DNA Design Based on Improved Ant Colony Optimization Algorithm With Bloch Sphere

QIHANG ZHOU[1], XIAO WANG[2], AND CHANGJUN ZHOU[ID][1]

[1]College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua 321004, China
[2]Xingzhi College Zhejiang Normal University, Jinhua 321004, China

Corresponding author: Xiao Wang (tianzhu213@zjnu.cn)

ABSTRACT DNA computing and coding have good application prospects in data storage, data computing, data encryption and other fields. Meanwhile, it is very important to design a set of DNA coding set that meets a variety of constraints in today's research. The purpose of DNA coding is to find as many qualified code sets as possible under given conditions or to keep each DNA code in the DNA code set as far as possible from other codes. The former is used in this paper. The algorithm uses the Bloch system to initialize the DNA coding population, and uses the Ant Colony Algorithm to find the optimal DNA coding. At the same time, crossover and mutation operations are added to make the generated population more random and diverse. Experimental results show that the number of code sets obtained by this algorithm under certain specific conditions is better than the number of code sets obtained by other algorithms.

INDEX TERMS DNA computing, DNA design, bloch system, ant colony algorithm.

I. INTRODUCTION

DNA computing was proposed by Head T [1] in 1987. Adleman [2] proved head's conjecture with an innovative method in 1994. At the same time, the DNA coding set that meets the constraints plays an important role in various DNA fields. With the rapid development of DNA related technology, DNA coding technology is not only used in DNA computing, but also used in other technologies, such as data storage [3], DNA nanostructure [4], DNA microarray [5], image processing and encryption [6], [7]. Deng *et al.* [8] proposed an improved hybrid coding method of variable-length run-length limited (VL-RLL) coding and low-density parity-check (LDPC) coding based on DNA based data storage technology. The experimental results show that the hybrid coding method proposed in this paper has better performance than the current traditional DNA data storage technology. Immink *et al.* [9] proposed a sequence replacement method for K constraints and Q metadata. Experimental data show that this method has significant improvement over the existing replacement techniques. Calais *et al.* [10] proved that under typical concentration conditions, the maximum melting temperature and 14 base saturation provide a useful guidance for all technical

solutions, and the effectiveness of the optimization method is verified by experiments. Yin *et al.* [11] proposed a new nonlinear control strategy and an improved NOL-HHO algorithm. Experiments on several functions show that the algorithm can obtain a better lower bound of DNA storage and has stronger global search ability. Weber *et al.* [12] designed a DNA coding set with specified minimum distance and certain ability of error detection. Experiments show that the DNA coding set has good ability of error detection. Common constraints include Hamming distance constraint (HD), Reverse Complement Hamming distance constraint (RC) and GC constraint. The above three constraints are used in this paper, and they will be mentioned in the second chapter. According to the constraints mentioned above, many researchers and scholars are studying DNA coding ensemble now. Cao *et al.* [13] proposed a K-means multiple optimization algorithm (KMVO) based on Hamming distance constraint, GC constraint and other constraints. The algorithm can find better coding boundary than the previous multiple optimization algorithms (MVO). The experimental results also show that the algorithm can store more information more effectively in a given length, so as to improve the utilization of space. Kim *et al.* [14] proposed a DNA coding structure based on binary Hadamard matrix. The experimental data show that compared with the minimum complement Hamming

distance, the DNA coding structure with length of 8 or 16 in this paper is larger than that of all previous known results. Tulpan *et al.* [15] designed an algorithm for random, heuristic and linear structure of DNA strand satisfying Hamming distance constraint and complement constraints, and used GC constraints to preprocess DNA coding. The results show that the algorithm is better than other methods when using linear construction. Deaton *et al.* [16] proposed an algorithm that is stricter than Hamming distance and reverse Hamming distance constraints. Tulpan *et al.* [17], [18] proposed a random local search algorithm. Experiments show that the algorithm can find the DNA coding set with better quality.

The premise of DNA coding set design is to meet the corresponding constraints. The research on DNA coding can be divided into two aspects: (1) find the DNA coding with better quality under the condition of meeting the constraints [19]; (2) under the premise of meeting a certain quality, the more DNA coding sets found, and the better. The focus of this paper is the second case.

In this paper, we design an ant colony algorithm based on Bloch Sphere to find the coding set under various constraints. In this method, Bloch coordinates are regarded as three gene sequences, and each chromosome is composed of three genes. Then the ant colony algorithm is used to find the best initialization coding set. Compared with the random search algorithm, the search strategy based on genetic algorithm can search in a wider range, and the effect is better than the former. Through comparison, we can know that the results obtained by using the proposed search algorithm are better than the previous search algorithm.

II. CONSTRAINTS ON DNA CODING

A DNA [20] code of length n is composed of a group of $(x_1, \ldots, x_n)$ of $x_i \in \{A, G, C, T\}$ (T, C, G and A represent four nucleotides). The problem of DNA coding design is to find n coding sets that meet the constraints. Suppose that in the set s, the chromosome of X is $5' - x_1 x_2 \ldots x_n - 3'$, and the chromosome of Y is $5' - y_1 y_2 \ldots y_n - 3$. Given a distance D, the following constraint conditions will appear:

A. HAMMING DISTANCE CONSTRAINT (HD)

Hamming distance is the sum of the number of different bases in the same position of two codes with the same length. For different X and y, hamming distance is expressed as $H(x, y) \geq d$ where $H(x, y)$ represents Hamming distance of coded X and y, and D represents distance. The calculation formula of Hamming distance is as follows:

$$H(X, Y) = \sum_{i=1}^{n} h(x_i, y_i) \tag{1}$$

$$h(x_i, y_i) = \begin{cases} 0, & x_i = y_i \\ 1, & x_i \neq y_i \end{cases} \tag{2}$$

In formula 2, the Hamming distance is calculated for the i-th of code X and the i-th of code y (that is, the difference in the same position between the two codes is compared),

The same code is 0, and the different code is 1. For example, if the code x is 100000 and the code y is 011111, the Hamming distance of the two codes is 6. In Formula 1, n is the number of codes.

