Cuckoo Optimization Algorithm in Cutting Conditions During Machining

Ahmad Esfandiari
Department of Engineering, Islamic Azad University, Sari Branch, Sari, Iran
esfandiari@iausari.ac.ir

Received: 2014/03/15; Accepted: 2014/04/19

Abstract

Optimization of cutting conditions is a non-linear optimization with constraint and it is very important to the increase of productivity and the reduction of costs. In recent years, several evolutionary and meta-heuristic optimization algorithms were introduced. The Cuckoo Optimization Algorithm (COA) is one of several recent and powerful meta-heuristics which is inspired by the cuckoos and their lifestyle. In this paper, COA, Simulated Annealing (SA), Genetic Algorithm (GA) and Imperialist Competitive Algorithm (ICA) are first applied to five test functions and the performance of these algorithms is compared. These algorithms are then used to optimize the cutting conditions. The results showed that COA has more capabilities such as accuracy, faster convergence and better global optimum achievement than others.

Keywords: Cuckoo Optimization Algorithm (COA); cutting conditions; Imperialist Competitive Algorithm (ICA); Simulated Annealing (SA); Genetic Algorithm (GA)

1. Introduction

In manufacturing, the process of removing unwanted segment of metal workpiece in the form of chips is known as machining. Machining is one of the five groups of manufacturing processes which includes casting, forming, powder metallurgy and joining. The machining process will shape the workpiece as desired and it is usually done using machine and cutting tools. The machining cutting process can be divided into two major groups which are (i) cutting process with traditional machining (e.g., turning, milling, boring and grinding) and (ii) cutting process with modern machining (e.g., electrical discharge machining (EDM) and abrasive waterjet (AWJ)) [1].

The objective is the minimization of the product cost in conventional manufacturing. However, during the manufacturing of precision parts the achievement of very high quality standards becomes the primary objective [2]. The optimization of cutting parameters is the key component in planning of machining processes. Usually, the desired cutting parameters are determined based on the experience or by the use of standard handbooks [3]. However, deep analysis of cutting involves certain costs, particularly in case of small series. In case of individual machining it is particularly necessary to shorten as much as possible the procedure for determination of the optimum cutting parameters, otherwise the cost of analysis might exceed the economic efficiency which could be reached
if working with optimum conditions [4]. In today’s manufacturing world, the primary objective in machining operations is to produce high-quality products with low cost. In order to minimize the machining cost for machining economics problem, the optimization of cutting parameters is one of the most important issues since these parameters strongly affect the cost, productivity and quality [5, 6].

At this juncture, it is necessary to find a suitable method to select the appropriate cutting conditions during machining. In the recent years, analytical-direct space search procedures were followed to optimize the machining parameters. They were dynamic programming, geometric programming, stochastic programming, etc. These techniques are not appropriate to search the optimum cutting parameters[7, 8]. However these optimization techniques are either stuck at local optimum or take a long time to converge to a reasonable result. Thus, meta-heuristic algorithms have been developed to solve machining economics problems because of their power in global searching [9]. GA has been greatly used for the selection of the operating conditions in machining operations [10-33]. To simplify the modeling, simulated annealing [8, 29-32, 34, 35], fuzzy logic [36, 37], and NNs [38-44] have been used with the GA. Recently Ants Colony Optimization (ACO) [1, 33, 45-47] and Particle Swarm Optimization (PSO) [1, 29, 44, 48-51] and Imperialist Competitive Algorithm (ICA)[52] were applied to optimize the machining parameters.

The operation of cutting is defined as a multiple-objective optimization problem with limitation non-equations and with three conflicting objectives (production rate, operation cost and quality of machining). All the above mentioned objectives are represented as a function of the cutting speed, feed rate and depth of cutting [4].

In this study a recently developed meta-heuristic optimization algorithm named as Cuckoo Optimization Algorithm (COA) is introduced. The aim is to demonstrate the potential of COA for machining process optimization. To validate the proposed approach, compare is made against SA, GA and ICA methods.

