

## Research Article

# A Parallel Bioinspired Algorithm for Chinese Postman Problem Based on Molecular Computing

Zhaocai Wang,<sup>1</sup> Xiaoguang Bao,<sup>1</sup> and Tunhua Wu <sup>2</sup>

<sup>1</sup>College of Information, Shanghai Ocean University, Shanghai 201306, China

<sup>2</sup>School of Information Engineering, Wenzhou Business College, Wenzhou 325035, China

Correspondence should be addressed to Tunhua Wu; [apll188@163.com](mailto:apll188@163.com)

Received 28 September 2020; Revised 25 December 2020; Accepted 2 January 2021; Published 16 January 2021

Academic Editor: Paolo Gastaldo

Copyright © 2021 Zhaocai Wang 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 Chinese postman problem is a classic resource allocation and scheduling problem, which has been widely used in practice. As a classical nondeterministic polynomial problem, finding its efficient algorithm has always been the research direction of scholars. In this paper, a new bioinspired algorithm is proposed to solve the Chinese postman problem based on molecular computation, which has the advantages of high computational efficiency, large storage capacity, and strong parallel computing ability. In the calculation, DNA chain is used to properly represent the vertex, edge, and corresponding weight, and then all possible path combinations are effectively generated through biochemical reactions. The feasible solution space is obtained by deleting the nonfeasible solution chains, and the optimal solution is solved by algorithm. Then the computational complexity and feasibility of the DNA algorithm are proved. By comparison, it is found that the computational complexity of the DNA algorithm is significantly better than that of previous algorithms. The correctness of the algorithm is verified by simulation experiments. With the maturity of biological operation technology, this algorithm has a broad application space in solving large-scale combinatorial optimization problems.

## 1. Background

The Chinese postman problem (CPP) was first raised by Guan Meigu [1] in the 1960s, and it was based on the question of “how should a postman choose a route so that he can walk all the streets where he is responsible for delivering the letter and the shortest distance traveled?” It arises in numerous applications in many fields, such as municipal solid waste collection [2], winter gritting [3], material distribution and transportation [4], and intelligent transportation [5]. Over the past half century, many branches have been formed in the study of the Chinese postman problem, such as the directed Chinese postman problem (DCPP) [6], the mixed Chinese postman problem (MCP) [7], the windy postman problem (WPP) [8], and the hierarchical Chinese postman problem (HCPP) [9]. Because of its wide application, the Chinese postman problem has been extensively studied, including the design and improvement of calculation methods, the deformation of the problem, and

its application in various aspects. In 1965, Edmonds [10] gave the definition of the Chinese postman problem and then proposed an algorithm to solve the shortest distribution route by first finding the minimum weight matching of any graph to construct Euler graph. Later, in 2002, Yin et al. [11] used molecular biology technology to solve the Chinese postman problem. In 2003, Jiang and Li [12] solved the problem in an undirected graph with an evolutionary algorithm and achieved good results. Their algorithm can also be extended to the directed Chinese postman problem. In 2005, Fei and Cui [13] proposed a new search algorithm CPDPA based on the idea of the dynamic planning decision-making process, which realized the dynamic planning of the Chinese postman problem for the first time, and effectively solved the optimal path problem of the NK model. In 2009, Li and Wang [14] proposed a new solution algorithm, which used the polymerase chain reaction technology to eliminate the nonsolution, and then combined the surface-based DNA computing method with fluorescence labeling technology

and finally extracted the optimal solution from all feasible solutions. In 2011, Zoraida [15] used the thermodynamic properties of DNA and the chemical reaction operation with other biomolecules to improve the accuracy of the algorithm experiment. According to the specific operation steps, the optimal solution of the Chinese postman problem was finally obtained. In 2016, Lee [16] created the Euler circle algorithm for the Chinese postman problem. They selected the edge with the least weight from the adjacent edge of each vertex to get the minimum spanning tree and finally got the shortest path edge of the odd order vertex.

In addition, the arc routing optimization problem similar to the Chinese postman problem has also been widely concerned. Nossack et al. [17] introduce the first modeling of the windy rural postman problem with zigzag option and time windows, discuss two mixed-integer programming formulations for the problem at hand, and suggest exact solution approaches. Meng et al. [18] investigate a class of CTSP, called serial CTSP (S-CTSP), and present a population-based incremental learning (PBIL) approach to the problem. By adding a powerful local search operation, 2-opt, to the algorithm, they can further enhance its search ability. Wang and Lu [19] propose a memetic algorithm with competition (MAC) to solve the capacitated green vehicle routing problem (CGVRP). Carmine et al. [20] develop a two-stage solution approach for the Directed Rural Postman Problem with Turn Penalties (DRPP-TP) and describe an integer linear program that is combined with a local search algorithm. This combination produces high-quality solutions to the DRPP-TP in a reasonable amount of computing time. To solve the Chinese and windy postman problem with variable service costs (CPPVSC), Keskin et al. [8] proposed two heuristics for searching the solution. Xu et al. [21] develop a Delaunay-triangulation-based Variable Neighborhood Search (DVNS) algorithm to solve a general colored traveling salesman problem (GCTSP). Wang et al. [22] firstly introduce a multiobjective MTPDPTW-MP (MO-MTPDPTWMP) with three objectives to better describe the real-world scenario. Then, a multiobjective iterated local search algorithm with adaptive neighborhood selection (MOILS-ANS) is proposed to solve the problem.

In real life, as a classical optimization problem, the Chinese postman problem is widely used in practice. Therefore, it is of great practical value and significance to design an efficient algorithm to solve this problem. But with the increase of the number and scale of the problem, the computational efficiency and storage limit of the traditional algorithm are greatly limited, and it is more and more impossible to solve the problem in the polynomial time level. As a new intelligent algorithm, DNA computing has both the advantages of parallel computing and the amazing storage capacity, which has become one of the potential tools to solve the problem efficiently.

The Chinese postman problem is the mail delivery distance in a certain area. The postman starts from the post office every day, goes through all the streets in the area, and then returns to the post office. The question is how he should arrange the route to deliver the letters so as to minimize the

total distance. In the language of graph theory, the Chinese postman problem is to find a generalized Euler tour that satisfies the sum of the weight of all edges and the sum is minimized, and the starting (ending) vertex is fixed. Given an undirected connected graph  $G = (V, E)$ ,  $V$  represents the vertex set and  $E$  represents the set of arcs (connecting by the vertices in  $V$ ), and  $\|V\| = n$  and  $\|E\| = m$ . Let  $W = (w_{ij})$  be the cost (distance) matrix associated with  $E$  and  $w_{ij} = w_{ji}$ . In general, we assume that  $v_1$  is both starting and ending node, and the a generalized Euler tour is  $A$ . The mathematical definition of the Chinese postman problem is as follows:

$$\begin{aligned} & \text{Min} \sum_{e_{ij} \in A} w_{ij} x_{ij} \\ & \text{s.t.} \quad \sum_{e_{ij} \in A} x_{ij} = \sum_{e_{ki} \in A} x_{ki} \\ & \quad \forall v_i \in V x_{ij} \geq 1 \text{ and interger } \forall e_{ij} \in A. \end{aligned} \quad (1)$$

The optimal route of the Chinese postman problem generally has the following limitations:

- (1) Route is continuous, starting, and returning to the same designated vertex
- (2) Each edge has been traversed at least once by the route
- (3) The weight sum of the route is the minimum

We use Figure 1 as an example for the Chinese postman problem. The optimal route starts from the vertex  $v_1$ , then passes through all the edges at least once, and finally returns  $v_1$ , and the sum of the weight sum of the route is minimized. Through mathematical calculation, the optimal route to meet the requirements can be obtained as follows:

$$\{e_{12}, e_{23}, e_{34}, e_{45}, e_{53}, e_{36}, e_{65}, e_{52}, e_{26}, e_{64}, e_{46}, e_{61}, e_{13}, e_{32}, e_{21}\}. \quad (2)$$

In this paper, we will use the DNA algorithm of the Adleman–Lipton model to solve the Chinese postman problem. The feasibility and computational complexity of the algorithm are analyzed, which proves that the algorithm has higher computational efficiency, and the correctness of the algorithm is also verified by simulation experiments.

