IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 5, 2017, accepted September 30, 2017, date of publication October 9, 2017,
date of current version November 7, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2760882

DNA Code Design Based on the Bloch Quantum Chaos Algorithm

QINGJI GUO[1], BIN WANG[1], CHANGJUN ZHOU[1], XIAOPENG WEI[1,2], AND QIANG ZHANG[2]
[1]Key Laboratory of Advanced Design and Intelligent Computing, Ministry of Education, Dalian University, Dalian 116622, China
[2]School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China

Corresponding author: Qiang Zhang (zhangq@dlut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61425002, Grant 61772100, Grant 61402066, Grant 61672121, Grant 61672051, Grant 61572093, Grant 61402067, Grant 61370005, and Grant 31370778; in part by the Program for Changjiang Scholars and Innovative Research Team in University under Grant IRT_15R07; in part by the Program for Liaoning Innovative Research Team in University under Grant LT2015002; in part by the Basic Research Program of the Key in Liaoning Province Educational Department under Grant LZ2014049 and Grant LZ2015004; in part by the Scientific Research Fund of Liaoning Provincial Education under Grant L2015015 and Grant L2014499; and in part by the Program for Liaoning Key Lab of Intelligent Information Processing and Network Technology in University.

ABSTRACT The design of DNA codes sets satisfying certain combinatorial constraints is important in reducing hybridization errors due to nonspecific hybridization between distinct codes and their complements in DNA computing, data encryption, and data storage. The DNA code design problem is to find the largest possible set of DNA codes. In this paper, we present a Bloch quantum chaos algorithm for the designing DNA codes sets. The algorithm uses the chaotic equation to initialize the Bloch coordinates of quantum bits, next adopts the whole interference crossover strategy to get crossover operation, then employs quantum non gate strategy to get mutation operation, finally utilizes the dynamic adjustment strategy to adjust the quantum rotation corner. Several experimental results in which our algorithm finds DNA codes sets match or exceed the best previously known constructions.

INDEX TERMS DNA code design, chaotic, whole interference crossover, Bloch quantum chaos algorithm.

I. INTRODUCTION

Sets of DNA codes that satisfy combinatorial constraints play an important role in information storage and retrieval. They are widely applied into DNA computing [1], DNA Microarray technology [2], Molecular bar codes for chemical libraries [3], image encryption [4], [5] and data storage [6]. The reliability of these applications is dependent on specific hybridization between a DNA code and its Watson-Crick complement, but nonspecific hybridization may also occur between a DNA code and the reverse of a distinct code [7]. Combinatorial constraints can effectively limit nonspecific hybridization in specific applications.

Common combinatorial constraints [8] are comprised by hamming distance constraints (HD), reverse Watson-Crick complement constraints (RC) and fixed GC content constraints (GC). The first two constraints are used to avoid undesirable hybridization between different DNA strands and the last constrain ensures that all the codes have similar thermodynamic characteristics to perform uniform computation [9]. The details of these constraints are defined in Section 2.

Based on the above constraints, many researchers have worked on the design of sets of DNA codes. Constructive lower bounds for DNA codes using coding theory was proposed by Marathe *et al.* [8]. Deaton *et al.* [10] proposed genetic algorithms for designing DNA codes that satisfy frame shifts constraint, which was stronger than the HD and RC. Tulpan *et al.* [11], [12] used a stochastic local search (SLS) algorithm to generate reliable DNA codes. In their approach, they used HD, RC , GC constrains and hybrid them. Because their algorithms framework has a good flexibility, it can be easily applied to other constraints. Kobayashi *et al.* [13] presented a template-map strategy for designing DNA codes. The template method is easy to construct specific DNA code, though the distance parameter cannot be large. Zhang *et al.* [14] advanced the improved dynamic genetic algorithm to design DNA code sets based on minimum free energy (MFE) constraints. Montemanni and Smith [15] introduced a variable neighborhood search (VNS) algorithm into design DNA code sets. Their method is composed seeding building, clique

search, hybrid search and iterated Greedy Search. This aims to change the neighborhood structure over time, and thus could obtain a larger search space. Kawashimo *et al.* [16] proposed dynamic neighborhood search (DNS) algorithm to design DNA sequence sets. Their approach is easy to handle many types of hamming distance based constraints. Gaborit and King [17] first presented linear constructions of code. Smith *et al.* [18] further extended this construction for nonlinear code and cyclic code. Simulated Annealing (SA) is a metaheuristic algorithm derived from thermodynamic principles. Gamal *et al.* [19] put forward SA to construct good source codes, error-correcting codes, and spherical codes. Montemanni *et al.* [20] applied SA to design DNA code sets. In recent work, computer algebra systems Niema [21] were also used to construct DNA codes that satisfy GC and HD constrains. Different mapping from fields [22], [23] or rings [24], [25] to DNA codes could produce different lower bounds. Aforementioned all the methods were based on the traditional coding theory and heuristic algorithm to construct DNA codes. Hong *et al.* [26] first introduced algebraic number theory into DNA codes design with large n (<1000) and number of codes (M < 7000) satisfying the GC constraint.

In this context, we present a bloch quantum chaos algorithm for the design of DNA codes sets. In this method, the bloch coordinates of each qubit are treated as three paratactic genes, each chromosome consist of three gene chains, and each of chains represents an optimization solution that is a DNA code. Quantum rotation corner is updated by the dynamic adjustment strategy. The bloch coordinates of qubits are updated by quantum rotation gates, and are mutated by quantum non-gates, are crossed by whole interference crossover strategy. By comparing the sizes of DNA codes sets obtainable by our method with previous work, our results match or exceed the best previously known constructions.