2. Cuckoo Optimization Algorithm (COA)

There are several evolutionary and meta-heuristic optimization algorithms such as simulated annealing (SA), genetic algorithm (GA), particle swarm optimization(PSO), Ant Colony optimization (ACO), ABC (artificial bee colony), artificial fish swarm and imperialist competitive algorithm (ICA). The cuckoo optimization algorithm (COA) is one of several recent and powerful meta-heuristics.

COA is a nature-inspired meta-heuristic introduced by Yang and Deb [53]. It is inspired by the cuckoos and their lifestyle. The COA proposed by Yang & Deb [54] is based on the obligate brood parasitic behavior of some cuckoo species in combination with the Lévy flight behavior of some birds and fruit flies.

Rajabioun [55] developed this algorithm and proved the efficiency of his algorithm via a benchmarking study.

The steps to reach the global optima solution in COA are given as follows:

First, like other evolutionary algorithms, COA starts with an initial population of cuckoos and they have some eggs to lay in some host bird’s nests. Of these eggs, a group that more closely resembles the host bird's eggs has a greater opportunity for growth and will mature into cuckoo. Other eggs are detected by host birds and are
destroyed. The number of mature eggs shows the suitability of the nest in that area. The more eggs survived in an area and can live, the more profit (interest) that area will have. The situation in which most eggs are saved is the mode that COA is trying to optimize. Cuckoos search the best area to maximize their survival eggs. After the chicks were hatched and were mature into cuckoo, they create communities and groups.

Now each group has its specific residence for living. The best residence for all groups will be next destination for the cuckoos in the other groups. Hence, all groups migrate to the best current available area. Each group will be living in a region near the current location. Considering the number of eggs that every cuckoo will lay and also the distance of cuckoos from the current optimal region for residence, a number of radius lying is calculated and formed. Then cuckoos start to randomly lie in the nests inside lying. This process continues to reach the best place to laying (the region with the highest profit). This optimum location is where the most number of cuckoos get together[55].Figure 1 shows the flowchart of the COA.

![Figure 1: Flowchart of Cuckoo optimization algorithm (COA)](image)

3. Problem formulation
3.1. General background

There are many factors related to machinability. These factors can be classified as type of machining operations (turning, facing, milling, etc.), parameters of machine tools (rigidity, horsepower, etc.), parameters of cutting tools (material, geometry, etc.), parameters of operations (cutting speed, feed rate, depth of cut, etc.), and characteristics of work piece (material, rigidity, geometry, etc.).

In defining the machinability of a workpiece we consider to types of factors:

a) Objective factors: these factors completely pertain to the behavioral performance of the workpiece in terms of process outputs such as power consumption, vibration, etc.

b) Subjective factors: these factors completely pertain to the manufacturer's interest or performance with respect to different objectives. They are based on trade off among minimizing cost of operation, maximizing production rate and quality of cutting, etc.
In this section, first we specify three machine variables. Then we discuss three objectives for machinability evaluation which are functions of machine variables as well as process outputs.

3.2. Machine variables

The machine variables are the independent operation parameters such as cutting speed \(v\), feed rate \(f\) and depth of cut \(a\). The cutting speed \(v\) is defined as the rate at which the uncut surface of the workpiece passes the cutting edge of the tool. In operations with rotational movement of the tool or workpiece, the operating parameter given for speed is usually rotating speed \(n\) in revolutions per minute. For turning operations, the feed rate \(f\) is the distance moved by the tool in an axial direction at each revolution of the workpiece. The depth of cut \(a\) is the thickness of metal removed from the workpiece measured in a radial direction.

3.3. Objectives for machinability

Machinability ratings can be based on evaluation of certain economic and technical objectives which are the consequences of the machining operation on a given workpiece material. In general there are three major objectives: production rate, operation cost and product quality [7].

1. Production rate. Production rate can be derived from the total operation time required to produce one item of product. It is a function of metal removal rate \(MRR\), tool life \(T\), set up time per part, machining time per part, tool change time, and idle time per part. Usually, the production rate is measured as the entire time necessary for the manufacture of a product \(T_p\). It is the function of the \(MRR\) and of the \(T\):

\[
T_p = T_s + V \frac{(1 + \frac{T_c}{MRR})}{MRR} + T_i
\]

Where \(T_s\), \(T_c\), \(T_i\) and \(V\) are the tool set up time, the tool change time, the time during which the tool does not cut and the volume of the removed metal. In some operations the \(T_s\), \(T_c\), \(T_i\) and \(V\) are constants so that \(T_p\) is the function of \(MRR\) and \(T\).

