T - T_r)^2 \quad (9)$$

$$MD = \frac{1}{n}\sum_{i=1}^{n} |T - T_r| \quad (10)$$

$$MMRE = \frac{1}{n}\sum_{i=1}^{n} \frac{|T - T_r|}{T} \quad (11)$$

where T is the real actual value and $T_r$is the estimated target value, n is the total number of measurements.

V. EXPERIMENTAL RESULTS

Based on the experimental data shown in Table I and Table II, the CSA method is applied into two cutting tool model with tuning parameter shown in Table III to determine the optimal coefficients for the empirical model respectively. After 5 runs for each Cases, the optimized parametric model by CSA obtained so far are shown in Table IV for Case 1 and Table V for Case 2.

$$T_r = 470.2968 a_p^{0.0324} f^{0.0828} v^{0.159} \quad (12)$$

$$T_r = 163.4713 a_p^{0.1647} f^{0.2728} v^{0.3377} r^{-0.0751} \quad (13)$$

Based on the different flock size tuned in the CSA computing, the convergence curves of the two developed parametric models by CSA obtained so far are shown in Figure 3 for Case 1 and Figure 4 for Case 2. Both figures only display the first 250 iterations in order to show the different convergence rate based on different population size. These figures demonstrate that the CSA method has a great convergence rate of determining optimal coefficients of the empirical model in the cutting tool regardless of the population sizes for both study Case. Even though they show the larger population sizes help to find the optimal solution within fewer iterations compared to the smaller population, the difference between curves is small.

Approach	Parametric models	VAF rank	MSE rank	ED rank	MAE rank	MMRE rank
PSO	$T_r = 470.3\, a_p^{0.0324} f^{0.0828} v^{0.159}$	1	2	2	3	4
GA	$T_r = 471\, a_p^{0.0301} f^{0.083} v^{0.159}$	2	3	3	3	5
CS	$T_r = 471\, a_p^{0.0324} f^{0.0828} v^{0.15897}$	1	1	1	1	2
CSA	$T_r = 470.2968\, a_p^{0.0324} f^{0.0828} v^{0.159}$	1	2	2	2	3

TABLIE IV: Parametric models optimized for P05 horny alloy steel Cutting Tool from literature [19], [21], [22].

Approach	Parametric models	VAF rank	MSE rank	ED rank	MAE rank	MMRE rank
PSO	$T_r = 181.23\, a_p^{0.140} f^{0.1944} v^{0.2761} r^{0.000073}$	2	2	2	3	3
GA	$T_r = 159.333\, a_p^{0.136} f^{0.2312} v^{0.3182} r^{-0.0317}$	4	4	4	4	4
CS	$T_r = 172\, a_p^{0.144} f^{0.2473} v^{0.3137} r^{-0.0386}$	3	3	3	2	2
CSA	$T_r = 163.4713\, a_p^{0.1647} f^{0.2728} v^{0.3377} r^{-0.0751}$	1	1	1	1	1

TABLIE V: Parametric models optimized for EN-31 steel alloy Chip Tool from literature [19], [21], [22].

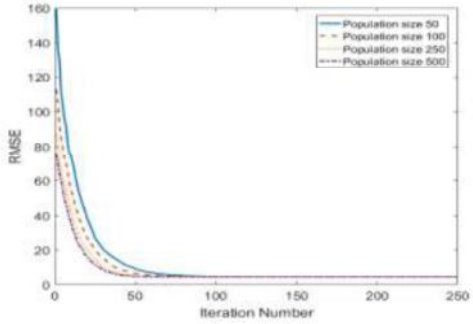


Fig. 3: Best so far convergence rate of CSA with different population sizes for P05 horny alloy steel Cutting Tool.

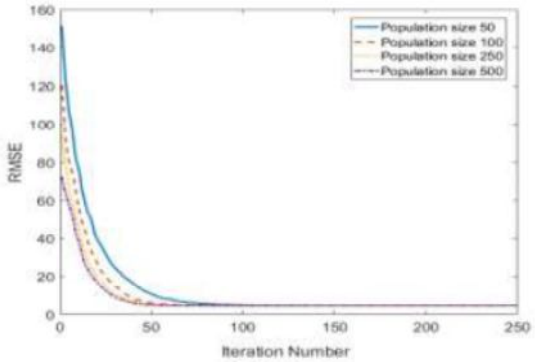


Fig. 4: Best so far convergence rate of CSA with different population sizes for EN-31 steel alloy Chip Tool.