II. CONSTRAINTS ON DNA CODES

A DNA code of length n is a set of codes $(x_1, \ldots, x_n)$ with $x_i \in \{A, G, C, T\}$(representing the four nucleotides in DNA). The problem of designing DNA code set is to construct the largest sets which satisfy combinatorial constraints. Let DNA code x and y are $x = 5' - x_1x_2 \ldots x_n - 3'$ and $y = 5' - y_1y_2 \ldots y_n - 3'$ in the set S, respectively. Given a distance parameter d, the constraints considered here are as follows:

A. HAMMING DISTANCE CONSTRAINT (HD)

For all pairs of distinct code x and y, the HD constraint specifies that $H(x, y) \geq d$; where $H(x, y)$ denotes the Hamming distance between codes x and y; that is, the number of positions i at which the *ith* letter in x differs from the *ith* letter in y. The hamming distance formula for this calculation is:

$$H(x, y) = \sum_{i=1}^{n} h(x_i, y_i), h(x_i, y_i) = \begin{cases} 0, & x_i = y_i \\ 1, & x_i \neq y_i \end{cases} \quad (1)$$

In the design of DNA code, Hamming distance is used to describe the degree of dissimilarity between two DNA sequences. A higher value of $H(x, y)$ indicates that the more number of different bases between the two DNA code x and y, then the same number of bases is less, thus the number of complementary bases between the code x and y^C(the Watson-Crick complement of DNA code y) is less. Eventually, the probability of a nonspecific hybridization between x and y^C is smaller.

For example: Let $x = ATGACT$ and $y = ACTAGC$, then $H(x, y) = 4$.

B. REVERSE COMPLEMENT HAMMING DISTANCE CONSTRAINT (RC)

For all pairs of code x and y, where x may equal y, the RC constraint specifies that $H(x, y^{RC}) \geq d$; where $H(x, y^{RC})$ denotes the reverse complement hamming distance between codes x and y. In addition, y^{RC} denotes the Watson-Crick complement of DNA code y; y^{RC} is the code which reverse y and replaced each A in y by T and vice versa, replaced each G in y by C and vice versa.

In the DNA computing experiment, single stranded DNA molecules are freely expanded in solution, so x may hybridize with the reverse code y of y^R. $H(x, y^{RC})$ is to used describe the degree of dissimilarity between DNA sequences x and y^{RC}. A higher value of $H(x, y^{RC})$ indicates the more number of different bases between the two DNA code x and y^{RC}, then the less number of complementary base pairs between DNA code x and y^R, so it is difficult to occur non-specific hybridization between DNA code x and y^R.

For example: Let $x = ATGACT$ and $y = GTACAC$, then $H(x, y^{RC}) = 4$.

C. GC CONSTRAINT

A fixed number w of the letters within each code is either G or C. Throughout; we assume that the number w is $\lfloor n/2 \rfloor$. For a code x the number of letters which are G or C is denoted $GC(x)$. the GC constraint specifies that $GC(x) = \lfloor n/2 \rfloor$. We use the following formula to calculate the content of $GC(x)$:

$$GC(x) = \frac{|C| + |G|}{|x|} \quad (2)$$

where $|G|$ and $|C|$ respectively represent the number of G and C in the code x; and $|x|$ is the length of code x.

For example: Let $x = ATGACC$, then $GC(x) = 50\%$.

D. FITNESS FUNCTION

The optimization problem is defined by the problem of minimum value. The model of the problem is as follows:

$$fitness(s_i) = \sum_{s_i, s_j \in W, s_i \neq s_j} \max\{0, d - H(s_i, s_j)\} + \sum_{s_i, s_j \in W} \max\{0, d - H(s_i, s_j^{RC})\} \quad (3)$$

The target is to repeatedly modify the code s_i towards a feasible solution (*i.e* $fitness(s_i) = 0$). If the GC constraint is considered, we check that each generated code accepts only codes with the specified GC content.

In this study, we have chosen to construct DNA code sets that satisfy two or three of above constrains. Let $A_4^{RC}(n, d)$ denotes the maximum numbers of a DNA code of length n that satisfies the HD and RC constraints for a given parameter d, and let $A_4^{GC,RC}(n, d)$ denotes the maximum numbers of a DNA code of length n that satisfies the HD, RC and GC constraints for a given parameter d and w.

III. THE PRINCIPLE OF THE BLOCH QUANTUM CHAOS ALGORITHM

In 1996, Narayanan and Moore [27] introduced the concept of quantum into genetic algorithm for the first time, and proposed quantum inspired genetic algorithm, which successfully solved the TSP problem. However, the quantum meaning of the algorithm is not very obvious, and it is similar to the isolation niche genetic algorithm. In 2000, Han and Kim [28] proposed a genetic quantum algorithm which was used to solve a class of combinatorial optimization problems effectively. At the same time he introduced the concept of quantum bits and quantum gates. Chaos is a common phenomenon in nature with confusing behaviors, but it has a delicate internal regularity. Chaotic motion has the property of ergodicity, which is more advantageous than the blind random search in optimizing search. The ergodicity can avoid the shortcomings of the evolutionary algorithm into the local optimal as much as possible [29]. Chaos optimization algorithm is more suitable for searching relatively small space, and the optimization effect may not very satisfactory when the search space is relatively large. In recent years, Chaos optimization algorithm has achieved good performance in solving the well-known traveling salesman, 0-1 Knapsack and image encryption [30], [31] problem. In this study, we integrate chaos optimization into quantum evolutionary algorithm to design DNA code.

A. CHAOTIC SYSTEM

Chaos is a widely existing nonlinear phenomenon. In a certain range, it does not iterate through all possible states according to its own ''rules''. In chaos optimization, we use the logistic model to generatechaos variable [32]. The logistic map is a mathematical map which statistics the insect number that changes with time under certain geographical scope and living conditions. That is $x_{n+1} = \mu x_n(1-x_n)$. x_n is a chaos variable, $x_n \in [0, 1]$, μ is a control parameter. If $\mu > 4$, then the model will diverges, so the range of μ is between 3.6 and 4. If $\mu = 4$, then the system enters the complete chaotic state.

B. THREE CHAINS ENCODING METHOD FOR QUANTUM CHROMOSOME

In classical calculations, information is represented by binary numbers 0 and 1, which are often referred to as bits. In quantum computing, $|0\rangle$ and $|1\rangle$ are used to represent the state of microscopic particles, which are called quantum bits, also known as quantum. ''$|\rangle$'' is called the Deakra mark which represents the meaning of state in quantum mechanics [33].

