

## A Novel Algorithm for Adjacent Vertex-Distinguishing Edge Coloring of Large-scale Random Graphs

Zhao Huanping<sup>1,\*</sup>, Zhu Peijin<sup>1</sup>, Li Jingwen<sup>2</sup> and Shi Huojie<sup>3</sup>

<sup>1</sup>School of Computer and Software, Nanyang Institute of Technology, Nanyang 473004, China

<sup>2</sup>School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China

<sup>3</sup>Department of Biological Systems Engineering, Washington State University, Pullman, MI 98101, United States

Received 28 May 2021; Accepted 30 June 2021

### Abstract

Graph coloring is one of the areas in graph theory with high research importance. Adjacent vertex-distinguishing edge coloring is a type of multi-conditional coloring in graph coloring, but existing associated studies lack analysis on constraint conditions. In this study, a novel algorithm was designed to increase the adjacent vertex-distinguishing edge coloring efficiency of large-scale random graphs. Sub-graphs were produced in this work by using a nondestructive segmentation algorithm to reduce the scale of random graphs, and random pre-coloring was performed on the edges of each sub-graph. Iteration was performed step by step in accordance with regulations by searching the conflict set of inaccurate edge coloring until the color met the requirements of the ultimate objective function. Next, the sub-graphs that were colored successfully were combined to realize adjacent vertex-distinguishing edge coloring of large-scale random graphs. Afterward, the accuracy of the adjacent vertex-distinguishing edge chromatic number was proven through theoretical analysis and experimental comparison. Several experiments were also performed on random graphs with less than 4000 vertexes and an edge density smaller than 0.1. Results show that when the number of vertexes is greater than 2000 and the edge density exceeds 0.07, the run time generally rings from 0.9 s to 1.5 s, whereas the run time for other random graphs is between 0.6 and 1.2 s. The algorithm can solve the adjacent vertex-distinguishing edge chromatic number of random graphs effectively, and the time complexity of the algorithm do not exceed  $O(n^3)$ . The proposed algorithm provides evidence for solving the shortest path of large-scale random graphs.

*Keywords:* Random graphs, Adjacent vertex-distinguishing edge coloring, Adjacent vertex-distinguishing edge chromatic number, Segmentation algorithm, Multi-objective optimization

### 1. Introduction

Graph coloring, a classical problem in graph theory, originates from the well-known “four-color conjecture.” Many problems in practical life, such as computer communication, traffic orientation, goods storage, and combined optimization, can be solved by transforming them into graph coloring [1-7]. Thus, graph coloring is one topic with important practical value and theoretical importance in graph theory. However, classical intelligent optimization algorithms, such as genetic algorithm and neural network, have shortages and limitations in the adjacent vertex-distinguishing edge coloring of large-scaled random graphs. Therefore, acquiring adjacent vertex-distinguishing edge coloring of large-scaled random graphs performance quickly and effectively is a key problem that needs to be solved.

Existing studies on graph coloring mainly cover those on theories and on algorithms. In modern times, some mathematical researchers emphasize graph coloring problems [8-11]. Zhang et al. [12] proposed the concept and conjecture of strong vertex-distinguishing total coloring for graphs based on adjacent vertex-distinguishing total coloring and strong adjacent vertex-distinguishing total coloring. Nevertheless, graph coloring is considered an NP-complete problem. Traditional intelligence algorithms [13], such as

genetic algorithm, ant colony algorithm, and neural network, are generally limited to solving graph coloring problems of single constraint, and they could obtain the expected coloring results under small-scale graphs. Ran and Zhang [14] achieved four-color graph coloring effectively by the improved heuristic ant colony algorithm. Zhang et al. [15] searched the initial solution of the genetic algorithm by using the ant colony algorithm and solved the vertex coloring problem involving multiple vertices by using the improved ant colony algorithm based on the genetic algorithm. Ordinary intelligent algorithms present serious limitations in solving the adjacent vertex-distinguishing edge coloring problem of random graphs when the graph scale is greater than 600.

Existing studies have rarely discussed the algorithm for adjacent vertex-distinguishing edge coloring of large-scale random graphs. In consideration of the adjacent vertex-distinguishing edge coloring characteristics of large-scale random graphs, sub-graphs were produced in this study by using a nondestructive segmentation algorithm to reduce the scale of graphs. Adjacent vertex-distinguishing edge coloring was performed on all sub-graphs in accordance with the multi-objective optimization function. Then, all sub-graphs after successful coloring were combined to complete the coloring. This approach effectively shortens the run time of the algorithm and increases the coloring efficiency.