- The metal removal rate (MRR). MRR can be expressed by analytical derivation as the product of the cutting speed, feeding and cutting depth:

\[
MRR = 1000 \cdot v \cdot f \cdot a
\]

- Tool life (T). The tool life is measured as the average time between the tool changes or tool sharpening. The relation between the tool life and the parameters is expressed with the well-known Taylor’s formula:

\[
T = \frac{K_T}{V \cdot f^{a_1} \cdot a^{a_2}}
\]

where the \(k_T\), \(a_1\), \(a_2\) and \(a_3\), which are always positive constant parameters, are determined statistically.
2. **Operation cost.** Operation cost can be expressed at cost per product \( C_p \), which a function of tool cost per piece, power cost, machining cost per minute, cost of labour, cost of overhead, etc. The \( C_p \) can be expressed as:

\[
C_p = T_p \frac{C_t + C_l + C_o}{T + T_p}
\]

(4)

Where \( T \) and \( T_p \) are cutting parameters and \( C_t, C_l \) and \( C_o \) are the tool cost, the labour cost and the overhead cost respectively. In some operations the \( C_t, C_l \) and \( C_o \) are independent of the cutting parameters.

3. **Production quality.** Product quality includes all aspects of the machine workpiece, such as surface roughness or finish, dimensional accuracy, freedom from burrs, scratches, cracks, breakout chatter marks, tooth marks, thermal damage, etc.

The most important criterion for the assessment of the surface quality is roughness calculated according to:

\[
R_a = kN^x_1 \cdot f^{x_2} \cdot a^{x_3}
\]

(5)

where \( x_1, x_2, x_3 \) and \( k \) are the constants relevant to a specific tool-workpiece combination [7, 52].

The objective of the optimization is to determine the optimal machining parameters including cutting speed, feed rate and depth of cut for minimizing of entire time necessary for the manufacture of a product, production cost and surface roughness. According to that, the objective function for cutting process can be mathematically stated as follows[4, 7, 21, 52, 56, 57]:

\[
F(T_p, C_p, R_a) = 0.42e^{-0.22T_p} + 0.36e^{-0.32C_p} + 0.17e^{-0.26R_a} + \frac{0.05}{(1+1.22T_p, C_p, R_a)}
\]

(6)

3.4. **Practical constraints**

Practical constraints always exist in any machining which has to be considered to get real time performance measures. In this work, different practical constraints are considered before calculating production time, cost and quality:

- **Parameters constraints**
  
  \[ v_{\text{min}} \leq v \leq v_{\text{max}}, f_{\text{min}} \leq f \leq f_{\text{max}}, a_{\text{min}} \leq a \leq a_{\text{max}} \]

  (7)

- **Cutting power and force constraints**
  
  The consumption of the power can be expressed as the function of the cutting force and cutting speed:

  \[
P = \frac{F \cdot v}{6122.45 \cdot \eta}
\]

  (8)

  where \( \eta \) is the mechanical efficiency of the machine and \( F \) is given by following formula:

  \[
  F = k_f \cdot f^{\beta_1} \cdot a^{\beta_2}
\]

  (9)

  The constraints of the power and cutting force are equal to:
$P(v, f, a) \leq P_{\text{max}}$, $F(v, f, a) \leq F_{\text{max}}$ (10)

4. Test functions

If a function has two or more local optima is multimodal. The search process must be able to avoid the regions around local minima in order to approximate, as far as possible, to the global optimum. The most complex case appears when the local optima are randomly distributed in the search space. The dimensionality of the search space is another important factor in the complexity of the problem. To compare the performance of the proposed COA with SA, GA, ICA, we used five classical test functions as given in Ref. [58]

4.1. Griewank function

The Griewank function has value 0 at its global minimum (0,0,…,0). Its initialization range is $[-600,600]$. The Griewank function of order $n$ is distinct by:

$$f(x) = \frac{1}{4000} \left( \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) \right) + 1$$ (11)