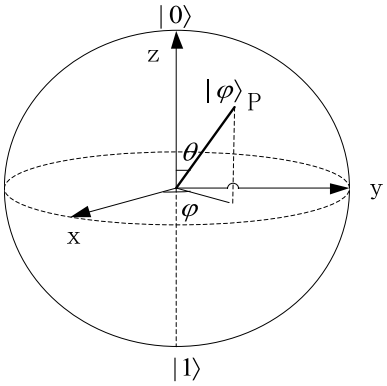


FIGURE 1. Bloch sphere representation of a qubit.

Quantum bits can not only represent two basic states of $|0\rangle$ and $|1\rangle$, but also the linear superposition states of these two states. On a three-dimensional bloch sphere, a qubit can be described as follows: $|\varphi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$, where $\cos\frac{\theta}{2}$ and $e^{i\varphi}\sin\frac{\theta}{2}$ are complex numbers $\left|\cos\frac{\theta}{2}\right|^2$ and $\left|e^{i\varphi}\sin\frac{\theta}{2}\right|^2$ represent the probability that a qubit are in states $|0\rangle$ and $|1\rangle$, respectively, and satisfy the normalization conditions as follows: $\left|\cos\frac{\theta}{2}\right|^2 + \left|e^{i\varphi}\sin\frac{\theta}{2}\right|^2 = 1$. As shown in the following figure, on the bloch sphere, a point P can be determined by two angle parameters θ and φ. At the same, we can also know that each qubit corresponds to one point on the bloch sphere. So, the qubit $|\varphi\rangle$ can be expressed by bloch coordinates as $|\varphi\rangle = [\cos\varphi\sin\theta, \sin\varphi\sin\theta, \cos\theta]^T$.

C. POPULATION INITIALIZATION

Using the Logistic map to generate r chaotic variables

$$x_{n+1}^i = \mu x_n^i(1 - x_n^i), \quad i = 1, 2, 3, 4, \ldots, r \tag{4}$$

where $\mu = 4$, i is the serial number of the chaotic variable. Set n=0, different initial values are given to r chaotic variables x_0^i $(i = 1, 2, \ldots r)$ respectively r Chaotic variables $x_1^i(i = 1, 2, \ldots r)$ are produced by the chaotic equations of (4).These r chaotic variables $x_1^i(i = 1, 2, \ldots r)$ initialize the first bloch spherical coordinates of qubit in the population.

Let $n = 2, 3, 4, \ldots, N-1$, then the remaining chromosomes are produced according to the method described above. Taking the Nth chromosome of initial results for example:

$$p_n = \begin{vmatrix} \cos\varphi_{in}\sin\theta_{in} \\ \sin\varphi_{in}\sin\theta_{in} \\ \cos\theta_{in} \end{vmatrix} \tag{5}$$

where $\varphi_{in} = 2\pi \times x_n^i$, $\theta_{in} = \pi \times x_n^i$, as can be seen from the above equation, a chromosome contains three gene chains, called x chain, y chain and z chain. Each gene chain represents an optimal solution. Thus, the three optimal solutions represented by the three chromosomes:

$$\begin{aligned} p_{ix} &= (\cos\varphi_{i1}\sin\theta_{i1}, \ldots, \cos\varphi_{in}\sin\theta_{in}) \\ p_{iy} &= (\sin\varphi_{i1}\sin\theta_{i1}, \ldots, \sin\varphi_{in}\sin\theta in) \\ p_{iz} &= (\cos\theta_{i1}, \ldots, \cos\theta_{in}) \end{aligned} \tag{6}$$

TABLE 1. The query table $\Delta\varphi$ of and $\Delta\theta$.

$\Delta\varphi$		$\Delta\theta$	
$A \neq 0$	$-\text{sgn}(A)$	$B \neq 0$	$-\text{sgn}(B)$
$A = 0$	0	$B = 0$	0

D. SOLUTION SPACE TRANSFORM

The code problem is transformed into a mathematical language description, and the following definitions are given: $f(x) : \{T, C, G, A\} \rightarrow \{0, 1, 2, 3\}$.

$$f(x) = \begin{cases} 0, & x = T \\ 1, & x = C \\ 2, & x = G \\ 3, & x = A \end{cases} \tag{7}$$

From the above formula, we can see that the range of optimization problem space $A = \{0, 1, 2, 3\}$ is. Because the range of the bloch spherical coordinates of the qubit is limited to the unit space $I^n = [-1, 1]^n$, we have to convert the unit space to the specific optimization problem space. In the population, each chromosome contains three bloch spherical coordinates. By using the linear transformation, the bloch sphere coordinates of the unit space can be transformed into the solution space of the optimization problem Ω. Each optimization variable in the solution space corresponds to the bloch sphere coordinates. If the *jth* qubit on p_i chromosome is $[x_{ij},\ y_{ij},\ z_{ij}]^T$, then the corresponding solution space transformation formula is as follows:

$$\begin{aligned} X_{ix}^j &= round\{\frac{1}{2}[b(1 + x_{ij}) + a(1 - x_{ij})]\} - 1 \\ Y_{iy}^j &= round\{\frac{1}{2}[b(1 + y_{ij}) + a(1 - y_{ij})]\} - 1 \\ Z_{iz}^j &= round\{\frac{1}{2}[b(1 + z_{ij}) + a(1 - z_{ij})]\} - 1 \end{aligned} \tag{8}$$

where $b = Max(A) + 1$ and $a = Min(A) + 1$. From the above definition, we can know that the design problem of DNA coding is a discrete problem. So we should convert the final result into an integer.

E. UPDATE OF QUANTUM ROTATING GATE

In the quantum evolutionary algorithm, population updates are achieved through quantum gates. Common quantum gates include XOR gates, controlled XOR gates, HadamardH gates, and rotary gates [33]. Choosing the right quantum gates to update the population is the key to the design of quantum evolutionary algorithms. The quality of the quantum gates directly affects the performance of the algorithm. The parameters of the quantum rotation gate can be adjusted arbitrarily and have strong versatility, so the quantum rotation gate is used to update the population. Quantum rotating gate can make each of the current population approaching the current optimal chromosomes. In this process of approaching, it is possible to produce better chromosomes. The quantum

TABLE 2. Before the whole interference crossover.