The structure of this paper is as follows. Section 2 introduces the research status of DNA computing and the basic knowledge of the Adleman–Lipton model. Section 3 uses a parallel DNA computing algorithm to solve the Chinese postman problem by different operations. The feasibility and computational complexity of the algorithm are analyzed in Section 4. Section 5 gives the simulation results of the algorithm. Finally, in Section 6, we make a summary evaluation of the algorithm and put forward the future research direction.

## 2. Biocomputing and Modeling

With the rise and development of molecular biology, the structure and function of biomacromolecules have been

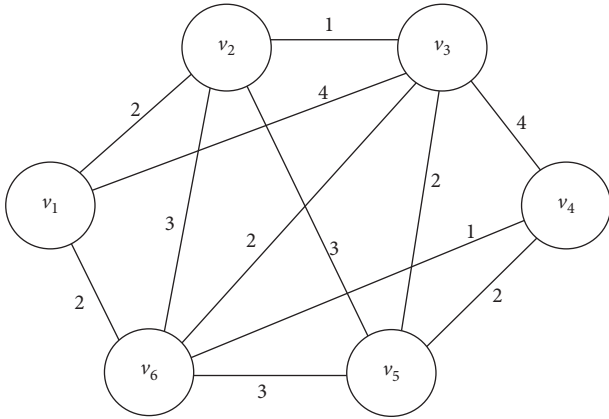


FIGURE 1: The graph of corresponding CPP with six vertices.

proposed gradually. It is found that, in the process of interaction of biomacromolecules, following the laws of biology, chemistry, and physics, a computing process like information transfer and processing process will be formed, accompanied by the emergence of biomacromolecule computing. The concept of biocomputing was first introduced by the famous “salesman problem”. Researchers solved this problem with special DNA tubes in seven nodes, opening a new era of DNA computing research. In 1994, Adleman [23] solved a well-known computational problem with the DNA computing method, that is, the directed Hamiltonian circuit problem. With the increase of variables, it becomes more and more difficult to solve the problem, but DNA computing can easily solve the problem. Adelman’s experiments prove the feasibility of DNA for specific purpose computing and verify the advantages of molecular computing in parallel, high-density compression solutions. Its novelty is that it creates a precedent for applying mathematical problems to molecular level computing and encourages the application of DNA computing in a broader field.

Subsequently, DNA computing has also been used by many scientists to solve various nondeterministic polynomial (NP) mathematical problems, and many typical biological computing models have been established at the same time. For example, in 1995, Lipton [24] solved the satisfiability (SAT) problem through DNA experiments, proving that DNA-based computing has a huge efficiency improvement compared with traditional electronic computers. In 1997, Ouyang et al. [25] proposed a method to solve the biggest group problem by using molecular biology technology. They map all possible clusters into a set of binary numbers to form a data pool and then delete and classify it. The effective solution of this problem provides strong evidence for DNA computing to solve NP-complete problem in the future. At present, DNA computing has become a hot topic in the research of new computing models. Through different operation and control technologies for DNA molecules, new algorithms can be formed to provide strong technical support for many problems without effective solutions [26–39]. For the Chinese postman problem, with the increase of data volume and

scale, the traditional algorithm will be more difficult to solve it. Therefore, we need to propose and try some high-performance computing algorithms; DNA algorithm is one of them.

DNA is a substance that stores genetic information in living cells. DNA is a very thin and long polymer compound composed of a series of deoxynucleotide chains, which in turn are composed of deoxyribose, phosphate, and nitrogen-containing bases. DNA algorithm mainly uses the bases in it, which contains 4 kinds of bases, namely, adenine (A), guanine (G), cytosine (C), and thymine (T). The four types of bases have Watson–Crick complementarity, that is, two opposing bases meet the complementary characteristics, where A complementary to T and C complementary to G. Billions of years of evolutionary cells have created and perfected enzymes with special functions that can copy information from DNA molecules and pass it on to other DNA molecules. The Adleman–Lipton model uses basic biological experimental operations and uses the strings of four different bases A, C, G, and T in DNA as information means to express, and different information is represented by different codes. This provides storage capacity for large amounts of information. A similarly biologically different enzyme is used for different operations to achieve a given goal. Although the fields are different, the principles are the same. In computation, we consider that the length of a single-stranded DNA is the number of nucleotides composed of a single strand. Therefore, if a single-stranded DNA contains 30 nucleotides, we consider the strand to be 30 in length and called 30 mer.

In the Adleman–Lipton model, for given test tubes (experimental test tubes) which contain a finite number of DNA strands consisting of A, G, C, and T, we can perform the following operations:

- (1) Merge ( $T_1, T_2$ ): for two given test tubes  $T_1$  and  $T_2$ , it stores the union  $T_1$  and  $T_2$  in  $T_1$  and leaves  $T_2$  empty.
- (2) Annealing ( $T$ ): for a given test tube  $T$ , it produces all feasible double strands in  $T$ .
- (3) Denaturation ( $T$ ): for a given test tube  $T$ , it dissociates each double strand in  $T$  into two single strands.
- (4) Separation ( $T_1, X, T_2$ ): for a given test tube  $T_1$  and a given set of strings  $X$ , it removes all single strands containing the string  $X$  from  $T_1$  and produces a test tube  $T_2$  with the removed strands.
- (5) Discard ( $T$ ): for a given test tube  $T$ , it discards the strands in tube  $T$ .
- (6) Copy ( $T_1, T_2$ ): for a given test tube  $T_1$ , it produces a test tube  $T_2$  with the same strands as  $T_1$ .
- (7) Cleavage ( $T, \gamma_0\gamma_1$ ): given a test tube  $T$  and strings  $\gamma_0\gamma_1$ , it cuts every strand containing  $[\gamma_0\gamma_1]$  into three strands:

$$[\dots, \alpha\gamma_0\gamma_1\beta, \dots] \Leftrightarrow [\dots, \alpha], [\gamma_0\gamma_1], [\beta, \dots]. \quad (3)$$

- (8) Sort( $T_1, T_2, T_3$ ): it moves the shortest strands of the test tube  $T_1$  to the tube  $T_2$  and the longest to the test tube  $T_3$ , and the rest is still stored in the test tube  $T_1$ .
- (9) Read( $T$ ): for a given tube  $T$ , the operation is used to describe a single molecule, which is contained in the tube  $T$ . Even if  $T$  contains many different molecules and each of which encodes a different set of bases, the operation can give an explicit description of exactly one of them.

Since the above nine operations can be realized by a series of continuous operations on the biological chains, it can reasonably be deduced that the complexity of each operation is  $O(1)$  [40]. The assumption has been used in previous studies to analyze the complexity of DNA computing [41–46]. We use these operations to implement algorithms to solve the Chinese postman problem.

### 3. Algorithm Design for the Chinese Postman Problem

In this section, the basic idea of the algorithm is introduced first, and then the symbols in the algorithm are set uniformly. Finally, Figure 1 is taken as an example of the Chinese postman problem to demonstrate the algorithm.