B. REVERSE COMPLEMENT HAMMING DISTANCE CONSTRAINT (RC)

In DNA coding, the code x and the code y may be equal. The Reverse Complement Hamming distance is expressed as $H\left(X, Y^{\text{RC}}\right) \geq d$, where $H\left(X, Y^{\text{RC}}\right)$ represents the Reverse Complement Hamming distance between the code x and the code y, and Y^{RC} is the Watson-Crick complement of Y.

In DNA computing experiments, single-stranded DNA molecules can diffuse freely in the solution, so they can be hybridized with the reverse code to describe the degree of difference between code x and code y. Experiments have shown that the greater the number of different bases between two DNA codes, the less complementary base data between them, and the less prone to non-specific hybridization.

C. GC CONSTRAINT (GC)

Every DNA code consists of four bases, T, C, G, and A, and G and C play a very important role in it. In DNA coding, the total number of bases G and C satisfies $\lfloor n/2 \rfloor$, that is $GC(x) = \lfloor n/2 \rfloor$. The calculation formula of GC is as follows:

$$GC(x) = \frac{|G| + |C|}{n} \tag{3}$$

where $|G|$ and $|C|$ represent the number of base G and base C of X in the DNA sequence respectively, and *n* represents the length of sequence X. For example, X=12343434, $GC(x) = 25\%$.

III. RELATED ALGORITHMS
A. BLOCH SPHERE CODING
1) BLOCH SPHERE CODING
In 2002, Ham and Kim [21] proposed to make use of the characteristics of chaos (such as randomness, convenience and regularity) to make the ethnic groups more diverse. Use Logistic mapping to generate r variables, the formula is as follows:

$$x_{i+1}^j = \mu x_i^j \left(1 - x_i^j\right), \quad j = 1, 2, 3, 4, \ldots, m \tag{4}$$

Among them, m is the number of variables, and the value of μ is generally between [0, 4]. If the value is greater than 4, the result will diverge, and n chromosomes will be generated through the above formula.

Bloch sphere is a unit of two-dimensional sphere, which is orthogonal to the corresponding points. The north and south poles of Bloch sphere correspond to the up and down states of electrons respectively, which can be expressed as 0 or 1, and vice versa. In a second-order quantum system, the possible state $|\psi\rangle$ can be represented by two bases which are mutually steamed, and these two bases can be represented by $|0\rangle$ and $|1\rangle$. In physics, $|0\rangle$ and $|1\rangle$ represent the only two results obtained by quantum measurement, which can be expressed

as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \alpha, \beta \in \mathbb{C}, \quad |\alpha|^2 + |\beta|^2 = 1$. On the three-dimensional Bloch sphere, a qubit can be expressed as $|\varphi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$, where $e^{i\delta}$ is called the global phase, which has the same effect on $|0\rangle$ and $|1\rangle$. The $e^{i\delta}$ in relative phase is different. Its influence formula is:$|\psi\rangle = \cos\theta|0\rangle + \sin\theta e^{i\phi}|1\rangle$, so the range of θ and ϕ can be determined:$0 \le \theta \le \frac{\pi}{2} \Rightarrow 0 \le 2\theta \le \pi, \quad 0 \le \phi < 2\pi$. The Bloch sphere can be obtained by drawing all the $\mathbb{R}^3$ distributed in three-dimensional space in 2θ and ϕ. The spherical formula is as follows:

$$x = \sin 2\theta \times \cos\phi \qquad (5)$$
$$y = \sin 2\theta \times \sin\phi \qquad (6)$$
$$z = \cos 2\theta \qquad (7)$$

The 3D Bloch sphere is shown below:

2) TRANSFORMATION OF SOLUTION SPACE
In DNA coding, it is necessary to express the 4 bases in real life with codes that can be read by a computer. At the same time, it is also necessary to establish a corresponding mathematical model. Its definition formula is as follows:

$$f(x) = \begin{cases} 0, & x = T \\ 1, & x = C \\ 2, & x = G \\ 3, & x = A \end{cases} \qquad (8)$$

From the above formula, we can see that the four bases T, C, G, and A of DNA in real life are represented by the four numbers 0, 1, 2, and 3 respectively, which facilitates the subsequent DNA coding work.

The value range of n chromosomes generated by formula (4) is between [-1,1], which does not meet the requirements of our DNA coding set, so we need to perform space transformation to convert n chromosomes into our own In the required space, the formula for solution space conversion is as follows:

$$X_{ix}^j = \text{round}\left\{\frac{1}{2}\left[b\left(1 + x_{ij}\right) + a\left(1 - x_{ij}\right)\right]\right\} - 1 \quad (9)$$
$$X_{iy}^j = \text{round}\left\{\frac{1}{2}\left[b\left(1 + y_{ij}\right) + a\left(1 - y_{ij}\right)\right]\right\} - 1 \quad (10)$$
$$X_{iz}^j = \text{round}\left\{\frac{1}{2}\left[b\left(1 + z_{ij}\right) + a\left(1 - z_{ij}\right)\right]\right\} - 1 \quad (11)$$

Because the number of bases selected in this paper is 4, the value of b is set to 4, and the value of a is set to 1. After the transformation of the above solution space formula, a set of DNA codes in the range [0, 4] can be obtained.