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

TABLE 3. After the whole interference crossover.

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

rotation gate U used is shown below:

$$U = \begin{bmatrix} \cos\Delta\varphi\cos\Delta\theta & -\sin\Delta\varphi\cos\Delta\theta & \sin\Delta\theta\cos(\varphi+\Delta\varphi) \\ \sin\Delta\varphi\cos\Delta\theta & \cos\Delta\varphi\cos\Delta\theta & \sin\Delta\theta\sin(\varphi+\Delta\varphi) \\ -\sin\Delta\theta & -\tan(\varphi/2)\sin\Delta\theta & \cos\Delta\theta \end{bmatrix} \tag{9}$$

$$\begin{aligned} U \begin{bmatrix} \cos\varphi\sin\theta \\ \sin\varphi\sin\theta \\ \cos\theta \end{bmatrix} \\ = \begin{bmatrix} \cos(\varphi + \Delta\varphi)\sin(\theta + \Delta\theta) \\ \sin(\varphi + \Delta\varphi)\sin(\theta + \Delta\theta) \\ \cos(\theta + \Delta\theta) \end{bmatrix} \end{aligned} \tag{10}$$

From Eq (10) can be seen, it is clear that the U causes the phase rotation of $\Delta\varphi$ and $\Delta\theta$. The value and direction of $\Delta\varphi$ and $\Delta\theta$ affect the convergence speed and efficiency of the algorithm. Han *et al* constructed [28] a query table to decide the direction of quantum rotation corner. The table listed all possible situations as an aid decision-making tool. This method involved multiple conditional judgments which would seriously affect the efficiency of the algorithm. As far as the direction of quantum rotation corner is concerned, Li *et al* presented a simple method to decide the direction of quantum rotation corner. Let $q_{0j}(x_{0j}, y_{0j}, z_{0j})$ be the bloch coordinates of the *jth* qubit in the current optimum chromosome. $q_{ij}(x_{ij},\ y_{ij},\ z_{ij})$ is the bloch coordinates of the *jth* qubit in the *ith* chromosome. Let

$$A = \begin{vmatrix} x_{0j} & x_{ij} \\ y_{0j} & y_{ij} \end{vmatrix} B = z_{0j} - z_{ij} \tag{11}$$

The direction of $\Delta\varphi$ and $\Delta\theta$ is determined by such rules as follows: (1) If $A \neq 0$ or $(B \neq 0)$, then the direction of $\Delta\varphi$ or $(\Delta\theta)$ is $-sgn(A)$ or $(-sgn(B))$; (2) If $A = 0$ or $(B = 0)$, then the direction of $\Delta\varphi$ or $(\Delta\theta)$ is arbitrary. The query table of $\Delta\varphi$ and $\Delta\theta$ is as follows.

With the values of $\Delta\varphi$ and $\Delta\theta$, although the scope $(0.005\pi,\ 0.1\pi)$ was given in the literature [27], there is no specific basis for selection. Therefore, the value of

TABLE 4. Parameters used in our algorithm.

Symbol	Meaning	Value
m	Population size	{50,100,1000,2000}
maxDT	Maximum number of iterations	{1000,5000,8000}
p	Mutation probability	0.25
λ	Fixed search step	$0.01*\pi$

```
for p in pop
    if p compatible with S
     S ← p ∪ S
   else if p incompatible with an element in the set S
      delete the element incompatible element p from the set S
      S ← p ∪ S
   end if
end for
```

FIGURE 2. The pseudo code of the updated set *S*.

$\Delta\varphi$ and $\Delta\theta$ should be based on specific practical issues. When their values are too large, the algorithm will converge early and then fall into the local optimal solution. Conversely, the algorithm will converge slowly and seriously affect the efficiency of the algorithm. We adopt the dynamic adjustment strategy to adjust the value of quantum rotation corner. The method is as follow:

$$\theta' = \lambda * \exp(-\frac{t}{\text{maxDT}}) \tag{12}$$

$$\varphi' = \lambda * \exp(-\frac{t}{\text{maxDT}}) \tag{13}$$

where λ is a fixed search step with the value 0.01π, t represents the number of iterations, maxDT represents the maximum number of iterations. In a word, the quantum rotation corner of the update formula is as follow:

$$\Delta\varphi = -sgn(A) * \lambda * \exp(-\frac{t}{\text{maxDT}}) \tag{14}$$

$$\Delta\theta = -sgn(B) * \lambda * \exp(-\frac{t}{\text{maxDT}}) \tag{15}$$

F. CROSSOVER OPERATION

In the basic quantum evolutionary algorithm, since only the quantum rotation gate update operation is carried out, it is easy to cause all the individuals to evolve toward the same target during the population evolution, and then fall into the local optimum. The chromosomal crossover operation not only improves the diversity of the population, but also keeps the good individuals in the population. These two properties are very important for the evolutionary algorithm. Common cross-over strategies are single-point crossover, two-point crossover, multi-point crossover and uniform crossover [34]. In order to combine the coherent characteristics of quantum bits, this paper adopts the whole interference crossover. This crossover operation requires that all chromosomes in the population participate in the exchange of information,