\*E-mail address: 34875716@qq.com

ISSN: 1791-2377 © 2021 School of Science, IHU. All rights reserved.

doi:10.25103/jestr.143.08

## 2. State of the art

Existing studies on graph coloring mainly focus on theoretical studies. In 1993, Burriss [16] introduced and studied vertex-distinguishing edge coloring (also known as strong edge coloring). Major conclusions on the vertex-distinguishing edge coloring of graphs were mainly summarized in previous studies [17-20]. In 2002, Zhang et al. [21] proposed the concept and conjecture of strong adjacent vertex-distinguishing edge coloring based on vertex-distinguishing edge coloring. Many relevant studies have been reported worldwide [22-28]. In 2007, Zhang et al. [29] added a constraint to the concept of adjacent vertex-distinguishing edge coloring and proposed the concept of strong adjacent vertex-distinguishing total coloring. Meanwhile, they gained the accurate value of strong adjacent vertex-distinguishing total coloring for special graphs and the upper boundary conjecture. For a simple graph with less than three orders,  $\chi_{st} \leq n + \lceil \log_2 n \rceil + 1$  may be used.

Studies on graph coloring algorithms can generally be divided into three types. The first type recognizes the law of graph coloring by combined construction, but this method is only applicable to unique graphs. The second type gives rough lower and upper boundaries of graph coloring based on the probability statistical method. However, such boundaries are markedly rough and, therefore, have some limitations. The third type processes graph coloring problems by using a computer, which can solve large-scale graph coloring problems by designing a reasonable high-efficiency algorithm based on the great operating capacity of computers. This approach can prove some conjectures. For example, based on the sequence approximation method, Appel and Haken [30] proved and solved the four-color graph coloring problem under computer assistance based on order approximation in 1976 (the computer operated for more than 1,200 hours). However, their method required a great deal of time in solving large-scale problems and the computation time was proportional to  $O(n^2)$ , where  $n$  represented the number of regions in the graph. By combining the advantages and disadvantages of the taboo search and genetic algorithms, Li and He [31] generated an initial solution by using the genetic algorithm to implement field-changing searching and update the vertex coloring by the taboo searching algorithm; their results showed an increase in the searching speed of the algorithm. Liao and Ma [32] analyzed graph coloring based on the heuristic searching ant algorithm and gained the expected results for small graphs. Yu [33] applied the simulated annealing algorithm to graph coloring, but the initial value and parameter determination of the algorithm could directly affect its performance. Any improper setting of parameters would lead to slow convergence and long implementation time. Yu et al. [34] provided the graph coloring model of uncertainty based on "DNA Origami." Li [35-36] studied the strong vertex-distinguishing total coloring and effectively calculated the vertex-distinguishing total coloring number for a graph with a fixed random number of vertices. Moreover, the time complexity of the solving algorithm was lower than  $O(n^3)$ . Cao et al. [37] realized vertex-distinguishing total coloring of graphs  $K_{2n+1} \setminus E(W_m)$  by using a heuristic algorithm and verified the vertex-distinguishing total chromatic number and relevant conjecture of graphs that Li Muchun et al. presented in Reference [38]. Although studies on vertex-distinguishing coloring of large-scale graphs have produced several useful outcomes [39-42], no

effective method is available to solve the coloring issues of large-scale random graphs.

Previous studies on graph coloring were based on  $K$ -vertex coloring. Several scholars have studied adjacent vertex-distinguishing edge coloring by adopting classical algorithms, but they focused on a relatively small scale of research objects. If these algorithms are applied to adjacent vertex-distinguishing edge coloring of large-scale random graphs, the run time and rate of convergence will increase to an unacceptable degree because the chromatic number increases continuously with graph scale and density. In the current study, nondestructive segmentation of large-scale graphs was performed by breaking two-degree or three-degree vertexes. Adjacent vertex-distinguishing edge coloring of small-scale sub-graphs after segmentation was completed with the assistance of iterative exchange of a complementary set of the coloring matrix, thus accelerating the convergence of the algorithm.

The remainder of this study is organized as follows. Section 3 describes the concept of adjacent vertex-distinguishing edge coloring and the constraint function and design of the algorithm for adjacent vertex-distinguishing edge coloring. Section 4 analyzes the experimental results and the proposed algorithm. Section 5 summarizes the conclusions.

## 3. Methodology

### 3.1 Relevant definitions

For any undirected graph  $G(V, E)$ ,  $V(G)$  refers to the vertex set of graph  $G$ ,  $E(G)$  is the edge set of graph  $G$ ,  $C(u)$  is the color set used by vertex  $u$  and relevant edges in graph  $G$ , and  $\bar{C}(u)$  is the complementary set of  $C(u)$ . The relevant definitions of the coloring of graph  $G$  are introduced as follows.