*3.1. Preliminary Ideas.* The basic idea to solve the Chinese postman problem by DNA computing is as follows: first, the corresponding vertices and edges of the Chinese postman problem are represented by specific symbols; then, all possible DNA chains in the problem are generated by biochemical reactions, and the chains that meet the constraint conditions of the problem are screened. Finally, through search and recognition, the optimal solution corresponding to the biological chain is obtained, and then, the optimal route of the problem is obtained.

Correspondingly, the algorithm is divided into four steps:

- Step 1: generate all possible random routes  
 Step 2: filter out routes that start at a fixed node (post office) and end at the same node  
 Step 3: select all routes that pass through all edges at least once  
 Step 4: find the shortest generalized Euler closed loop, get the shortest route, and output the optimal solution

*3.2. Graph Theory Expression with DNA Strands.* In this paper, in order to clarify and standardise the expression of our algorithm, it is necessary to define and explain the symbols and notations in the algorithm. Therefore, we use the symbols in Table 1 to illustrate.

In DNA computing, as the optimal path is mainly determined by the length of the feasible solution strands, it is necessary and critical to design the length of the problem element information strands. So, in this paper, we set  $\|\bar{O}c_{ij}\bar{O}\| = w_{ij} * l$  mer and  $\|\#\| = \|A_i\| = \|B_i\| = l$  mer, where  $l$  represent positive integers. Obviously, the length of the single-stranded DNA in the experiment depends largely on the size of the problem involved. At the same time, we use their complementary strand  $\overline{B_i A_i}$  to synthesize double strands. Generally, we assume that node  $v_i$  is the beginning and ending points of the route for the Chinese postman problem.

*3.3. Algorithm Description.* For the Chinese postman problem, we set the nodes as vertices  $V$ , the streets as edges  $E$ , and the weight of edge  $e_{ij}$  is  $w_{ij}$ .

Let us start with the initial tubes  $T_1, T_2$ , and  $T_3$ :

$$\begin{aligned}
 T_1 &= \left\{ \#A_1 \bar{O}c_{1j_1} \bar{O}B_{j_1}, \dots, A_i \bar{O}c_{ij} \bar{O}B_j, \dots, A_{i_p} \bar{O}c_{ip1} \bar{O}B_1 \# | e_{1j_1}, e_{ij}, e_{ip1} \in E \right\}, \\
 T_2 &= \{ \overline{B_i A_i} | i = 1, 2, \dots, n \}, \\
 T_3 &= \left\{ c_{1j_1}, c_{ij_2}, \dots, c_{ij}, \dots, c_{ip1} | e_{ij} \in E, \|c_{ij}\| = (w_{ij} - 2) * l \right\}.
 \end{aligned} \tag{4}$$

We use the Chinese postman problem in Figure 1 (beginning and ending nodes are  $v_1$ ) as an example to demonstrate the process of the algorithm. Let

$$\begin{aligned}
T_1 = & \{ \#A_1\bar{O}c_{12}\bar{O}B_2, \#A_1\bar{O}c_{13}\bar{O}B_3, \#A_1\bar{O}c_{16}\bar{O}B_6, A_2\bar{O}c_{21}\bar{O}B_1\#, \\
& \cdot A_2\bar{O}c_{23}\bar{O}B_3, A_2\bar{O}c_{25}\bar{O}B_5, A_2\bar{O}c_{26}\bar{O}B_6, A_3\bar{O}c_{31}\bar{O}B_1\#, A_3\bar{O}c_{32}\bar{O}B_2, \\
& \cdot A_3\bar{O}c_{34}\bar{O}B_4, A_3\bar{O}c_{35}\bar{O}B_5, A_3\bar{O}c_{36}\bar{O}B_6, A_4\bar{O}c_{43}\bar{O}B_3, A_4\bar{O}c_{45}\bar{O}B_5, \\
& \cdot A_4\bar{O}c_{46}\bar{O}B_6, A_5\bar{O}c_{52}\bar{O}B_2, A_5\bar{O}c_{53}\bar{O}B_3, A_5\bar{O}c_{54}\bar{O}B_4, A_5\bar{O}c_{56}\bar{O}B_6, \\
& \cdot A_6\bar{O}c_{61}\bar{O}B_1\#, A_6\bar{O}c_{62}\bar{O}B_2, A_6\bar{O}c_{63}\bar{O}B_3, A_6\bar{O}c_{64}\bar{O}B_4, A_6\bar{O}c_{65}\bar{O}B_5 \}, \\
T_2 = & \{ \overline{B_1A_1}, \overline{B_2A_2}, \overline{B_3A_3}, \overline{B_4A_4}, \overline{B_5A_5}, \overline{B_6A_6} \}, \\
T_3 = & \{ c_{12}, c_{13}, c_{16}, c_{21}, c_{23}, c_{25}, c_{26}, c_{31}, c_{32}, c_{34}, c_{35}, c_{36}, c_{43}, c_{45}, c_{46}, c_{52}, c_{53}, c_{54}, c_{56}, c_{61}, c_{62}, c_{63}, c_{64}, c_{65} \}.
\end{aligned} \tag{5}$$

In Figure 1, the edge set matrix composed of binary variables is as follows:

$$M(G_1) = [x_{ij}] = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}. \tag{6}$$

After the above seven operations, we generate DNA strands to represent different postman routes. Taking the problem in Figure 1 as an example, the chain  $\#A_1\bar{O}c_{12}\bar{O}B_2A_2\bar{O}c_{25}\bar{O}B_5A_5\bar{O}c_{54}\bar{O}B_4A_4\bar{O}c_{43}\bar{O}B_3A_3\bar{O}c_{31}\bar{O}B_1\#$  represents the route  $\{e_{12}, e_{25}, e_{54}, e_{43}, e_{31}\}$ .

For example, in Figure 1, route  $\{e_{12}, e_{23}, e_{34}, e_{45}, e_{53}, e_{36}, e_{64}, e_{45}, e_{56}, e_{63}, e_{32}, e_{26}, e_{61}\}$  should be excluded because it does not pass through edges  $e_{13}$  and  $e_{25}$ . After the operations, all DNA single strands in the tube  $T_5$  represent the routes start and end vertex  $v_1$  and pass through all edges at least once.

Finally, we can get the exact solutions strands for the Chinese postman problem.

**3.3.1. Generate a Data Pool and Filter Out the Routes That Start and End with the Specified Vertex.** is provided in Algorithm 1.

**3.3.2. Path through All Edges.** is provided in Algorithm 2

**3.3.3. Eliminate the Influence of Vertex Chain Length.** is provided in Algorithm 3

**3.3.4. Reconfirm That the Route Passes All Edges At Least Once.** is provided in Algorithm 4

**3.3.5. Get the Solution.** is provided in Algorithm 5

## 4. Feasibility and Time Complexity of the Proposed DNA Algorithm

We should prove that the proposed DNA algorithm can find the DNA chain corresponding to the optimal solution in a certain length range and solve the Chinese postman problem in a certain time complexity.