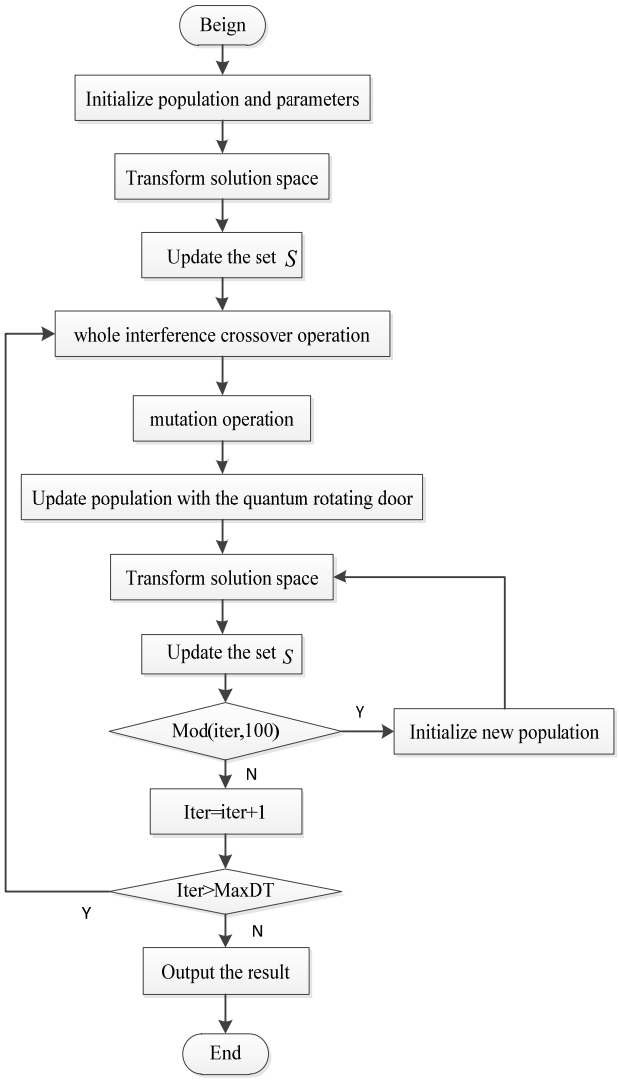


FIGURE 3. Flow chart illustrating the algorithm.

TABLE 5. The meaning of subscripts.

Subscripts	Meaning
t	template-map strategy construction from [13]
s	stochastic local search algorithm from [11] or [12] ;
p	from Proposition 1 of [36]
c	lexicographic construction from [17] or [36]
m	miscellaneous(non-linear, non-lexicographic) design from [17]
d	cited in [38] as personal communication from Dan Tulpan;
a	stochastic local search algorithm from [35]
b	stochastic local search reported in[35] due to X *et al.*
v	variable neighborhood search algorithm from [15] or [20]
u	the computer algebra system from [21]

which can greatly increase the diversity of the population, but it increases the computational complexity at the same time. Suppose the population size is 5, each chromosome has 4 qubits, before the crossover the results are shown as Table II, after the crossover the results are shown as Table III.

TABLE 6. Lower Bounds for $A_c^{GC,RC}(n, d, w)$.

n	\d	3	4	5	6	7	8	9	10	11	12	13	14
4	B	$6^{s,l}$	2^{p}	—	—	—	—	—	—	—	—	—	—
	A	6	2	—	—	—	—	—	—	—	—	—	—
5	B	15^{l}	3^{l}	1^{p}	—	—	—	—	—	—	—	—	—
	A	15	3	1	—	—	—	—	—	—	—	—	—
6	B	44^{a}	16^{l}	4^{l}	2^{p}	—	—	—	—	—	—	—	—
	A	43	16	4	2	—	—	—	—	—	—	—	—
7	B	135^{a}	36^{b}	11^{m}	2^{l}	1^{p}	—	—	—	—	—	—	—
	A	132	35	11	2	1	—	—	—	—	—	—	—
8	B	528^{m}	128^{d}	28^{b}	$12^{s,m}$	2^{l}	1^{p}	—	—	—	—	—	—
	A	439	102	26	11	2	1	—	—	—	—	—	—
9	B	1354^{m}	275^{a}	67^{a}	21^{v}	8^{m}	2^{l}	1^{p}	—	—	—	—	—
	A	1343	**280**	66	19	8	2	1	—	—	—	—	—
10	B	4542^{m}	860^{u}	210^{u}	54^{c}	17^{v}	8^{l}	2^{p}	2^{p}	—	—	—	—
	A	**4558**	**871**	183	**55**	16	8	2	2	—	—	—	—
11	B	.	2457^{m}	477^{a}	117^{a}	37^{v}	14^{v}	5^{m}	2^{m}	1^{p}	—	—	—
	A	.	**2478**	**478**	**118**	35	13	5	2	1	—	—	—
12	B	.	.	1381^{v}	924^{c}	87^{v}	29^{v}	12^{v}	$4^{s,l}$	2^{p}	2^{p}	—	—
	A	.	.	**1384**	312	84	25	11	4	2	2	—	—
13	B	.	.	3974^{v}	924^{c}	206^{v}	62^{v}	23^{v}	10^{v}	4^{m}	2^{m}	1^{p}	—
	A	.	.	3787	793	194	59	21	8	4	2	1	—
14	B	.	.	.	2963^{c}	749^{c}	180^{c}	49^{v}	20^{v}	8^{v}	4^{m}	2^{p}	2^{p}
	A	.	.	.	2095	505	135	45	17	6	4	2	2

G. MUTATION OPERATION

In the quantum algorithm theory, the mutation of quantum chromosomes is done by means of quantum non-gate. If the qubit $|\varphi\rangle$ is described by the vector $[\cos\theta \ \ \sin\theta]^T$ in unit circle, then the result of the mutation is as follow:

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} \sin\theta \\ \cos\theta \end{bmatrix} \tag{16}$$

From the results of the mutation, it can be seen that the effect of non-gate is to exchange the probability amplitudes of qubit and the phase θ is mutated to $\pi/2-\theta$. We can extend the effect of the quantum non-gate from the plane unit circle to the three-dimensional bloch sphere, and set the mutation operator on the bloch sphere. Let V be the mutation operator on the bloch sphere.

$$V = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \tag{17}$$

$$V \begin{bmatrix} \cos\varphi\sin\theta \\ \sin\varphi\sin\theta \\ \cos\theta \end{bmatrix} = \begin{bmatrix} \cos(\pi/2-\varphi)\sin(\pi/2-\theta) \\ \sin(\pi/2-\varphi)\sin(\pi/2-\theta) \\ \cos(\pi/2-\theta) \end{bmatrix} \tag{18}$$

TABLE 7. The meaning of subscripts.

