

## Research Article

# Improved Clonal Selection Algorithm Based on Biological Forgetting Mechanism

Chao Yang <sup>1,2,3</sup> Bing-qiu Chen <sup>1</sup> Lin Jia,<sup>1</sup> and Hai-yang Wen<sup>1</sup>

<sup>1</sup>School of Computer Science and Information Engineering, Hubei University, Hubei Province, Wuhan 430062, China

<sup>2</sup>Hubei Education Informatization Engineering Technology Research Center, Hubei Province, Wuhan 430062, China

<sup>3</sup>Hubei Key Laboratory of Applied Mathematics, Faculty of Mathematics and Statistics, Hubei University, Wuhan 430062, China

Correspondence should be addressed to Chao Yang; [stevenyc@hubu.edu.cn](mailto:stevenyc@hubu.edu.cn)

Received 3 December 2019; Revised 24 February 2020; Accepted 10 March 2020; Published 7 April 2020

Academic Editor: Lingzhong Guo

Copyright © 2020 Chao Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The antibody candidate set generated by the clonal selection algorithm has only a small number of antibodies with high antigen affinity to obtain high-frequency mutations. Among other antibodies, some low-affinity antibodies are replaced by new antibodies to participate in the next clonal selection. A large number of antibodies with high affinity make it difficult to participate in clonal selection and exist in antibody concentration for a long time. This part of inactive antibody forms a “black hole” of the antibody set, which is difficult to remove and update in a timely manner, thus affecting the speed at which the algorithm approximates the optimal solution. Inspired by the mechanism of biological forgetting, an improved clonal selection algorithm is proposed to solve this problem. It aims to use the abstract mechanism of biological forgetting to eliminate antibodies that cannot actively participate in high-frequency mutations in the antibody candidate set and to improve the problem of insufficient diversity of antibodies in the clonal selection algorithm, which is prone to fall into the local optimal. Compared with the existing clonal selection and genetic algorithms, the experiment and time complexity analysis show that the algorithm has good optimization efficiency and stability.

## 1. Introduction

As a heuristic algorithm to solve complex problems with a high mutation rate, the clonal selection algorithm has two important characteristics: it has efficient optimization performance [1] and it is difficult to get into local optimum [2]. As a result, it has attracted the attention of scholars in related fields. Kim and Bentley [3] used a dynamic cloning selection algorithm to solve the problem of anomaly detection in the changing environment. In recent years, the cloning selection algorithm inspired by biological immunity has also been widely used in power industries such as power plant addressing [4], electricity price prediction [5], hybrid shop scheduling [6], car flow organization [7] and other power industries, playing an active role in the improvement of clustering [8] as well as machine learning algorithms [9–11].

The classical clonal selection algorithm has problems of algorithm efficiency, convergence rate, and lack of sufficient theoretical support [12]. Therefore, people use the cloning

selection algorithm to solve practical problems but also put forward many useful improvements to the algorithm itself. For example, Gálvez et al. [13] proposed an elitist clonal selection algorithm for complex multimodal and multi-variable continuous nonlinear optimization problems. The algorithm replaces the selection and cloning mechanism of the classical clonal selection algorithm with the antibody with the best affinity for antigen and selects the optimal set of the first  $n$  antibodies. This algorithm is better than the related alternative method of genetic algorithm in the automatic node adjustment for the B-spline curve. Gong et al. [14] proposed an improved cloning selection algorithm based on the Baldwin effect, which makes the algorithm more effective and robust by promoting the evolutionary exploration of good genotypes. Rao and Vaisakh proposed [15] a multi-objective adaptive cloning selection algorithm to solve the optimal power flow problem. Pareto optimization was found by using the crowded distance, and the best strategy was selected based on the fuzzy mechanism. In terms of

theoretical analysis, Hong et al. [16] analyzed the convergence of elitist clonal selection algorithm from the perspective of the state transition probability matrix, Markov chain, and other random process-related theories and proposed a method to evaluate the convergence property of the algorithm, which has a certain reference value. This algorithm is better than the genetic algorithm in terms of automatic node adjustment of B-spline curves.

It can be seen from the above literature that the current clonal selection algorithm flow mainly includes steps of selection, cloning, mutation, reelection, and replacement. At present, a large number of improved algorithms are intended to improve the steps of selection, cloning, and mutation, without considering the improvement and optimization of the replacement and update steps of antibody-concentrated antibodies by clonal selection algorithm. In order to improve the update efficiency of the antibody set, this paper tries to find a new scheme to replace the replacement steps used in the current clonal selection algorithm, so as to improve the overall search accuracy and convergence stability of the algorithm.

The antibody candidate set is a set of antibodies produced by the clonal selection algorithm during initialization. The clonal selection algorithm sorts the affinity of the antibodies in the set and selects the first  $n$  antibodies with high affinity to clone and mutate. Since  $n$  depends on manual experience, it is generally set to 10% to 50% of the population size [12]; this selection mechanism inevitably leads to a large number of antibodies with general affinity “retention” in the antibody candidate set. Some were replaced by new antibodies due to low affinity, and the other part did not satisfy the conditions of being replaced and they remained in the antibody candidate set for a long time. This was not conducive to the rapid update of the antibody set and affected the efficiency of the algorithm in finding the optimal solution.

This kind of antibody that stays in the candidate set of antibodies for a long time can neither converge nor update quickly, jumping out of the local optimal interval. Therefore, there is a phenomenon in which antibodies cannot be selected and update iterations in a short period of time, which is vividly called “black hole” in this paper.

In this paper, an improved clonal selection algorithm based on the mechanism of biological forgetting (FCSA) is proposed. Aiming at the black hole formed by antibodies whose affinity does not meet the mutation and update conditions, the forgetting mechanism is applied to the antibody candidate set update process of the algorithm. The aim is to update these inactive and long-lasting antibodies in the candidate antibody set and enhance the diversity of antibodies in the antibody set, thereby increasing the convergence speed of the algorithm.