4.2. Rastrigin function

The Rastrigin function has value 0 at its global minimum (0,0,…,0). Initialization range for the function is $[-5.12, 5.12]$. Finding the minimum of this function is a fairly difficult problem due to its large search space and its large number of local minima. It is defined by:

$$f(x) = \sum_{i=1}^{n} \left( x_i^2 - 10 \cos(2\pi x_i) + 10 \right)$$ (12)

4.3. Rosenbrock function

The third function is Rosenbrock function that its value is 0 at its global minimum (1,1,…,1). Initialization range for the function is $[-2.048, 2.048]$. It is defined by:

$$f(x) = \sum_{i=1}^{n} \left( 100(x_{i+1}^2 - x_i)^2 + (1 - x_i)^2 \right)$$ (13)

4.4. Ackley function

The forth function is Ackley function whose value is 0 at its global minimum (0,0,…,0). Initialization range for the function is $[-32.768, 32.768]$. The Ackley function is distinct by:

$$f(x) = 20 + e - 20e^{\frac{\left( x - e^{\sum_{i=1}^{n} x_i^2} \right)}{\sum_{i=1}^{n} x_i^2}} - \frac{1}{e} \sum_{i=1}^{n} \cos(2\pi x_i)$$ (14)

4.5. Salomon function

The fifth function is Salomon function whose value is 0 at its global minimum (0, 0,…, 0). Initialization range for the function is $[-100, 100]$. It is defined by:

$$f(x) = 1 - \cos(2\pi \sqrt{\sum_{i=1}^{n} x_i^2}) + 0.1 \sqrt{\sum_{i=1}^{n} x_i^2}$$ (15)

5. Results and discussion

This section illustrates some comparisons between the COA (cuckoo optimization algorithm) and SA (simulated annealing), GA (genetic algorithm) and ICA (imperialist competitive algorithm) using some test functions as depicted in section 4. Then each of the above four algorithms are applied on the cutting conditions problem. Each of the algorithms has been performed several times, and the best results have been considered.
The following parameters have been used to each of the four abovementioned algorithms.

**SA setting:** Fast annealing function, reannealing interval of 100, exponential temperature function, initial temperature of 100 are used for SA.

**GA setting:** The following GA parameters were used to reach the best results: initial population=100; probability of crossover=0.5; probability of mutation=0.4; Rank scaling function, Tournament selection function, single point crossover function and Gaussian mutation function.

**COA and ICA setting:** The parameters of the COA and ICA are summarized in Table 1 and Table 2 respectively.

### 5.1. Results of test functions

Tables 3, 4, 5, 6 and 7 illustrate the empirical performance of the SA, GA, ICA and COA algorithms on five test functions for different iterations. For Griewank and Rastrigin functions four variables, for Rosenbrock and Salomon functions three variables and for Ackley function five variables have been considered. As evident from the results, COA has more accuracy, faster convergence and better global optimum achievement than others. Also after COA, the GA is better than others.

#### Table 1: parameters of COA

<table>
<thead>
<tr>
<th></th>
<th>5</th>
<th>2</th>
<th>4</th>
<th>1</th>
<th>9</th>
<th>10</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial population</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum number of eggs for each cuckoo</td>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum number of eggs for each cuckoo</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum iteration</td>
<td></td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of clusters</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion coefficient (Lambda)</td>
<td></td>
<td></td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum number of cuckoos</td>
<td></td>
<td></td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Radius coefficient</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

#### Table 2: parameters of ICA

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of initial countries</td>
<td>200</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of empires</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Number of iterations</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>Assimilation coefficient β</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.55</td>
</tr>
<tr>
<td>Assimilation angle γ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>Revolution rate</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.35</td>
</tr>
</tbody>
</table>