Subscripts	Meaning
s	stochastic local search from [37]
sa	simulated annealing algorithm from [37]
v	variable neighborhood search algorithm from [37]
l	Linear and nonlinear constructions from [37]

From (18), we can calculate the specific form of the mutation operator V.

$$V = \begin{bmatrix} 0 & \cot\theta & 0 \\ \cot\theta & 0 & 0 \\ 0 & 0 & \tan\theta \end{bmatrix} \tag{19}$$

The actual effect of this mutation is a sort of phase rotation of qubit. The process of mutation is as follows: first set the probability of mutation, according to the probability of mutation from the current population to select a number of quantum chromosomes to apply quantum non-gate transformation. The mutation operation does not compare with the current optimal chromosome,

TABLE 8. Lower Bounds for $A_4^{RC}(n, d)$.

n\d		3	4	5	6	7	8	9	10	11	12	13	14
4	B	6^{sa}	2^{sa}	—	—	—	—	—	—	—	—	—	—
	A	6	2	—	—	—	—	—	—	—	—	—	—
5	B	32^{s}	4^{s}	2^{s}	—	—	—	—	—	—	—	—	—
	A	32	4	2	—	—	—	—	—	—	—	—	—
6	B	62^{s}	28^{v}	4^{s}	2^{s}	—	—	—	—	—	—	—	—
	A	61	28	4	2	—	—	—	—	—	—	—	—
7	B	196^{s}	42^{s}	12^{sa}	2^{v}	2^{v}	—	—	—	—	—	—	—
	A	**197**	41	11	2	2	—	—	—	—	—	—	—
8	B	620^{s}	128^{l}	29^{v}	16^{sa}	2^{s}	2^{s}	—	—	—	—	—	—
	A	**643**	125	**30**	11	2	2	—	—	—	—	—	—
9	B	1952^{sa}	345^{v}	80^{sa}	22^{v}	8^{s}	2^{v}	2^{v}	—	—	—	—	—
	A	**2145**	**371**	79	21	8	2	2	—	—	—	—	—
10	B	8064^{l}	2016^{l}	496^{l}	120^{l}	17^{s}	8^{sa}	2^{v}	2^{v}	—	—	—	—
	A	7190	1145	220	55	17	8	2	2	—	—	—	—
11	B	.	4832^{l}	607^{v}	136^{v}	40^{v}	15^{s}	6^{v}	2^{v}	2^{vv}	—	—	—
	A	.	3524	**611**	**139**	40	15	6	2	2	—	—	—
12	B	.	32640^{l}	4032^{l}	2016^{l}	120^{l}	31^{s}	12^{v}	4^{s}	2^{s}	2^{v}	—	—
	A	.	10686	1711	348	90	29	10	4	2	2	—	—
13	B	.	.	5469^{v}	2016^{l}	240^{v}	70^{v}	24^{v}	10^{s}	4^{v}	2^{v}	2^{v}	—
	A	.	.	4774	900	220	63	22	10	4	2	2	—
14	B	.	.	.	8192^{l}	2016^{l}	521^{l}	120^{l}	32^{l}	8^{s}	4^{v}	2^{v}	2^{v}
	A	.	.	.	2404	519	143	44	17	6	3	2	2

and the rotation direction is positive, and the purpose is to increase the diversity of the population, thus reducing the possibility of falling into the local optimal solution.

IV. ALGORITHM DESCRIPTION

The step of a bloch quantum chaos algorithm can be described as follows:

Step1. Initialize population and parameters. Set the current optimization iteration $t \leftarrow 0$, the mutation probability p_m, the fixed search step λ, the size of population m, the maximum optimization iterations maxDT. Generate an initial population by Chaos equation.

Step2. Transform solution space. Three approximate solutions are transformed from unit space $I^n = [-1, 1]^n$ to solution space Ω in each chromosome. Finally compute the fitness of all individuals in the population.

Step3. Update the set S. If the fitness function $fitness(s) == 0$, then the individual is added to the set S. Otherwise, if all individuals fitness function $fitness(s) \neq 0$, then randomly select an individual to add the set S.

Step4. Perform the whole interference crossover operation.

Step5. Perform the mutation operation.

Step6. Update population with the quantum rotating door.

Step7. The updated population is transformed into the solution space of the optimization problem, which is denoted as *pop*, and the fitness of all individuals in the population is computed.

Step8. Update the set S. The update process is as follows: If a individual from set *pop* is compatible with all elements in set S, then we should add it to S. If it's incompatible with an individual in the set S, we should add it to S and remove the incompatible individual. Other circumstances are not considered. Compatibility means that the combination constraints are satisfied. The main idea of the random search algorithm as derived from the literature [35]. The pseudo code of the updated set S is shown in Fig 2.

Step9. If the current optimization iteration is an integer multiple of 100, then regenerate the population by Chaos equation and go back to Step 7; otherwise go to step10.

Step10. The number of iterations increases by 1.

Step11. Determine whether the algorithm termination condition is satisfied. Output the optimal solution if it is satisfied; otherwise, go to step 4.

A flow chart illustrating the algorithm is shown in Fig.3.

V. SIMULATION RESULTS

A. ALGORITHM PARAMETERS

Simulations were conducted on an Intel(R) Core(TM) i3-4160 CPU 3.60GHz 4GB RAM machine with the parameter settings as listed in Table IV.

B. RESULTS

In the following tables, lower bounds are given for $A_4^{GC,RC}(n,d,w)$, $A_4^{RC}(n,d)$ and for $4 \le n \le 14$, $3 \le d \le n$. For each combination, we reported the best result obtained by the algorithms, together with the best-known result available from the literature. The entry A presents our results. The entry B presents the best result of previous study. In case of $A_4^{GC,RC}(n,d,w)$, note that for the bloch quantum chaos algorithm the constant GC-content w is always taken to be $\lfloor n/2 \rfloor$, while for the B entries it is in the range from $\lfloor n/2 \rfloor - 1$ to $\lfloor n/2 \rfloor + 1$. The symbol "." denotes that this instance of the algorithm did not run. The symbol "—" denotes that the distance constraint d is greater than the code length n. Entries in bold are new best lower bounds found by our algorithms.