## 2. Overview of the Biological Forgetting Mechanism

**2.1. Biological Forgetting Research Background.** In 1885, Ebbinghaus [17] put forward the Ebbinghaus memory curve, which described the amount of forgetting over time. Over

the next 100 years, numerous scholars have proposed theories to explain how forgetting occurs. At present, there are various opinions as to the causes of forgetting, and the widely supported explanations include decay theory [18] and interference theory [19]. Decay theory holds that the decay of forgetting, which is similar to the decay of radioactive elements, is a process in which memories disappear spontaneously over time. According to the interference theory, forgetting is caused by the conflict between memory blocks. The similarity, quantity, and intensity of memory blocks determine the degree of forgetting. The more similar the contents between memory blocks are, the more times they are remembered, and the higher the memory intensity is, the less likely they are to be forgotten.

**2.2. Forgetting Mechanism Affected by Rac1 Activity.** Although both attenuation theory and interference theory are supported by experimental results, they lack an objective mechanism to explain why attenuation theory is only applicable to long-term memory, while interference theory is more applicable to short-term memory.

Shuai et al. [20] studied the small G protein Rac1 in the brain of drosophila and found that Rac1 activity affected the degree of forgetting in a drosophila olfactory avoidance experiment. Specifically, after Rac1 activity was inhibited in the brain of drosophila, the forgetting rate slowed down; after Rac1 activity was enhanced by stimulation, the forgetting rate increased. Changes in Rac1 activity in drosophila fly brain during memory decay and memory interference mentioned in Section 2.1 of this paper also support the above conclusions.

Therefore, Shuai et al. [20] proposed a set of forgetting mechanisms affected by Rac1 activity. The activity of Rac1 would also change according to the memory process with differing time lengths. When memory was disturbed for a short time, Rac1 activity also increased rapidly, making the previous outdated memory forgotten. Rac1 takes longer to activate when the time span is large, suggesting that memory declines over time.

Liu et al.'s [21] studies on Rac1 protein activity in mouse hippocampal neurons and object recognition memory in mice further support the role of Rac1 activation in forgetting.

**2.3. Observations about Biological Forgetting.** In the complex biological system, the process of forgetting often occurs. This paper argues that forgetting is a process of information loss, and information loss is meaningful under certain circumstances. In reference [22], the odor type that would cause the fruit fly to receive an electric shock was repeatedly changed, and the fruit fly would remember the odor that would receive an electric shock the most recently, so as to avoid this type of odor. This suggests that in complex external situations, memories that cannot be adapted to the current environment need to be forgotten. Above, we suggest that the forgetting mechanism is meaningful for behaviors requiring rapid adaptation to the new environment, such as cloning selection for a high-mutation environment.

At the same time, the idea of attenuation theory of biological forgetting is introduced into the replacement process of the clonal selection algorithm. The number of iterations of antibodies in the antibody candidate set is taken as the time length, and whether the antibody participates in high-frequency mutation in a certain iteration is taken as the basis of whether the antibody is remembered at the point in time, so as to realize the purpose of replacing the antibody with weak memory degree when the time span is large. Moreover, since the current replacement mechanism of clonal selection is still to replace the  $d$  antibodies with the worst affinity of antibodies, the antibody can be dynamically replaced according to its own characteristics after the introduction of attenuation theory.

### 3. Clonal Selection Algorithm Based on Forgetting Mechanism

#### 3.1. Clonal Selection Algorithm and Forgetting Mechanism

**3.1.1. Introduction to Clonal Selection Algorithm.** In 2002, De Castro and Von Zuben [23], inspired by biological as well as artificial immunology, proposed a clonal selection algorithm to solve the problems of multimodal and combinatorial optimization by using the principle of clonal selection. The main flow of this algorithm is

*Step 1.* Initialize the size of the antibody set, iteration number, cloning number, and other relevant parameters, randomly select an antigen from the antigen set, and generate the candidate antibody set, which is composed of a memory set and residual set.

*Step 2.* Calculate the affinity between each antibody and antigen in the candidate antibody concentration, and select the first  $n$  antibodies with the highest affinity.

*Step 3.* Clone the  $n$  antibodies, and the number of antibody clones is positively correlated with their affinity to the antigen.

*Step 4.* Carry out mutation treatment on the antibody set generated after cloning, and the higher the affinity, the lower the probability of antibody mutation.

*Step 5.* Calculate the antibody affinity after mutation, select the antibody with the highest affinity, and compare it with the antibody in the current memory set. Then, select the antibody with the highest affinity and put it into the memory set.

*Step 6.* Randomly select  $d$  new antibodies to replace the  $d$  antibodies with the worst affinity to the antigen in the remaining set.

*Step 7.* Skip to step 2 for the next iteration. When the number of iterations meets the termination condition, the algorithm terminates.

It can be seen that the cloning selection algorithm selects the first  $n$  antibodies with the highest concentration affinity of candidate antibodies in step 2, and the subsequent steps primarily involve cloning and mutation of these  $n$  antibodies. The rest of the antibody candidate set does not

participate in the epicycle mutation, leading to the following situations: Antibody A does not belong to the first  $n$  antibodies with the highest affinity in one iteration of the algorithm, and it is not replaced in step 6. In the subsequent rounds of iterations, antibody A does not always become the  $d$  antibody with the lowest affinity with the antigen. In this case, antibody A can no longer adapt to new changes, cannot participate in the high mutation process of the algorithm, and cannot be replaced as the worst  $d$  antibody in time, thus affecting the update rate of antibodies in the whole candidate antibody concentration.

**3.1.2. Clonal Selection Algorithm Inspired by Forgetting Mechanism.** In order to more intuitively explain the state of the candidate antibodies in a certain round of iteration, the affinity between the antigen and the antibody is expressed in the form of distance. According to the previous section, after all antibodies are sorted according to their affinity, the clonal selection algorithm selects the  $n$  best antibodies in the antibody candidate set for cloning and mutation. As shown in Figure 1, in the reselect step, the original antibody and the antibody are mixed and screened for the next iteration. At this time, there are antibodies that were not selected as clones in the previous round. We divide the best  $n$  antibodies into the first layer and the worst  $d$  antibodies among the remaining antibodies into the third layer, and then, the second layer contains the intermediate antibodies.