#### Table 3: performance of SA, GA, ICA and COA on Griewank function with 4 variables

<table>
<thead>
<tr>
<th></th>
<th>X_1</th>
<th>X_2</th>
<th>X_3</th>
<th>X_4</th>
<th>Best Cost</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0.005</td>
<td>0.05</td>
<td>0.005</td>
<td>0.005</td>
<td>2.6066e-005</td>
<td>2000</td>
</tr>
<tr>
<td>GA</td>
<td>-0.002</td>
<td>0.005</td>
<td>0.005</td>
<td>0.005</td>
<td>1.9908e-005</td>
<td>51</td>
</tr>
<tr>
<td>ICA</td>
<td>0.0017</td>
<td>-0.0065</td>
<td>-0.0191</td>
<td>-0.0215</td>
<td>1.3031e-004</td>
<td>100</td>
</tr>
<tr>
<td>COA</td>
<td>1.4520e-009</td>
<td>-2.1100e-009</td>
<td>2.5300e-010</td>
<td>0</td>
<td>0</td>
<td>25</td>
</tr>
</tbody>
</table>

#### Table 4: performance of SA, GA, ICA and COA on Rastrigin function with 4 variables

<table>
<thead>
<tr>
<th></th>
<th>X_1</th>
<th>X_2</th>
<th>X_3</th>
<th>X_4</th>
<th>Best Cost</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0</td>
<td>6.0205e-004</td>
<td>5078</td>
</tr>
<tr>
<td>GA</td>
<td>0</td>
<td>0.001</td>
<td>0</td>
<td>0.001</td>
<td>1.9020e-004</td>
<td>71</td>
</tr>
<tr>
<td>COA</td>
<td>-4.9000e-011</td>
<td>3.0990e-009</td>
<td>1.1000e-011</td>
<td>0</td>
<td>0</td>
<td>21</td>
</tr>
</tbody>
</table>
Table 5: performance of SA, GA, ICA and COA on Rosenbrock function with 3 variables

<table>
<thead>
<tr>
<th></th>
<th>X₁</th>
<th>X₂</th>
<th>X₃</th>
<th>Best Cost</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>1.009</td>
<td>1.018</td>
<td>1.035</td>
<td>4.0536e-004</td>
<td>2034</td>
</tr>
<tr>
<td>GA</td>
<td>1.007</td>
<td>1.013</td>
<td>1.027</td>
<td>2.2068e-004</td>
<td>160</td>
</tr>
<tr>
<td>ICA</td>
<td>1.0047</td>
<td>1.0095</td>
<td>1.0190</td>
<td>1.1241e-004</td>
<td>300</td>
</tr>
<tr>
<td>COA</td>
<td>1.0023</td>
<td>1.0054</td>
<td>1.0115</td>
<td>1.4261e-004</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 6: performance of SA, GA, ICA and COA on Ackley function with 5 variables

<table>
<thead>
<tr>
<th></th>
<th>X₁</th>
<th>X₂</th>
<th>X₃</th>
<th>X₄</th>
<th>X₅</th>
<th>Best Cost</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5.1832e-006</td>
<td>1800</td>
</tr>
<tr>
<td>GA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4.6650e-004</td>
<td>64</td>
</tr>
<tr>
<td>ICA</td>
<td>9.5000e-008</td>
<td>-1.6750e-006</td>
<td>2.1710e-006</td>
<td>-5.6800e-007</td>
<td>-4.1800e-007</td>
<td>5.0682e-006</td>
<td>100</td>
</tr>
<tr>
<td>COA</td>
<td>1.0000e-008</td>
<td>0</td>
<td>0</td>
<td>-2.6600e-006</td>
<td>-1.0560e-005</td>
<td>1.9489e-005</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 7: performance of SA, GA, ICA and COA on Salomon function with 3 variables

<table>
<thead>
<tr>
<th></th>
<th>X₁</th>
<th>X₂</th>
<th>X₃</th>
<th>Best Cost</th>
<th>Iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0585e-006</td>
<td>1500</td>
</tr>
<tr>
<td>GA</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.6660e-006</td>
<td>81</td>
</tr>
<tr>
<td>COA</td>
<td>0</td>
<td>1.0680e-005</td>
<td>1.0000e-008</td>
<td>1.0707e-006</td>
<td>40</td>
</tr>
</tbody>
</table>

5.2. Results of cutting conditions problem

The best cost for the four above algorithms and their iterations there are in Table 8. SA is reached to best cost 0.2172 at iterations 1951. ICA and GA have achieved to this cost at iterations 27 and 65, respectively, while the COA has reached to this at iterations 3. The results of the four algorithms show that SA has the lowest rate of convergence, ICA converge to the optimal solution faster than the GA and the COA is also much faster than all of them.