B. IMPROVED ANT COLONY ALGORITHM
Ant colony optimization (ACO)[22], also known as ant algorithm, is a probabilistic algorithm to find the optimal path, and also a simulated evolutionary algorithm. Its inspiration comes from the behavior of finding the optimal path in the process of searching for food in the animal kingdom.

Ant colony algorithm was first applied to TSP problem (Traveling Salesman Problem). After several years of development, ant colony algorithm can now be seen in other fields [23, 24]. Although the ant colony algorithm has achieved good performance in many fields, the most successful is its application in combinatorial optimization problems.

1) FUNDAMENTAL
Biologists have discovered that ants are polymorphic social creatures without vision. They rely on the pheromone on the way to find the best path. For example, if the ants are at an intersection that they have never passed before, they will randomly choose a path to travel, and at the same time release pheromone on the changed path. When the ants pass through the intersection, they will choose the one with a greater probability. The path of pheromone advances. At the same time, the pheromone on the path will evaporate over time, which will result in more and more ants passing through the optimal path, and the pheromone will become more and more concentrated. The probability of choosing this intersection will become greater and greater. Correspondingly, the pheromone on the path that is not chosen by the ant will become weaker and weaker, and the probability of the ant choosing this road will become less and less. In the end, the ants will choose the best path.

2) ANT COLONY TRANSFER STRATEGY
The DNA coding set S_k is structured as follows: First initialize a certain number of DNA code sets V(If there is a GC constraint, you only need to ensure that the GC content is $GC(x) = \lfloor n/2 \rfloor$ when initializing the DNA code set).

Ant k randomly selects a code from the code set V and adds it to S_k, and then constructs the candidate code set candidates in the set S_k. The definition formula is as follows:

$$candidates = H\left(v_i, v_j\right) \ge d, \quad H\left(v_i, v_j^R\right) \ge d \quad (12)$$

Because the constraints used in the above formula are Hamming distance constraints and Reverse Complement Hamming distance constraint, the corresponding constraints can also be selected according to the actual situation. Each code in the candidate code and all codes in the set S_k meet the corresponding constraints. Then calculate the probability of each code in the candidate code set, and then use roulette to select one of the codes to add to the set S_k, so as to repeat until the candidate set is empty. At this time, the first ant has completed its mission, and then the second ant also performs the corresponding operation until the last ant completes the operation. The formula of ant transfer strategy [25] with candidate codes added to set S_k is as follows:

$$\eta_i = d_i / \sum_{j \in \text{Candidates}} d_j \qquad (13)$$

$$p(v_i) = \frac{(\tau_i)^\alpha (\eta_i)^\beta}{\sum_{v_j \in \text{Candidates}} (\tau_j)^\alpha (\eta_j)^\beta} \qquad (14)$$

where d_i is the sum of the distances (Hamming distance and inverse Hamming distance) between the ith code and other codes, and η_i is the heuristic function. α is the pheromone factor and β is the heuristic function factor, The larger the value of β, the larger the proportion in formula (11). τ_i is the pheromone concentration of the ith code. From the above two formulas, we can see that the two major factors that affect our selection probability are pheromone concentration τ_i and heuristic function η_i, that is to say, the higher the pheromone concentration, the greater the probability of the path being selected. However, with the increase of coding length, the calculation time of $p(v_i)$ will increase, so a new formula is needed to reduce the calculation time. On the basis of the original ant transfer strategy, the simplified formula is simplified and a simplified version of ant transfer strategy is proposed. The simplified version of ant transfer strategy [26] is shown in formula (12).

$$p(v_i) = \frac{(\tau_i)^\alpha}{\sum_{v_j \in \text{Cand idates}} (\tau_j)^\alpha} \tag{15}$$

Compared with formula (11), formula (12) does not need to calculate the distance between codes, which greatly improves the calculation efficiency and shortens the calculation time. The basic ant colony algorithm will calculate the distance of all DNA codes when calculating the ant transfer strategy, and select the code with the highest probability to add to the candidate code, but this operation will greatly improve the computing time of the computer. The improved ant transfer strategy only needs to calculate the pheromone concentration, and the ant only needs to rely on the pheromone concentration At the same time, it also proves the universality of the improved ant transfer strategy, which can be combined with other algorithms to improve the robustness of the hybrid algorithm and reduce the running time of the algorithm.

3) UPDATE LOCAL PHEROMONE

The communication between ant colonies is completed by pheromone, which plays a crucial role in the cooperation of ant colony. When any ant K completes a coding set S_k, it will update the local pheromone. The update formula [27] is as follows:

$$\tau(v_i) = \begin{cases} \tau(v_i) - \tau_{\min} \cdot \dfrac{|S_{bs}|}{|S_k|} & v_i \in S_k \\ \tau(v_i) & \text{otherwise} \end{cases} \tag{16}$$

In the process of updating pheromones, the content of pheromones that do not belong to the coding set S_k remains unchanged, while those that belong to the coding set S_k remain unchanged The pheromone content in code a is reduced. The reason may be that when the nth ant chooses the path with higher pheromone concentration, it is more likely to choose the path with higher pheromone concentration. Therefore, when the local update is completed, the pheromone concentration not belonging to code set S_k is higher than that belonging to code set S_k, which leads to the ant preferentially choosing those not previously selected Thus, the diversity

of solution space is increased. The value of pheromone is generally fixed in a range, and when the local pheromone is updated, the pheromone content of the code will also change. At this time, it is necessary to determine whether the pheromone content of the code exceeds the specified range. If the content of a pheromone in the coding is less than the minimum value set before, the pheromone content is set to the minimum value. On the contrary, if pheromone content in the encoding is higher than the maximum, the pheromone content is set to the maximum. From the above formula, we can see that updating the local pheromone can be divided into two parts: the first part updates the pheromone, and the second part processes the pheromone beyond the specified range.