Since the total number  $P$  of antibodies remains unchanged, after the completion of one iteration of the CSA, all of the antibodies in the third layer will be replaced with new antibodies, and the antibodies in the antibody candidate set will be reordered according to the affinity with the antigen, so some of the antibodies in the entire set of antibody candidates will migrate. In the first layer, a part of the antibody will migrate to the first tier II. In addition, some of the antibodies in layer II migrate to layer III. Analyze the layer II antibodies at this time, including the antibodies migrated into layer I and layer III and the original antibodies not migrated out of layer II. After multiple rounds of iteration, there are still some unmigrated native antibodies in layer II which have not migrated to the high-frequency mutations involved in the algorithm in layer I and have not been replaced with new antibodies in layer III. It is suggested that the above unmigrated antibodies that always exist in layer II are a memory that cannot adapt to the current environment and should be updated along with the antibody in layer III.

Inspired by the forgetting mechanism, for each antibody in the candidate antibody set, calculate the number of times it is selected as the top  $n$  affinity and the number of iterations in the candidate antibody set and use these two as antibody characteristics. These antibody characteristics were used as the basis for the change of Rac1 activity, and the activation degree of Rac1 determined whether to eliminate the “stranded” antibody from the concentration of candidate antibodies.

As shown in Figure 2, antibodies in the candidate set of antibodies are divided into two layers according to the size of affinity. Layer II contains all remaining antibodies.

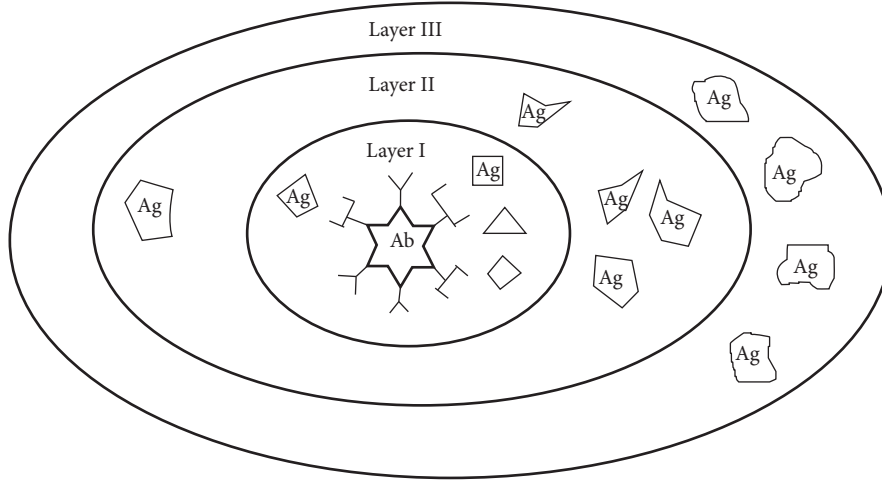


FIGURE 1: Antigen and antibody distribution structure.

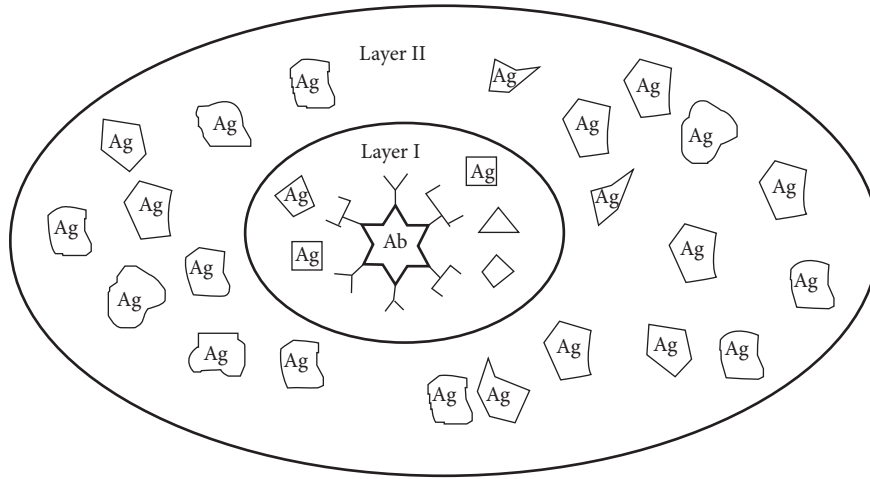


FIGURE 2: Antigen-antibody distribution structure affected by Rac1.

For each antibody in the candidate set, the Rac1 activity of the antibody in layer I is significantly lower than that in layer II. When the Rac1 activity of the layer II antibody exceeds the activity threshold, the antibody is replaced and the entire antibody candidate set is updated. In this way, the clonal selection algorithm avoids the antibody “black hole” formed by the partially unmigrated original antibodies in layer II.

**3.2. Abstract Definition of Forgetting Mechanism.** The following definitions relate to the forgetting mechanism:

- (1) Antibody survival time is the number of iterations that antibodies have participated in the antibody candidate set.
- (2) Time  $T$  is the execution time of the clonal selection algorithm. In this paper,  $T$  refers to the number of algorithm iterations.
- (3) Appropriate memory is the attribute of each candidate antibody. In an iteration, if the antibody belongs to the best  $n$  antibodies, it is regarded as a suitable memory.

(4) Appropriate memory strength is the appropriate memory accumulated by candidate antibodies during  $T$  time.

(5) Rac1 protein activity is the index affecting antibody forgetting, determined by the survival time of antibody in candidate antibody concentration and the strength of appropriate memory. Rac1 protein activity is proportional to the survival time of antibodies and inversely proportional to the degree of appropriate memory.

**3.3. Improved Clonal Selection Algorithm.** In this paper, an improved clonal selection algorithm (FCSA) inspired by the forgetting mechanism is proposed. Its core implementation idea is to replace the receptor editing mechanism [24] in the CSA with a unique forgetting mechanism.

The specific implementation method is as follows: In each iteration of the algorithm, the appropriate memory strength and survival time of each antibody candidate set are recorded. After several iterations of the algorithm, antibody



forgetting was determined based on whether Rac1 protein activity reached the threshold.