**Theorem 1.** *The proposed DNA algorithm can solve the Chinese postmen problem.*

*Proof.* We use DNA strands with different symbols to express the corresponding nodes and edges of the problem and use biochemical reactions with complementary strands to generate all possible routes strands in the test tube. Therefore, after Algorithm 1, each DNA strand represents a feasible path starting from a specified fixed vertex and ending it. Another constraint of the feasible solutions is that each edge needs to pass at least once, so we screen out feasible chains that pass through all specified edges by Algorithm 2. The chains  $L_1$  after Algorithm 2 can be described as

$$\begin{aligned}
& \#A_1\bar{O}c_{1j_1}\bar{O}B_{j_1}A_{j_1}\bar{O}c_{j_1j_2}\bar{O}B_{j_2}, \dots, \bar{O}A_{j_p}c_{j_pj_{p+1}}\bar{O}B_{j_{p+1}}, \dots, A_q\bar{O}c_{q1}\bar{O}B_1\# \\
& j_p, j_{p+1} \in \{1, 2, \dots, n\}.
\end{aligned} \tag{7}$$

In order to select the optimal solution in a feasible chain, the weight of the feasible chain needs to be compared. Because the original chain contains information passing

through the vertex, we filter out the vertex information in the chain through Algorithm 3. Algorithm 4 excludes the infeasible chains of doping caused by the synthesis of

TABLE 1: Notations and symbols.

Symbol	Description	Symbol	Description
$V$	Vertex set	$v_k$	The $k$ -th vertex
$E$	Edge set	$\#$	Starting and ending of DNA strand
$W$	Weight set	$A_k, B_k$	DNA string of the $k$ -th vertex
$n$	Number of vertices	$A_i \bar{O} c_{ij} \bar{O} B_j$	DNA string of the edge $e_{ij}$
$m$	Number of edges	$\bar{B}_i A_i$	Complementary strands of adjacent routes

We can get all the possible routes strands (from  $v_1$  to  $v_1$ ) of the Chinese postman problem in a test tube.

- (1-1) Merge( $T_1, T_2$ );
- (1-2) Annealing( $T_1$ );
- (1-3) Denaturation( $T_1$ );
- (1-4) Separation( $T_1, \{\#A_1\}, T_4$ );
- (1-5) Discard( $T_1$ );
- (1-6) Separation( $T_4, \{B_1\# \}, T_5$ );
- (1-7) Discard( $T_4$ ).

ALGORITHM 1: Generate various route strands.

complementary chains in Algorithm 3. At this time, the chains  $L_2$  which pass through all edges at least once and only contain edge weight information are retained in test tube  $T_5$ . After the step, the chain  $L_2$  can be expressed as follows:

$$\#A_1 \bar{O} c_{1j_1} \bar{O} \bar{O} c_{j_1 j_2} \bar{O} \bar{O}, \dots, \bar{O} c_{j_p j_{p+1}} \bar{O}, \dots, \bar{O} c_{q_1} \bar{O} B_1 \#. \quad (8)$$

In addition, we set  $\|\bar{O} c_{ij} \bar{O}\| = w_{ij} * l$  mer and  $\|\#\| = \|A_i\| = \|B_j\| = l$  mer. The length of chain  $L_2$  is given by

$$\begin{aligned} \|L_2\| &= \|\#\| + \|A_1\| + \|\bar{O} c_{1j_1} \bar{O}\| + \|\bar{O} c_{j_1 j_2} \bar{O}\| + \dots + \|\bar{O} c_{j_p j_{p+1}} \bar{O}\| + \dots + \|\bar{O} c_{q_1} \bar{O}\| + \|B_1\| + \|\#\| \\ &= l + l + w_{1j_1} * l + w_{j_1 j_2} * l + \dots + w_{j_p j_{p+1}} * l + \dots + w_{q_1} * l + l + l \\ &= 4l + \left( w_{1j_1} + w_{j_1 j_2} + \dots + w_{j_p j_{p+1}} + \dots + w_{q_1} \right) * l \\ &= 4l + \sum_{\bar{O} c_{ij} \bar{O} \in T_5} w_{ij} * l. \end{aligned} \quad (9)$$

Therefore, the length of  $L_2$  chain changes in the same direction as the weight of the edge through the path it represents. In Algorithm 5, we obtain the optimal solution to the problem by comparing the lengths of the DNA chains.  $\square$

**Theorem 2.** *The Chinese postman problem can be solved at the  $O(n^2)$  time complexity level.*

*Proof.* Since each biochemical operation is completed in  $O(1)$  time complexity [41–46], according to the steps we designed, the Chinese postman problem can be performed in a limited number of steps and time complexity. The time complexity  $T$  of the DNA algorithm is as follows:

$$\begin{aligned} T(\text{Algorithm 1}) &= O(1); \\ T(\text{Algorithm 2}) &= O(n^2); \\ T(\text{Algorithm 3}) &= O(n); \\ T(\text{Algorithm 4}) &= O(n^2); \\ T(\text{Algorithm 5}) &= O(1); \\ T &= T(\text{Algorithm 1}) + T(\text{Algorithm 2}) \\ &\quad + T(\text{Algorithm 3}) + T(\text{Algorithm 4}) \\ &\quad + T(\text{Algorithm 5}) \\ &= O(1) + O(n^2) + O(n) + O(n^2) + O(1) \\ &= O(n^2). \end{aligned} \quad (10)$$

In the Chinese postman problem, each edge should be traversed at least once. In other words, a route cannot be a feasible path if it does not go through all the edges. We judge the information of the edge that the route passes by and screen out the routes that do not meet the restriction.

```

For  $i = 1$  to  $i = n - 1$ 
  For  $j = i + 1$  to  $j = n$ 
(2-1) If  $(x_{ij} = 1)$ 
(2-2-1) Separation  $(T_5, \{A_i \bar{O}c_{ij} \bar{O}B_j\}, T_6)$ ;
(2-2-2) Separation  $(T_5, \{A_j \bar{O}c_{ji} \bar{O}B_i\}, T_7)$ ;
(2-2-3) Merge  $(T_6, T_7)$ ;
(2-2-4) Discard  $(T_7)$ ;
(2-2-5) Discard  $(T_5)$ ;
(2-2-6) Copy  $(T_6, T_5)$ ;
(2-2-7) Discard  $(T_6)$ ;
  Else
(2-3) Continue.
  End for
End for

```

ALGORITHM 2: Judge the edges of the route.

The feasible solution chains contain the information of the chain length passing through the vertex. The optimal solution based on the total chain length will be affected by the number of vertices the route passes through. Therefore, it is necessary to exclude the vertex chain information contained in the original feasible chain.

```

For  $i = 1$  to  $i = n$ 
(3-1) Cleavage  $(T_5, B_i A_i)$ ;
(3-2) Separation  $(T_5, B_i A_i, T_8)$ ;
(3-3) Discard  $(T_8)$ ;
(3-4) Merge  $(T_5, \bar{O}\bar{O})$ ;
(3-5) Annealing  $(T_5)$ ;
(3-6) Denaturation  $(T_5)$ ;
(3-7) Separation  $(T_5, \bar{O}\bar{O}, T_9)$ ;
(3-8) Discard  $(T_5)$ ;
(3-9) Copy  $(T_9, T_5)$ ;
(3-10) Discard  $(T_9)$ .
End for
In Figure 1, the edge route  $\{e_{13}, e_{36}, e_{65}, e_{54}, e_{46}, e_{62}, e_{25}, e_{53}, e_{32}, e_{21}, e_{16}, e_{64}, e_{43}, e_{31}\}$  can be coded as  $\#A_1 \bar{O}c_{13} \bar{O}\bar{O}c_{36} \bar{O}\bar{O}c_{65} \bar{O}\bar{O}c_{54} \bar{O}\bar{O}c_{46} \bar{O}\bar{O}c_{62} \bar{O}\bar{O}c_{25} \bar{O}\bar{O}c_{53} \bar{O}\bar{O}c_{32} \bar{O}\bar{O}c_{21} \bar{O}\bar{O}c_{16} \bar{O}\bar{O}c_{64} \bar{O}\bar{O}c_{43} \bar{O}\bar{O}c_{31} \bar{O}B_1 \#$ .