4) UPDATE GLOBAL PHEROMONE

After completing an iteration, all ants build DNA coding set, and then we need to update the global pheromone. Because the pheromone in natural environment evaporates with time, the algorithm also increases evaporation to simulate the phenomenon in nature. ρ is pheromone volatilization factor. $\Delta\tau(v_i)$ is the new pheromone content, Q is the pheromone constant, and L_k is the sum of the codes that meet the constraint conditions. According to different rules, ant colony algorithm can be divided into three models: Ant Cycle model, Ant Quantity model and Ant Density model. The model used in this paper is Ant Cycle model. Ant Cycle model means that ants release pheromones after completing a path cycle. The formula for updating global pheromone is as follows:

$$\tau(v_i) = (1 - \rho) \cdot \tau(v_i) + \Delta\tau(v_i) \tag{17}$$

$$\Delta\tau(v_i) = \sum_{k=1}^{m} \Delta\tau_{v_i}^k \tag{18}$$

$$\Delta\tau\left(v_i^k\right) = \begin{cases} \dfrac{Q}{L_k}, & v_i \in S_k \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

From the above three formulas, we can see that global pheromone updating includes two processes: increasing and evaporating. In nature, the pheromone content released by ants in the two places is not wireless superposition, because pheromone will slowly dissipate with the passage of time, and the evaporation of pheromone on the path that few ants pass will be higher than the increase, so the pheromone content on the path will be less and less until exhausted, in contrast, the increase of pheromone on the path that many ants pass So the pheromone on this path will be more and more. After completing iteration, the evaporation of pheromone on all codes is $\rho \cdot \tau(v_i)$, which leads to ants slowly forgetting the previous path and choosing the path they haven't found to search, thus increasing the diversity of understanding space search.

C. CROSSOVER

In evolutionary algorithm, if there is only update operation, the population will evolve in the same direction, which leads
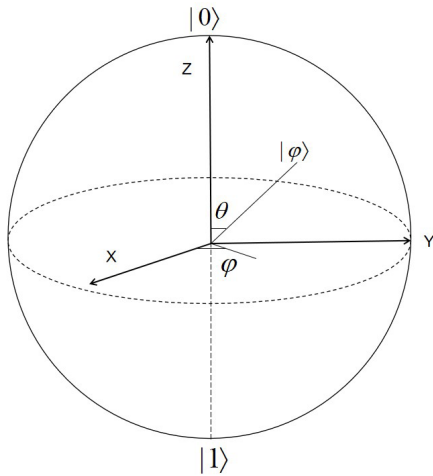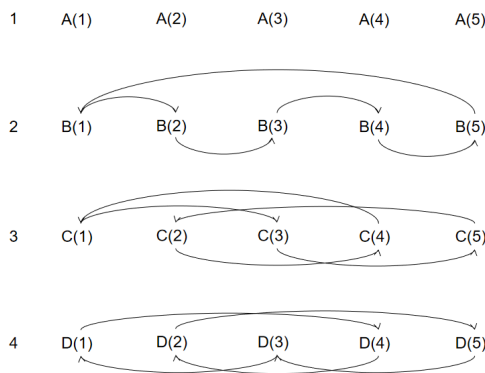
FIGURE 1. 3D Bloch sphere.

FIGURE 2. Results before crossover.

1	A(1)	A(2)	A(3)	A(4)	A(5)
2	B(5)	B(1)	B(2)	B(3)	B(4)
3	C(4)	C(5)	C(1)	C(2)	C(3)
4	D(3)	D(4)	D(5)	D(1)	D(2)

FIGURE 3. Results after crossover.

1	A(1)	A(2)	A(3)	A(4)	A(5)
2	B(1)	B(2)	B(3)	B(4)	B(5)
3	C(1)	C(2)	C(3)	C(4)	C(5)
4	D(1)	D(2)	D(3)	D(4)	D(5)

FIGURE 4. Results before mutation.

to premature algorithm and reduces the ability of searching solution space. Crossover operation can not only make the population more diverse, but also retain excellent individuals to the next generation. These two aspects play an important role in evolutionary algorithm. The common crossover strategies of evolutionary algorithm [28] are single point crossover, two-point crossover, multi-point crossover and uniform crossover. In order to make the population more diverse, the crossover method used in this paper is full crossover, that is, each gene of the population participates in the crossover, but this operation greatly increases the calculation time. Suppose the population size is 4, and each chromosome has 5 genes. The whole crossover process is shown in Figure 1, and the crossover result is shown in Figure 2.

D. MUTATION

Mutation operation is also an indispensable part of evolutionary algorithm. Mutation operation can prevent the algorithm from converging to the local optimum too prematurely to a certain extent. However, in order to ensure the stability of the algorithm, the mutation probability of the mutation operation is generally very small. The mutation method used in this article is to randomly select a gene in a chromosome to become another gene in the same chromosome when the

mutation requirements are met. Assuming that the population size is 4 and the number of genes per chromosome is 5, Figure 3 is the result before mutation, Figure 4 is the result after mutation, and the red part in Figure 4 is the value after mutation.

E. CALCULATE FITNESS

The constraints in this paper are combination constraints (Hamming distance constraint, Reverse Complement Hamming distance constraint, etc.), so the adaptive function of the algorithm is set to satisfy the Reverse Complement Hamming distance constraint. If the fitness function value is zero, all the code in the collection satisfies the combination constraint. The formula is as follows:

$$fitness(W) = \sum \max \left\{ 0, d - H\left(w_1, w_2^{RC}\right) \right\} \quad (20)$$

F. ALGORITHM DESCRIPTION

1) Use Bloch Sphere Coding to initialize the population.
2) Ant K starts to work and randomly selects a code from code s to add it to C_k.
3) Each ant constructs a candidate set based on its own set.
4) According to the ant transfer strategy (roulette), select a set from the candidate set to join the C_k.
5) Determine whether the candidate set is empty. If not, return to step 4. Otherwise, go to step 6.
6) Update the local pheromone.
7) Determine whether all ants have completed the update, if not, return to step 4, otherwise go to step 8.