**3.3.1. Affinity Calculation.** To simplify the calculation, the target test function of antibody affinity to antigen is the function value, which can be expressed as

$$\text{Aff}_{ab_i} = f(x_1, x_2, \dots, x_D), \quad (1)$$

where  $\{x_i | i = 1, 2, \dots, D\}$  is the antibody and  $D$  is the dimension of the antibody.

**3.3.2. Cloning Method.** According to the affinity corresponding to the antibody and antigen, the cloning method performs the cloning operation on the antibody. The higher the affinity, the greater the number of antibodies that will be cloned. The specific cloning formula is

$$Abc = \left\{ ab_{ij} \left| \begin{array}{l} i \in [0, \text{population\_size} - 1] \\ j = \max \left( 1, \text{int} \left( \frac{\text{Aff}_{ab_i} + a}{a} * \text{max\_clone} \right) \right) \right. \right\}, \quad a > 0, \quad (2)$$

where  $\text{population\_size}$  represents the number of antibodies in the candidate set of antibodies,  $\text{Aff}_{ab_i}$  represents the affinity between the antibody  $ab_i$  and the antigen,  $\text{max\_clone}$  represents the initial clone number, and  $j$  represents the clone number of the antibody  $ab_i$ .

**3.3.3. Variation Method.** The mutation method aims at the  $ab_i$  of the cloned antibody and determines the mutation degree of the antibody according to the affinity between the antibody and the antigen. The higher the affinity, the less the possibility and degree of the mutation.

The specific variation formula is

$$ab = \{x_i | i = 1, 2, \dots, D\},$$

$$x_i = \begin{cases} x_i + \text{random}(-a, a), & \text{random}(0, 1) < e^{-(r * \text{aff}_{ab} / \text{max\_aff})}, \\ x_i, & \text{random}(0, 1) > e^{-(r * \text{aff}_{ab} / \text{MAX\_aff})}, \end{cases} \quad (3)$$

where  $r$  is the mutation rate,  $a$  is the variation range,  $a > 0$ , and  $\text{max\_aff}$  is the maximum affinity of the concentrated antibody.

**3.3.4. Forgetting Method.** The method of forgetting determines the necessity of antibody forgetting based on the survival time of the antibody, the appropriate memory intensity, and the activity of the Rac1 protein.

The specific forgetting formula is

$$ab = \{x_i | i = 1, 2, \dots, D\},$$

$$x_i = \begin{cases} \text{random}(\text{min Domain}, \text{max Domain}), & \frac{ab_{\text{time}}}{ab_{\text{strength}}} > c, \\ x_i, & \text{otherwise}, \end{cases} \quad (4)$$

where  $ab_{\text{time}}$  is the antibody survival time,  $ab_{\text{strength}}$  is the appropriate memory strength, and  $c$  is the Rac1 protein activity threshold.

**3.3.5. Algorithm Flow.** The flow of the improved algorithm proposed in this paper is shown in Algorithm1.

The suspension conditions of the algorithms in Algorithm 1 can be determined according to specific needs. Common termination conditions include reaching the maximum value of the function evaluation and reaching the maximum number of generations.

In the algorithm, Rac1 protein activity is an inherent property of each candidate antibody, which is calculated based on antibody survival time and appropriate memory strength when the antibody is first selected into the candidate set. And it changes dynamically with the execution of the algorithm. When the property value of the antibody reaches the threshold, it means that the antibody has not mutated in a better direction within the time we expect, and it is not sufficiently competitive with other candidate antibodies. So in the algorithm, the antibody that meets the threshold value will perform the forgetting operation.

## 4. Experiment and Algorithm Evaluation

Thirteen kinds of testing were done to select CEC test function optimization algorithm functions as experimental test functions, respectively, using the test function of the presented algorithm (FCSA) proposed in [13], the elitist clonal selection algorithm (ECSA) proposed in literature [14], Baldwinian learning in clonal selection algorithm (BCSA), and search accuracy of the genetic algorithm (GA) for testing. The experimental steps are as follows.

First, initialize various parameters of the algorithm. The termination criterion in this experiment is to run the GA, BCSA, ECSA, and FCSA until the number of function evaluations reaches the maximum value of 350,000.

Second, find the optimal solution of the test function. Three algorithms were executed to obtain the optimal solution generated by each execution of the algorithm. The average optimal solution, maximum optimal solution, and minimum optimal solution after 100 executions were analyzed.

The purpose of the experiment in this paper is to analyze the effectiveness of the forgetting mechanism applied to the clonal selection algorithm. The performance of the algorithm is mainly evaluated by the quality of the results obtained when the suspension conditions are consistent. This

## FCSA

Input:  $N$  (the size of the population),  $n$  (the number of antibodies selected for cloning),  $n_c$  (the number of clones),  $m$  (the degree of variation),  $c$  (Rac1 protein activity threshold)

Output: the best antibody

- (1) Begin
- (2) Randomly generate  $N$  antibodies to form the initial candidate set
- (3) **while not** meet algorithm termination conditions **do**
- (4)   Calculate the affinity  $\text{Aff}_{ab_i}$  of each antibody for antigen in the candidate set and record antibody survival time  $T_{ab_i}$
- (5)   Sort the antibodies in the candidate set according to their affinity, and put the best  $n$  antibodies into the antibody set  $Ab_s$
- (6)   **for**  $ab_i$  **in**  $Ab_s$
- (7)     Update the value of the appropriate memory of antibody  $ab_i$ :  $S_{ab_i} + 1$ . See CLONING METHOD, clone antibody  $ab_i$  according to  $n_c$  and  $\text{Aff}_{ab_i}$ , and put all antibodies obtained by cloning into antibody set  $Ab_c$
- (8)   **end for**
- (9)   **for**  $ab_i$  **in**  $Ab_c$
- (10)     See VARIATION METHOD, according to the degree of variation  $m$  and the affinity of the antibody for the antigen  $\text{Aff}_{ab_i}$  to mutate  $ab_i$
- (11)     **if** antibody  $ab_i$  is a variant antibody
- (12)       The  $ab_i$  survival time  $T_{ab_i} = 0$ , The appropriate memory intensity  $S_{ab_i} = 1$
- (13)     **end if**
- (14)   **end for**
- (15)   Select the  $N$  antibodies with the highest antigen affinity in  $Ab_c$  and  $Ab$  to replace the  $N$  antibodies in  $Ab$
- (16)   See FORGETTING METHOD, calculate the Rac1 protein activity of each antibody in  $Ab$  according to the ratio of  $T_{ab_i}$  to  $S_{ab_i}$
- (17)   **if** antibody  $ab_i$  Rac1 protein activity  $>$  threshold
- (18)     forget the antibody  $ab_i$
- (19)   **end if**
- (20)   **end while**
- (21) Choose the best antibody as the final output