```

ALGORITHM 3: Generate weight chain of edges.

Because the complementary chain of vertex  $\bar{B}_i \bar{A}_i$  is used to generate the weight information chain of the edge, some paths that do not pass through all the edges will also be doped in tube  $T_5$ , so the chains should be excluded. We judge the path passing through the edges again and screen out the feasible path chains satisfying that all the edges pass through at least once.

```

For  $i = 1$  to  $i = n - 1$ 
  For  $j = i + 1$  to  $j = n$ 
(4-1) If  $(x_{ij} = 1)$ 
(4-2-1) Separation  $(T_5, \{\bar{O}c_{ij} \bar{O}\}, T_{10})$ ;
(4-2-2) Separation  $(T_5, \{\bar{O}c_{ji} \bar{O}\}, T_{11})$ ;
(4-2-3) Merge  $(T_{10}, T_{11})$ ;
(4-2-4) Discard  $(T_{11})$ ;
(4-2-5) Discard  $(T_5)$ ;
(4-2-6) Copy  $(T_{10}, T_5)$ ;
(4-2-7) Discard  $(T_{10})$ ;
  Else
(4-3) Continue.
  End for
End for

```

ALGORITHM 4: Judge the edges of the route again.

In many different routes, the optimal solution of the Chinese postman problem has the minimum weight value. We search for the shortest DNA strand in the last test tube to indicate the result of the Chinese postman problem.

(5-1) Sort( $T_5, T_{12}, T_{13}$ );

(5-2) Read( $T_{12}$ ).

ALGORITHM 5: Search the optimal route.

TABLE 2: Sequences chosen to represent  $A_i$ ,  $B_i$ , #,  $\bar{O}$ , and  $c_{ij}$  ( $i, j \in \{1, 2, \dots, 7\}$ ) for the Chinese postman problem.

Bit	3' - 5' DNA sequence
$A_1$	AACTGA
$A_2$	TGTGAG
$A_3$	GACTTC
$A_4$	CATTTG
$A_5$	ACTCGC
$A_6$	CTACGT
#	AGCAGA
$c_{12}$	TCGTCACG
$c_{16}$	AGTGACGC
$c_{25}$	ATTAAGCGCCAATC
$c_{34}$	AATGACAGCCAACCGAGATC
$c_{36}$	GCGTTTCA
$c_{46}$	AT
$c_{21}$	ATCAGCTA
$c_{61}$	TATCACAG
$c_{52}$	AATGCTAGAGTGCT
$c_{43}$	TTCTGAACACTGACACCAGA
$c_{63}$	ATCATGCT
$c_{64}$	TG
$B_1$	TAAATC
$B_2$	CCAATA
$B_3$	ATACGC
$B_4$	TGCAAG
$B_5$	GATCTC
$B_6$	TGAAGC
$\bar{O}$	GG
$c_{13}$	CATAGCACGCCTGCCTCTGT
$c_{23}$	TC
$c_{26}$	CCGAGATCCTCAAT
$c_{35}$	ATAGCGCT
$c_{45}$	AGCCAATG
$c_{56}$	CGACTAATATCGTC
$c_{31}$	GTGCGCACGAGCCTTTACCA
$c_{32}$	GC
$c_{62}$	TCCTCGCTAGATCG
$c_{53}$	ACGTCATG
$c_{54}$	CTATTTCA
$c_{65}$	GTGTACGACCCTAT

Therefore, according to the analysis, the algorithm can solve the Chinese postman problem at the  $O(n^2)$  time complexity level.  $\square$

## 5. Simulation Experiments of DNA Algorithms

In order to verify the feasibility of the algorithm, simulation operations are very necessary and meaningful. The calculation of DNA depends on the accuracy of biochemical molecular operation; otherwise, it will lead to the

accumulation and expansion of errors in biochemical reactions. At the same time, the operations should exclude the DNA sequences that may promote the unexpected hybridization of the probe library. Therefore, the design of appropriate DNA sequence is the necessary basis to ensure the accuracy of DNA calculation. In order to achieve these objectives, the experiments comply with the seven limitations of DNA manipulation proposed in [47]. These limitations include the following: no more than 7 base pairs of any 8 base pairs between library sequences and their own or any other library sequences; no occurrence of 5 or more consecutive identical nucleotides appear in all library and probe sequences; no more than 7 matches of any probe sequence with any 8 base pairs of any library sequence; the probe will only bind weakly where it is not intended to bind, and so on. Similarly, the experiments use the sequence design method in references [40, 47–52].

In this paper, the computational molecular biology tool Biopython is used as the system development platform, and Braich's program is used to generate "appropriate DNA sequences" suitable for biological computing algorithm, which can be used to solve the CPP. Braich's program and other simulations are running on a Windows 7 machine, with an Intel Core-XP CPU and 8 GB main memory, and the compiler is Python 3.6. Our modified program is used to construct a random sequence of 4 bases for each bit of the library and to check whether the library chain meets the seven constraints of DNA sequence [47]. When generating one DNA chain, the first step is to determine whether the chain satisfies the restrictions specified in references [40, 47–52]. If the produced DNA sequence does not meet the restrictions, the program will continue to generate another new sequence. When these restrictions can be met, the sequence is accepted for subsequent biochemical reactions. Using this method, we can get the "appropriate DNA sequences" which meet the restriction conditions to improve the accuracy of biochemical reactions.

Therefore, taking the Chinese postman problem in Figure 1 as an example, the program generates random sequences to form  $A_i$ ,  $B_i$ , #,  $\bar{O}$ , and  $c_{ij}$  ( $l = 6$  mer and  $\|\bar{O}\| = 2$  mer) as shown in Table 2. Table 3 demonstrates the DNA node sequence composed by Braich's methods. In Table 4, the enthalpy, entropy, and free energy of the binding of each probe to the corresponding region in the chains have also been calculated. Their average deviation and standard deviation levels are also shown in Table 4. Then, DNA solution strands for the Chinese postman problem have been found. Table 5 demonstrates the DNA solution strands for the Chinese postman problem.



TABLE 3: Sequences chosen to represent the elements  $e_{ij}$  for the Chinese postman problem.

