Soft Comput
DOI 10.1007/s00500-014-1459-6

METHODOLOGIES AND APPLICATION

A novel optimization algorithm inspired by the creative thinking process

Xiang Feng · Ru Zou · Huiqun Yu

Abstract Creative thinking, which plays an essential role in the progress of human society, has an outstanding problem-solving ability. This paper presents a novel creativity-oriented optimization model (COOM) and algorithm (COOA) inspired by the creative thinking process. At first, COOM is constructed by simplifying the procedure of creative thinking while retaining its main characteristics. And then, COOA is presented for continuous optimization problems. It is a realization of COOM. As a new nature-inspired algorithm, COOA is different from other similar algorithms in terms of the basic principle, mathematical formalization and properties. Features of the COOM and the corresponding algorithm include a powerful processing ability for the complex problems, namely high-dimensional, highly nonlinear and random problems. The proposed approach also has the advantages in terms of the higher intelligence, effectiveness, parallelism and lower computation complexity. The properties of COOA, including convergence and parallelism, are discussed in detail. The numerous simulations on the CEC-2013 real-parameter optimization benchmark functions' problems have shown the effectiveness and parallelism of the proposed approach.

Keywords Creativity-oriented optimization algorithm · Nature-inspired algorithm · Creative thinking · Numerical function optimization

Communicated by V. Loia.

X. Feng (✉) · R. Zou · H. Yu
Department of Computer Science and Engineering,
China University of Science and Technology,
Meilong Road 130, 200237 Shanghai, People's Republic of China
e-mail: xfeng@ecust.edu.cn

Published online: 19 September 2014

1 Introduction

1.1 Background and significance

At the interface between natural sciences and computer science, great discoveries await researchers who seek them. Natural computing is such a highly interdisciplinary field that it connects the natural sciences with computing science (Kari and Rozenberg 2008). A commonly used research method in this field is to mimicking natural phenomena or biological models, and then design corresponding algorithms. Nowadays, nature-inspired algorithms have become a popular choice to solve the complex problems in real world because of their intelligence and simplicity. However, with the development of human society, the scale and complexity of optimization problems are expanding promptly. Existing algorithms could not always meet the demands. It is necessary to continually exploring new intelligent and effective algorithms. Creative thinking is seen as the most complex and abstract of the high-order cognitive skills, and it can produce solutions for complex problems through unexpected insights (DeHaan 2011). Based on the mindset of learning from nature (Kephart 2011), it will be challenge and meaningful to mimic the creative thinking process to present a new nature-inspired algorithm.

1.2 Related work

Creativity is seen as a marvel of the human mind (Boden 2009). Edward de Bono said that, without creativity, there will be no progress (Bono 2007). Creative thinking is the core of creativity, and it always leads to ideas that are novel and valuable. In the past few decades, the study of creative thinking has drawn a lot of attentions. Up to now, there have been a large number of researches about creative thinking in psychology, neurology, sociology, etc. These researches can be defined within two different theoretical frameworks (DeHaan 2011). One framework is from the individual level. Most researches in this framework consider the divergent thinking and convergent thinking as two important thinking modes of creative thinking (Guilford 1967; Runco 2010; DeHaan 2011; Chermahini and Hommel 2012; Mumford et al. 2012; Runco and Acar 2012). The "insight" discussed in Kounios and Beeman (2009) can be seen as an inspiration thinking mode of creative thinking, and Sternberg (1999), Balter (2010), Vartanian et al. (2013) and Lee and Therriault (2013) indicates that the "experience" or "working memory" is an important influence factor to creative thinking. The other framework is from the social aspect. Researches in this framework mainly present that knowledge sharing and social interactions play important roles in fostering creativity (Ashton-James and Chartrand 2009; Wang et al. 2010; Sternberg 2010; Fink et al. 2012; Carmeli et al. 2013). All researches above provide a theoretical guidance for us to model the creative thinking process.

Up to now, it seems that no nature-inspired algorithm is proposed based on the creative thinking process. However, existing researches of nature-inspired algorithms can provide references for our work. The inspirations of these algorithms come from different nature systems. For instance, the evolutionary algorithms (EAs) (Das and Suganthan 2011; Peterson 2011; Jong 2012; Ahmed and Deb 2013) are inspired by the nature evolution theory, swarm intelligence (SI) algorithms (Krause et al. 2010; Neyoy et al. 2013; Melin et al. 2013) are based on the various behaviors among social creatures, the artificial immune system (AIS) (Zhang and Qian 2011) is constructed by simulating the biological immune system, the simulated annealing (SA) (Sousa et al. 2012) is presented according to the statistics mechanics of particles in the molten metal, etc. In recent years, a number of new nature-inspired algorithms have been proposed, such as artificial bee colony algorithm (ABC) (Karaboga et al. 2014), gravitational search algorithm (GSA) (Precup et al. 2012), bacterial foraging optimization (Xu and Chen 2014), mosquito host-seeking algorithm (MHSA) (Feng et al. 2013) and root growth algorithm (RGA) (Zhang et al. 2014). Each of these algorithms has its own characteristics. Their modeling processes and effective mechanisms could inspire us when developing our own algorithm.

Nevertheless, limitations still exist in the current nature-inspired algorithms. To begin with, most of the existing nature-inspired algorithms do not have intelligent enough study objects. Therefore, these algorithms will be more and more difficult to satisfy the increasingly overwhelming requirements for intelligence in real world. Moreover, under their current mechanisms, the existing population-based algorithms need too many internal communications, which will limit the parallelism of them, especially in the realistic applications that have high interactive costs. Our purpose is to put forward a high intelligent and high parallelism algorithm inspired by the creative thinking process.

1.3 Motivation

As the most intelligent creature in nature, human beings have strong problem-solving ability. Creative thinking plays a crucial role in it. There must exist many effective mechanisms in a creative thinking process. This motivates us to get inspirations from it:

1. From the individual level, the effective thinking modes in a creative thinking process could be designed into different optimizing operators.
2. From the social level, knowledge sharing and social interactions can greatly promote the generation of creative ideas. This inspires us to introduce collective behaviors into our model.
3. Due to the high intelligence of human beings, intelligent behaviors could be easily introduced into our algorithm. Meanwhile, since the thinkers do not need frequent communication with each other, there exists inherent parallelism. In conclusion, the intelligence and parallelism of our algorithm can be guaranteed.

To sum up, we can expect to propose an excellent nature-inspired algorithm through mimicking the human's creative thinking process.

1.4 Contribution

The main contribution of this paper is as follows.

1. A creativity-oriented optimization model (COOM) inspired by the creative thinking process is constructed. Since the real creative thinking process is very complex and will be impacted by various factors, it is unpractical to construct an accurate model of the creative thinking process for us. So we simplify the creative thinking process from a computer science perspective to solve the optimization problems.
2. Based on COOM, and also learning from the existing nature-inspired algorithms, specific mathematical operations for COOM to solve the continuous function optimization problems are designed. Then a creativity-oriented optimization algorithm (COOA) is presented. The properties of COOA, including convergence and parallelism, are analyzed theoretically.
3. To test the effectiveness of COOA, it is applied to solve the CEC-2013 real-parameter benchmark functions. Experimental results are compared with four state-of-the-art algorithms.

4. The parallelism of COOA is tested on a cluster.

As a newly proposed nature-inspired algorithm, COOA has many features in common with other similar algorithms. They all have a certain adaptive, self-learning ability and inherent parallelism. It makes COOA applicable to many of the same problems that can be solved by other nature-inspired algorithms, namely, high-dimension, nonlinearly, large-scale complex problems. Furthermore, compared with other similar algorithms, the individuals in COOA are much more intelligent and independent, so the interactions between them can be less. It will greatly improve the parallelism of COOA, especially when the communication costs among the individuals are high. Meanwhile, the expansibility of COOA is much better, as we can easily introduce some intelligent behaviors of human into it.

1.5 Organization

The structure of the rest of this paper is organized as follows. In Sect. 2, we define a global optimization problem. The biological model of COOM is proposed in Sect. 3, and its extended algorithm COOA is constructed. The mathematical model of COOM for real-parameter function optimization problems is designed in Sect. 4. In Sect. 5, we discuss the convergence and parallelism of COOA. The effectiveness and parallelism of COOA are tested through simulation in Sect. 6, and conclusions are drawn in Sect. 7.

2 Global optimization problem

Since the minimization and maximization problems can be transformed into each other easily, we define a global optimization problem as (1) and (2).

$$\arg\min_{\vec{X}\in S} f(\vec{X}) \tag{1}$$

$$\vec{X} = (x_1, x_2, \ldots, x_n) \in S \tag{2}$$

where $f(\vec{X})$ is the objective function of the optimization problem, whose variable number is n. We use $\vec{X}(x_1, x_2, \ldots, x_n)$ to represent the solution of the problem. S is the feasible region. The goal of global optimization is to find a solution $\vec{X}^*$, such that:

$$\forall \vec{X} \in S,\ f(\vec{X}^*) \le f(\vec{X}) \tag{3}$$

3 The biological model of COOM

As there exist many similarities between the creative thinking process and the optimization problem-solving process (as Table 1 shows), we try to mimic the creative thinking process to solve optimization problems. First, we simplified the creative thinking process and constructed a COOM based on the following three hypotheses.

Table 1 Optimization problem-solving vs creative thinking

Serial number	Optimization	Creative thinking
1	Solution	Idea
2	The best solution	The most creative idea
3	Optimization process	Creative thinking process
4	Control parameters	Restrictions
5	The utility of the optimization problem	The effect of the creative idea

Hypothesis 1 Creative thinking is the unity of various thinking activities, in which divergent thinking and convergent thinking dominates, and inspiration thinking also plays an important role.

Hypothesis 2 The idea generation process of creative thinking will be affected by the history experience, and a creative idea should not be the same with the history ideas.

Hypothesis 3 Creative thinking can be improved through social interaction.

The three hypotheses above are consistent with the current studies of creative thinking. Hypothesis 1 simplifies the model of creative thinking. Hypothesis 2 points out the role of historical experience in creative thinking. Hypothesis 3 is based on the previous researches that studied creative thinking from the social aspect, which provides a theoretical foundation for us to introduce the collective thinking into COOM. COOM is the foundation of COOA. It includes divergent thinking model, convergent thinking model, inspiration thinking model, collective thinking model and experience model. We will discuss their biological models in Sects. 3.1, 3.2, 3.3, 3.4 and 3.5. COOA is proposed in Sect. 3.6.

3.1 Divergent thinking model

Most of the current researches show that divergent thinking plays an indispensable role in the creative thinking process (Guilford 1967; Runco and Acar 2012). This thinking mode often leads to originality, which is the central feature of creativity (Runco and Acar 2012). In this phase, people always try to think along different directions to generate a plurality of ideas around the current idea, as shown in Fig. 1. According to the quality and complexity of divergent thinking, Guilford (1967) divided it into three levels: fluency, flexibility and originality. Fluency refers to the ability to generate a number of ideas in a short time. Flexibility is to generate diverse ideas by thinking from different directions. Originality is usually

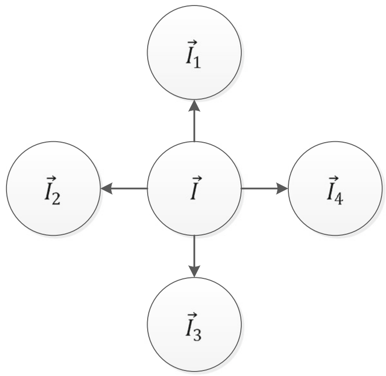

Fig. 1 Divergent thinking

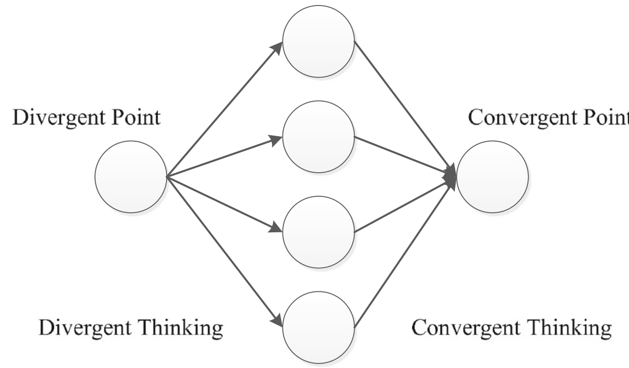

Fig. 2 Convergent thinking

defined in terms of novelty, and it should be valuable too. Originality is the highest level of divergent thinking.