ALGORITHM 1: Forgotten-based clonal selection algorithm.

article counts the mean and standard deviation of the results of multiple runs of the algorithm to evaluate the quality of the results. These two indicators reflect the concentration trend and the degree of dispersion of the experimental data, respectively. Therefore, this paper uses these two indicators to verify the effectiveness of the improved algorithm.

Finally, we obtained the results of GA, CSA, and FCSA running at 1000 generations and plotted them as line charts. The purpose is to analyze the accuracy and speed of the algorithm by characterizing the relationship between generations and algorithm results.

Among them, the algorithm parameters are set as shown in Table 1, and the execution environment of the algorithm is shown in Table 2.

**4.1. Test Function.** The test functions selected in this paper are shown in Table 3. Their common feature is that they have a global minimum value [25], and the function image is complex with multiple local minimum values. Conversely, the opposite value of the test function is the global maximum.

Consider the test functions in Table 3. Since antibodies with high affinity are generally selected when comparing the affinity of antigens and antibodies, this paper takes the opposite value corresponding to these test functions as the global maximum value, which is equivalent to the global minimum value of the test function in Table 3. The opposite of the trial function is the global maximum.

TABLE 1: Initialization parameters.

Algorithm parameter	GA	CSA	FCSA	ECSA	BCSA
Cross rate	0.5	—	—	—	—
Mutation rate	0.13	2	2	2	2
Initial clone number	—	5	5	5	5
Rac1 threshold	—	—	3	—	—

TABLE 2: Execution environment.

OS	Windows 10 professional edition
CPU	Intel(R) Core(TM) i3-3217U CPU @ 1.80GHZ
RAM	12.0 GB
Compiler version	Python 3.6

**4.2. Experimental Results.** The results of our experiment are shown in Table 4. The closer the average value of the optimal solution obtained by each algorithm is to the reference value, the higher the accuracy of the algorithm under the termination condition. For the case that the test functions  $f_1$ ,  $f_6$ ,  $f_8$ ,  $f_9$ , and  $f_{12}$  have high-dimensional solutions, in order to verify the convergence of the FCSA in high-dimensional test function, set  $d = 50$  and  $d = 100$  and compare the results with ECSA and BCSA, as shown in Tables 5 and 6.

After replacing the updating operator in BCSA with the forgetting operator, set  $d = 50$  and  $d = 100$  to test the updating ability of the forgetting operator compared with the previous updating operator in different dimensions.

TABLE 3: Test function.

Test function	Expression	Optimum
Ackley Function	$f_1(x) = -20 \exp(-0.2 \sqrt{(1/d) \sum_{i=1}^d x_i^2}) - \exp((1/d) \sum_{i=1}^d \cos(2\pi x_i)) + 20 + \exp(1)$	0
Bukin Function n. 6	$f_2(x) = 100 \sqrt{ x_2 - 0.01x_1^2  + 0.01 x_1 + 10 }$	0
Cross-in-Tray Function	$f_3(x) = -0.0001 ( \sin x_1 \sin x_2 \exp((100 - (\sqrt{x_1^2 + x_2^2}/\pi))  + 1)^{0.1}$	-2.06261
Drop-Wave Function	$f_4(x) = -(1 + \cos(12\sqrt{x_1^2 + x_2^2})/0.5(x_1^2 + x_2^2) + 2)$	-1
Eggholder Function	$f_5(x) = -(x_2 + 47) \sin(\sqrt{ x_2 + (x_1/2) + 47 }) - x_1 \sin(\sqrt{ x_1 - x_2 - 47 })$	-959.6407
Griewank Function	$f_6(x) = \sum_{i=1}^d x_i^2/4000 - \prod_{i=1}^d \cos(x_i/\sqrt{i}) + 1$	0
Holder Table Function	$f_7(x) = - \sin(x_1) \cos(x_2) \exp( 1 - (\sqrt{x_1^2 + x_2^2}/\pi) ) $	-19.2085
Levy Function	$f_8(x) = \sin^2(\pi w_1) + \sum_{i=1}^{d-1} (w_i - 1)^2 [1 + 10 \sin^2(\pi w_i + 1)] + (w_d - 1)^2 [1 + \sin^2(2\pi w_d)]$ , where $w_i = 1 + (x_i - 1/4)$ , for all $i = 1, \dots, d$	0
Rastrigin Function	$f_9(x) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$	0
Schaffer Function n. 2	$f_{10}(x) = 0.5 + (\sin^2(x_1^2 - x_2^2) - 0.5/[1 + 0.001(x_1^2 + x_2^2)])^2$	0
Schaffer Function n. 4	$f_{11}(x) = 0.5 + (\cos(\sin( x_1^2 - x_2^2 )) - 0.5/[1 + 0.001(x_1^2 + x_2^2)])^2$	0.5
Schwefel Function	$f_{12}(x) = 418.9829d - \sum_{i=1}^d x_i \sin(\sqrt{ x_i })$	0
Shubert Function	$f_{13}(x) = (\sum_{i=1}^5 i \cos((i+1)x_1 + i)) (\sum_{i=1}^5 i \cos((i+1)x_2 + i))$	-186.7309

TABLE 4: Results of GA, CSA, and FCSA when  $D = 2$ .