**Definition 1:** For graph  $G(V, E)$ , mapping  $f: E(G) \rightarrow \{1, 2, \dots, k\}$  meets  $f(e) \neq f(e')$  for any adjacent edge  $e$  and  $e'$ . Hence,  $f$  is a  $k$ -normal edge coloring of  $G$  and recorded as  $k$ -PEC.

**Definition 2:** Mapping  $f$  of one  $k$ -PEC of  $G(V, E)$  also meets any  $uv \in E(G)$  and  $C(u) \neq C(v)$ . Hence,  $f$  is an adjacent vertex-distinguishing edge coloring of the graph and recorded as  $k$ -AVDEC.  $\chi'(G) = \min\{k \mid k\text{-AVDEC of } G\}$  is the adjacent vertex-distinguishing edge chromatic number of  $G$ , where  $C(u) = \{f(uv) \mid uv \in E(G)\}$ .

**Conjecture 1:** If a connected graph has two adjacent maximum vertexes with at least three match exponents, it has

$$\chi_{as}(G) \geq \Delta(G) + 1 \quad (1)$$

### 3.2 Construction of the constraint function

According to the concept of adjacent vertex-distinguishing edge coloring of a graph, obtaining different color sets of adjacent vertexes is required, except for normal edge coloring. Moreover, adjacent vertex-distinguishing edge coloring of graph  $G$  with  $n$  vertexes requires two constraints, namely, (a) the adjacent edges have different colors and (b) the color sets of adjacent vertexes are different. In accordance with these constraints, the constraint function can be defined as follows.

#### 3.2.1 Edge constraint function

$G(V,E)$  involves mapping  $f: V(G) \rightarrow \{1,2,\dots,k\}$ , and the edge constraint function is defined as follows. Suppose that  $e_i, e_j \in E(G)$  and  $e_i, e_j$  are adjacent edges. Let

$$f_1(e_i, e_j) = \begin{cases} 1 & f(e_i) = f(e_j) \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Then,

$$F_1(e_i, e_j) = \sum_{e_i, e_j \in E(G)} f_1(e_i, e_j) \quad (3)$$

where  $F_1(e_i, e_j)$  denotes the number of edges that do not meet Constraint (a). Constraint (b) is met only when  $F_1(e_i, e_j) = 0$ .

### 3.2.2 Constraint function of the color set

Graph  $G(V,E)$  involves mappings  $f: \{V(G), E(G)\} \rightarrow \{1,2,\dots,k\}$  and  $C(v_i) = \{f(v_i, v_j) | v_i, v_j \in E(G)\}$ , where  $i, j = 1, 2, \dots, n$ . The constraint function of the color set is defined as follows.

For any  $u, v \in V(G)$ , let

$$f_2(u, v) = \begin{cases} 1 & C(v) = C(u) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Then,

$$F_2(u, e) = \sum_{u, v \in V(G)} f_2(u, v) \quad (5)$$

where  $F_2(u, e)$  is the number of color sets of the vertex that do not meet Constraint (b). Constraint (b) is met only when  $F_2(u, e) = 0$ .

### 3.2.3 Total objective function

$$F_{as} = F_1 + F_2 \quad (6)$$

where  $F_{as}$  is the quantity that does not meet four coloring conditions, and the coloring is successful only when  $F_{as} = 0$ .

### 3.3 Algorithm description

The basic idea of the proposed novel algorithm is to perform nondestructive segmentation of large-scale random graphs and color the sub-graphs subsequently. The colored sub-graphs are reduced and combined, thus obtaining the coloring results of large-scale random graphs.

The proposed novel algorithm is composed of Algorithms 1 and 2. Algorithm 1 is the segmentation algorithm of random graphs, and Algorithm 2 is the adjacent vertex-distinguishing edge coloring algorithm of random graphs.

**Algorithm 1:** Segmentation algorithm of random graphs

**Input:** adjacent matrix of large-scale random graphs

**Output:** sub-graphs, segmented vertexes, and segmented sets of large-scale random graphs

**Step 1)** Input the adjacent matrix color  $[ ]$  of large-scale random graphs.

**Step 2)** Search the two-degree and three-degree vertexes and input into the arrays degree\_2[] and degree\_3[], respectively.

**Step 3)** Disconnect the two-degree vertexes of large-scale random graphs in accordance with the rules of the algorithm. The breaking vertexes generate new vertexes, and the vertexes are added because of the vertex of the original graph. Search the sub-graphs  $(p_1, p_2, \dots, p_n)$ , which are stored independently, by using the depth-first traversal algorithm.

**Step 4)** Judge the segmentation threshold of these sub-graphs, which is the proportion of the total vertexes and edges of single sub-graphs in the large-scale random graphs. The range of the segmentation threshold for graphs with less than 3 000 vertexes is  $[0.05, 0.1]$ . The sub-graphs after segmentation generally have an equal scale. If this condition still cannot be met, continue to break the three-degree vertexes and return to Step 4.