1	A(1)	A(1)	A(3)	A(4)	A(5)
2	B(1)	B(2)	B(3)	B(5)	B(5)
3	C(4)	C(2)	C(3)	C(4)	C(5)
4	D(1)	D(2)	D(3)	D(4)	D(3)

FIGURE 5. Results after mutation.

8) Update the local optimal solution and determine whether the maximum number of iterations is reached, if not reached, update the global optimal solution and the global pheromone and return to step 2, otherwise go to step 9.
9) Obtain the global optimal solution S_{bs} and use it as the input of genetic algorithm.
10) Use Bloch Sphere Coding to initialize the set S.
11) Perform crossover, mutation and fitness calculation operations on the set S and update the code set.
12) Determine whether the set s is empty, if it is not empty, go to step 11, otherwise go to step 13.
13) Determine whether the maximum number of iterations has been reached, if not, go to step 10, otherwise go to step 14.
14) The final set S_{bs} is output, and the operation ends.

The algorithm flow chart is shown in Figure 5.

IV. EXPERIMENT

A. EXPERIMENTAL ENVIRONMENT
The CPU of this experiment is i7-10070, 16GB memory. MATLAB is used to experiment and compare the results.

B. EXPERIMENTAL DATA
In this manuscript, Bloch spherical coding method is used, which has more randomness and diversity than the traditional random coding method. Therefore, the coding method used in this manuscript is Bloch spherical coding (the population generated by Bloch spherical coding is the experimental data).

C. PARAMETER SETTINGS
The combination of ant colony algorithm and genetic algorithm is used to solve the problem of DNA coding set design. The CPU of this experiment is i7-10070, 16GB memory. MATLAB is used to experiment and compare the results. Table 1 shows the parameters of ant colony algorithm, and table 2 shows the parameters of genetic algorithm.

D. EXPERIMENTAL PROCESS
In this experiment, we first use Bloch spherical coding to initialize the population (the population obtained by Bloch

TABLE 1. Parameter details of ant colony algorithm.

Symbol	Meaning	Value
Count	Initialize collection size	100
M	Size of ant colony	20
ρ	Pheromone volatilization factor	0.01
Tau	Pheromone matrix	Ones(100,1)
α	Pheromone importance factor	[1,4]
β	Heuristic function factor	[0,5]
Q	Pheromone constant	1.5
Max_iter	Maximum number of iterations	[100,500]

TABLE 2. Genetic algorithm parameter details.

Symbol	Meaning	Value
Size	Size	100
Maxgen	Maximum number of iterations	10000
Pc	Crossover probability	0.7
Pm	Variation probability	0.1

TABLE 3. The meaning of corner mark.

Corner Mark	Meaning
sa	Simulated annealing algorithm [15]
f	Random search algorithm [15]
s	Random search algorithm [17,18]
m	Hybrid algorithm [29]
l	Domain or ring construction [30]
e	Evolutionary algorithms [31]
v	Dynamic search algorithm [15,31,32]
c	Dictionary method [29,32]
a	Random search algorithm [33]
d	The lower bound of the theory [34]
p	Deduction of proposition [32]
r	Linear and nonlinear construction [35]
u	Computer algebra system [36]

spherical coding can have randomness and diversity), then remove the same coding and put the coding into the improved ant colony algorithm to get the optimal one or more DNA coding sets, and then use genetic algorithm (crossover and mutation) to expand the DNA coding set, After many cycles, the final DNA coding set is obtained.

E. MEASURE STANDARD
In this experiment, Bloch spherical coding, improved ant colony algorithm and genetic algorithm are used to obtain DNA coding set, so the evaluation index of this experiment is to obtain as many DNA coding sets as possible under certain conditions (given constraints and DNA coding length).