Bit	3' – 5' DNA sequence
$A_1\bar{O}c_{12}\bar{O}B_2$	AACTGAGGTCGTACGGGCCAATA
$A_1\bar{O}c_{13}\bar{O}B_3$	AACTGAGGCATAGCACGCCTGCCTCTGTGGATACGC
$A_1\bar{O}c_{16}\bar{O}B_6$	AACTGAGGAGTGACGCGGTGAAGC
$A_2\bar{O}c_{23}\bar{O}B_3$	TGTGAGGGTCCGATACGC
$A_2\bar{O}c_{25}\bar{O}B_5$	TGTGAGGGATTAAAGCGCCAATCGGGATCTC
$A_2\bar{O}c_{26}\bar{O}B_6$	TGTGAGGGCCGAGATCCTCAATGGTGAAGC
$A_3\bar{O}c_{34}\bar{O}B_4$	GACTTCGGAATGACAGCCAACCGAGATCGGCATTTG
$A_3\bar{O}c_{35}\bar{O}B_5$	GACTTCGGATAGCGTGGGATCTC
$A_3\bar{O}c_{36}\bar{O}B_6$	GACTTCGGGCGTTTCAGGTGAAGC
$A_4\bar{O}c_{45}\bar{O}B_5$	CATTTGGGAGCCAATGGGGATCTC
$A_4\bar{O}c_{46}\bar{O}B_6$	CATTTGGGATGGTGAAGC
$A_5\bar{O}c_{56}\bar{O}B_6$	ACTCGCGGCGACTAATATCGTCCGGTGAAGC
$A_2\bar{O}c_{21}\bar{O}B_1$	TGTGAGGGATCAGCTAGGTAATC
$A_3\bar{O}c_{31}\bar{O}B_1$	GACTTCGGGTGCGCACGAGCCTTTACCAGGTAATC
$A_6\bar{O}c_{61}\bar{O}B_1$	CTACGTGGTATCACAGGGTAAATC
$A_3\bar{O}c_{32}\bar{O}B_2$	AGACTTCGGGCGGCCAATA
$A_5\bar{O}c_{52}\bar{O}B_2$	ACTCGCGGAATGCTAGAGTGCTGGCCAATA
$A_6\bar{O}c_{62}\bar{O}B_2$	CTACGTGGTCCCTCGCTAGATCGGGCCAATA
$A_4\bar{O}c_{43}\bar{O}B_3$	CATTTGGGTTCTGAACACTGACACCAGAGGATACGC
$A_5\bar{O}c_{53}\bar{O}B_3$	ACTCGCGGACGTCATGGGATACGC
$A_6\bar{O}c_{63}\bar{O}B_3$	CTACGTGGATCATGCTGGATACGC
$A_5\bar{O}c_{54}\bar{O}B_4$	ACTCGCGGCTATTCAGGTGCAAG
$A_6\bar{O}c_{64}\bar{O}B_4$	CTACGTGGTGGGTGCAAG
$A_6\bar{O}c_{65}\bar{O}B_5$	CTACGTGGGTGTACGACCCTATGGGATCTC

TABLE 4: The energies over all probe/library strand interactions.

Vertex	Enthalpy energy $H$	Entropy energy $S$	Free energy $G$
$A_1\bar{O}c_{12}\bar{O}B_2$	101.4	250.9	23.9
$A_1\bar{O}c_{13}\bar{O}B_3$	116.9	294.4	26.6
$A_1\bar{O}c_{16}\bar{O}B_6$	99.7	252.6	23.2
$A_2\bar{O}c_{23}\bar{O}B_3$	96.1	246.3	23.3
$A_2\bar{O}c_{25}\bar{O}B_5$	106.7	269.3	26.3
$A_2\bar{O}c_{26}\bar{O}B_6$	100.7	249.9	23.7
$A_3\bar{O}c_{34}\bar{O}B_4$	102.9	263.2	23.8
$A_3\bar{O}c_{35}\bar{O}B_5$	112.5	279.6	27.3
$A_3\bar{O}c_{36}\bar{O}B_6$	103.5	273.3	23.4
$A_4\bar{O}c_{45}\bar{O}B_5$	102.7	263.9	22.7
$A_4\bar{O}c_{46}\bar{O}B_6$	113.5	294.5	28.0
$A_5\bar{O}c_{56}\bar{O}B_6$	106.0	277.8	22.7
$A_2\bar{O}c_{21}\bar{O}B_1$	105.5	266.5	24.5
$A_3\bar{O}c_{31}\bar{O}B_1$	111.2	284.5	25.1
$A_6\bar{O}c_{61}\bar{O}B_1$	105.1	271.2	23.2
$A_3\bar{O}c_{32}\bar{O}B_2$	110.6	278.6	26.5
$A_5\bar{O}c_{52}\bar{O}B_2$	99.6	250.7	23.7
$A_6\bar{O}c_{62}\bar{O}B_2$	97.1	238.9	23.1
$A_4\bar{O}c_{43}\bar{O}B_3$	105.6	251.3	24.2
$A_5\bar{O}c_{53}\bar{O}B_3$	104.3	249.7	24.5
$A_6\bar{O}c_{63}\bar{O}B_3$	103.2	249.3	25.3
$A_5\bar{O}c_{54}\bar{O}B_4$	116.5	297.6	25.6
$A_6\bar{O}c_{64}\bar{O}B_4$	102.5	248.6	24.2
$A_6\bar{O}c_{65}\bar{O}B_5$	96.1	244.7	23.1
Average	104.618	263.727	24.427
Standard deviation	5.5650	16.8164	1.5088

TABLE 5: DNA sequences chosen to represent the solutions to the Chinese postman problem in Figure 1.

3' – AAGCAGAAACTGAGGTCGTCACGGGGGTCGGGGAATGACAG
CCAACCGAGATCGGGGAGCCAATGGGGGCGTTTCAGGGGGTG
TACGACCCATGGGGAATGCTAGAGTGCTGGGGCCGAGATCCT
CAATGGGGTGGGGGATGGGGTATCACAGGGGGCATAGCACGCC
TGCCTCTGTGGGGGCGGGGATCAGCTAGGTAATCAGCAGA – 5'

TABLE 6: Comparison of the computational complexity of different algorithms.

Authors	Time complexity
Edmonds and Johnson [53]	$O(n^3)$
Kaj Holmberg [54]	$O(n^3)$
Kundeti et al. [55]	$O( E ^2 \log^2( V ))$
Gutin et al. [56]	$O(2^k n^3)$
Nilofer and Rizwanullah [57]	$O(n^2) - O(n^4)$
Our algorithm	$O(n^2)$

## 6. Conclusions

Traditional silicon electronic computers can deal with simple types of computing problems, but they cannot deal with complex big data problems efficiently due to the limitations of the chip's microprocessing ability and other factors. DNA molecules not only have the natural characteristics of specific hybridization and miniaturization but also have the ability of high-density information storage and powerful parallel computing, which makes DNA computing having natural advantages in dealing with difficult problems. In this paper, an algorithm based on the Adleman–Lipton model is proposed to solve the Chinese postman problem. DNA algorithm has the advantages of simple interpretation, simple coding, and high precision. Secondly, the computational complexity of the algorithm increases linearly with the scale of the problem. Due to the good parallelism of DNA computing, this method solves the Chinese postman problem with  $n$  vertices in  $O(n^2)$  time complexity. Compared with the previous algorithms, it has made great progress and improvement. For example, Edmonds [10] and Johnson et al. [53] give a perfect matching algorithm about the minimum weight of CPP with the  $O(n^3)$  time complexity. Kaj Holmberg [54] proposed a heuristic idea and improved it with the Frederickson method; the time complexity of the method is also  $O(n^3)$ . Kundeti et al. [55] showed that the cyclic CPP on bidirected graphs can be solved without reducing it to bidirected flow with  $O(|E|^2 \log^2(|V|))$  time complexity. Gutin et al. [56] prove that the MCP parameterized by the number of arcs is also fixed-parameter tractable, and solve it in time  $O(2^k n^3)$ , where  $n$  is the number of vertices in  $G$ . Nilofer and Rizwanullah [57] propose a combination of minimum matching algorithms, which create an Euler walk and Dijkstra algorithm to calculate short paths, and the computational complexity of the algorithms is between  $O(n^2)$  and  $O(n^4)$ . The comparison results are shown in Table 6. By comparing the results, we can see that different algorithms will have a great impact on the computational efficiency due to the differences in computational complexity when solving

large-scale Chinese traveling salesman problems. Especially with the increase of  $n$ , the difference will become more clear. Therefore, the improvement of computational efficiency is obviously conducive to the solution of large-scale problems.