The calculation formula of the segmentation threshold is

$$q(G_i) = \frac{n_i + e_i}{n + e} \quad (7)$$

where  $n_i, e_i$  is the number of vertexes and edges of sub-graph  $i$  and  $n$  and  $e$  are number of vertexes and edges of a large-scale random graph.

**Step 5)** Place the segmented vertexes and segmented sets of the large-scale random graphs in arrays point\_g[] and point\_sets[].

**Step 6)** Input all sub-graphs into the algorithm for adjacent vertex-distinguishing edge coloring. In the coloring process, the adjacent edges of the original graph must be controlled and colored differently, and the colored sub-graphs must be reduced. The vertexes produced from the original graph are combined into one (adjacent vertex-distinguishing edge coloring colors only the edges and not the vertex), thus realizing coloring of the original graph.

**Step 7)** Examine whether the final coloring results are accurate. If they are accurate, output the adjacent matrix; otherwise, return to Step 1.

**Algorithm 2** Adjacent vertex-distinguishing edge coloring algorithm

**Input:** adjacent matrix of  $G$  of random graphs

**Output:** adjacent vertex-distinguishing edge coloring matrix

**Step 1)** Normal edge coloring: In accordance with the initial chromatic number, select colors randomly from the colors that are not greater than the chromatic number for random pre-coloring of edges in the graph. Next, determine the conflicts and perform iterative adjustment on edge colors in the edge coloring conflicts. After several iterations, obtain the normal edge coloring results of graphs.

**Step 2)** Establish statistics on color set  $C(u)$  of different vertexes and the corresponding complementary set  $\bar{C}(u)$ . Generate the ranking set manyc[ ] in accordance with the number of elements in  $\bar{C}(u)$ .

**Step 3)** Calculate the value of the constraint function ( $F_2$ ) of the color set. Initialize  $\bar{C}(v_i)$  and set it to 0. When a conflict exists between a pair of color sets, that is,  $\bar{C}(v_i) = \bar{C}(v_j)$ ,  $\text{sum\_com}[i] = \text{sum\_com}[j]$ , and  $v_i, v_j \in E(G)$ , add 1. Detect all color sets one by one and calculate the value of  $F_2$ . If  $F_2 \neq 0$ , go to Step4.

**Step 4)** Solve the color set conflict. It has to replace elements in the complementary set and elements in the matrix. Compare the first vertex in the ranking set manyc[ ] to all follow-up vertexes. If an edge exists between two vertexes and their complementary sets have the intersection  $\bar{C}(v_i) \cap \bar{C}(v_j) = \{a_1, a_2, \dots, a_i\}$ , replace the color in the matrix. Let

$f_2(u,v) = a_i$  and modify the values of  $\bar{c}(v_i)$  and  $\bar{c}(v_j)$ . After the replacement, determine whether the color sets of adjacent vertexes have the same colors. If  $F_2 \neq 0$ , a conflict still exists, and another replacement is needed. This task continuous for several color iterations until  $F_2 = 0$ .

**Step 5)** Complete the coloring. Determine whether edge coloring and the color set are normal. If they are, output the final coloring results; otherwise, repeat Step 5.

**4. Result analysis and discussion**

**4.1 Algorithm test**

In accordance with the steps of the algorithms, the test results of graph G with 15 vertexes are obtained.

**Step 1)** First, an undirected graph with 15 vertexes is generated randomly. The value is 1 when edges are present; otherwise, the value is 0. The adjacent matrix of this undirected graph is shown in Fig. 1.

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_3 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_4 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_5 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_7 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_8 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ v_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ v_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ v_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ v_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ v_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Fig. 1. Adjacent matrix of random graph G

**Step 2)** Statistics on the degree arrays of different vertexes are established, and the initial chromatic number k is determined. The numbers of vertexes under different degrees are listed in Table 1.

**Table 1.** Number of vertexes of the graph under different degrees

Number of degrees	4	3	2	1
Number of vertexes	3	5	2	5

Given that the adjacent vertex-distinguishing edge chromatic number is not less than the maximum degree +1,

the initial chromatic number is k=5.

**Step 3)** The number of two-degree vertexes (V3 and V8) is recorded. With the graph segmentation algorithm, new vertexes v16 and v17 are generated by breaking V3 and V8. The adjacent matrixes of G1 and G2 after segmentation are shown in Figs. 2 and 3.

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 & v_{16} \\ v_1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ v_3 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_4 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ v_5 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ v_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ v_7 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ v_8 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ v_{16} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Fig. 2. Adjacent matrix of G1

$$\begin{pmatrix} v_{17} & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{17} & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ v_9 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ v_{10} & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ v_{11} & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ v_{12} & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ v_{13} & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Fig. 3. Adjacent matrix of G2

**Step 4)** Adjacent vertex-distinguishing edge coloring is performed on G1 and G2. The following text introduces the detailed process of adjacent vertex-distinguishing edge coloring of G2.