The effect of divergent thinking in COOM is to exploit the local region of the current idea.

3.2 Convergent thinking model

Convergent thinking is another mode of thinking. In some ways, it is opposite to divergent thinking. As previously mentioned, the aim of divergent thinking is to generate a number of new ideas based on the current idea. The thinking direction of it is uncertain. However, in convergent thinking, it requires a certain thinking direction, which is consistent with our optimization goal most of the time. The target of this stage is to choose a most creative idea as the current idea. On the other hand, convergent thinking and divergent thinking is collaborative. Divergent thinking can provide numerous reference ideas for convergent thinking. Meanwhile, the existence of convergent thinking makes divergent thinking meaningful, as it guides the thinking process to converge to the expected goal. Otherwise, no matter how many ideas are generated in the divergent thinking process, they cannot formulate a creative result. To sum up, divergent thinking and convergent thinking are both opposite and unified, they should work together to promote the creative thinking process from occurring (Lee and Therriault 2013). The collaboration diagram of divergent thinking and convergent thinking is shown in Fig. 2.

To conclude, the effect of convergent thinking is to select a most creative idea from the ideas generated in divergent thinking, with the purpose of guiding the thinking process to converge to the optimal solution.

3.3 Inspiration thinking model

Inspiration thinking is also a basic thinking mode of human. We may often observe an "Aha! Moment" (Kounios and Beeman 2009) in real life. Inspiration thinking plays an important role in human's creative thinking process, it refers to a man's sudden perspicacity and understanding to a problem (Kounios and Beeman 2009). When we think about a problem for a long time, we often tend to generate inadvertently "inspiration". This is a reflection of the inspiration thinking. Although the inspiration thinking is not a logical way of thinking, it can produce really unimaginable results sometimes. This is an important feature of creative thinking. Inspiration thinking can significantly improve a person's creative thinking ability. In this paper, we assume that inspiration thinking will be triggered under certain conditions during the creative thinking process.

In COOM, the inspiration thinking can provide a strategy to get rid of local optimum.

3.4 Collective thinking model

In today's world, science and technology are developing rapidly. Social division becomes more and more refined. As the personal knowledge and ability are very limited, it is often difficult for a single person to solve a large complex problem. Simonton (2000) and some researchers believe that, in modern society, creativity becomes more cooperative. Current existing researches about creative thinking from the social aspect present that knowledge sharing and social interaction play important roles in fostering creativity (DeHaan 2011). So we introduce a collective thinking mode to improve the likelihood of generating a creative idea. In collective thinking, as the thinkers have different range of knowledge, they can learn from each other and generate a variety of ideas. Figure 3 is the schematic diagram of collective thinking.

Collective thinking is conductive to the generation of new creative ideas. At the same time, the speed to solve the optimization problem is accelerated through the mutual learning among the thinkers.

3.5 Experience model

"Working memory" is helpful for human to generate creative ideas (Balter 2010; Lee and Therriault 2013). During

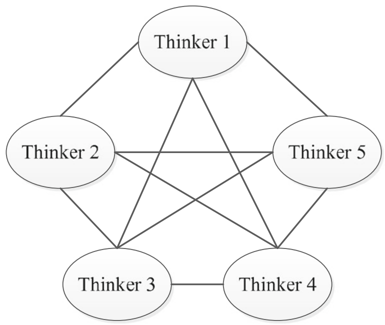

Fig. 3 Collective thinking

the process of problem-solving, human will generate a variety of ideas, and these ideas will be remembered. Through accumulation, an experience model is built, which can provide reference and guidance for the latter creative thinking process. A characteristic of creative thinking is to avoid generating a same idea as before. This is the function of the experience model in COOM.

3.6 COOA

Considering the relationships between COOM's sub-models, we get the flow diagram of COOA as Fig. 4 shows. For the optimization problem with a fitness function $f(\vec{X})$, we use $\vec{I}$ to represent the solution of the problem. Then the pseudo-code of COOA is shown in Algorithm 1.

In COOA, the thinkers have relatively high independence and intelligence. Each of these thinkers has a certain ability to solve problems alone. For many optimization problems, COOA does not require frequent collective thinking. Therefore, we can design a parallel structure for COOA as shown in Fig. 5. Each thinker thinks independently most of the time, after a certain period of independent thinking, they begin to think collectively. Correspondingly, the creativity-oriented optimization algorithm shown in Algorithm 1 can be expended to a parallel creativity-oriented optimization algorithm, as Algorithm 2 shows.

4 The mathematical model of COOM

4.1 Divergent thinking model solving

In this model, $\vec{I}$ is used to represent the current idea of a thinker. During the divergent thinking process, this thinker will randomly generate a plurality of "idea updating vectors" $\Delta\vec{I}$. Combining these vectors with his current idea, this thinker can get multiple new ideas. In this paper, the number of $\Delta\vec{I}$ generated by the thinker in a divergent thinking process is defined as a fixed value. $DNum$ is used to represent it. So the thinker can get his new ideas according to (4):

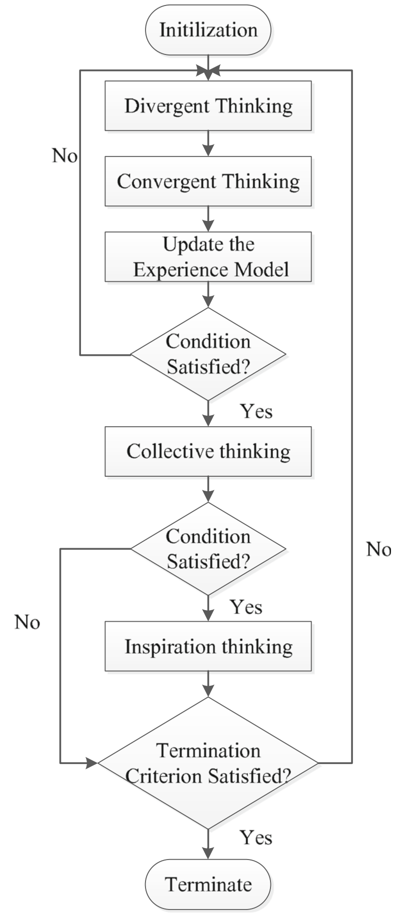

Fig. 4 Flowchart of COOA

Algorithm 1 Creativity-oriented optimization algorithm

Input: COOA's parameters and the fitness function $f(\vec{X})$.
Output: The best idea $\vec{I}$ generated by the thinkers.

```
Initialization:Each thinker randomly generates an idea.
repeat
  for each thinker[1, m] do
    Divergent Thinking( );
    Convergent Thinking( );
    Update the Experience Model;
    if the condition is satisfied then
      Inspiration thinking();
    end if
  end for
  if the condition is satisfied then
    Collective thinking ( )
  end if
until termination criterion is met.
```

$$\vec{I}_k^{\text{new}} = \vec{I} + \Delta\vec{I}_k, (k = 1, 2, \ldots, DNum) \tag{4}$$

The "idea updating vector" $\Delta\vec{I}[v_1, v_2, \ldots, v_n]$ is an n-dimensional vector, where $v_1, v_2, \ldots, v_n$ are Gaussian random variables whose mean is 0 and variance is σ^2. σ^2 is called the "creative variance". When σ^2 is larger, the new

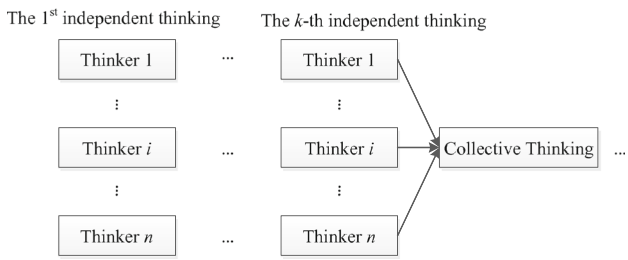

Fig. 5 The parallel architecture of COOA

Algorithm 2 Parallel Creativity-Oriented Optimization Algorithm

Input: COOA's parameters and the fitness function $f(\vec{X})$.
Output: The best idea $\vec{I}$ generated by the thinkers.
Initialization:Each thinker randomly generates an idea. ——parallel
repeat
  for each $thinker[1, m]$ **do**
    Divergent Thinking( ); ——parallel
    Convergent Thinking( ); ——parallel
    Update the Experience Model; ——parallel
    if the condition is satisfied **then**
      Inspiration thinking(); ——parallel
    end if
  end for
  if a certain period of independent thinking is over **then**
    Collective thinking( )
  end if
until termination criterion is met.

ideas can be quite different from the original idea, so it is conductive to jump out of local extreme points. When σ^2 is smaller, it can search near the current idea, which is helpful to improve the accuracy of the solution. To prompt the thinker to produce a variety of ideas, we assume the thinker has three adaptive "creative variances". σ_k^2 represents the kth "creative variance" of the thinker, where $k \in \{1, 2, 3\}$, $\sigma_{\min}^2$ and $\sigma_{\max}^2$ represent the lower bound and upper bound of all the thinkers' "creative variances", respectively. A predefined scaling factor $Sfactor \in (0, 1)$ is used to control the "creative variances". Each time, when a thinker gets a better idea after the divergent thinking, all his three "creative variances" will increase (i.e., $\sigma_k^2 \leftarrow \sigma_k^2 / Sfactor$). Otherwise, the three "creative variances" will remain unchanged until $INum$ consecutive divergent thinking of this thinker are invalid, then they will decrease (i.e., $\sigma_k^2 \leftarrow \sigma_k^2 * Sfactor$). Algorithm 3 is the "creative variance" updating algorithm of a thinker.

4.2 Convergent thinking model solving

Convergent thinking is to deal with the ideas generated in the process of divergent thinking. In this phase, the thinker will evaluate the ideas generated in divergent thinking at first, and then pick out the ideas whose fitness value is better than his current idea. At last, the thinker should choose the most

Algorithm 3 Adaptive creative variance algorithm

Input: The current σ_k^2, the last fitness value $f(\vec{I}_l)$ and the current fitness value $f(\vec{I})$, the number of consecutive invalid divergent thinking t.
Output: The updated "creative variance" σ_k^2.
if $f(\vec{I}) < f(\vec{I}_l)$ **then**
  $\sigma_k^2 = \sigma_k^2 / Sfactor$
  if $\sigma_k^2 > \sigma_{\max}^2$ **then**
    $\sigma_k^2 = \sigma_{\max}^2$
  end if
else
  $t = t + 1$.
end if
if $t == INum$ **then**
  $\sigma_k^2 = \sigma_k^2$*Sfactor
  if $\sigma_k^2 < \sigma_{min}^2$ **then**
    $\sigma_k^2 = \sigma_{min}^2$
  end if
  $t = 0$.
end if

creative idea to replace his current idea. In this model, "originality" is used to represent the "creativity" of the idea. As discussed in Harnad (2006), something creative must be both valuable and novel. So, the "originality" of a new idea is defined as follows: if the fitness value of the new idea is worse than that of the current idea, then the "originality" of this new idea is defined as zero. Otherwise, the "originality" of this new idea is defined as the Manhattan distance between this new idea and the current idea. $DisM(\vec{I}_a, \vec{I}_b)$ is used to represent the Manhattan distance between idea $\vec{I}_a$ and idea $\vec{I}_b$. In this paper, $f(\vec{I})$ and $O(\vec{I})$ represent the fitness value and the "originality" of idea $\vec{I}$, respectively. Then the mathematical model of convergent thinking can be depicted as (5) and (6).

$$O(\vec{I}_i) = \begin{cases} DisM(\vec{I}_i, \vec{I}_t) & \text{if} \quad f(\vec{I}_i) < f(\vec{I}_t) \\ 0 & \text{otherwise} \end{cases} \qquad i = 1, 2, \ldots, DNum \tag{5}$$

$$\vec{I}_{t+1} = \begin{cases} \vec{I}_k & \text{if} \quad O(\vec{I}_k) > 0 \\ \vec{I}_t & \text{otherwise} \end{cases} \qquad k = \underset{i \in \{1,2,\ldots,DNum\}}{\arg\max} \; O(\vec{I}_i) \tag{6}$$

4.3 Inspiration thinking model solving

Inspiration thinking is a non-logical way of thinking, while it is also an essential part in our creative thinking model. Meanwhile, the thinking direction of inspiration thinking is highly uncertain. Therefore, in this paper, the inspiration thinking is defined as such a process: when this thinking mode is triggered, the thinker will think divergently with a "creative variance" equals to $\sigma_{\max}^2$. A thinker's inspiration thinking mode will be triggered when his σ_k^2 $(k = 1, 2, 3)$ decrease.