ALGs	GA		CSA		FCSA	
	Mean	Std	Mean	Std	Mean	Std
$f_1$	$-5.0923e-001$	$5.1699e-001$	$-4.1302e-001$	$3.4025e-001$	<b><math>-1.9901e-001</math></b>	<b><math>2.0546e-001</math></b>
$f_2$	$-7.3939e+000$	$4.7669e+000$	$-1.2512e+000$	$6.3021e-001$	<b><math>-9.7695e-001</math></b>	<b><math>5.3091e-001</math></b>
$f_3$	<b><math>2.0601e+000</math></b>	$5.0685e-003$	$2.06255e+00$	$6.1577e-005$	$2.06258e+00$	<b><math>2.9248e-005</math></b>
$f_4$	$9.2563e-001$	$4.3546e-002$	$9.6047e-001$	$2.3679e-002$	<b><math>9.7154e-001</math></b>	<b><math>2.0048e-002</math></b>
$f_5$	$9.3533e+002$	<b><math>1.4155e-002</math></b>	$9.5836e+002$	$1.7805e+000$	<b><math>9.5926e+002</math></b>	$6.3550e-001$
$f_6$	<b><math>-1.6845e-002</math></b>	$2.7008e-002$	$-2.5327e-002$	$1.5556e-002$	$-2.3604e-002$	<b><math>1.3131e-002</math></b>
$f_7$	$1.8907e+001$	$3.8513e-001$	$1.9203e+001$	$6.1897e-003$	<b><math>1.9206e+001</math></b>	<b><math>3.1154e-003</math></b>
$f_8$	$-9.7077e-003$	$2.2512e-002$	$-7.2359e-004$	$7.3241e-004$	<b><math>-3.3312e-004</math></b>	<b><math>3.5434e-004</math></b>
$f_9$	$-1.4031e+000$	$9.6354e-001$	$-1.5528e-001$	$1.5445e-001$	<b><math>-7.9310e-002</math></b>	<b><math>7.6989e-002</math></b>
$f_{10}$	<b><math>-2.0083e-004</math></b>	<b><math>6.4135e-004</math></b>	$-2.7670e-003$	$3.2928e-003$	$-1.1855e-003$	$1.6805e-003$
$f_{11}$	$-5.00096e-01$	$3.4944e-006$	<b><math>-5.00094e-01</math></b>	$1.7190e-006$	<b><math>-5.00094e-01</math></b>	<b><math>1.5678e-006</math></b>
$f_{12}$	<b><math>-2.5903e-003</math></b>	<b><math>4.8844e-003</math></b>	$-4.4234e-002$	$4.5491e-002$	$-3.0112e-002$	$3.1088e-002$
$f_{13}$	$1.6615e+002$	$2.3853e+001$	$1.8632e+002$	$3.9038e-001$	<b><math>1.8649e+002</math></b>	<b><math>2.9016e-001</math></b>

TABLE 5: Results of BCSA, ECSA, and FCSA when  $D = 50$ .

ALGs	BCSA		ECSA		FCSA	
	Mean	Std	Mean	Std	Mean	Std
$f_1$	$-1.2838e+001$	$2.3931e-001$	$-2.0213e+001$	<b><math>1.1482e-001</math></b>	<b><math>-1.2749e+001</math></b>	$2.3649e-001$
$f_6$	$-7.5633e+002$	<b><math>3.3740e+001</math></b>	$-7.4943e+002$	$3.4107e+001$	<b><math>-7.4891e+002</math></b>	$3.6480e+001$
$f_8$	$-2.5742e+002$	$1.7774e+001$	$-2.5256e+002$	$1.9350e+001$	<b><math>-2.4812e+002</math></b>	<b><math>1.6304e+001</math></b>
$f_9$	$-6.3444e+002$	$1.9519e+001$	$-6.3492e-001$	$1.9929e+001$	<b><math>-6.2661e+002</math></b>	<b><math>1.6450e+001</math></b>
$f_{12}$	<b><math>-1.4640e+004</math></b>	$3.1504e+002$	$-1.4646e+004$	$3.0211e+002$	$-1.4721e+004$	<b><math>2.7377e+002</math></b>

TABLE 6: Results of BCSA, ECSA, and FCSA when  $D = 100$ .

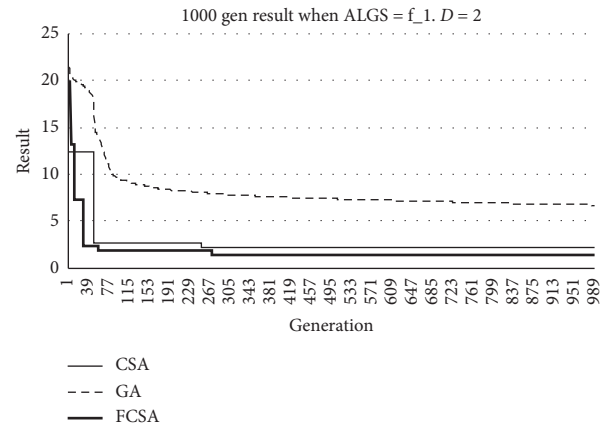
ALGs	BCSA		ECSA		FCSA	
	Mean	Std	Mean	Std	Mean	Std
$f_1$	$-1.3768e+001$	$1.1996e-001$	<b><math>-1.3756e+001</math></b>	$1.2061e-001$	$-1.3697e+001$	<b><math>1.1556e-001</math></b>
$f_6$	$-1.9237e+003$	<b><math>5.7814e+001</math></b>	$-1.9198e+003$	$6.4516e+001$	<b><math>-1.9070e+003</math></b>	$6.5405e+001$
$f_8$	$-7.1533e+002$	<b><math>2.5226e+001</math></b>	$-7.0166e+002$	$4.0096e+001$	<b><math>-6.9408e+002</math></b>	$2.7792e+001$
$f_9$	$-1.4416e+003$	$2.6102e+001$	$-1.4370e+003$	$3.2709e+001$	<b><math>-1.4283e+003</math></b>	<b><math>2.2737e+001</math></b>
$f_{12}$	<b><math>-3.2839e+004</math></b>	$4.5766e+002$	$-3.2866e+004$	$3.7953e+002$	$-3.2949e+004$	<b><math>3.6880e+002</math></b>

Finally, the comparison results of the three algorithms of GA, CSA, and FCSA in the 1000th generation are shown in Figure 3. The abscissa represents the current generation number of the algorithm, the ordinate represents the current result of the algorithm, the dashed line represents GA, and the thin solid line represents CSA. The line represents FCSA.