For both Cases, the solutions converged to the excellent region for an optimal solution within 100 iterations with a population size of 50, and within 50 iterations with a population size of 500. The corresponding RMSEs at the optimal solution reported are 4.5234 for Case 1 and 25.29 for Case 2, respectively. Even if one more decision variable involved, the CSA algorithm still took the same number of iterations to find the optimal solution in Case 2 under the same population size compared to Case 1.

As mentioned in the previous section, two developed models estimated by CSA are evaluated by Equations 7-11. Other developed models estimated by PSO [19], LS, GA [21] and CS [22] are also evaluated with the same performance measurement methods and the comparative performance tables for Case 1 and Case 2 are shown as Table VI and VII respectively. The best value of performance results is highlighted (bold) in term of each performance measurement method.

The rank-based on different measurement methods is also presented in the Table IV and Table V. For Case 1, the performance results of CSA outperformed LS and GA, whereas CS beats it with a slight difference. On the other hand, the performance results of CSA in Case 2 illustrate the ascendancy compared to other optimization approaches. In additions, Figure 3 and Figure 4 demonstrate the outstanding convergence rate regardless of population size. Therefore, the CSA approach is a promising method for solving the optimization problems with the advantage of convergence rate.

VI. CONCLUSIONS

In the article, the metaheuristic algorithm, crow search algorithm, is introduced to solve the optimization problems. The less parameter tuning is an excellent advantage of CSA, which helps users to implement the CSA in solving optimization problems

quickly. The CSA can handle engineering optimization problems with complex objective functions, such as nonlinear and multi-modal. The results of two empirical model of cutting tool Case studies demonstrate that CSA is a practical approach to solve optimization problems. Through the performance comparison between the CSA approach with some popular metaheuristic algorithms, it illustrates the outstanding performance of the CSA approach in developing a complex empirical model. The better convergence rate of CSA in comparison shows the ascendancy once again. In order to prove the advantages and excellent performance of CSA, a more comparative study with other optimization algorithms in different optimization problems need to be done in the future, especially problems with more complicated objective function with a large number of decision variables.

Meanwhile, the computing time is another essential criteria which should be taken into consideration in further comparisons. More importantly, constraints in the problems can be considered into the CSA during computing. Based on this advantage of the CSA approach, problems with constraints should be used for studying the performance of CSA. In term of the CSA approach, the optimal values of parameter setting (fight length and awareness probability) selected based on different problems should be studied as well.

Criteria	LS [21]	PSO [19]	GA [21]	CS [22]	CSA
VAF (%)	99.52	**99.522**	99.514	**99.522**	**99.522**
MSE	25.229	20.463	20.929	**20.461**	20.463
ED	15.069	13.571	13.725	**13.57**	13.571
MAE	3.5815	3.7683	3.9048	**3.7526**	3.7664
MMRE	**0.0042223**	0.0044736	0.0046434	0.0044559	0.0044712

TABLIE VI: Comparative performance evaluation results for P05 horny alloy steel Cutting Tool.

Criteria	LS [21]	PSO [19]	GA [21]	CS [22]	CSA
VAF (%)	77.11	89.409	87.7	89.362	**91.818**
MSE	1964.9	827.62	965.57	860.14	**639.37**
ED	217.16	140.94	152.23	143.68	**123.87**
MAE	38.736	23.772	25.722	22.773	**20.61**
MMRE	0.10903	0.079903	0.091778	0.077904	**0.067776**

TABLIE VII: Comparative performance evaluation results for EN-31 steel alloy Chip Tool.

REFERENCES

[1] A. Sheta, M. Braik, and H. Al-Hiary, "Modeling the Tennessee Eastman chemical process reactor using bio-inspired feedforwardneural network (BI-FF-NN),"The International Journal of Advanced Manufacturing Technology, Apr 2019.