4.4 Collective thinking model solving

In this section, we focus on the construction of the collective thinking's mathematical model, the number of thinkers is NT.

First, we define an "influence matrix" A, A is an $NT \times NT$ dimensional matrix. $\alpha_{i,j}$ is the element in the ith row and jth column of A. It represents the current influence of thinker i to thinker j. We use (7) and (8) to calculate $\alpha_{i,j}$:

$$M_{i,j} = \begin{cases} 0 & \text{if } i = j \\ \frac{f(\vec{I}_j)-f(\vec{I}_i)}{\left|f(\vec{I}_j)\right|} & \text{if } i \neq j \end{cases} \tag{7}$$

$$\alpha_{i,j} = \frac{M_{i,j}}{\sum_{i=1}^{n} |M_{i,j}|} \tag{8}$$

where $f(\vec{I}_i)$ is the fitness value of thinker i's current idea and $M_{i,j}$ is an intermediate variable. When $\alpha_{i,j} > 0$, it represents that the current idea of thinker i is better than the idea of thinker j, thinker i has a positive impact on thinker j. When $\alpha_{i,j} < 0$, it indicates that the current idea of thinker i is not as good as the idea of thinker j, thinker j will tend to refer to the opposite direction of the thinker i's current idea.

After calculating the "influence matrix", we will construct our collective thinking model. The steps of our collective thinking are described as follows:

4.4.1 Choose learning object

In actuality, a thinker may learn from a number of thinkers. The "learn" here means both imitating good ideas and also trying to avoid poor ideas. For simplicity, the number of the learning objects is limited to be three. We assume that, a thinker is inclined to choose those thinkers who have higher influence on him as the learning objects. So, in our COOA, all the thinkers use the roulette strategy to select their learning objects. The probability that thinker i selects thinker j as his learning object can be calculated as (9).

$$P_{i,j} = |\alpha_{i,j}| \tag{9}$$

4.4.2 Generate new ideas

For thinker i, suppose that the three learning objects chosen by him are $\vec{I}_a$, $\vec{I}_b$, $\vec{I}_c$, where $a, b, c \in \{1, 2, \ldots, \text{TN}\}$, and none of a, b, c equals to i. First, (10) is used to generate an idea $\vec{V}_i$. Then, combining $\vec{V}_i(v_{i,1}, v_{i,2}, \ldots, v_{i,D})$ with the current idea $\vec{I}_i(I_{i,1}, I_{i,2}, \ldots, I_{i,D})$, a new idea $\vec{U}_i(u_{i,1}, u_{i,2}, \ldots, u_{i,D})$ is generated. D is the dimension of the variable, and the generation rule of $\vec{U}_i$ is (11). $crand \in (0, 1)$ is an uniformly distributed random number, $r \in (0, 1)$ is a predefined parameter, and $rnbr(i) \in 1, 2, \ldots, D$ is randomly generated.

$$\vec{V}_i = \begin{cases} \vec{I}_a + \alpha_{a,i} * (\vec{I}_a - \vec{I}_i) + (\alpha_{b,i} - \alpha_{r,i}) * (\vec{I}_b - \vec{I}_r), \\ \quad \text{if} \quad \alpha_{a,i} > 0 \\ \vec{I}_i + \alpha_{a,i} * (\vec{I}_a - \vec{I}_i) + (\alpha_{b,i} - \alpha_{r,i}) * (\vec{I}_b - \vec{I}_r), \\ \quad \text{otherwise} \end{cases} \tag{10}$$

$$u_{i,k} = \begin{cases} v_{i,k} & \text{if } (crand \leq r) \text{ or } (k = rnbr(i)) \\ I_{i,k} & \text{if } (crand > r) \text{ or } (k \neq rnbr(i)) \end{cases}$$
$$k = 1, 2, \ldots, D \tag{11}$$

4.4.3 Idea determine

In this phrase, the thinker i will compare the new idea $\vec{U}_i$ with his current idea $\vec{I}_i$, and use (12) to update his current idea.

$$\vec{I}_i = \begin{cases} \vec{U}_i & \text{if} \quad f(\vec{U}_i) \leq f(\vec{I}_i) \\ \vec{I}_i & \text{otherwise} \end{cases} \tag{12}$$

4.5 Experience model solving

To implement the role of the experience model described in Sect. 3.5, we need to construct a memory table named "experience table". The function of this table is to record the ideas generated over a period of time. The length of the table could be different according to specific problems. The experience model constructed in this paper is shown as (13).

$$D(\vec{I}) = \omega_T \sum_{i=1}^{l} \exp\left(-\frac{\|\vec{I} - \vec{I}_i\|}{2\sigma_T^2}\right) \tag{13}$$

where ω_T is the weight coefficient, σ_T^2 is the density variance, $\vec{I}$ represents the current idea, $\vec{I}_i$ is the history ideas stored in the "experience table", and l is the number of history ideas stored in the "experience table". The value of ω_T is usually set as $1/l$. Each time, when the current idea is updated, it will replace the oldest idea in the "experience table". The value of $D(\vec{I})$ indicates the degree of similarities between the current idea and the history ideas. It reflects the level of local optimum. In this paper, the "experience model" is mainly used to evaluate the ideas generated in the divergent thinking and collective thinking. If the $D(\vec{I})$ of an idea is larger than a random variable $\eta \in (0, 1)$ under the uniform distribution, then this idea will not be accepted. The experience model can effectively reduce the computation time for some problems that have complex evaluation process.

5 Properties of COOA

In this section, we will discuss the convergence and parallelism of COOA. Section 5.1 elucidates the convergence property of COOA, and Sect. 5.2 shows the parallelism of it.

5.1 The convergence property of COOA

Exploration and exploitation are the two cornerstones of problem-solving by search (Črepinšek et al. 2013). In PSO, the adjustment equations of the particles' positions and velocities can be seen as the exploitation behavior, while the spread of particles plays a role as exploitation. For COOA, the corresponding relationships between the thinking models and the exploitation and exploration behaviors are listed as follows:

1. The divergent thinking and convergent thinking process perform the exploitation behavior.
2. The inspiration thinking and collective thinking process are equivalent to the exploration behavior.

In the following part of this section, we will analyze the convergence property of COOA.

For a minimization problem $\langle S, f\rangle$, where S is the solution space composed by the domain of the variables in the optimization problem, and f is the objective function of the problem. $\vec{X}_t$ is used to represent the solution obtained after the tth iteration, and $\vec{X}_{t+1}$ is the solution obtained after the $(t+1)$th iteration. $f(\vec{X}_t)$ is the value of the objective function when the solution is $\vec{X}_t$.

Hypothesis 4 The solution obtained after the $(t+1)$th iteration cannot be worse than that obtained after the tth iteration.

Hypothesis 5 Define the solution obtained after the tth iteration as $\vec{X}_t$, if it is not the best solution, then the probability that the algorithm can get a better solution in the $(t+1)$th iteration is $P_{t+1} \in (0, 1)$.

Theorem 1 *An algorithm is convergent if it satisfies Hypotheses 4 and 5.*

Proof According to Hypothesis 5, the probability that the algorithm can find a solution which is better than the current solution is $P^* = 1 - \prod_{k=1}^{n}(1 - P_k)$, then $\lim\limits_{n\to\infty} P^* = 1$. It means that the algorithm can always find a better solution until it converges. Combining with the Hypothesis 4, we can prove that this algorithm is convergent. □

Theorem 2 *COOA satisfies Hypothesis 4.*

Proof Define the current idea of thinker i as $\vec{I}_{i,t}$, the optimal idea generated is $\vec{O}_{i,t}$, then the idea chosen by thinker i is:

$$\vec{I}_{i,t+1} = \begin{cases} \vec{O}_{i,t} & \text{if } f(\vec{O}_{i,t}) \le f(\vec{I}_{i,t}) \\ \vec{I}_{i,t} & \text{if otherwise} \end{cases} \tag{14}$$

To sum up, COOA satisfies Hypothesis 4. □

Typically, a single solution space S can be divided into two areas: R_+ and R_-, where $R_+ \cup R_- = S$ and $R_+ \cap R_- = \varnothing$,

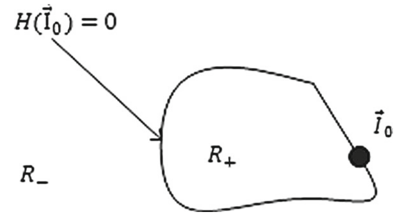

Fig. 6 Two-dimension single model solution space

as shown in Fig. 6. $\vec{I}_o$ is used to represent the history optimal idea, then R_+ is the collection of ideas that are better than $\vec{I}_o$, and R_- is the collection of ideas that are not as good as $\vec{I}_o$. This is to say, $\forall \vec{I}_a \in R_+$, $\vec{I}_b \in R_-$, it has $F(\vec{I}_a) < F(\vec{I}_o) < F(\vec{I}_b)$, and $H(\vec{I}_o) = 0$.

It shows that, if there exits at least one idea $\vec{I}_c \in R_+$, whose generated probability $P(\vec{I}_c) > 0$, then COOA will satisfy Hypothesis 5, and thus we can prove that COOA is convergent.