Moreover, our algorithm uses DNA as the basic element and unit of calculation, which has the characteristics of low energy consumption and large storage capacity compared with other algorithms. As DNA molecules are very tiny, a cubic meter of DNA solution can store nearly one trillion binary data. In addition, the energy consumption required for the reaction is very low. To complete the same computing operation, the energy consumption is one billion of that of supercomputers. It is conceivable that, with the expansion of the scale of the problem, there will be enough molecules to represent the element information in the problem. At the same time, the parallel chemical reaction of molecules has obvious advantages over the serial operation of computer. The parallel efficiency of DNA computing is amazing. The number of parallel operations of DNA strands can reach 10 to the 14th power per second. Of course, these advantages are based on the further development of DNA molecular manipulation technology. This also leads to the fact that although DNA computing has the advantage in computing time of molecular reactions, the preparation work of algorithm operation and the preparation work between experiments are seriously wasted. But from the perspective of development, with the maturity of DNA operation technology, the advantages of high parallelism, huge storage capacity, and low energy consumption response of DNA computing will be brought into play in the future. Now, DNA computing algorithms are mainly used to solve NP problems with exponential feasible solutions. Especially in the era of big data in the future, the scale of the problem to be solved, the amount of data and the number of feasible solutions are huge. In this case, ordinary computing algorithms are powerless. DNA computing just takes advantage of its own advantages to play its own advantages in such problems. In addition, the effective solution of this problem has certain reference significance and application performance for resource planning.

However, our algorithm also has some shortcomings. Our algorithm to solve this problem is only compared with the classical algorithm, and the conclusion of comparison is relatively limited. The reason is that the main purpose of this paper is to show the characteristics and advantages of the algorithm in reducing the computational complexity of the problem algorithm, so we only compare it with the previous literature studies that have obviously proposed the computational complexity. The vast majority of literature studies mainly reveal the efficiency performance of the algorithm, while fewer literature studies clearly give the analysis of

algorithm complexity. Therefore, our comparative literature studies are limited. On the other hand, DNA computing algorithm is mainly based on the biological DNA molecular chemical reaction to achieve the output of the algorithm function. Because the computer programs are executed in sequence, it is impossible to realize the parallel chemical reaction operation of molecules in the DNA algorithm. Therefore, the Python program designed by our simulation analysis can only realize the biological experiment results we designed. Compared with other algorithms (ant colony algorithm [33], particle swarm optimization algorithm [34], collaborative optimization algorithm [35], and genetic algorithm [36]), DNA algorithm is not ideal because of the different computing mechanism. Therefore, in the past papers on DNA computing, most of them explained the feasibility of the algorithm from the theoretical point of view [40, 43–45, 47, 48]. Of course, the problems raised by the reviewers are also the research direction of our efforts in the future, and we hope that the research work in this area can make a breakthrough.

Because the various DNA operations used in the algorithm model can already be operated in the laboratory, the method can be put into practical application. Now, DNA computer has broadened people's vision, accelerated the development of bioinformatics, brought corresponding problems, and put forward challenges to many fields. We believe that more efficient DNA algorithms can be used to solve more problems in future research.

## Data Availability

All data included in this study are available upon request by contact with the corresponding author.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

The project was supported by the National Natural Science Foundation of China (Grant no. 11701363). It was also funded by the Research Start-up Project of Wenzhou Business School (RC202002).

## References

- [1] M. G. Guan, "Odd-even-point graphical operation method," *Acta Mathematica Sinica*, vol. 3, pp. 263–266, 1960.
- [2] V. Yadav and S. Karmakar, "Sustainable collection and transportation of municipal solid waste in urban centers," *Sustainable Cities and Society*, vol. 53, Article ID 101937, 2020.
- [3] O. Quirion-Blais, A. Langevin, and M. Trépanier, "A case study of combined winter road snow plowing and de-icer spreading," *Canadian Journal of Civil Engineering*, vol. 44, no. 12, pp. 1005–1013, 2017.
- [4] H. F. Rahman and I. Nielsen, "Scheduling automated transport vehicles for material distribution systems," *Applied Soft Computing*, vol. 82, Article ID 105552, 2019.
- [5] G. Li, R. Lin, M. Li, R. Sun, and S. Piao, "A master-slave separate parallel intelligent mobile robot used for autonomous pallet transportation," *Applied Sciences*, vol. 9, no. 3, p. 368, 2019.
- [6] E. Gregory and M. Gutin, "Chinese Postman Problem on edge-colored multigraphs," *Discrete Applied Mathematics*, vol. 217, pp. 196–202, 2017.
- [7] Ero, E. lu and M. Azizo, lu, "Exact solution approaches for the directed bi-objective Chinese postman problem," *Journal of Industrial and Engineering Chemistry*, vol. 29, pp. 15–30, 2018.
- [8] M. E. Keskin and M. Yilmaz, "Chinese and windy postman problem with variable service costs," *Soft Computing*, vol. 23, no. 16, pp. 7359–7373, 2019.
- [9] M. K. Codur and M. Yilmaz, "A time-dependent hierarchical Chinese postman problem," *Central European Journal of Operations Research*, vol. 17, pp. 1–30, 2020.
- [10] J. Edmonds, "Chinese postman problem," *Operational Research*, vol. 17, 1965.
- [11] Z. Yin, F. Zhang, and J. Xu, "A Chinese Postman Problem based on DNA computing," *Journal of Chemical Information and Computer Sciences*, vol. 42, no. 2, pp. 222–224, 2002.
- [12] H. Jiang and L. S. Li, "Genetic algorithm for Chinese postman problems," *Wuhan University Journal of Natural Science*, vol. 8, no. 1, pp. 316–318, 2003.
- [13] R. Fei and D. Cui, "Research on Chinese postman problem decision process algorithm," *Journal of Computer Research and Development*, vol. 42, no. 2, pp. 294–299, 2005.
- [14] W. Li and L. Wang, "DNA computation for Chinese postman problem," *Journal of Computer Applications*, vol. 29, no. 7, pp. 1880–1883, 2009.
- [15] B. S. E. Zoraida, "DNA algorithm employing temperature gradient for chinese postman problem. process automation, control and computing (pacc)," in *Proceedings of the 2011 International Conference on, 2011*, Bonn, Germany, March 2011.
- [16] S.-U. Lee, "An eulerian cycle algorithm for Chinese postman problem," *Journal of the Korea Society of Computer and Information*, vol. 21, no. 7, pp. 47–52, 2016.
- [17] J. Nossack, B. Golden, E. Pesch, and R. Zhang, "The windy rural postman problem with a time-dependent zigzag option," *European Journal of Operational Research*, vol. 258, no. 3, pp. 1131–1142, 2017.
- [18] X. Meng, J. Li, M. Zhou, X. Dai, and J. Dou, "Population-based incremental learning algorithm for a serial colored traveling salesman problem," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 2, pp. 277–288, 2018.
- [19] L. Wang and J. Lu, "A memetic algorithm with competition for the capacitated green vehicle routing problem," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 2, pp. 516–526, 2019.
- [20] C. Carmine, D. Benjamin, W. Xingyin et al., "A two-stage solution approach for the directed rural postman problem with turn Penalties," *European Journal of Operational Research*, vol. 272, pp. 754–765, 2019.
- [21] X. Xu, J. Li, and M. Zhou, "Delaunay-triangulation-based variable neighborhood search to solve large-scale general colored traveling salesman problems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, 2020.
- [22] J. Wang, Y. Sun, Z. Zhang, and S. Gao, "Solving multitrip pickup and delivery problem with time windows and manpower planning using multiobjective algorithms," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 4, pp. 1134–1153, 2020.
- [23] L. Adleman, "Molecular computation of solutions to combinatorial problems," *Science*, vol. 266, no. 5187, pp. 1021–1024, 1994.