In the test results of the Ackley Function, the optimal interval of FCSA in the process of 100 executions of the algorithm is  $(-1, 0)$ , the optimal interval of CSA is  $(-1.5, 0)$ , and the optimal interval of GA is  $(-2.5, 0)$ . The convergence degree of the GA is worse than that of CSA and FCSA, and FCSA has the best degree of convergence.

Among the test results of the Bukin Function  $n. 6$ , drop-wave Function, Rastrigin Function, and Schaffer Function  $n. 4$ , the FCSA has the best optimization accuracy and convergence stability, while the GA has a poor optimization accuracy and convergence degree compared with CSA.

In the test results of functions, Cross-in-Tray Function, Holder Table Function, Levy Function, and Shubert Function, the CSA and FCSA algorithms converge stably to the

FIGURE 3: Results of GA, CSA, and FCSA when  $D=2$  and  $ALGs = f_1$ .

global optimal, among which FCSA has the optimal average search accuracy and stability, while the GA still has deviation points and the convergence is not stable.



TABLE 7: Results of the BCSA algorithm combined with the forgetting mechanism when  $D = 50, 100$ .

ALGs	$D = 50$		$D = 100$	
	Mean	Std	Mean	Std
$f_1$	$-8.5121e + 000$	$1.4299e - 001$	$-9.1801e + 000$	$9.4149e - 002$
$f_6$	$-7.5371e + 002$	$3.7047e + 001$	$-1.9109e + 003$	$7.2792e + 001$
$f_8$	$-2.4475e + 002$	$1.9376e + 001$	$-6.8859e + 002$	$3.1488e + 001$
$f_9$	$-6.2326e + 002$	$1.9837e + 001$	$-1.4202e + 003$	$3.0524e + 001$
$f_{12}$	$-1.4873e + 004$	$2.7891e + 002$	$-3.3390e + 004$	$3.6963e + 002$

According to the test results of Eggholder Function, the convergence stability of the CSA and FCSA is worse than GA, but the optimization accuracy is better than GA.

In the test results of Griewank Function, the optimization accuracy of the CSA and FCSA is better than that of GA, and GA has a few deviation points as well as poor convergence stability. With the improved CSA, when  $d = 50$ , FCSA is the best in algorithm stability, and when  $d = 100$ , FCSA is the best in algorithm optimization accuracy.

In the test results of Schaffer Function  $n. 2$  and Schwefel Function, the GA has better convergence stability and optimization accuracy than the CSA and FCSA. For Schwefel Function, FCSA shows a more stable degree of convergence in higher dimensions, while BCSA shows more accurate convergence in higher dimensions.

As can be seen from the experimental results in Table 7, the mean value and standard deviation of the optimal solution in the test functions  $f_1$ ,  $f_6$ ,  $f_8$ , and  $f_9$ , after the forgetting mechanism is introduced into the BCSA algorithm, are both better than the results of the original BCSA in Tables 5 and 6.

On the other hand, it can be seen from Figure 3 that the clone selection algorithm converges around 300 generations and is very close to the global minimum of the function (the global minimum is 0). In particular, the improved algorithm proposed in this paper is more accurate when compared with the comparison algorithm, and it can always find better results when the generation is the same.

**4.3. Experiment Analysis.** According to Section 4.2 of this paper, respectively, using GA, CSA, and FCSA on 13 kinds of test functions for the optimal solution (global maximum) experimental data, the following results can be seen. Using the Ackley Function, Bukin Function, Eggholder Function, Levy Function, Rastrigin Function, Shubert Function, etc., when using this algorithm in the initial experimental environment which is the same as the CSA and GA optimization under the condition of higher precision, convergence is stable and reliable. The GA can only converge stably on the test functions Schaffer Function  $n. 2$  and Schwefel Function, but it cannot converge stably on the other 11 test functions, and it is easy to fall into local optimum.

Overall, the experimental results show that FCSA has higher optimization accuracy and stability than CSA, and FCSA has higher optimization accuracy and convergence stability than GA in most test functions.

It can be seen from the high-dimensional experiments of BCSA, ECSA, and FCSA that FCSA has more advantages over ECSA and BCSA in terms of convergence stability and

accuracy. Due to the characteristics of the test function itself, the higher the dimension, the more complex the function change is, which leads to decreased optimization accuracy and stability of the algorithm.

By applying the forgetting mechanism to BCSA, the number of antibodies to be replaced by the original manual definition is changed to the number of antibodies to be replaced by the affinity attribute of antibodies. The forgetting mechanism has a positive effect on improving the convergence speed and convergence stability of such algorithms.

## 5. Conclusion

To solve the problem that the CSA cannot in a timely way eliminate antibodies that are not adapted to the new environment and thus form an antibody black hole, we see that by changing the receptor editing mechanism of the clonal selection algorithm to a new forgetting mechanism, the antibody candidate set can be replaced and updated under the regulation of Rac1 protein. Experiments show that FCSA is an effective improved algorithm compared with ECSA and BCSA in terms of optimization efficiency, optimization accuracy, and convergence stability.

Because FCSA changes the substitution step in the current clonal selection algorithm, it is better than the existing improved clonal selection algorithm. However, from the experimental performance in high-dimensional test function, FCSA still has the problem of low optimization precision. In the future, the FCSA will be combined with the existing improved clonal selection algorithm to further optimize the precision and stability of high-dimensional optimization.

We also note that Luo et al. [26] proposed a clonal selection method for dynamic multimodal optimization problems and proved the effectiveness of the method. When the global peaks of the problem change with time, how to use the forgetting mechanism to quickly adapt to the new height of global peaks and forget the outdated experience in time will be our future research direction.