**Step 4.1)** The results after normal edge coloring are shown in Fig. 4.

$$\begin{pmatrix} v_{17} & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{17} & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 \\ v_9 & 2 & 0 & 4 & 5 & 1 & 0 & 0 & 4 \\ v_{10} & 0 & 4 & 0 & 2 & 5 & 0 & 0 & 3 \\ v_{11} & 0 & 5 & 2 & 0 & 1 & 0 & 0 & 3 \\ v_{12} & 0 & 1 & 5 & 1 & 0 & 4 & 0 & 4 \\ v_{13} & 0 & 0 & 0 & 0 & 4 & 0 & 3 & 5 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 \end{pmatrix}$$

Fig. 4. Normal edge coloring results of G2

**Step 4.2)** The ranking set  $manyc[8]=\{17,14,15,10,11,13,9,12\}$  is obtained in accordance with the coloring matrix. One vertex is selected from  $manyc[8]$  to compare with the follow-up vertexes. On the basis of the rule of the algorithms, the first exchange is performed for color sets that have conflicts.

$$V_{17} \leftrightarrow V_9; V_{14} \leftrightarrow V_{13}, V_{15} \leftrightarrow V_{13}, V_{10} \leftrightarrow V_{11}, V_{10} \leftrightarrow V_{12}, V_{11} \leftrightarrow V_9, V_{12} \leftrightarrow V_{13}$$

The exchange results are shown in Fig. 5.

$$\begin{pmatrix} v_{17} & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{17} & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 1 \\ v_9 & 3 & 0 & 4 & 2 & 1 & 0 & 0 & 4 \\ v_{10} & 0 & 4 & 0 & 3 & 2 & 0 & 0 & 3 \\ v_{11} & 0 & 2 & 3 & 0 & 1 & 0 & 0 & 3 \\ v_{12} & 0 & 1 & 2 & 1 & 0 & 5 & 0 & 4 \\ v_{13} & 0 & 0 & 0 & 0 & 5 & 0 & 1 & 2 & 3 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 5. First exchange results

**Step 4.3)** On the basis of the first exchange results, the second exchange is performed according to the rules of the algorithm.

$$V_{17} \leftrightarrow V_9, V_{14} \leftrightarrow V_{13}, V_{15} \leftrightarrow V_{13}, V_{10} \leftrightarrow V_{11}, V_{10} \leftrightarrow V_{12}, V_{11} \leftrightarrow V_{12}, V_{12} \leftrightarrow V_{13}$$

The exchange results are shown in Fig. 6.

$$\begin{pmatrix} v_{17} & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_{17} & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 1 \\ v_9 & 5 & 0 & 4 & 3 & 1 & 0 & 0 & 4 \\ v_{10} & 0 & 4 & 0 & 5 & 3 & 0 & 0 & 3 \\ v_{11} & 0 & 3 & 5 & 0 & 4 & 0 & 0 & 3 \\ v_{12} & 0 & 1 & 3 & 4 & 0 & 5 & 0 & 4 \\ v_{13} & 0 & 0 & 0 & 0 & 5 & 0 & 4 & 1 & 3 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 1 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 6. Second exchange results

**Step 4.4)** After the exchange, a conflict occurs between the color sets of V9 and V12. According to the rule of the algorithm, exchange  $V_{12} \leftrightarrow V_{13}$ . Similarly, a conflict exists between the color sets of V10 and V12. According to the rule of the algorithm, exchange  $V_{11} \leftrightarrow V_9$ .

The adjacent vertex-distinguishing edge coloring outcomes obtained after solving the conflicts between color sets are shown in Fig. 7.

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 \\ v_1 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 5 & 0 & 4 & 2 & 1 & 0 & 0 & 0 \\ v_3 & 0 & 4 & 0 & 5 & 3 & 0 & 0 & 0 \\ v_4 & 0 & 2 & 5 & 0 & 4 & 0 & 0 & 0 \\ v_5 & 0 & 1 & 3 & 4 & 0 & 2 & 0 & 0 \\ v_6 & 0 & 0 & 0 & 0 & 2 & 0 & 4 & 1 \\ v_7 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ v_8 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Fig. 7. Adjacent vertex-distinguishing edge coloring results of G2

**Step 5)** Adjacent vertex-distinguishing edge coloring of G1 is performed (Fig. 8).

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 & v_{16} \\ v_1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 2 & 0 & 3 & 1 & 0 & 0 & 0 & 0 & 0 \\ v_3 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_4 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 3 & 4 \\ v_5 & 0 & 0 & 0 & 2 & 0 & 1 & 3 & 0 & 0 \\ v_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ v_7 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ v_8 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ v_{16} & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Fig. 8. Adjacent vertex-distinguishing edge coloring results of G12

**Step 6)** The graphs after G1 and G2 coloring are reduced and combined, yielding the coloring outcomes for original graph G (Fig. 9).