Lemma 1 *When $\vec{I}_c \subset R_+$, $\{\vec{\beta}_i\}$ is the collection of the ideas in the "experience table", and $\vec{I}_o$ is the expected optimal idea, if the experience model $D(\vec{I}_c)|_{\vec{\beta}_i} \in (0, 1)$, then (15) is founded.*

$$\|\vec{I}_c - \vec{I}_o\| > 2\sigma_T \ln(l\omega_T) \tag{15}$$

Proof According to the experience model shown in (13), $D(\vec{I}_c)|_{\vec{\beta}_i}$ can be rewritten as:

$$D(\vec{I})|_{\{\vec{\beta}_i\}} = \omega_T \sum_{i=1}^{l} \exp\left(-\frac{r_i^2}{2\sigma_T^2}\right), \text{ where } \quad r_i^2 = \|\vec{\beta}_i^2 - \vec{I}\| \tag{16}$$

As there does not exist any old idea in R_+, (16) can be rewritten as (17):

$$D(\vec{I})|_{\{\vec{\beta}_i\}} = \omega_T \sum_{i=1}^{l} \exp\left(-\frac{a_i d_{\min}^2}{2\sigma_T^2}\right), \text{ where}$$
$$d_{\min}^2 = \min_i \|\vec{\beta}_i - \vec{I}\| \text{ and } \quad a_i \subset [1, \infty]. \tag{17}$$

For a given value of $\vec{I}$, the maximum value of the experience model will be obtained when all the elements in $\{a_i\}$ equal to 1.

$$\max_{\{\vec{\beta}_i\}, \vec{I} \subset R_+} D(\vec{I})|_{\{\vec{\beta}_i\}} = \max_{\{a_i\}, \vec{I} \subset R_+} D(\vec{I})|_{\{\vec{\beta}_i\}}$$
$$= l\omega_T \exp\left(-\frac{d_{\min}^2}{2\sigma_T^2}\right) \tag{18}$$

As can be seen, the old ideas distribute in a hypersphere whose center is $\vec{I}$ and radius is $d_{\min}$. Therefore, a new idea $\vec{I} \in R_+$ as long as (19) is satisfied.

$$l\omega_T \exp\left(-\frac{a_i d_{\min}^2}{2\sigma_T^2}\right) < 1 \tag{19}$$

Then we can see that COOA will constantly approach to the optimal solution if: $\|\vec{\beta}_i - \vec{I}\| = d_{\min}^2 > 2\sigma_T^2 \ln(l\omega_T)$. □

Lemma 2 *If the solution space S is a bounded closed region in R^n, and the objective function is a continuous function on S, then COOA satisfies Hypothesis 5.*

Proof According to Fig. 6, the whole solution space can be divided into two parts. One is the neighborhood of the optimal point $\vec{X}^*$, S_0 is used to represent it, where $S_0 = \{\vec{X} \in \Omega \big| |f(\vec{X}) - f(\vec{X}^*)| < \varepsilon\}$, $\varepsilon > 0$. The other part $S_1 = S - S_0$. Then the current solution can be divided into two categories: (1) at least has one solution in S_0 and (2) all the solutions are in S_1.

If it belongs to the first category, we can know that there at least has one solution in S_0 in the subsequent solutions based on Theorem 2. Then according to the mechanism of our divergent thinking model, we can prove that COOA will eventually converge to the optimal solution.

If it belongs to the second category, as $f(\vec{X})$ is continuous on S, define $\vec{X}_0$ as the optimal point in the solution space, then there exists $d > 0$. If (20) is satisfied, we can get (21). Recording the collection constructed by $\vec{X}$ which satisfies (21) as Q, then $Q \subset S_0$.

$$\|\vec{X} - \vec{X}_0\|_\infty \leq d \tag{20}$$

$$\left| f(\vec{X}) - f(\vec{X}^*) \right| < \frac{\varepsilon}{2} \tag{21}$$

As

$$\begin{aligned} P\{(\vec{X} + \Delta\vec{X}) \in Q\} &= \prod_{i=1}^{n} P\{|x^i + \Delta x^i - x_0^i| \leq r\} \\ &= \prod_{i=1}^{n} P\{|x_0^i - x^i - d \leq \Delta x^i \leq x_0^i - x^i + d| \leq r\} \end{aligned} \tag{22}$$

where x_0^i and x^i represent the ith component of point x_0 and point x, respectively. Because Δx^i is a Gaussian random variable, $\Delta x^i \sim N(0, \sigma_i^2)$, so:

$$P\{(\vec{X} + \Delta\vec{X}) \in Q\} = \prod_{i=1}^{n} \int_{x_0^i - x^i - d}^{x_0^i - x^i + d} \frac{1}{\sqrt{2\pi}\sigma_i} \exp \frac{-y^2}{2\sigma_i^2} \,\mathrm{d}y \tag{23}$$

It can be seen that $0 < P\{(\vec{X}+\Delta\vec{X}) \in Q\} < 1$, this indicates the probability that the point gets belongs to Q is in the range of $(0, 1)$. The area S_0 is the collective of points that are better than the current point. Then we can prove that COOA satisfies Hypothesis 5 under the condition of this lemma. □

Theorem 3 *COOA has a closed bounded region in the solution space S, and it is global convergent when the object function is a continuous function on S.*

Proof According to Lemma 2, we can prove that the algorithm satisfies Hypothesis 5. Combining Theorems 2 and 1, we can prove that COOA has a closed bounded region in the solution space S, and it is convergent when the objective function is continuous on S. □

Theorem 4 *In COOA, collective thinking does not impact the global convergence property, but it can accelerate the convergence speed of COOA.*

Proof Since we use a greedy strategy in our collective thinking stage, every time after the collective thinking, the current ideas of the entire group will not be worse than before. For a continuous solution space, we can consider that the collective thinking process is a stochastic process to compress the solution space. It does not affect the global convergence of creative thinking, but it can speed up the convergence. □

5.2 The parallel property of COOA

For the parallel structure shown in Fig. 5, we use t_1 to represent the average time a thinker spends on an independent thinking, and t_2 is used to represent the average time a collective thinking takes. Assuming that, without affecting the accuracy of COOA, the thinkers can take a collective thinking after each one has thought k times independently. For a group with n thinkers, the time required for iterating m times is $T_s = m(nt_1 + t_2)$ when the parallel structure is not used as Fig. 5 shows. However, the corresponding time is $T_p = m(t_1 + t_2/k)$ when we use the parallel structure. Therefore, the theoretical speed-up ratio is:

$$S_p = \frac{T_s}{T_p} = \frac{nt_1 + t_2}{t_1 + t_2/k} \tag{24}$$

Thus, we can see that COOA's parallel effect is directly related with the number of thinkers, the time once independent thinking required, the time once collective thinking consumed, and the independent thinking times between two continuous collective thinking.

6 Experiment and analysis

In this section, COOA's effectiveness and parallelism will be tested through the function optimization problems. All the experiments were run on the HPCC of Lenovo Shenteng 1800, which has eight computation nodes and a console node. Each computation node is a high-performance server whose memory is 24 G and has two 2.4 GHz quad-core CPU. All the servers' operating system is Red Hat Enterprise Linux 7. The platform for COOA's effectiveness test is matlab R2009b. The specific experiment and its analysis are described in Sect. 6.1. Our parallel validation program is

Table 2 Parameter settings of COOA in this paper

Parameters	NP	$DNum$	$\sigma^2_{\min}$	$\sigma^2_{\max}$	$Sfactor$	$INum$	$IntervalNum$	l	σ^2_T	r
Value	12	6	1e−10	1e+4	0.95	3	4	50	1e−4	0.5

Table 3 Summary of results in 10D

Function	$f(x^*)$	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
1	0	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	1.92e−04±1.16e−03
2	0	6.48e+03±5.06e+03	**0.00e+00±0.00e+00**	4.15e+02±9.60e+02	**0.00e+00±0.00e+00**	9.93e+05±7.58e+05
3	0	4.59e+04±6.50e+04	**2.48e−01±1.23e+00**	4.96e+03±4.66e+04	5.69e−01±1.81e+00	2.13e+07±3.13e+07
4	0	1.33e+01±2.14e+01	**0.00e+00±0.00e+00**	6.50e−02±6.38e−01	**0.00e+00±0.00e+00**	8.80e+03±2.50e+03
5	0	1.24e−04±6.91e−05	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	2.94e−03±8.06e−03
6	0	**1.48e+00±3.49e+00**	**5.41e+00±4.76e+00**	6.18e+00±4.73e+00	6.74e+00±6.74e+00	1.52e+00±2.91e+00
7	0	1.07e+01±1.11e+00	**2.27e+00±4.45e+00**	5.63e+00±7.94e+00	5.45e+08±5.38e+09	3.27e+01±8.31e+00
8	0	**2.03e+01±9.30e−02**	**2.03e+01±1.03e-−01**	2.05e+01±1.06e−01	**2.03e+01±1.32e-−01**	2.04e+01±7.66e−02
9	0	**1.53e+00±9.67e−01**	2.29e+00±7.19e−01	2.37e+00±1.41e+00	1.50e+01±3.65e+00	4.98e+00±9.13e−01
10	0	1.37e−02±3.00e−02	1.42e−02±9.58e−03	1.25e−01±9.20e−02	**1.33e−02±1.39e−02**	1.47e+00±6.44e−01
11	0	1.36e+00±5.21e+00	**9.75e−02±2.96e−01**	2.48e+00±1.61e+00	2.31e+02±2.67e+02	1.97e+00±1.18e+00
12	0	1.17e+01±5.47e+00	**7.80e+00±4.10e+00**	1.06e+01±4.76e+00	3.49e+02±3.81e+02	2.64e+01±7.93e+00
13	0	**1.19e+01±8.61e+00**	1.21e+01±6.40e+00	2.01e+01±7.94e+00	2.94e+02±3.99e+02	3.56e+01±8.30e+00
14	0	4.19e+01±2.92e+01	**3.64e+00±4.39e+00**	1.25e+02±1.12e+02	1.88e+03±4.25e+02	5.27e+01±3.86e+01
15	0	**4.04e+02±2.76e+02**	7.36e+02±2.60e+02	7.26e+02±2.61e+02	1.80e+03±3.92e+02	8.92e+02±2.18e+02
16	0	**4.76e−02±4.14e−02**	4.04e−01±3.14e−01	5.43e−01±4.49e−01	4.51e−01±5.06e−01	1.18e+00±2.19e−01
17	0	2.07e+01±4.87e+00	**1.03e+01±1.55e−01**	1.32e+01±1.85e+00	9.55e+02±3.42e+02	1.60e+01±2.07e+00
18	0	**1.48e+01±6.53e+00**	2.46e+01±4.68e+00	1.93e+01±4.67e+00	9.01e+02±3.10e+02	5.41e+01±6.33e+00
19	0	6.97e−01±2.43e−01	**3.95e−01±1.25e−01**	6.44e−01±2.09e−01	1.19e+00±5.00e−01	7.65e−01±2.52e−01
20	0	**2.33e+00±5.92e−01**	2.65e+00±4.48e−01	2.87e+00±5.25e−01	4.68e+00±3.79e−01	3.50e+00±1.96e−01
21	0	**2.90e+02±1.22e+02**	3.83e+02±5.50e+01	3.98e+02±1.99e+01	3.68e+02±8.71e+01	3.73e+02±6.75e+01
22	0	4.57e+02±2.74e+02	**4.93e+01±5.33e+01**	1.33e+02±1.04e+02	2.32e+03±4.25e+02	7.27e+01±4.98e+01
23	0	**5.11e+02±2.89e+02**	5.78e+02±3.16e+02	8.82e+02±3.08e+02	2.25e+03±4.48e+02	1.15e+03±2.58e+02
24	0	**1.38e+02±3.07e+01**	2.02e+02±1.76e+01	2.02e+02±1.56e+01	3.83e+02±1.57e+02	2.01e+02±2.54e+01
25	0	**1.94e+02±3.40e+01**	2.02e+02±1.91e+00	2.00e+02±1.23e+01	2.62e+02±4.92e+01	2.12e+02±1.09e+01
26	0	**1.26e+02±2.25e+01**	1.26e+02±3.69e+01	1.47e+02±4.36e+01	2.57e+02±1.13e+02	1.58e+02±2.40e+01
27	0	3.14e+02±6.11e+01	3.37e+02±5.23e+01	**3.06e+02±2.76e+01**	4.23e+02±1.35e+02	4.27e+02±4.91e+01
28	0	**2.00e+02±1.02e+02**	3.17e+02±6.87e+01	3.07e+02±5.78e+01	1.21e+03±1.21e+03	3.01e+02±1.28e+02

written in C for MPI, and the MPI version is MPICH2. We will discuss the parallelism test in Sect. 6.2.

6.1 The effectiveness test and analysis of COOA

The benchmark functions used in this paper are the 28 CEC-2013 real-parameter functions. According to the performance measures described in the protocol of CEC-2013 testbed (Liang et al. 2013), COOA was executed on each function for 51 independent runs at three problem dimensionalities (10D, 30D and 50D). The max number of function evaluations was set as $10{,}000 \times D$, where D is the dimension of the function. At each time COOA is run on the functions, it terminates when the function evaluations reaching the max number or the error value $(f_i(\vec{X}) - f_i(\vec{X}^*))$ is smaller than 10^{-8}. Error values and standard deviations smaller than 10^{-8} are taken as zero. The experimental results are compared with four state-of-the-art algorithms (SMADE, MDE-pBX, CMAES and CCPSO2) listed in Caraffini et al. (2013).