[2] F. Glover, "Future paths for integer programming and links to artificial intelligence," Computers Operations Research, vol. 13, no. 5, pp. 533 – 549, 1986, applications of Integer Programming. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0305054886900481

[3] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," ACM Comput. Surv., vol. 35, no. 3, pp. 268–308, Sep. 2003. [Online].Available:http://doi.acm.org/10.1145/937503.937505

[4] X.-S. Yang, "Metaheuristic optimization," Scholarpedia, vol. 6, no. 8, p. 11472, 2011.

[5] M. Gen and R. Cheng, Genetic algorithms and engineering optimization. John Wiley & Sons, 2000, vol. 7.

[6] X.-S. Yang and A. Hossein Gandomi, "Bat algorithm: a novel approach for global engineering optimization," Engineering Computations, vol. 29, no. 5, pp. 464–483, 2012.

[7] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," Engineering applications of artificial intelligence, vol. 20, no. 1, pp. 89–99, 2007.

[8] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," Engineering with computers, vol. 29, no. 1, pp. 17–35, 2013.A

[9] X.-S. Yang and S. Deb, "Engineering optimisation by cuckoo search," arXiv preprint arXiv:1005.2908, 2010.

[10] Z. W. Geem, J. H. Kim, and G. Loganathan, "A new heuristic optimization algorithm: Harmony search," SIMULATION, vol. 76, no. 2, pp. 60–68, 2001. [Online]. Available: https://doi.org/10.1177/003754970107600201

[11] A. Sheta, H. Faris, and I. Aljarah, "Estimating arma model parameters of an industrial process using meta-heuristic search algorithms," International Journal of Engineering & Technology, vol. 07, p. 187âAS194, Apr 2018.

[12] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," Computers Structures, vol. 169, pp. 1 – 12, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045794916300475

[13] H. Prior, A. Schwarz, and O. Güntürkün, "Mirror-induced behavior in the magpie (pica pica): Evidence of self-recognition," PLOS Biology, vol. 6, no. 8, pp. 1–9, 08 2008. [Online]. Available: https://doi.org/10.1371/journal.pbio.0060202

[14] P. Rincon, "Crows and jays top bird iq scale," BBC News, 2005. [Online]. Available: http://news.bbc.co.uk/2/hi/science/nature/4286965.stm

[15] F. Tan and R. Creese, "A generalized multi-pass machining model for machining parameter selection in turning," The International Journal of Production Research, vol. 33, no. 5, pp. 1467–1487, 1995.

[16] I. Mukherjee and P. K. Ray, "A review of optimization techniques in metal cutting processes," Computers & Industrial Engineering, vol. 50, no. 1-2, pp. 15–34, 2006.
[17] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," Journal of the society for Industrial and Applied Mathematics, vol. 11, no. 2, pp. 431–441, 1963.
[18] N. Clayton and N. Emery, "Corvid cognition," Current Biology, vol. 15, no. 3, pp. R80 – R81, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0960982205000801
[19] L. Yi-jian, Z. Jian-ming, and W. Shu-qing, "Parameter estimation of cutting tool temperature nonlinear model using pso algorithm," Journal of Zhejiang University-SCIENCE A, vol. 6, no. 10, pp. 1026–1029, 2005.
[20] Y. Fang and Y. Liu, "Parameter estimation of cutting tool temperature nonlinear model using a novel simplified e. coli foraging optimization algorithm," pp. 2659–2662, Sep. 2008.
[21] J. Zhou, Y. Liu, and Q. Yu, "GA algorithm for cutting experiment data drawing," Journal of Southwest Petroleum Institute, vol. 29, no. 3, pp. 62–63, 1998.
[22] H. Faris and A. Sheta, "A comparison between parametric and non-parametric soft computing approaches to model the temperature of a metal cutting tool," International Journal of Computer Integrated Manufacturing, vol. 29, no. 1, pp. 64–75, 2016.
[23] L. Abhang and M. Hameedullah, "Chip-tool interface temperature prediction model for turning process," International Journal of Engineering Science and Technology, vol. 2, no. 4, pp. 382–393, 2010.
[24] M. Shin and A. L. Goel, "Empirical data modeling in software engineering using radial basis functions," IEEE Transactions on Software Engineering, vol. 26, no. 6, pp. 567–576, 2000.

★★★