Achieved best costs associated with number of iterations are shown in Figures 2 to 5. Figure 6 shows the initial empires, empires at iteration 20, empires at iterations 30 and 70 (maximum iteration) in ICA. ICA has reached to global optima at iterations 27. Also figures 7 and 8 depict the minimum and mean cost of GA and ICA versus iteration, respectively.

The results of implementation show the COA successfully compared with other algorithms. This Algorithm has very good results in speed and accuracy of convergence and simplicity of computations.
### Table 8: Comparison of best costs of objective function

<table>
<thead>
<tr>
<th>Iteration</th>
<th>SA</th>
<th>GA</th>
<th>ICA</th>
<th>COA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.4882</td>
<td>0.6193</td>
<td>0.2307</td>
<td>0.2589</td>
</tr>
<tr>
<td>2</td>
<td>0.3144</td>
<td>0.6892</td>
<td>0.2301</td>
<td>0.2304</td>
</tr>
<tr>
<td>3</td>
<td>0.4355</td>
<td>0.5623</td>
<td>0.2213</td>
<td>0.2172</td>
</tr>
<tr>
<td>4</td>
<td>0.5033</td>
<td>0.5351</td>
<td>0.2285</td>
<td>--------</td>
</tr>
<tr>
<td>5</td>
<td>0.2874</td>
<td>0.4522</td>
<td>0.2237</td>
<td>--------</td>
</tr>
<tr>
<td>6</td>
<td>0.2618</td>
<td>0.4938</td>
<td>0.2302</td>
<td>--------</td>
</tr>
<tr>
<td>7</td>
<td>0.4041</td>
<td>0.4665</td>
<td>0.2267</td>
<td>--------</td>
</tr>
<tr>
<td>8</td>
<td>0.2208</td>
<td>0.3716</td>
<td>0.2268</td>
<td>--------</td>
</tr>
<tr>
<td>9</td>
<td>0.2724</td>
<td>0.3996</td>
<td>0.2229</td>
<td>--------</td>
</tr>
<tr>
<td>10</td>
<td>0.2398</td>
<td>0.3141</td>
<td>0.2250</td>
<td>--------</td>
</tr>
<tr>
<td>13</td>
<td>0.2305</td>
<td>0.2833</td>
<td>0.2261</td>
<td>--------</td>
</tr>
<tr>
<td>14</td>
<td>0.2514</td>
<td>0.2827</td>
<td>0.2228</td>
<td>--------</td>
</tr>
<tr>
<td>20</td>
<td>0.2578</td>
<td>0.2825</td>
<td>0.2201</td>
<td>--------</td>
</tr>
<tr>
<td>27</td>
<td>0.2283</td>
<td>0.2531</td>
<td>0.2172</td>
<td>--------</td>
</tr>
<tr>
<td>30</td>
<td>0.2225</td>
<td>0.2358</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>50</td>
<td>0.2214</td>
<td>0.2177</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>65</td>
<td>0.2200</td>
<td>0.2172</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>70</td>
<td>0.2378</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>90</td>
<td>0.2286</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>100</td>
<td>0.2266</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>1565</td>
<td>0.2172</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>

**Figure 2**: Minimum cost of SA versus iteration  
**Figure 3**: Minimum cost of GA versus iteration  
**Figure 4**: Minimum cost of ICA versus iteration  
**Figure 5**: Minimum cost of COA versus iteration
6. Conclusion

In this paper, we have presented cuckoo optimization algorithm (COA) to optimize the cutting condition during machining and the results of this algorithm have been compared with simulated annealing (SA), genetic algorithm (GA) and imperialist competitive algorithm (ICA). The results achieved in this experience show that COA has high global searching ability, high computation efficiency and high optimization speed, and can converge to the optimum solution rapidly and the complexity of algorithm is lower.

Further research can also use the performance of this algorithm for multi-objective optimization.
References