F. EXPERIMENTAL RESULT
Table 3 shows the meaning of the upper corner of each experimental data, that is, the experimental results of each algorithm. Table 4 and table 5 are the maximum number of sets obtained by each algorithm of $A_4^{RC}(n, d)$ and $A_4^{GC,R\ C}(n, d, w)$ respectively. The value range of coding length is [4, 13], and the range of distance D is [3, n]. A in each column represents the previous experimental results (the letter in the upper right corner of each result represents the method used, see Table 3 for details), B represents the experimental results of this paper. The symbol "-" indicates
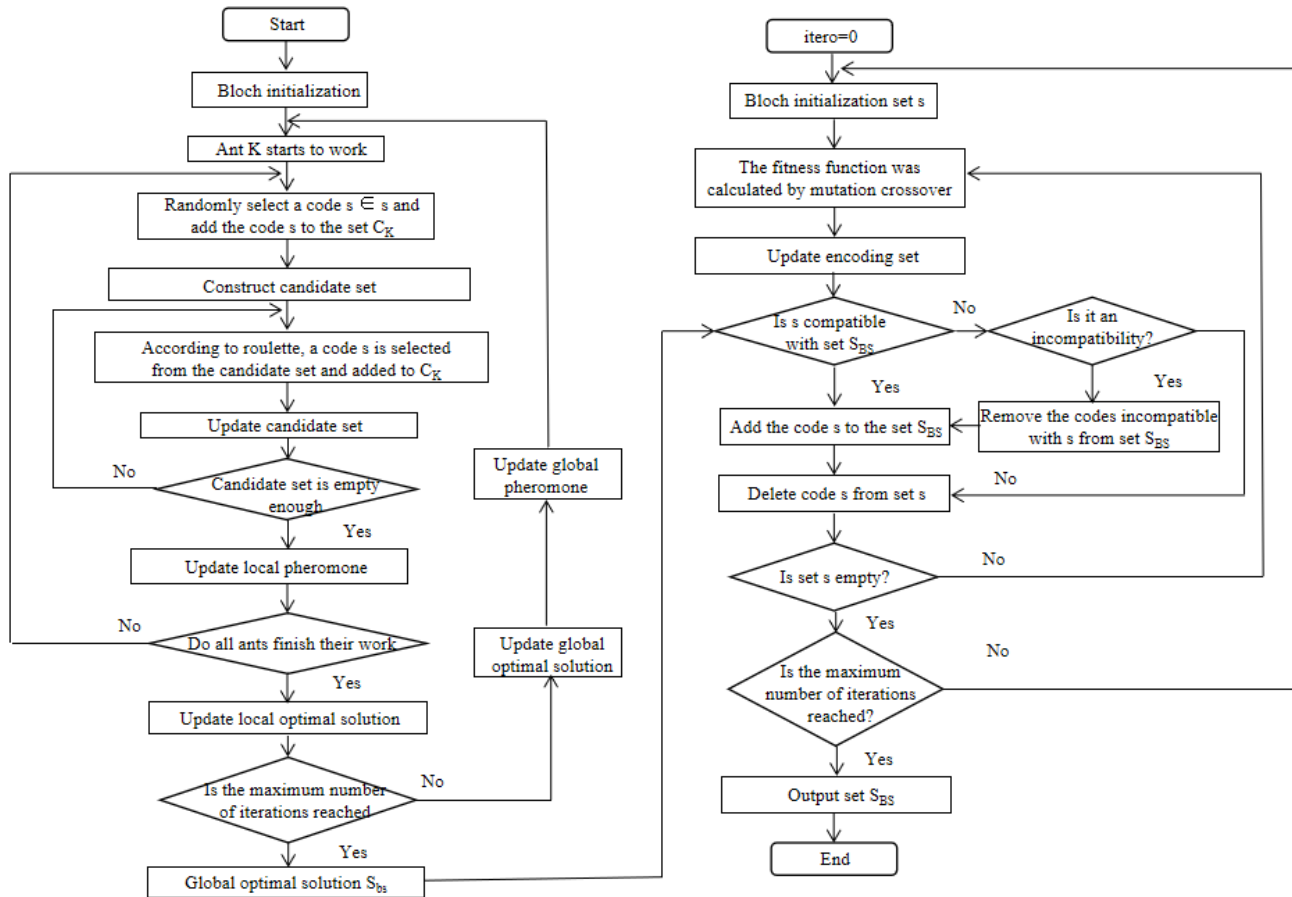
FIGURE 6. Algorithm flowchart.

that the value of distance D is greater than the length of the code. The number in bold indicates that the experiment in this paper is better than the previous experiment, and the symbol "." indicates that the experiment has not been run. In this paper, the value range of w is $\lfloor n/2 \rfloor$.

In Table 4, most of the previous research results come from Variable Neighborhood Search algorithm (VNS), simulated annealing algorithm, linear construction and nonlinear construction. It can be seen from table 4 and table 5 that some results in this experiment, such as $A_4^{RC}(12, 6)$ and $A_4^{GC,R\,C}(12, 6, 6)$, are far from the existing experimental results, and some results are equal to the previous results, such as $A_4^{RC}(9, 7) = 8$. There are also some cases where the number of codes in the new algorithm is larger than that of the existing codes in the range of coding length [4, 13], such as $A_4^{RC}(7, 3) = 197$, $A_4^{RC}(8, 3) = 633$, $A_4^{RC}(9, 3) = 2077$, $A_4^{RC}(11, 6) = 138$, $A_4^{GC,R\,C}(10, 3, 5) = 4552$, $A_4^{GC,R\,C}(11, 4, 5) = 2569$, etc. This shows that the algorithm can find more codes that meet the constraints in some cases. The reason why the algorithm based on Bloch sphere coding, improved ant colony algorithm and genetic algorithm can get

TABLE 4. The maximum number of codes obtained by $A_4^{RC}(n, d)$ set.

1	1	3	4	5	6	7	8	9	10	11	12
4	A	6^{sa}	2^{sa}	-	-	-	-	-	-	-	-
	B	6	2	-	-	-	-	-	-	-	-
5	A	32^s	4^s	2^s	-	-	-	-	-	-	-
	B	30	4	2	-	-	-	-	-	-	-
6	A	62^f	28^v	4^s	2^s	-	-	-	-	-	-
	B	61	20	4	2	-	-	-	-	-	-
7	A	196^f	42^s	12^{sa}	2^v	2^v	-	-	-	-	-
	B	**197**	41	11	2	2	-	-	-	-	-
8	A	620^f	128^l	29^v	16^{sa}	2^s	2^s	-	-	-	-
	B	**633**	124	29	12	2	2	-	-	-	-
9	A	1952^{sa}	345^v	80^{sa}	22^v	8^s	2^v	2^v	-	-	-
	B	**2077**	**359**	78	21	8	2	2	-	-	-
10	A	8064^l	2016^l	496^l	120^v	17^f	8^{sa}	2^v	2^v	-	-
	B	6897	1089	206	53	16	8	2	2	-	-
11	A	.	4832^l	607^v	136^v	40^v	15^f	6^v	2^v	2^v	-
	B	.	3520	603	**138**	40	13	6	2	2	-
12	A	.	.	4032^l	2016^l	120^l	31^s	12^v	4^f	2^f	2^v
	B	.	.	1689	336	75	24	10	4	2	2

better results is that: (1) the method of randomly generated coding is replaced by Bloch sphere coding, which makes the generated population more random, so as to achieve a larger search range, and also has a certain impact on the subsequent search solution space (2) compared with random

TABLE 5. The maximum number of codes obtained by $A_4^{GC,RC}(n,d,w)$ set.