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 & v_7 & v_8 & v_9 & v_{10} & v_{11} & v_{12} & v_{13} & v_{14} & v_{15} \\ v_1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_2 & 2 & 0 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_3 & 0 & 3 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_4 & 0 & 1 & 4 & 0 & 2 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_5 & 0 & 0 & 0 & 2 & 0 & 1 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_7 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ v_8 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 \\ v_9 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 4 & 2 & 1 & 0 & 0 \\ v_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 5 & 3 & 0 & 0 \\ v_{11} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 5 & 0 & 4 & 0 & 0 & 0 \\ v_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 & 4 & 0 & 2 & 0 & 0 \\ v_{13} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 4 & 1 \\ v_{14} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ v_{15} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Fig. 9. Adjacent vertex-distinguishing edge coloring results of G

**Step 7)** Examine whether the coloring results of G after combination meet the requirements. The examination shows that the results conform to Conjecture 1:  $\chi_{as}(G) \geq \Delta(G) + 1$ . Therefore, the coloring is accurate.

### 4.2 Experimental results

The relations among the number of vertexes, chromatic number, and edge density of several graphs were tested implementing the proposed novel algorithm on a computer with 64-bit Windows OS, 2 GB of memory, 500 G of hard disk, and VC6.0. The relation graphs were derived after many tests (Figs. 10 and 11). Chromatic number was determined to be positively related with the number of vertexes and edge density of coloring graphs.

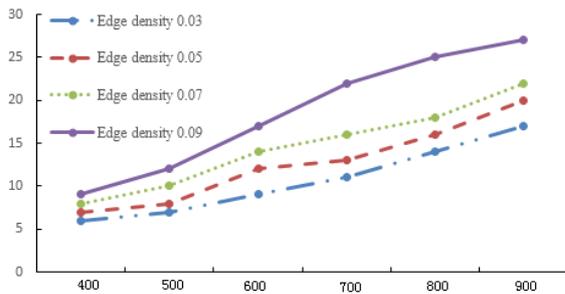


Fig. 10. Number of vertexes, edge density, and chromatic number (number of vertexes: <1000)

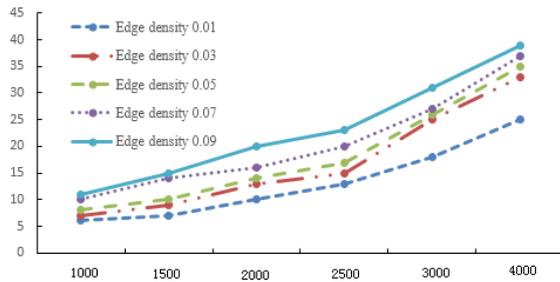


Fig. 11. Number of vertexes, edge density, and chromatic number (number of vertexes: 1000–4000)

### 4.3 Algorithm analysis

#### 4.3.1 Validity

The proposed novel algorithm implements random pre-coloring on the edges in a graph and examines the conflicts in edge coloring, that is, whether certain edges are colored the same. If yes, color iteration and exchange are performed based on the agreed rules of the algorithm until edge coloring meets the requirement  $F_1=0$ . Next, the algorithm judges whether the color sets of adjacent vertexes are similar. If they are, the colors are adjusted using the algorithm that solves the conflicts between color sets to make  $F_2=0$ . In this manner, all sub-objective functions can meet the requirements, and the adjacent vertex-distinguishing edge coloring is completed. The proposed algorithm achieves the total objective  $F_{as}=F_1+F_2$  through stepwise optimization of different sub-objectives. This procedure conforms to Conjecture 1, namely,  $\chi_{as}(G) \geq \Delta(G)+1$ . Hence, the proposed algorithm is valid.

#### 4.3.2 Time complexity

The proposed novel algorithm involves the following major steps:

- (a) An  $n \times n$  adjacent matrix is generated, and the time

complexity is  $T_1(n) = \sum_{i=1}^n \sum_{j=1}^n Random = O(n^2)$ .