In the Table VI, subscripts in B entries indicate with which algorithm the result was found. The meaning of the superscripts is as follows:

Let $A_4^{GC}(n,d,w)$ denote the maximum numbers of a DNA code of length n that satisfies the HD and GC constraints for a given parameter d and w. According to the following inequality proposed by King [36]:

$$\begin{cases} A_4^{GC,RC}(n,d,w) \le \dfrac{1}{2} A_4^{GC}(n,d,w) \\ 0 < d \le n, 0 \le w \le n \end{cases} \tag{20}$$

With the last word, one of the new lower bounds for $A_4^{GC,RC}(n,d,w)$ obtained here yield new lower bounds for $A_4^{GC}(n,d,w)$ via the above inequality: $A_4^{GC}(9,4,4) \ge 560$. The previous best lower bounds for these were $A_4^{GC}(9,4,4) \ge 555$, obtained via variable neighborhood search algorithm.

In Table VIII, subscripts in B entries indicate with which algorithm the result was found. The meaning of the superscripts is as follows:

It can be found from Table VIII that the SLS, VNS and SA algorithm constructed better lower bounds for $A_4^{RC}(n,d)$ with the code length $n \le 8$ and distance constraint $d \ge n/2$ than linear and nonlinear of construction. Otherwise, linear and nonlinear of construction produced many more DNA codes with the code length $n > 8$ and distance constraint $d < n/2$ than algorithmic methods. Linear and nonlinear of construction and algorithmic methods can be considered complementary to obtain a better lower bound, which can better meet the needs of different DNA code applications.

VI. CONCLUSIONS

In this paper, we proposed a novel quantum evolutionary algorithm to construct DNA code sets. In this algorithm, the chromosome is encoded by the bloch coordinates of qubits, and each chromosome comprises three gene chains, and each of chains is treated as an optimization solution. We use chaotic equation to initialize the bloch coordinates of quantum bits, and adopt the dynamic adjustment strategy to adjust the quantum rotation corner. The whole interference crossover strategy is introduced into the algorithm to update population. $A_4^{GC}(n,d,w)$ and $A_4^{RC}(n,d)$ with the code length $4 \le n \le 14$ and distance constraint $3 \le d \le n$. By comparing our experimental results with the previous works, the results improve the previously known code and matched the best-known code for several instances. In addition, combine the lower bounds for $A_4^{GC,RC}(9,4,4)$ with King's inequality, we can deduce the new lower bounds for $A_4^{GC}(9,4,4)$.

AUTHOR CONTRIBUTIONS

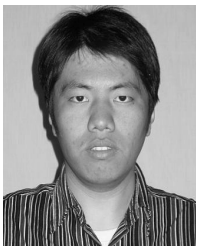
Conceived and designed the experiments: C. Zhou, Q. Zhang. Performed the experiments: Q. Guo, C. Zhou. Analyzed the data: Q. Zhang, X. Wei. Contributed reagents/materials/analysis tools: B. Wang, Q. Guo, Q. Zhang. Wrote the paper: B. Wang, Q. Guo.

REFERENCES

[1] L. M. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021–1024, 1994.
[2] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary DNA microarray," *Science*, vol. 270, no. 5235, pp. 467–470, 1995.
[3] S. Brenner and R. A. Lerner, "Encoded combinatorial chemistry," *Proc. Nat. Acad. Sci. USA*, vol. 89, no.12, pp. 5381–5383, 1992.
[4] X.-Y. Wang, Y.-Q. Zhang, and X.-M. Bao, "A novel chaotic image encryption scheme using DNA sequence operations," *Opt. Lasers Eng.*, vol. 73, pp. 53–61, Oct. 2015.
[5] X. Wang, H. Liu, and A. Kadir, "Image encryption using DNA complementary rule and chaotic maps," *Appl. Soft Comput.*, vol. 12, no. 5, pp. 1457–1466, 2012.
[6] D. Limbachiya and M. K. Gupta. (May 2015). "Natural data storage: A review on sending information from now to then via nature." [Online]. Available: https://arxiv.org/abs/1505.04890
[7] H. Bi, J. Chen, R. Deaton, M. Garzon, H. Rubin, and D. H. Wood, "*In vitro* selection of non-crosshybridizing oligonucleotides for computation," *Natural Comput.*, vol. 2, no. 4, pp. 417–426, 2003.
[8] A. Marathe, A. E. Condon, and R. M. Corn, "On combinatorial DNA word design," *J. Comput. Biol.*, vol. 8, no. 3, pp. 201–219, 2001.
[9] D. Limbachiya, B. Rao, and M. K. Gupta. (Jul. 2016). "The art of DNA strings: Sixteen years of DNA coding theory." [Online]. Available: https://arxiv.org/abs/1607.00266
[10] R. Deaton, M. H. Garzon, R. C. Murphy, and J. R. Koza, "Genetic search of reliable encodings for DNA based computation," in *Proc. Late Breaking Papers Genet. Program.*, 1996, pp. 9–15.
[11] 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.
[12] D. C. Tulpan and H. H. Hoos, "Hybrid randomised neighbourhoods improve stochastic local search for DNA code design," in *Advances in Artificial Intelligence*, vol. 2671. 2003, pp. 418–433.
[13] S. Kobayashi, T. Kondo, and M. Arita, "On template method for DNA sequence design," in *Proc. Int. Workshop DNA-Based Comput., DNA Comput.*, 2002, pp. 205–214.
[14] Q. Zhang, B. Wang, X. Wei, X. Fang, and C. Zhou, "DNA word set design based on minimum free energy," *IEEE Trans. Nanobiosci.*, vol. 9, no. 4, pp. 273–277, Dec. 2010.
[15] 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, p. 311, Sep. 2008.
[16] S. Kawashimo, H. Ono, K. Sadakane, and M. Yamashita, "DNA sequence design by dynamic neighborhood searches," in *DNA Computing*. Berlin, Germany: Springer-Verlag, 2006, pp. 157–171.
[17] P. Gaborit and O. D. King, "Linear constructions for DNA codes," *Theor. Comput. Sci.*, vol. 334, pp. 99–113, Apr. 2005.