1	1	3	4	5	6	7	8	9	10	11	12
4	A	$6^{s,l}$	2^p	-	-	-	-	-	-	-	-
	B	6	2	-	-	-	-	-	-	-	-
5	A	15^l	3^l	1^p	-	-	-	-	-	-	-
	B	15	3	1	-	-	-	-	-	-	-
6	A	44^a	16^l	4^l	2^p	-	-	-	-	-	-
	B	42	13	4	1	-	-	-	-	-	-
7	A	135^a	36^a	11^m	2^l	2^p	-	-	-	-	-
	B	129	34	11	2	1	-	-	-	-	-
8	A	528^m	128^d	28^a	$12^{s,m}$	2^l	1^p	-	-	-	-
	B	434	101	23	8	2	**2**	-	-	-	-
9	A	1354^m	275^a	67^a	21^v	8^m	2^l	1^p	-	-	-
	B	1298	**280**	65	18	7	2	1	-	-	-
10	A	4542^m	860^u	210^u	54^c	17^v	8^l	2^p	2^p	-	-
	B	**4552**	**865**	167	**55**	14	6	2	2	-	-
11	A	.	2457^m	477^a	117^a	37^v	14^v	5^m	2^m	1^p	**-**
	B	.	**2569**	477	117	29	11	5	2	1	-
12	A	.	.	1381^v	924^c	87^v	29^v	12^v	$4^{s,l}$	2^p	2^p
	B	.	.	**1382**	236	65	23	10	4	2	2

search, the coding set obtained by crossover and mutation of genetic algorithm can approach the optimal space, while random search has no such evolutionary mechanism, which is blind search. Although the algorithm can obtain better coding set in some cases, the coding set obtained in many cases is not better than the previous one, which indicates that the algorithm still has a lot of room for improvement. At the same time, the latest intelligent optimization can be added to the algorithm to obtain better performance.

V. CONCLUSION

In this paper, we combine Bloch sphere coding, ant colony algorithm and genetic algorithm to design a DNA coding method that meets the constraints. Bloch spherical coding method is more diverse than the traditional random generation method, which makes the search space of the algorithm larger and more able to find the DNA coding set that meets the constraints; ant colony algorithm has better stability and global search ability; genetic algorithm makes the population better adapt to the new environment and expand the existing solutions through mutation crossover and other operations Space.

Through experiments, we find that the size of DNA coding set found by the proposed algorithm in some cases is better than the existing results. This also proves the feasibility of the algorithm to a certain extent, and provides a strong support for the future research of DNA coding set.

REFERENCES

[1] T. Head, "Formal language theory and DNA: An analysis of the generative capacity of specific recombinant behaviors," *Bull. Math. Biol.*, vol. 49, no. 6, pp. 737–759, Nov. 1987.

[2] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021–1024, 1994.

[3] N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E. M. LeProust, B. Sipos, and E. Birney, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, pp. 77–80, Jan. 2013.

[4] S. D. Perrault and W. M. Shih, "Virus-inspired membrane encapsulation of DNA nanostructures to achieve *in vivo* stability," *Acs Nano.*, vol. 8, no. 5, pp. 5132–5140, 2014.

[5] A. Alonso, V. Larraga, and P. J. Alcolea, "The contribution of DNA microarray technology to gene expression profiling in *Leishmania* spp.: A retrospective view," *Acta Tropica*, vol. 187, pp. 129–139, Nov. 2018.

[6] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Process.*, vol. 155, pp. 44–62, Feb. 2019.

[7] X. Chai, Y. Chen, and L. Broyde, "A novel chaos-based image encryption algorithm using dna sequence operations," *Opt. Lasers Eng.*, vol. 88, pp. 197–213, Jan. 2017.

[8] L. Deng, Y. Wang, M. Noor-A-Rahim, Y. L. Guan, Z. Shi, E. Gunawan, and C. L. Poh, "Optimized code design for constrained DNA data storage with asymmetric errors," *IEEE Access*, vol. 7, pp. 84107–84121, 2019.

[9] K. A. S. Immink and K. Cai, "Design of capacity-approaching constrained codes for DNA-based storage systems," *IEEE Commun. Lett.*, vol. 22, no. 2, pp. 224–227, Feb. 2018.

[10] T. Calais, V. Baijot, M. D. Rouhani, D. Gauchard, Y. J. Chabal, C. Rossi, and A. Estève, "General strategy for the design of DNA coding sequences applied to nanoparticle assembly," *Langmuir*, vol. 32, no. 37, pp. 9676–9686, Sep. 2016.

[11] Q. Yin, B. Cao, X. Li, B. Wang, Q. Zhang, and X. Wei, "An intelligent optimization algorithm for constructing a DNA storage code: NOL-HHO," *Int. J. Mol. Sci.*, vol. 21, no. 6, p. 2191, Mar. 2020.

[12] J. H. Weber, J. A. M. De Groot, and C. J. Van Leeuwen, "On single-error-detecting codes for dna-based data storage," *IEEE Commun. Lett.*, vol. 25, no. 1, pp. 41–44, Jan. 2021.

[13] B. Cao, S. Zhao, X. Li, and B. Wang, "K-means multi-verse optimizer (KMVO) algorithm to construct DNA storage codes," *IEEE Access*, vol. 8, pp. 29547–29556, 2020.

[14] Y.-S. Kim, H. Park, and S.-H. Kim, "DNA codes with constant GC-content constructed from Hadamard matrices," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E100.A, no. 11, pp. 2408–2415, 2017.