- (b) Pre-coloring of edges: The worst time complexity of this step is equal to the time complexity of edge pre-coloring when G is a complete graph with n vertexes, i.e.,

$$T_2(n) = \sum_{i=1}^n \sum_{j=1}^n pre\_c = O(n^2)$$

- (c) The conflict in color sets between adjacent vertexes is solved. For G with n vertexes, the conflict sets are adjusted in the algorithm. The worst time complexity is

$$T_3(n) = \sum_{i=1}^n \sum_{j=1}^n exchange = O(n^3)$$

Comprehensive analysis shows that the run time of the novel algorithm is primarily determined by graph G, number of vertexes (n), and number of edges (m). In the worst condition, the time complexity of the novel algorithm is  $T(n) = O(n^3)$ .

## 5. Conclusions

Summarizing practical problems into graph coloring problems often involves random graphs. Hence, studying the algorithm for adjacent vertex-distinguishing edge coloring of random graphs has a high practical value. The proposed novel algorithm segments large-scale random graphs by nondestructive segmentation and completes adjacent vertex-distinguishing edge coloring of random graphs through independent coloring of sub-graphs. The following major conclusions were obtained from this study.

- (1) The analysis of constraints in the adjacent vertex-distinguishing edge coloring of graphs showed that  $\chi_{as}(G) \geq \Delta(G)+1$  if two adjacent maximum-degree vertexes exist.

- (2) Graph segmentation can increase the coloring efficiency. After segmentation, the ratio between the number of vertexes of sub-graphs and that of the large-scale graph is the segmentation threshold, which is in the range of [0.05, 0.1]. Image segmentation is successful, and the segmented sub-graphs basically have an equal scale.

- (3) The color sets are distinguished by a complementary set of different vertexes. The number of complementary sets is sufficient to generate a ranking set for handling the conflicts of color sets. When a conflict occurs between color sets, the elements in the complementary set can be selected for heuristic coloring to increase the coloring efficiency.

In consideration of the adjacent vertex-distinguishing edge coloring characteristics of large-scale random graphs, the proposed novel algorithm generates a multi-objective function according to the constraints and reduces the scale of graphs through the segmentation algorithm. The sub-graphs are combined after successful coloring, thus realizing the coloring of the large-scale random graphs. Moreover, several constraints in the novel algorithm can be increased to obtain other coloring results for random graphs, such as adjacent vertex-distinguishing total coloring and vertex-distinguishing total coloring. When the scale of a graph is relatively large, the run time of the novel algorithm increases by leaps. Hence, the time complexity of the novel algorithm should be reduced further to increase the coloring efficiency.

## Acknowledgements

The work was supported by the National Natural Science Foundation of China (Grant No. 10771091) and the Science and Technology Research Project of Henan Province (Grant No. 142102210554).

This is an Open Access article distributed under the terms of the Creative Commons Attribution License.