In this paper, the parameter settings of COOA are listed in Table 2. It is a comparatively good parameter combination that have been tuned by us.

6.1.1 Experimental results and analysis

Tables 3, 4 and 5 summarize the results obtained for COOA with 10, 30 and 50 dimensions, respectively. Corresponding results of the four comparison algorithms come from

Table 4 Summary of results in 30D

Function	$f(x^*)$	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
1	0	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**
2	0	1.03e+05±1.68e+05	**0.00e+00±0.00e+00**	9.56e+04±6.16e+04	**0.00e+00±0.00e+00**	9.95e+05±5.24e+05
3	0	7.84e+06±8.38e+06	9.82e+03±4.94e+04	1.80e+07±3.12e+07	**1.53e+01±1.23e+02**	5.59e+08±5.57e+08
4	0	6.67e+02±1.24e+03	**0.00e+00±0.00e+00**	1.32e+01±5.46e+01	**0.00e+00±0.00e+00**	5.57e+04±2.06e+04
5	0	6.67e−04±2.57e−04	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	2.62e−08±6.05e-08
6	0	1.31e+01±1.19e+01	**2.67e+00±7.85e+00**	1.99e+01±2.22e+01	3.05e+00±8.33e+00	2.19e+01±2.27e+01
7	0	4.19e+01±1.43e+01	**3.25e+01±1.61e+01**	5.70e+01±1.77e+01	9.83e+03±6.44e+04	1.15e+02±3.11e+01
8	0	**2.09e+01±1.04e−03**	2.10e+01±4.80e−02	2.11e+01±5.94e−02	2.09e+01±6.66e−02	2.10e+01±4.60e−02
9	0	**1.60e+01±1.88e+00**	2.23e+01±3.57e+00	2.22e+01±4.80e+00	4.45e+01±6.99e+00	2.84e+01±2.08e+00
10	0	4.56e−02±6.08e−02	1.84e−02±1.34e−02	1.64e−01±1.20e−01	**1.73e−02±1.25e−02**	1.48e−01±6.90e−02
11	0	1.01e+01±2.14e+01	1.09e+01±4.18e+00	4.62e+01±1.44e+01	9.47e+01±2.36e+02	**1.19e−01±2.90e−01**
12	0	**5.36e+01±1.91e+01**	5.72e+01±1.70e+01	6.94e+01±2.00e+01	7.11e+02±9.66e+02	2.12e+02±5.24e+01
13	0	**1.13e+02±3.46e+01**	1.28e+02±3.50e+01	1.49e+02±3.66e+01	1.64e+03±1.66e+03	2.44e+02±3.44e+01
14	0	2.03e+03±5.62e+02	1.33e+02±1.27e+02	1.17e+03±3.95e+02	5.21e+03±7.37e+02	**4.48e+00±2.87e+00**
15	0	**2.65e+03±4.90e+02**	4.10e+03±8.47e+02	3.95e+03±6.57e+02	5.37e+03±6.73e+02	3.85e+03±4.52e+02
16	0	1.29e−01±1.22e−01	1.31e−01±7.57e−02	1.25e+00±6.19e−01	**1.26e−01±8.87e−02**	2.16e+00±3.76e−01
17	0	1.37e+02±1.95e+01	3.48e+01±1.52e+00	7.05e+01±1.24e+01	3.95e+03±7.89e+02	**3.07e+01±3.03e+00**
18	0	1.15e+02±2.91e+01	8.33e+01±2.06e+01	**8.26e+01±1.89e+01**	4.23e+03±8.31e+02	2.31e+02±5.43e+01
19	0	3.93e+00±1.25e+00	2.55e+00±5.18e−01	9.54e+00±5.54e+00	3.66e+00±9.52e−01	**7.77e−01±1.58e−01**
20	0	1.06e+01±1.77e−01	**1.05e+01±8.07e−01**	1.07e+01±7.75e−01	1.50e+01±6.45e−02	1.35e+01±5.50e−01
21	0	2.92e+02±7.30e+01	3.27e+02±8.65e+01	3.40e+02±7.62e+01	3.05e+02±9.01e+01	**2.37e+02±6.71e+01**
22	0	3.05e+03±3.11e+02	1.79e+02±4.50e+01	1.17e+03±4.92e+02	6.97e+03±1.06e+03	**9.87e+01±6.70e+01**
23	0	**2.96e+03±5.41e+02**	4.22e+03±8.74e+02	4.70e+03±7.70e+02	6.76e+03±6.82e+02	4.99e+03±6.31e+02
24	0	**2.02e+02±7.10e+00**	2.32e+02±2.57e+01	2.31e+02±8.60e+00	8.19e+02±6.15e+02	2.80e+02±6.34e+00
25	0	**2.40e+02±6.46e+01**	2.78e+02±9.90e+00	2.79e+02±1.38e+01	3.46e+02±1.45e+02	2.98e+02±6.94e+00
26	0	**2.00e+02±5.44e−02**	2.15e+02±5.25e+01	2.26e+02±5.15e+01	5.51e+02±5.14e+02	2.00e+02±6.76e−01
27	0	**3.74e+02±5.91e+01**	6.47e+02±1.37e+02	6.50e+02±1.04e+02	8.53e+02±2.42e+02	1.04e+03±8.09e+01
28	0	**3.00e+02±2.02e−05**	3.88e+02±3.23e+02	3.09e+02±1.50e+02	1.96e+03±3.40e+03	4.35e+02±5.10e+02

Caraffini et al. (2013), and they are listed in the three tables too. The best results of the five algorithms are shown in bold. To analyze the experimental results, the nonparametric statistical technology is used, which has become a widespread technique in computational intelligence in recent years (Derrac et al. 2011). It is encouraged to use this technique to analyze the results obtained by a new proposed algorithm for continuous optimization problems (García et al. 2009). In this section, we use the Friedman test and Wilcoxon's test to analyze the performance of COOA among the five algorithms, and the significance is checked at a level of 0.05. All the p values in this paper were computed using SPSS (the version is 19.0.0). Figures 7, 8 and 9 depict the Friedman ranks of the five algorithms on each of the 28 functions with 10D, 30D and 50D, respectively. A smaller rank indicates a better performance of the corresponding algorithm on that problem. Each figure consists of three sub-figures, corresponding to the three categories of functions in the 28 CEC-2013 benchmark functions: uni-modal functions (f_1–f_5), multi-modal functions (f_6–f_{20}) and composition functions (f_{21}–f_{28}).

At first, the performance of COOA on the functions of different dimensions is discussed. Table 6 lists the average ranks of the five algorithms for $D = 10$, 30 and 50, respectively. The best ranks are shown in bold. As can be seen from the table, the performance of the five algorithms can be sorted by the average ranks into the flowing order: COOA, SMADE, MDE-pBX, CCPSO2 and CMAES, both on the 10-dimensional functions and the 50-dimensional functions. The average ranks of SMADE and COOA are equal on the 30-dimensional functions, and they are better than the other three algorithms. Table 7 depicts the p values of applying Wilcoxon's test between COOA and other four algorithms on the functions with different dimensions. R^+ represents the sum ranks for the problems in which COOA outperformed the comparison algorithm. The p values under the significance level are shown in bold. From the results shown in the

Table 5 Summary of results in 50D

Function	$f(x^*)$	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
1	0	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**
2	0	3.85e+05±1.34e+05	**0.00e+00±0.00e+00**	4.37e+05±1.64e+05	**0.00e+00±0.00e+00**	1.85e+06±9.33e+05
3	0	4.31e+07±2.84e+07	3.81e+05±1.35e+06	8.45e+07±1.46e+08	**9.64e+02±4.49e+03**	1.98e+09±2.09e+09
4	0	3.96e+02±3.20e+02	**0.00e+00±0.00e+00**	3.05e+01±6.62e+01	**0.00e+00±0.00e+00**	1.00e+05±3.57e+04
5	0	8.72e−04±3.73e−04	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**	**0.00e+00±0.00e+00**
6	0	4.34e+01±1.74e+01	**4.30e+01±6.28e+00**	5.48e+01±2.12e+01	4.32e+01±7.10e+00	4.35e+01±1.36e+01
7	0	7.60e+01±9.45e+00	4.32e+01±1.66e+01	6.59e+01±1.06e+01	**4.19e+01±1.70e+01**	1.37e+02±2.31e+01
8	0	**2.10e+01±5.46e−02**	2.11e+01±3.85e−02	2.12e+01±4.44e−02	2.11e+01±9.75e−02	2.11e+01±4.49e−02
9	0	**3.45e+01±2.65e+00**	4.36e+01±4.06e+00	4.32e+01±7.71e+00	7.66e+01±7.77e+00	5.79e+01±4.39e+00
10	0	2.32e−02±7.14e−02	2.47e−02±1.48e−02	1.34e−01±1.23e−01	**2.24e−02±1.49e−02**	1.24e−01±4.62e−02
11	0	2.03e+01±3.89e+01	4.81e+01±1.49e+01	1.24e+02±2.87e+01	2.19e+02±4.56e+02	**4.31e−01±5.74e−01**
12	0	**1.30e+02±3.57e+01**	1.57e+02±4.52e+01	1.58e+02±3.25e+01	2.25e+03±1.37e+03	4.46e+02±7.92e+01
13	0	**3.04e+02±3.92e+01**	3.35e+02±5.63e+01	3.24e+02±4.74e+01	3.36e+03±1.09e+03	5.49e+02±6.67e+01
14	0	5.07e+03±6.96e+02	3.41e+02±2.05e+02	2.65e+03±8.86e+02	8.82e+03±1.04e+03	**6.45e+00±3.20e+00**
15	0	**5.72e+03±5.81e+02**	8.54e+03±9.77e+02	7.46e+03±7.95e+02	9.09e+03±9.43e+02	7.95e+03±7.11e+02
16	0	8.72e−02±1.77e−01	8.96e−02±4.24e−02	1.75e+00±7.40e−01	**8.01e−02±4.72e−02**	2.39e+00±5.90e−01
17	0	2.27e+02±4.35e+01	6.57e+01±5.27e+00	1.75e+02±3.72e+01	6.97e+03±1.07e+03	**5.14e+01±2.84e−01**
18	0	2.36e+02±4.76e+01	1.93e+02±3.46e+01	**1.85e+02±3.40e+01**	7.08e+03±9.14e+02	4.94e+02±1.08e+02
19	0	9.67e+00±3.45e+00	5.43e+00±1.07e+00	4.25e+01±2.66e+01	6.32e+00±1.18e+00	**1.40e+00±2.19e−01**
20	0	2.08e+01±2.28e−01	**1.92e+01±8.86e−01**	2.00e+01±9.04e−01	2.50e+01±9.73e−02	2.28e+01±7.85e−01
21	0	**3.16e+02±3.23e+02**	8.46e+02±3.43e+02	9.22e+02±3.06e+02	8.12e+02±3.73e+02	3.27e+02±2.64e+02
22	0	6.03e+03±1.08e+03	3.39e+02±2.24e+02	3.09e+03±9.98e+02	1.19e+04±1.26e+03	**7.58e+01±8.58e+01**
23	0	**4.75e+03±7.78e+02**	9.89e+03±1.90e+03	8.88e+03±1.20e+03	1.18e+04±8.52e+02	1.05e+04±1.11e+03
24	0	**2.02e+02±9.46e+00**	3.00e+02±1.20e+01	2.87e+02±1.47e+01	1.64e+03±1.05e+03	3.56e+02±9.89e+00
25	0	**2.75e+02±1.01e+01**	3.68e+02±1.36e+01	3.69e+02±1.78e+01	4.94e+02±1.88e+02	3.96e+02±1.19e+01
26	0	2.22e+02±5.10e+01	2.91e+02±9.70e+01	3.50e+02±7.93e+01	6.04e+02±7.11e+02	**2.09e+02±3.92e+01**
27	0	**7.75e+02±5.47e+01**	1.18e+03±1.67e+02	1.24e+03±1.56e+02	1.28e+03±2.51e+02	1.79e+03±8.78e+01
28	0	**4.00e+02±2.65e+02**	1.07e+03±1.27e+03	4.33e+02±3.26e+02	3.27e+03±5.60e+03	6.33e+02±8.95e+02

table, we can draw the following conclusions: COOA outperforms MDE-pBX, CMAES and CCPSO2 significantly, and it is comparable with SMADE.