[15] D. Tulpan, D. H. Smith, and R. Montemanni, "Thermodynamic post-processing versus GC-content pre-processing for DNA codes satisfying the Hamming distance and reverse-complement constraints," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 11, no. 2, pp. 441–452, Mar. 2014.

[16] R. Deaton, R. C. Murphy, M. Garzon, D. R. Franceschetti, and S. E. Stevens, "Genetic search of reliable encodings for DNA based computation," in *Proc. 1st Annu. Conf. Genetic Program.*, 1996, p. 421.

[17] D. C. Tulpan, H. H. Hoos, and A. E. Condon, "Stochastic local search algorithms for DNA word design," in *DNA Computing* (Lecture Notes in Computer Science), vol. 2568. 2002, pp. 229–241.

[18] D. C. Tulpan and H. H. Hoos, "Hybrid randomised neighborhoods improve stochastic local search for DNA code design," in *Advances in Artificial Intelligence*, vol. 2671. 2003, pp. 418–433.

[19] J. M. Chaves-González and M. Vega-Rodríguez, "DNA strand generation for DNA computing by using a multi-objective differential evolution algorithm," *Biosystems*, vol. 116, pp. 49–64, Feb. 2014.

[20] J. D. Watson and F. H. Crick, "A structure for deoxyribose nucleic acid," *Nature*, vol. 421, no. 6921, pp. 397–398, 2003.

[21] K. H. Han and J. H. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proc. Congr. Evol. Comput.*, 2002, pp. 1354–1360.

[22] W. Deng, J. Xu, and H. Zhao, "An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem," *IEEE Access*, vol. 7, pp. 20281–20292, 2019.

[23] Y. Q. Gao, H. B. Guan, Z. W. Qi, Y. Hou, and L. Liu, "A multi-objective ant colony ststem algorithm for virtual machine placement in cloud computing," *J. Comput. Syst. Sci.*, vol. 79, no. 8, pp. 1230–1242, 2013.

[24] M. Mahi, Ö. K. Baykan, and H. Kodaz, "A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem," *Appl. Soft Comput.*, vol. 30, pp. 484–490, May 2015.

[25] H. Jiang, J. Zhang, and J. Xuan, "A hybrid ACO algorithm for the next release problem," in *Proc. Int. Conf. Softw. Eng. Data Ming.*, 2017, vol. 43, no. 14, pp. 166–171.

[26] Y. Wang and J. Xie, "An adaptive ant colony optimization algorithm and simulation," *Acta Simulata Systemica*, vol. 14, no. 1, pp. 31–43, 2002.

[27] J. E. Bell and P. R. McMullen, "Ant colony optimization techniques for the vehicle routing problem," *Adv. Eng. Inform.*, vol. 18, no. 1, pp. 41–48, 2004.

[28] A. Kaveh and V. Kalatjari, "Topology optimization of trusses using genetic algorithm, force method and graph theory," *Int. J. Numer. Methods Eng.*, vol. 58, no. 5, pp. 771–791, 2003.

[29] P. Gaborit and O. D. King, "Linear constructions for DNA codes," *Theor. Comput. Sci.*, vol. 334, nos. 1–3, pp. 99–113, Apr. 2005.

[30] R. Montemanni, D. H. Smith, and N. Koul, "Three metaheuristics for the construction of constant GC-content DNA codes," *Lect. Notes Manage. Sci.*, vol. 6, no. 14, pp. 167–175, Jul. 2014.

[31] R. Montemanni and D. H. Smith, "Construction of constant GC-content DNA codes via a variable neighbourhood search algorithm," *J. Math. Model. Algorithms*, vol. 7, no. 3, pp. 311–326, Sep. 2008.

[32] O. D. King, "Bounds for DNA codes with constant GC-content," *Electron. J. Combinatorics*, vol. 10, no. 1, pp. 33–38, Sep. 2003.

[33] Y. M. Chee and S. Ling, "Improved lower bounds for constant GC-content DNA codes," *IEEE Trans. Inf. Theory*, vol. 54, no. 1, pp. 391–394, Jan. 2008.

[34] O. D. King. (Jan. 2008). *Tables of Lower Bounds for DNA Codes with Constant GC-Content*. [Online]. Available: http://llama.med.harvard.edu/~king/dnacodes.html

[35] D. H. Smith, N. Aboluion, and R. Montemanni, "Linear and nonlinear constructions of DNA codes with Hamming distance d and constant GC-content," *Discrete Math.*, vol. 311, no. 13, pp. 1207–1219, 2011.

[36] A. Niema, "The construction of DNA codes using a computer algebra system," Ph.D. dissertation, Dept. Math. Stat., Fac. Adv. Technol., Univ. Glamorgan, Pontypridd, U.K., 2011.

XIAO WANG was born in Wuyi, China, in 1977. She received the master's degree in mechanical and electronic engineering from the Zhejiang University of Technology. She is a Lecturer in Xingzhi College Zhejiang Normal University and has published more than ten papers. Her research interests include intelligent computing and computer aided design.

QIHANG ZHOU was born in Jiangshan, China, in 1996. He received the bachelor's degree from the Xingzhi College Zhejiang Normal University, in 2019. He is currently pursuing the master's degree in computer science and technology with Zhejiang Normal University, Jinhua, Zhejiang, China. His research interests include DNA coding design, machine learning, and deep learning.

CHANGJUN ZHOU was born in Shangrao, China, in 1977. He received the Ph.D. degree in mechanical design and theory from the School of Mechanical Engineering, Dalian University of Technology, Dalian, in 2008. He is currently a Professor with Zhejiang Normal University. His research interests include intelligence computing, pattern recognition, and DNA computing. He has published 60 articles in these areas.

• • •