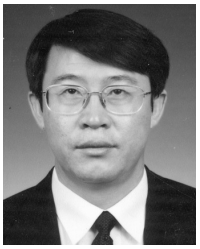
[18] D. H. Smith, N. Aboluion, R. Montemanni, and S. Perkins, "Linear and nonlinear Constructions of DNA codes with Hamming distance *d* and constant *GC*-content," *Discrete Math.*, vol. 312, no. 13, pp. 1207–1219, 2012.
[19] A. E. Gamal, L. Hemachandra, I. Shperling, and V. Wei, "Using simulated annealing to design good codes," *IEEE Trans. Inf. Theory*, vol. IT-33, no. 1, pp. 116–123, Jan. 1987.
[20] 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, pp. 167–175, Jul. 2014.
[21] A. Niema, "The construction of DNA codes using a computer algebra system," Ph.D. dissertation, Dept. Math. Stat. Faculty Adv. Technol., Univ. Glamorgan, Wales, U.K., 2011.
[22] T. Abualrub, A. Ghrayeb, and X. N. Zeng, "Construction of cyclic codes over *GF*(4) for DNA computing," *J. Franklin Inst.*, vol. 343, pp. 448–457, Jul./Aug. 2006.
[23] L. C. B. Faria, A. S. L. Rocha, J. H. Kleinschmidt, R. Palazzo, and M. C. Silva-Filho, "DNA sequences generated by BCH codes over GF(4)," *Electron. Lett.*, vol. 46, pp. 203–204, Feb. 2010.
[24] F. Ma, Y. Cao, and J. Gao, "On cyclic DNA codes over F_4 [u]/(u^2+1)," *Int. J. Res. Rev. Appl. Sci.*, vol. 24, no. 3, pp. 101–105, 2015.
[25] S. Pattanayak and A. K. Singh. (Aug. 2015). "On cyclic DNA codes over the ring Z_4+uZ_4." [Online]. Available: https://arxiv.org/abs/1508.02015
[26] H. Hong, L. Wang, H. Ahmad, J. Li, Y. Yang, and C. Wu, "Construction of DNA codes by using algebraic number theory," *Finite Fields Their Appl.*, vol. 37, pp. 328–343, Jan. 2016.
[27] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proc. IEEE Int. Conf. Evol. Comput.*, May 1996, pp. 61–66.
[28] K.-H. Han and J.-H. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proc. Congr. Evol. Comput.*, vol. 2. Jul. 2002, pp. 1354–1360.
[29] W. Liang, L. Zhang, and M. Wang, "The chaos differential evolution optimization algorithm and its application to support vector regression machine," *J. Softw.*, vol. 6, no. 7, pp. 1297–1304, 2011.
[30] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Process.*, vol. 92, no. 4, pp. 1101–1108, Apr. 2012.
[31] X.-Y. Wang, F. Chen, and T. Wang, "A new compound mode of confusion and diffusion for block encryption of image based on chaos," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 15, no. 9, pp. 2479–2485, 2010.
[32] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems*. New York, NY, USA: Wiley, 1989, pp. 65–72.
[33] P. Li and S. Li, "Quantum-inspired evolutionary algorithm for continuous space optimization based on Bloch coordinates of qubits," *Neurocomputing*, vol. 72, nos. 1–3, pp. 581–591, 2008, doi: 10.1016/j.neucom.2007.11.017.
[34] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proc. IEEE Int. Conf. Evol. Comput.*, May 2002, pp. 61–66.
[35] 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.
[36] O. D. King, "Bounds for DNA codes with constant GC-content," *Electron. J. Combinat.*, vol. 10, no. 1, pp. 33–46, 2003.
[37] 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. Bioinform.*, vol. 11, no. 2, pp. 441–452, Mar. 2014.
[38] O. D. King. (Jan. 2008). *Tables of Lower Bounds for DNA Codes With Constant GC-Content*. [Online]. Available: http://llama.med.harvard.edu/ËIJking/dnacodes.html

QINGJI GUO is currently working toward the M.S. degree at Dalian University, Dalian, China. His research interest is biological computing.

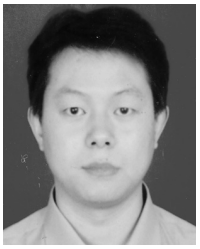


BIN WANG received the B.S. degree in computer science and technology from Dalian University in 2006, and the Ph.D. degree in mechanical design and theory from the Dalian University of Technology in 2013. He is currently a Lecturer with Dalian University. He has (co-) authored about 41 papers published. His research areas include image encryption, DNA computing, DNA sequence design, DNA cryptography, and biological network.

CHANGJUN ZHOU received the B.E. degree in fluid power transmission and control from the Department of Mechanical Engineering, Dalian Railway Institute, Dalian, in 2000, and 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 Dalian University. He has authored or coauthored 38 papers in his research areas. His research interests include DNA computing, image encryption, and intelligence computing.



XIAOPENG WEI received the B.E. degree in mechanical engineering and the M.E. and the Ph.D. degrees in CAD&CG from the School of Mechanical Engineering, Dalian University of Technology, Dalian, in 1982, 1986, and 1993, respectively. In 1982, he joined the Dalian University of Technology, as a Teaching Assistant, where he is currently a Professor. He holds a postdoctoral position with the School of Mechanical Engineering, Dalian University of Technology, from 1993 to 1995. In 1995, he was a Visiting Scholar with The University of Hong Kong, Hong Kong. In 1997, he was a Visiting Professor with Alberta University, Canada. In 1999, he was a Visiting Professor with the Queensland University of Technology, Australia. He has authored or coauthored over 150 papers in his research areas. His current research interests are neural networks, artificial intelligence, DNA computing, mechanical design, and CAD&CG.



QIANG ZHANG received the B.E. degree in electronic engineering and the M.E. and the Ph.D. degrees in circuits and systems from the School of Electronic Engineering, Xidian University, in 1994, 1999, and 2002, respectively. He holds a postdoctoral position with the School of Mechanical Engineering, Dalian University of Technology, Dalian, China. In 2003, he joined the Liaoning Key Laboratory of Intelligent Information Processing, Dalian University, as a Lecturer, where he is a Professor. He is currently with the School of Computer Science and Technology, Dalian University of Technology. He has authored or coauthored over 90 papers in his research areas. His current research interests are neural networks, artificial intelligence, DNA computing, and intelligent robot. He has been a Board Chairman of the Liaoning Graphics Society since 2015.