And then, the average performance of the five algorithms on the benchmark functions over all dimensions is analyzed. Table 8 depicts the average Friedman ranks per function over all dimensions, and the best results are shown in bold. The p value computed through Friedman test is 3.46e−05, which strongly suggests the existence of significant difference among the five algorithms. And the performance of the five algorithms can be sorted by the average ranks into the following order: COOA, SMADE, MDE-pBX, CCPSO2 and CMAES. The results of applying Wilcoxon's test between COOA and other four algorithms are listed in Table 9. According to the significance level we chose, COOA is significantly better than MDE-pBX, CMAES and CCPSO2, and it is comparable with SMADE.

At last, we focus on the performance of COOA on different categories of functions (uni-modal functions, multi-modal functions and composition functions). Through analyzing the results listed in Table 8, the average Friedman ranks on different categories of functions are shown in Table 10. The results indicate that COOA performs differently on different categories of functions. The performance of COOA is only better than CCPSO on the uni-modal functions, while on the multi-modal functions, it is just not as good as SMADE. Furthermore, COOA performs best on the composition functions. In fact, referring to Table 8, we can find that the average performance of COOA is the best on all the composition functions expect f_{22}. Wilcoxon's test between COOA and other four algorithms on different categories of functions are also taken, results are listed in Table 11, and the p values smaller than 0.05 are shown in bold. COOA performs significantly better than MDE-pBX, CMAES and CCPSO2 on the multi-modal functions; it performs significantly better than MDE-pBX and CMAES on the composition functions. In other cases, there do not exist significance under the current significance level.

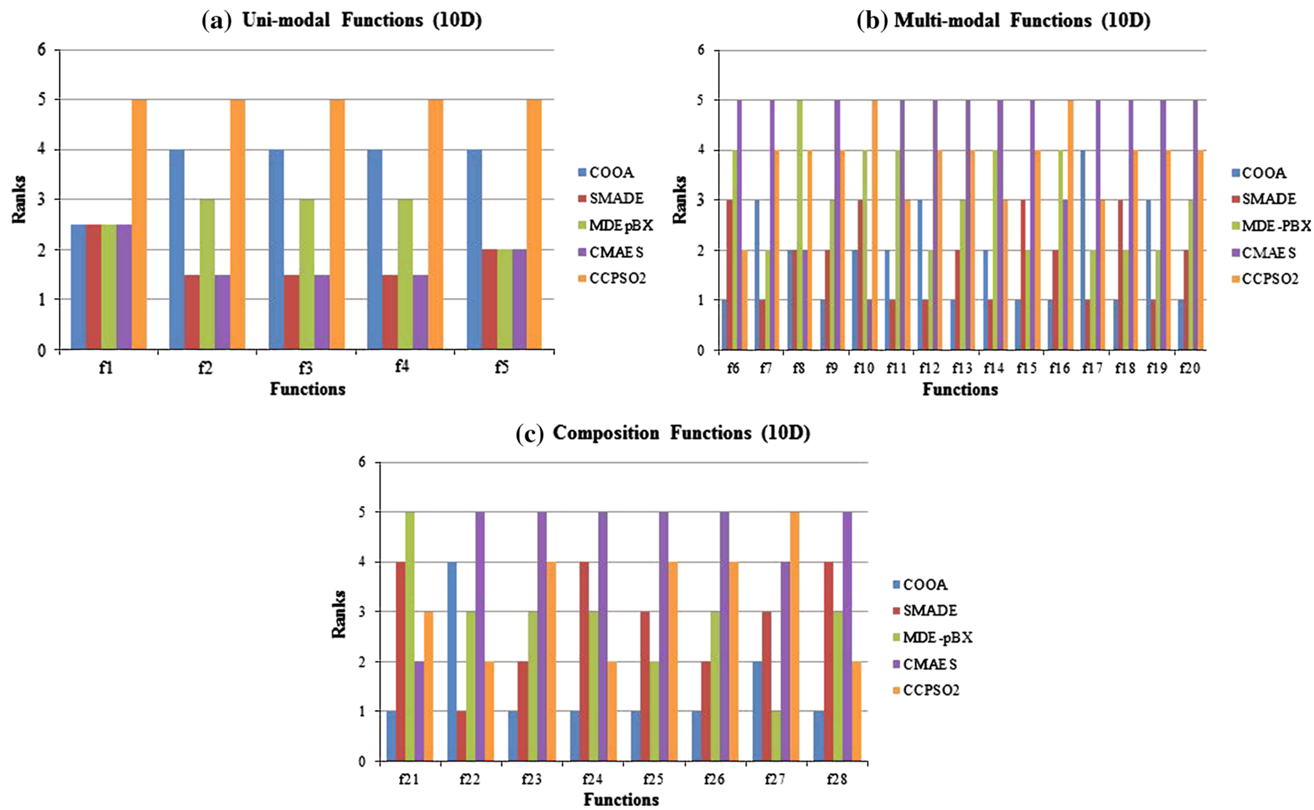

Fig. 7 Summary of Friedman ranks in 10D

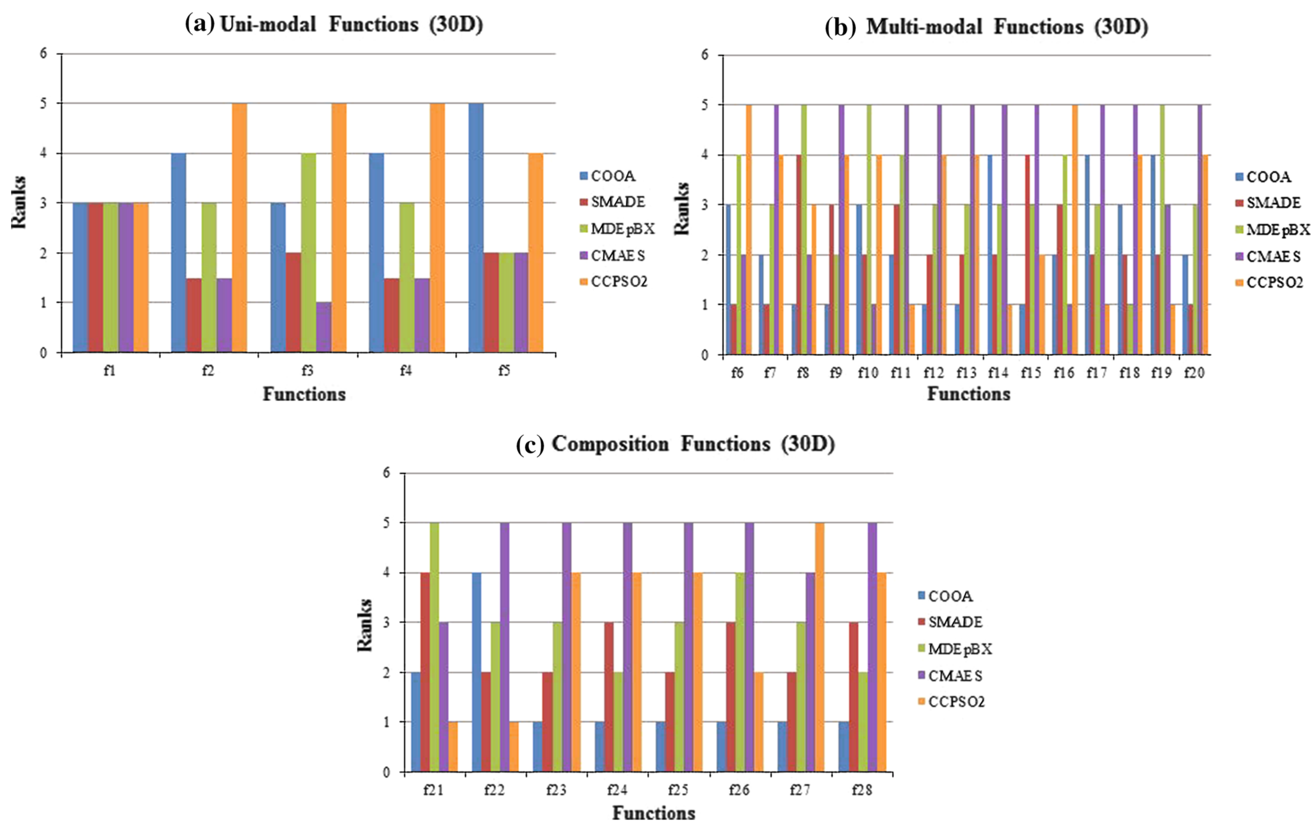

Fig. 8 Summary of Friedman ranks in 30D

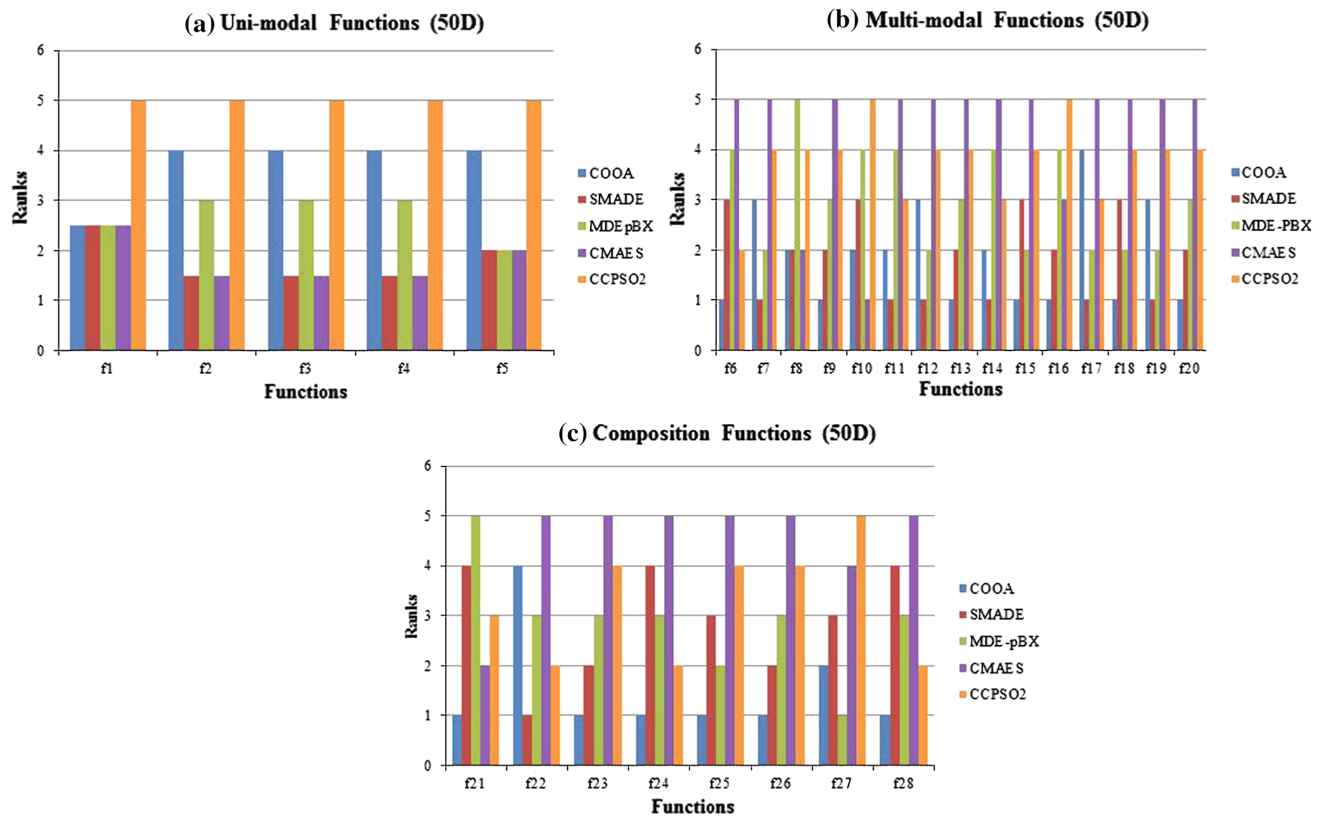

Fig. 9 Summary of Friedman ranks in 50D

Table 6 Average ranks for $D = 10$, 30 and 50, respectively

Dimensions	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
$D = 10$	**2.09**	2.14	2.95	3.96	3.86
$D = 30$	**2.32**	**2.32**	3.25	3.75	3.36
$D = 50$	**2.36**	2.45	3.31	3.73	3.34

From the above discussions, we can conclude that our new proposed algorithm has an excellent performance on the 28 CEC-2013 benchmark functions. Moreover, we should realize that the benchmark results should always be taken with a grain of salt. Different parameter values in the optimization algorithms may result in significant changes in their performance. Considering that we did not make too much special effort to tune the parameters of our algorithm, the performance of COOA in function optimization has great potential for ascension. In the following section, we will make an analysis of the parameters of COOA.