- [24] R. Lipton, "DNA solution of HARD computational problems," *Science*, vol. 268, no. 5210, pp. 542–545, 1995.
- [25] Q. Ouyang, P. D. Kaplan, S. Liu, and A. Libchaber, "DNA solution of the maximal clique problem," *Science*, vol. 278, no. 5337, pp. 446–449, 1997.
- [26] Z. Song, J. Xu, J. Xu, and B. Zhen, "Mixed-coexistence of periodic orbits and chaotic attractors in an inertial neural system with a nonmonotonic activation function," *Mathematical Biosciences and Engineering*, vol. 16, no. 6, pp. 6406–6425, 2019.
- [27] Z. Ji, Z. Wang, X. Bao, X. Wang, and T. Wu, "Research on water resources optimal scheduling problem based on parallel biological computing," *Desalination and Water Treatment*, vol. 111, pp. 88–93, 2018.
- [28] Z. Ji, Z. Wang, A. Deng, W. Huang, and T. Wu, "A new parallel algorithm to solve one classic water resources optimal allocation problem based on inspired computational model," *Desalination and Water Treatment*, vol. 160, pp. 214–218, 2019.
- [29] S. Yao, L. Ding, Z. Song, and J. Xu, "Two bifurcation routes to multiple chaotic coexistence in an inertial two-neural system with time delay," *Nonlinear Dynamics*, vol. 95, no. 2, pp. 1549–1563, 2019.
- [30] Z. Wang, J. Pu, L. Cao, and J. Tan, "A parallel biological optimization algorithm to solve the unbalanced assignment problem based on DNA molecular computing," *International Journal of Molecular Sciences*, vol. 16, no. 10, pp. 25338–25352, 2015.
- [31] Z. Song, B. Zhen, and D. Hu, "Multiple bifurcations and coexistence in an inertial two-neuron system with multiple delays," *Cognitive Neurodynamics*, vol. 14, no. 3, pp. 359–374, 2020.
- [32] H. Zhao, J. Zheng, J. Xu, and W. Deng, "Fault diagnosis method based on principal component analysis and broad learning system," *IEEE Access*, vol. 7, pp. 99263–99272, 2019.
- [33] 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.
- [34] W. Deng, R. Yao, H. Zhao, X. Yang, and G. Li, "A novel intelligent diagnosis method using optimal LS-SVM with improved PSO algorithm," *Soft Computing*, vol. 23, no. 7, pp. 2445–2462, 2019.
- [35] W. Deng, H. Zhao, L. Zou, G. Li, X. Yang, and D. Wu, "A novel collaborative optimization algorithm in solving complex optimization problems," *Soft Computing*, vol. 21, no. 15, pp. 4387–4398, 2017.
- [36] R. Li, Y. Chang, and Z. Wang, "Study on optimal allocation of water resources in Dujiangyan irrigation district of China based on improved genetic algorithm," *Water Supply*, vol. 24, 2020.
- [37] Z. Wang, X. Ren, Z. Ji, W. Huang, and T. Wu, "A novel bio-heuristic computing algorithm to solve the capacitated vehicle routing problem based on Adleman-Lipton model," *Bio-systems*, vol. 184, Article ID 103997, 2019.
- [38] X. Ren, X. Wang, Z. Wang, and T. Wu, "Parallel DNA algorithms of generalized traveling salesman problem-based bioinspired computing model," *International Journal of Computational Intelligence Systems*, vol. 14, no. 1, pp. 228–237, 2020.
- [39] W. Li, E. M. Patrikeev, and D. Xiao, "A DNA Algorithm for the maximal matching problem," *Automation and Remote Control*, vol. 76, no. 10, pp. 1797–1802, 2015.
- [40] K. H. Zimmermann, Z. Ignatova, and M. P. Israel, *DNA Computing Models*, Springer-Verlag, Berlin, Germany, 2008.
- [41] Z. Wang, Z. Ji, X. Wang, T. Wu, and W. Huang, "A new parallel DNA algorithm to solve the task scheduling problem based on inspired computational model," *Biosystems*, vol. 162, pp. 59–65, 2017.
- [42] W. Li, D. Xiao, and L. He, "DNA ternary addition," *Applied Mathematics and Computation*, vol. 182, no. 2, pp. 977–986, 2006.
- [43] Z. Wang, D. Huang, J. Tan, T. Liu, K. Zhao, and L. Li, "A parallel algorithm for solving the  $n$ -queens problem based on inspired computational model  $n$ -Queens Problem Based on Inspired Computational Model," *BioSystems*, vol. 131, pp. 22–29, 2015.
- [44] X. Zheng, J. Xu, and W. Li, "Parallel DNA arithmetic operation based on  $n$ -moduli set  $n$ -moduli set," *Applied Mathematics and Computation*, vol. 212, no. 1, pp. 177–184, 2009.
- [45] C. A. A. Sanches and N. Y. Soma, "A polynomial-time DNA computing solution for the Bin-Packing Problem," *Applied Mathematics and Computation*, vol. 215, no. 6, pp. 2055–2062, 2009.
- [46] R. Maazallahi, A. Niknafs, and P. Arabkhedri, "A polynomial-time DNA computing solution for the  $N$ -queens problem," *Procedia-Social and Behavioral Sciences*, vol. 83, pp. 622–628, 2013.
- [47] R. S. Braich, C. Johnson, P. W. K. Rothmund, D. Hwang, N. Chelyapov, and L. M. Adleman, "Solution of a satisfiability problem on a gel-based DNA computer," in *Proceedings of the Sixth International Conference on DNA Computation (DNA 2000)*, *Lecture Notes in Computer Science*, pp. 27–42, Leiden, The Netherlands, June 2001.
- [48] C. A. A. Sanches and N. Y. Soma, "A general resolution of intractable problems in polynomial time through DNA Computing," *BioSystems*, vol. 150, pp. 119–131, 2016.
- [49] T. Hosoya, I. Kawamata, S.-I. M. Nomura, and S. Murata, "Pattern formation on discrete gel matrix based on DNA computing," *New Generation Computing*, vol. 37, no. 1, pp. 97–111, 2019.
- [50] H. Zhang and X. Liu, "A CLIQUE algorithm using DNA computing techniques based on closed-circle DNA sequences," *BioSystems*, vol. 105, no. 1, pp. 73–82, 2011.
- [51] R. S. Braich, C. Johnson, P. W. K. Rothmund, N. Chelyapov, and L. M. Adleman, "Solution of a 20-variable 3-SAT problem on a DNA computer," *Science*, vol. 296, no. 5567, pp. 499–502, 2002.
- [52] R. B. A. Bakar, J. Watada, and W. Pedrycz, "DNA approach to solve clustering problem based on a mutual order," *Bio-Systems*, vol. 91, no. 1, pp. 1–12, 2008.
- [53] J. Edmonds, E. L. Johnson, and J. Matching, "Matching, Euler tours and the Chinese postman," *Mathematical Programming*, vol. 5, no. 1, pp. 88–124, 1973.
- [54] K. Holmberg, "Heuristics for the rural postman problem," *Computers & Operations Research*, vol. 37, no. 5, pp. 981–990, 2010.
- [55] V. Kundeti, S. Rajasekaran, H. Dinh, and H. E. I. U. Dinh, "An efficient algorithm for Chinese postman walk on bi-directed de bruijn graphs," *Discrete Mathematics, Algorithms and Applications*, vol. 4, no. 2, Article ID 1250019, 2012.
- [56] G. Gutin, M. Jones, and B. Sheng, "Parameterized complexity of the  $k$ -arc Chinese postman problem," *Journal of Computer and System Sciences*, vol. 84, pp. 107–119, 2017.

- [57] M. Nilofer and M. Rizwanullah, "An implementation of Chinese postman problem with priorities," *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 3, pp. 3301–3305, 2020.