At the same time, as an effective updating mechanism, the forgetting mechanism can also be applied to other heuristic algorithms that need to update the population of algorithms.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China (61977021), National Social Science Fund (15CGL074), Intelligent Information Processing and Real Time Industrial System Hubei Provincial Key Laboratory Open Fund Project (znxx2018MS05), and Open project of Hubei Key Laboratory of Applied Mathematics (HBAM201902).

## References

- [1] W. Zhang, K. Gao, W. Zhang, X. Wang, Q. Zhang, and H. Wang, "A hybrid clonal selection algorithm with modified combinatorial recombination and success-history based adaptive mutation for numerical optimization," *Applied Intelligence*, vol. 49, no. 2, pp. 819–836, 2019.
- [2] G. Yang, H. Jin, and X. Zhu, "Optimization algorithm based on differential evolution and clonal selection mechanism," *Computer Engineering and Applications*, vol. 49, no. 10, pp. 50–52, 2013.
- [3] J. Kim and P. Bentley, "Immune memory and gene library evolution in the dynamic clonal selection algorithm," *Genetic Programming and Evolvable Machines*, vol. 5, no. 4, pp. 361–391, 2004.
- [4] D. Zhang, K. Wang, Y. Long, and X. Zhao, "Multi-objective clonal selection algorithm applying to biomass power plant location model," *Journal of Geomatics*, vol. 43, no. 2, pp. 19–23, 2018.
- [5] M. Rafiei, T. Niknam, and M. H. Khooban, "Probabilistic electricity price forecasting by improved clonal selection algorithm and wavelet preprocessing," *Neural Computing and Applications*, vol. 28, no. 12, pp. 1–13, 2016.
- [6] G. Lou and Z. Cai, "Improved hybrid immune clonal selection genetic algorithm and its application in hybrid shop scheduling," *Cluster Computing*, vol. 22, no. S2, pp. 3419–3429, 2019.
- [7] Y. Jing and Z. Zhang, "A study on car flow organization in the loading end of heavy haul railway based on immune clonal selection algorithm," *Neural Computing and Applications*, vol. 31, no. 5, pp. 1455–1465, 2019.
- [8] Z. Zareizadeh, M. S. Helfroush, A. Rahideh, and K. Kazemi, "A robust gene clustering algorithm based on clonal selection in multiobjective optimization framework," *Expert Systems with Applications*, vol. 113, no. 15, pp. 301–314, 2018.
- [9] S. Kamada and T. Ichimura, "A generation method of immunological memory in clonal selection algorithm by using restricted boltzmann machines," in *Proceedings of the IEEE International Conference on Systems*, October 2016.
- [10] S. Mohapatra, P. M. Khilar, and R. Ranjan Swain, "Fault diagnosis in wireless sensor network using clonal selection principle and probabilistic neural network approach," *International Journal of Communication Systems*, vol. 32, no. 16, p. e4138, 2019.
- [11] C. Yavuz Burcu, Y. Nilufer, and O. Ozkan, "Prediction of protein secondary structure with clonal selection algorithm and multilayer perceptron," *IEEE ACCESS*, vol. 6, pp. 45256–45261, 2018.
- [12] W. Luo and X. Lin, "Recent advances in clonal selection algorithms and applications," in *Proceedings of the IEEE Symposium Series on Computational Intelligence*, November 2018.
- [13] A. Gálvez, A. Iglesias, A. Avila, C. Otero, R. Arias, and C. Manchado, "Elitist clonal selection algorithm for optimal choice of free knots in B-spline data fitting," *Applied Soft Computing*, vol. 26, pp. 90–106, 2015.
- [14] M. Gong, L. Jiao, and L. Zhang, "Baldwinian learning in clonal selection algorithm for optimization," *Information Sciences*, vol. 180, no. 8, pp. 1218–1236, 2010.
- [15] B. S. Rao and K. Vaisakh, "Multi-objective adaptive clonal selection algorithm for solving optimal power flow considering multi-type FACTS devices and load uncertainty," *Applied Soft Computing*, vol. 23, pp. 286–297, 2014.
- [16] L. Hong, C. L. Gong, J. Z. Wang, and Z. C. Ji, "Convergence rate estimation of elitist clonal selection algorithm," *Acta Electronica Sinica*, vol. 43, no. 5, pp. 916–921, 2015, in Chinese.
- [17] H. Ebbinghaus, *Memory*, Columbia University, New York, NY, USA, 1913.
- [18] T. J. Ricker, E. Vergauwe, and N. Cowan, "Decay theory of immediate memory: from Brown (1958) to today (2014)," *Quarterly Journal of Experimental Psychology*, vol. 69, no. 10, pp. 1969–1995, 2016.
- [19] M. Anderson, "Rethinking interference theory: executive control and the mechanisms of forgetting," *Journal of Memory and Language*, vol. 49, no. 4, pp. 415–445, 2003.
- [20] Y. Shuai, B. Lu, Y. Hu, L. Wang, K. Sun, and Y. Zhong, "Forgetting is regulated through rac activity in *Drosophila*," *Cell*, vol. 140, no. 4, pp. 579–589, 2010.
- [21] Y. Liu, S. Du, L. Lv et al., "Hippocampal activation of Rac1 regulates the forgetting of object recognition memory," *Current Biology*, vol. 26, no. 17, pp. 2351–2357, 2016.
- [22] T. Tully, S. Boynton, C. Brandes et al., "Genetic dissection of memory formation in *Drosophila melanogaster*," *Cold Spring Harbor Symposia on Quantitative Biology*, vol. 55, no. 1, pp. 203–211, 1990.
- [23] L. N. De Castro and F. J. Von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 239–251, 2002.
- [24] B. Ulutas and A. A. Islier, "Dynamic facility layout problem in footwear industry," *Journal of Manufacturing Systems*, vol. 36, no. 36, pp. 55–61, 2015.
- [25] S. Surjanovic and D. Bingham, "Virtual library of simulation experiments: test functions and datasets," 2018, <http://www.sfu.ca/~ssurjano/optimization.html>.
- [26] W. Luo, X. Lin, T. Zhu, and P. Xu, "A clonal selection algorithm for dynamic multimodal function optimization," *Swarm and Evolutionary Computation*, vol. 50, 2019.