6.1.2 Analysis of parameters

There are ten parameters in COOA, and the roles of them will be discussed in this section. And then, we will give out a suitable parameter combination for solving the CEC-2013 benchmark functions.

Table 7 Wilcoxon's test between COOA and other algorithms for $D = 10$, 30 and 50, respectively

COOA vs	SMADE	MDE-pBX	CMAES	CCPSO2
$D = 10$				
R^+	202	309	333	384
R_-	204	97	73	22
p value	0.969	**0.009**	**0.001**	**2.732e−5**
$D = 30$				
R^+	199	314	318	295
R_-	207	92	88	111
p value	0.922	**0.007**	**0.006**	**0.025**
$D = 50$				
R^+	204	299	314	227
R_-	202	107	92	119
p value	0.981	**0.018**	**0.007**	**0.041**

NT: The number of thinkers NT is a very important parameter of COOA. A larger NT can achieve a better problem-solving ability, but it will also increase the computation complexity. A proper NT need to be chosen according to the specific problem. Compared with the individuals in most of the commonly used population-based algorithms, the thinkers in COOA are much more intelligent. As a result, COOA can get

Table 8 Average ranks per function over all dimensions

Function	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
f_1	**2.83**	**2.83**	**2.83**	**2.83**	3.67
f_2	3.67	**1.50**	3.33	**1.50**	5.00
f_3	3.33	1.67	3.67	**1.33**	5.00
f_4	4.00	**1.50**	3.00	**1.50**	5.00
f_5	4.67	**2.17**	**2.17**	**2.17**	3.83
f_6	2.33	**1.67**	4.33	3.00	3.67
f_7	3.00	**1.33**	2.67	3.67	4.33
f_8	**1.33**	2.67	4.33	3.00	3.67
f_9	**1.00**	2.67	2.33	5.00	4.00
f_{10}	2.33	2.67	4.67	**1.00**	4.33
f_{11}	2.00	2.33	4.00	5.00	**1.67**
f_{12}	**1.67**	**1.67**	2.67	5.00	4.00
f_{13}	**1.00**	2.33	2.67	5.00	4.00
f_{14}	3.33	**1.67**	3.33	5.00	**1.67**
f_{15}	**1.00**	3.67	2.33	5.00	3.00
f_{16}	**1.67**	2.67	4.00	**1.67**	5.00
f_{17}	4.00	**1.67**	2.67	5.00	**1.67**
f_{18}	2.33	2.33	**1.33**	5.00	4.00
f_{19}	3.67	**1.67**	4.00	3.67	2.00
f_{20}	2.00	**1.33**	2.67	5.00	4.00
f_{21}	**1.33**	4.00	5.00	2.67	2.00
f_{22}	4.00	1.67	3.00	5.00	**1.33**
f_{23}	**1.00**	2.33	2.67	5.00	4.00
f_{24}	**1.00**	3.33	2.33	5.00	3.33
f_{25}	**1.00**	2.33	2.67	5.00	4.00
f_{26}	**1.33**	2.67	3.67	5.00	2.33
f_{27}	**1.33**	2.33	2.33	4.00	5.00
f_{28}	**1.00**	3.67	2.33	5.00	3.00
Mean	**2.26**	2.30	3.11	3.82	3.52
Std	1.18	0.74	0.87	1.45	1.15

Table 9 Wilcoxon's test between COOA and other algorithms over all dimensions

COOA vs	SMADE	MDE-pBX	CMAES	CCPSO2
R^+	201	313	319	339
R_-	205	93	87	67
p values	0.957	**0.005**	**0.002**	**0.002**

a comparable performance as other similar algorithms while just using a much more smaller size of population.

$DNum$: $DNum$ is the amount of ideas generated by each thinker in a divergent thinking process. A larger $DNum$ will lead to a more effective divergent thinking, it means that the thinker will have a greater probability to think out a better idea after this divergent thinking. However, it also causes too many unnecessary idea evaluations, which is very time consuming in some kinds of problems.

Table 10 Average ranks on uni-modal functions, multi-modal functions, composition functions, respectively

The function category	COOA	SMADE	MDE-pBX	CMAES	CCPSO2
Uni-modal	3.70	1.93	3.00	**1.87**	4.50
Multi-modal	2.18	**2.16**	3.20	4.07	3.40
Composition	**1.50**	2.79	3.00	4.58	3.12

Table 11 Wilcoxon's test between COOA and other algorithms on different categories of functions

COOA vs	SMADE	MDE-pBX	CMAES	CCPSO2
Uni-modal functions				
R^+	3	4	3	2
R_-	12	11	12	13
p value	0.066	0.197	0.066	0.104
Multi-modal functions				
R^+	31	100	102	99
R_-	5	20	18	21
p value	0.889	**0.013**	**0.004**	**0.027**
Composition functions				
R^+	31	35	36	31
R_-	5	1	0	5
p value	0.079	**0.020**	**0.011**	0.068

$\sigma_{\min}^2, \sigma_{\max}^2, \sigma_{i,k}^2$: Among them, $\sigma_{\min}^2$ is the lower bound of the "creative variance", while $\sigma_{\max}^2$ is the upper bound of the "creative variance". In general, we expect a thinker can have a broad mind in the early thinking process. Accordingly, when solving an optimization problem, the individuals in the algorithm should search through the whole space to get a better solution in the early time. The initials $\sigma_{i,1}^2, \sigma_{i,2}^2$ are set equal to $\sigma_{\min}^2$ and $\sigma_{\max}^2$, respectively. And the initial $\sigma_{i,3}^2$ is set to $(\sigma_{i,1}^2 + \sigma_{i,2}^2)/2$.

$Sfactor$: The parameter $Sfactor$ is the scale factor of the "creative variance", which is used to control the adaptive "creative variance" in this paper. To fully search the solution space, the "creative variance" should change gently. From this aspect, the $Sfactor$ should be set to a comparatively large value in the range of (0, 1). From another aspect, if the $Sfactor$ is too large, the number of invalid searches may increase. Therefore, we should find a balance.

$INum$: This is another parameter to adjust the "creative variance" of each thinker. For a thinker, if his divergent thinking cannot help him generate a better idea $INum$ times continually, then his "creative variance" will decrease. The parameter $INum$ can control the updating speed of a thinker's "creative variance". A larger $INum$ will improve the optimization accuracy of COOA, but it will slow down the convergence speed. However, if $INum$ is too small, COOA may converge prematurely.

$IntervalNum$: Each time, when the thinkers have thought independently $IntervalNum$ times, they will take a collective thinking. A larger value of $IntervalNum$ will improve the parallelism of COOA, while a smaller value of $IntervalNum$ will increase the optimization speed of COOA. We should set $IntervalNum$ according to the specific problems.

l: l represents the memory capacity of the "experience table". According to the "experience model" discussed in Sect. 4.5, we can see that, the more history ideas a thinker remembered, the better he will be guided by the "experience model". But if l is too large, it will increase the amount of calculation, and we also need to consider the available memory space.

σ_T^2: It is another parameter of the experience model. In accordance with (13), a smaller σ_T^2 can get a better search accuracy, while a larger σ_T^2 can greatly reduce the number of noneffective idea evaluations.

r: This parameter is similar to the crossover rate in DE (Das and Suganthan 2011). It should be set according to the interaction of decision variables in the given problem. When more decision variables are interacted with each other, a larger r will be a better choice. But a smaller r could be more effective when fewer decision variables are interrelated.

According to the analysis of parameters above, the general range of COOA's parameters advised by us for the CEC-2013 real-parameter benchmark problems is listed in Table 12. Since COOA has so many parameters, and the best parameter combinations may always be different on different functions, it will be very time consuming to find the best parameter combinations for each function. So, in this paper, we used a simple method to tune the parameters roughly. Considering the number of benchmark functions and parameters in our test, we only present an instance here. The tuned parameter is $Sfactor$, and the object function is f_9(10D). In this tuning experiment, the values of $Sfactor$ were changed as 0.80, 0.85, 0.90, 0.95 and 0.99, the values of other parameters were set as the same in Table 2. The corresponding test was run 51 times for each value of $Sfactor$. The mean error values and standard deviations are listed in Table 13. Figure 10 depicts the profile of mean error values with different values of $Sfactor$. We can see from the results that the effect of $Sfactor$ is consistent with our previous analysis of this parameter. Different values of $Sfactor$ can get significantly different results. In this paper, other parameters of COOA were all tuned through this simple method. Our future work may use the fuzzy logic technology to tune the parameters.

Table 12 The general range of COOA's parameters for the CEC-2013 benchmark problems

Parameters	Range
NT	[6,50]
$DNum$	[3,20]
$\sigma^2_{\min}$	[1e−12,1e−2]
$\sigma^2_{\max}$	[1e+2,1e+4]
$Sfactor$	(0.5,1)
$INum$	(1,30)
$IntervalNum$	(1,10)
l	(30,200)
σ_T^2	(1e−8,0.1)
r	(0.1,0.9)

6.2 COOA's parallelism test and analysis

In this section, COOA's parallelism will be tested and analyzed. Algorithm 2 is used, and the parameters are set as the same listed in Table 2. As the number of thinkers is set to 12 in this paper, theoretically speaking, when the processors used are more than 12, the parallel effect will not improve. Therefore, we only test the parallel effect of COOA when the

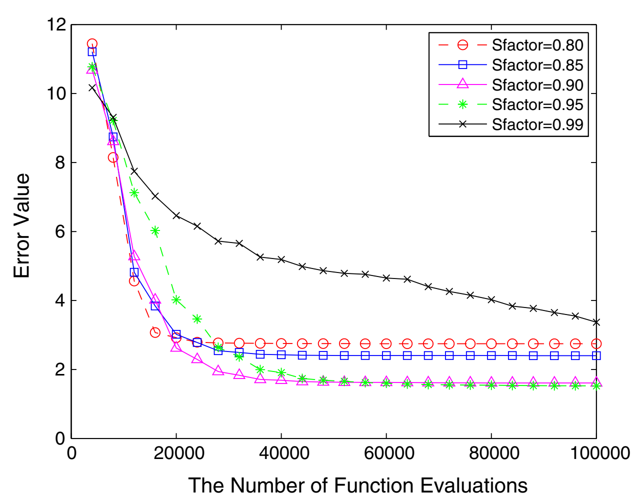

Fig. 10 Effect of parameter $Sfactor$ on f_9(10D)

Table 13 Experimental results of COOA on f_9(10D) under varying values of $Sfactor$

Function (f_9)	$Sfactor = 0.80$	$Sfactor = 0.85$	$Sfactor = 0.90$	$Sfactor = 0.95$	$Sfactor = 0.99$
Mean	2.75e+00	2.40e+00	1.61e+00	1.53e+00	3.28e+00
Std	1.09e+00	7.83e−01	9.80e−01	9.67e−01	8.90e−01

Table 14 Results of parallel test

Scale	1 processor	2 processors		4 processors		8 processors		12 processors	
Iterations	Time (s)	Time (s)	Speed-up ratio	Time (s)	Speed-up ratio	Time (s)	Speed-up ratio	Time (s)	Speed-up ratio
100	1.6226	0.8873	**1.8325**	0.4547	**3.5685**	0.2330	**6.9639**	0.2016	**8.0486**
200	3.2023	1.7261	**1.8552**	0.8873	**3.6090**	0.4635	**6.9089**	0.4037	**7.9341**
400	7.0325	3.6810	**1.9104**	1.9400	**3.6250**	0.9714	**7.2395**	0.7916	**8.8906**
800	13.8937	7.4635	**1.8616**	3.8524	**3.6065**	2.0015	**6.9416**	1.3769	**10.0905**
1600	27.1483	15.0479	**1.8041**	7.3816	**3.6778**	4.0631	**6.6817**	2.6265	**10.3363**
3200	55.0280	30.5084	**1.8036**	15.4109	**3.5707**	8.2019	**6.7091**	5.3327	**10.3189**
6400	117.7521	62.0643	**1.8973**	32.7204	**3.5987**	16.8873	**6.9728**	10.8990	**10.8039**

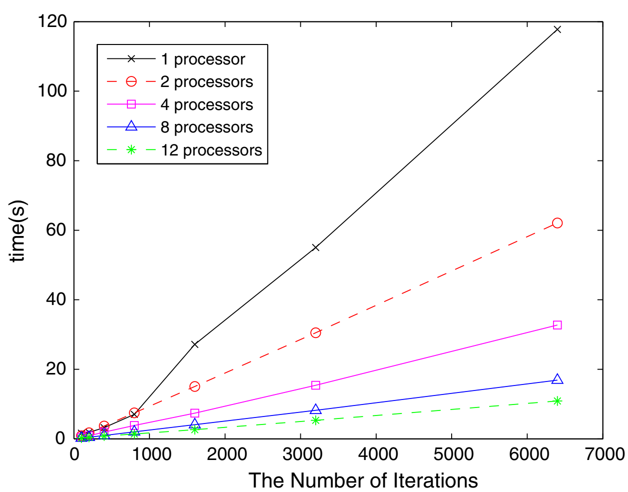

Fig. 11 Results of parallel test

number of processors used is 1, 2, 4, 8 and 12, respectively. For each experiment, we record the running time under a certain number of iterations, and then calculate the speed-up ratio of COOA. We choose f_{28}(50D) as the parallel test function, whose evaluation time is the longest in the 28 CEC-2013 benchmark functions when tested on our experiment platform. The results of the test are listed in Table 14.

When comprehensively analyzed, the results listed in Table 14, we can see that COOA's parallel processing speed continuous to accelerate with the addition of processors. It is able to maintain stable and high speed-up ratio when the scale of the problem expands. According to the data shown in Table 14, we draw up the relation diagram between the computation time and the size of the problem as Fig. 11 shows. From Fig. 11, we can be more intuitive to see the parallel acceleration effect of COOA.

7 Conclusion

Inspired by the human's creative thinking process, a novel COOA is proposed in this paper. Biological and mathematical models are built based on the existing researches about creative thinking. The COOM consists of five sub-models (divergent thinking model, convergent thinking model, inspiration thinking model, experience model and collective thinking model). Each of them corresponds to an effective thinking mode in human's creative thinking process, and they play different roles in an optimization process. The divergent thinking and convergent thinking models perform the local optimization, while the inspiration thinking model and experience model enhance the ability of avoiding the local optimum. Meanwhile, the collective thinking model can both improve the optimization effectiveness and accelerate the optimization speed. The properties of COOA, including convergence and parallelism, are analyzed in detail. Furthermore, to test the effectiveness of COOA, it is applied to solve the CEC-2013 real-parameter benchmark optimization problems. Experimental results are compared with four state-of-the-art algorithms: SMADE, MDE-pBX, CMAES and CCPSO2. Two non-parametric tests (the Friedman test and the Wilcoxon's test) are used to analyze the performance of COOA, and the significance is checked at a level of 0.05. Conclusions from the statistical analysis show that the average performance of COOA over all functions is significantly better than that of others. It also can be seen that COOA outperforms the other algorithms on all the 8 composition functions except f_{22}. In the end, a parallel test for COOA is carried out on a cluster. Experimental results indicate that COOA has an excellent parallelism property.

Our future research direction will be twofold. First, we will focus on the improvement of COOA. It could be done from two aspects. On the one hand, we will improve the biological and mathematical models of COOA. For instance, the divergent thinking model should be modified to generate ideas that are much more creative, and the experience model could be improved to be more effective to protect COOA from local optima. Meanwhile, other intelligent behaviors and some special influencing factors of creativity could also be introduced into COOM to make it more perfect. On the other hand, the fuzzy logic technology will be applied to tune the parameters of COOA. Second, considering the outstanding performance of COOA, we will attempt to apply it to solve some real-world complex problems. Since our current paper is just a preliminary work, there exists further research opportunities in the future.

Acknowledgments This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 60905043, 61073107 and 61173048, the Innovation Program of Shanghai Municipal Education Commission, and the Fundamental Research Funds for the Central Universities.

References

Ahmed F, Deb K (2013) Multi-objective optimal path planning using elitist non-dominated sorting genetic algorithms. Soft Comput 17(7):1283–1299

Ashton-James CE, Chartrand TL (2009) Social cues for creativity: the impact of behavioral mimicry on convergent and divergent thinking. J Exp Soc Psychol 45(4):1036–1040

Balter M (2010) Did working memory spark creative culture? Science 328(5975):160–163

Boden MA (2009) Computer models of creativity. AI Mag 30(3):23

Caraffini F, Neri F, Cheng J, Zhang G, Picinali L, Iacca G, Mininno E (2013) Super-fit multicriteria adaptive differential evolution. In: 2013 IEEE congress on evolutionary computation (CEC). IEEE, New York, pp 1678–1685

Carmeli A, Gelbard R, Reiter-Palmon R (2013) Leadership, creative problem-solving capacity, and creative performance: the importance of knowledge sharing. Hum Resour Manag 52(1):95–121

Chermahini SA, Hommel B (2012) Creative mood swings: divergent and convergent thinking affect mood in opposite ways. Psychol Res 76(5):634–640

Črepinšek M, Liu S-H, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput Surv (CSUR) 45(3):35

Das S, Suganthan PN (2011) Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31

De Bono E (2007) How to have creative ideas: 62 exercises to develop the mind. Random House, München, pp 1–224

De Jong K (2012) Evolutionary computation: a unified approach. In: Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference companion. ACM, New york, pp 737–750

DeHaan RL (2011) Teaching creative science thinking. Science 334(6062):1499–1500

Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol Comput 1(1):3–18

Feng X, Lau F, Yu H (2013) A novel bio-inspired approach based on the behavior of mosquitoes. Inf Sci 233:87–108

Fink A, Koschutnig K, Benedek M, Reishofer G, Ischebeck A, Weiss EM, Ebner F (2012) Stimulating creativity via the exposure to other people's ideas. Hum Brain Mapp 33(11):2603–2610

García S, Fernández A, Luengo J, Herrera F (2009) A study of statistical techniques and performance measures for genetics-based machine learning: accuracy and interpretability. Soft Comput 13(10):959–977

Guilford JP (1967) The nature of human intelligence. McGraw-Hill, New York, pp 1–538

Harnad S (2006) Creativity: method or magic? Hung Stud 20(1):163–177

Karaboga D, Gorkemli B, Ozturk C, Karaboga N (2014) A comprehensive survey: artificial bee colony (abc) algorithm and applications. Artif Intell Rev 42(1):21–57

Kari L, Rozenberg G (2008) The many facets of natural computing. Commun ACM 51(10):72–83

Kephart JO (2011) Learning from nature. Science 331(6018):682–683

Kounios J, Beeman M (2009) The aha! moment the cognitive neuroscience of insight. Curr Dir Psychol Sci 18(4):210–216

Krause J, Ruxton GD, Krause S (2010) Swarm intelligence in animals and humans. Trends Ecol Evol 25(1):28–34

Lee CS, Therriault DJ (2013) The cognitive underpinnings of creative thought: a latent variable analysis exploring the roles of intelligence and working memory in three creative thinking processes. Intelligence 41(5):306–320

Liang J, Qu B, Suganthan P, Hernández-Díaz AG (2013) Problem definitions and evaluation criteria for the cec 2013 special session on real-parameter optimization. Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report 201212

Melin P, Olivas F, Castillo O, Valdez F, Soria J, Valdez M (2013) Optimal design of fuzzy classification systems using pso with dynamic parameter adaptation through fuzzy logic. Expert Syst Appl 40(8):3196–3206

Mumford MD, Medeiros KE, Partlow PJ (2012) Creative thinking: processes, strategies, and knowledge. J Creat Behav 46(1):30–47

Neyoy H, Castillo O, Soria J (2013) Dynamic fuzzy logic parameter tuning for aco and its application in tsp problems. In: Recent advances on hybrid intelligent systems. Studies in computational intelligence, vol 451. Springer, Berlin, pp 259–271

Peterson LE (2011) Covariance matrix self-adaptation evolution strategies and other metaheuristic techniques for neural adaptive learning. Soft Comput 15(8):1483–1495

Precup R, David R, Petriu EM, Preitl S, Radac M (2012) Novel adaptive gravitational search algorithm for fuzzy controlled servo systems. IEEE Trans Ind Inform 8(4):791–800

Runco MA (2010) Creativity: theories and themes: research, development, and practice. Academic Press, New York, pp 1–520

Runco MA, Acar S (2012) Divergent thinking as an indicator of creative potential. Creat Res J 24(1):66–75

Simonton DK (2000) Creativity: cognitive, personal, developmental, and social aspects. Am Psychol 55(1):151

Sousa T, Morais H, Vale Z, Faria P, Soares J (2012) Intelligent energy resource management considering vehicle-to-grid: a simulated annealing approach. IEEE Trans Smart Grid 3(1):535–542

Sternberg RJ (1999) Handbook of creativity. Cambridge University Press, Cambridge, pp 1–170

Sternberg RJ (2010) Innovation: lighting the creative spark. Nature 468(7321):170–171

Vartanian O, Jobidon M-E, Bouak F, Nakashima A, Smith I, Lam Q, Cheung B (2013) Working memory training is associated with lower prefrontal cortex activation in a divergent thinking task. Neuroscience 236:186–194

Wang H-C, Cosley D, Fussell SR (2010) Idea expander: supporting group brainstorming with conversationally triggered visual thinking stimuli. In: Proceedings of the 2010 ACM conference on computer supported cooperative work. ACM, New York, pp 103–106

Xu X, Chen H-L (2014) Adaptive computational chemotaxis based on field in bacterial foraging optimization. Soft Comput 18(4):797–807

Zhang H, Zhu Y, Chen H (2014) Root growth model: a novel approach to numerical function optimization and simulation of plant root system. Soft Comput 18(3):521–537

Zhang Z, Qian S (2011) Artificial immune system in dynamic environments solving time-varying non-linear constrained multi-objective problems. Soft Comput 15(7):1333–1349