## References

1. Han, G. L., Du, W., "The application research of parking space allocation based on graph coloring model". *Telecom Express*, (7), 2019, pp.27-32.
2. Liu, W., Liao, R. B., "PCI automatic allocation method based on improved graph coloring algorithm". *Computer engineering and design*, 39(2), 2018, pp.352-357.
3. Pratap, A., Misra, R., Gupta, U., "Randomized Graph Coloring Algorithm for Physical Cell ID Assignment in LTE-A Femtocellular Networks". *Wireless Personal Communications*, 91(3), 2016, pp.1-23.
4. Oylum, x., eker, et al. "A decomposition approach to solve the selective graph coloring problem in some perfect graph families". *Networks*, 73(2), 2019, pp.145-169.
5. Ren, Y. L., Wang, L., Liang, D. Y., "Research on Railway Shunting Problem Based on Graph Coloring". *Advances in Applied Mathematics*, 10(2), 2021, pp.402-415.
6. Xu, F. M., Qian, H., Cao, H. Y., et al. "Seni-dynamic pilot allocation based on graph coloring in massive MIMO systems". *Telecommunications Science*, 36(11), 2020, pp.39-46.
7. Zheng, T. Y., Zhang, H., Zhou, C., "D2D Resource Allocation and Power Control Based on Improved Graph Coloring". *Telecommunication Engineering*, 384(11), 2020, pp.82-87.
8. Balister, N., Riordan, M., Schelp, R. H., "Vertex-distinguishing Proper Edge-Colorings". *J. Graph Theory*, 42(2), 2003, pp.95-109.
9. Bo, Y. H., Wang, L. X., "2-distance Coloring of Sparse Planar Graphs". *Advances in Mathematics*, 48(2), 2019, pp.145-155.
10. Dong, X. Y., Ma, D. J., "Strong edge coloring of Goldberg snark". *Journal of Northeast Normal University (Natural Science Edition)*, 50(4), 2018, pp.16-19.
11. Li, T., Chen, X. E., Wang, Z. W., "General Vertex-Distinguishing Total coloring of Corona graph of  $2K_2VK_1$ ". *Journal of Jilin University (Science Edition)*, 57(3), 2019, pp.544-552.
12. Zhang, Z. F., Li, J. w., C, X. E., "On adjacent vertex distinguishing total coloring of graphs". *ScienceChina (Ser A)*, 48(3), 2005, pp. 289-299.
13. Wei, J., "Application of Graph Coloring Algorithm Based on Ant Colony Optimization in Intelligent Community". *Journal of Jishou University (Natural Science Edition)*, 40(2), 2019, pp. 27-30,33.
14. Ran, C. S., Zhang, T., "Improved coloring algorithm based on ant colony for application in intelligence community". *Computer Engineering and Design*, 37(1), 2016, pp.152-153.
15. Zhang, X. P., Zhang, Y. Q., "An Ant Colony Genetic Algorithm for Graph Colouring Problem". *Computer Application and Software*, 31(11), 2014, pp.207-209.
16. Burris, A. C., "Vertex-Distinguishing Edge-Colorings". *PhD thesis Memphis State University*, 12(7), 1993, pp. 118 -129.
17. Dong, X. Y., "Strong Edge Coloring of Flower Snark". *Journal of Changchun Normal University*, 38(2), 2019, pp.4-8.
18. Whitehead, O., Zageglia, N., Salvi, O., "vability of the extended Fibonacci cubes", *Discrete Math*, 26(6), 2003, pp.431-440.
19. Balister, N., Riordan, O. M., Schelp, H., "Vertex-distinguishing edge coloring of graphs". *Graph Theory*, 56(42), 2003, pp.95-109.
20. Cui, J. F., "Discussing the Upper Bounds on Vertex-Distinguishing Edge coloring of a graph by probability method". *Journal of Capital Normal University (Natural Sciences Edition)*, 40(1), 2019, pp.12-14.
21. Zhang, Z. F., Lin, L. Z., Wang, J. F., "Adjacent strong edge colorings of graphs". *Applied Mathematics Letters*, 15(3), 2002, pp. 623-626.
22. Tian, S. L., Dong, X. F., Liu, R. L., "Adjacent vertex-distinguishing coloring of the semistrong product of graphs". *Operations Research Transactions*, 21(3), 2017, pp.119-125.
23. Balister, P. N., Guori, E., Lehel, J., "Adjacent vertex distinguishing edge colorings". *Discrete Mathematics*, 21(1), 2006, pp. 237-250.
24. Baril, J. L., Kheddouci, H., Togni, O., "Adjacent vertex distinguishing edge coloring of meshes and hypercubes". *Australasian Journal of Combinatory*, 35, 2006, pp. 89-102.
25. Hocquard, H., Montassier, M., "Adjacent vertex-distinguishing edge coloring of graphs with maximum degree at least five", *Electronic Notes in Discrete Mathematics*, 38, 2011, pp.457-462.
26. Hocquard, H., Montassier, M., "Adjacent vertex distinguishing edge coloring of graphs with maximum degree  $\Delta$ ". *Journal of Combinatorial Optimization*, 26(1), 2013, pp.152-160.
27. Li, Y. Y., "Adjacent vertex-distinguishable V-total coloring of some join graphs". *Mathematics in Practice and Theory*, 48(23), 2018, pp. 129-135.
28. Caro, Y., Hansberg, A., Montejano, A., "Unavoidable chromatic patterns in 2-colorings of the complete graph". *Journal of Graph Theory*, 97(4), 2020, pp.1-24.
29. Zhang, Z. F., Cheng, H., Yao, B., Li, J. W., "On the adjacent-vertex-strongly-distinguishing total coloring of graphs". *Science in China A Mathematics*, 51(3), 2008, pp.427-436.
30. Appel, K., Haken, W., "The solution of the four-color-mapprobem". *Scientific American*, 10(8), 1977, pp.108-121.
31. Li, S. Z., He, S. F., "Hybrid optimization algorithm based on graph coloring problem". *Application Research of Computers*, (1), 2016, pp.90-100.
32. Liao, F. X., Ma, L., "Heuristic Search-based Ant Algorithm of Solving Graph Coloring Problem". *Computer Engineering*, 33(16), 2007, pp. 191-195.
33. Yu, P. J., "The Study of Graph Coloring Based on Simulated Annealing Genetic Algorithm". *GuiZhou University*, 2009.
34. Yu, Y., Su, S., Zhao, J., "A "DNA origami"-based approach to the solution of graph coloring problem". *Journal of Nanjing University (Natural Science)*, 52(4), 2016, pp. 656-660.
35. Li, J. W., Li, X. H., Dong, W., "Algorithm for vertex distinguishing total coloring of random graphs". *Application Research of Computers*, 32(6), 2015, pp. 1707-1710.
36. Cao, D. T., Li, J. W., Jiang, D. D., "Adjacent vertex-distinguishing equitable V-total coloring algorithm of graph based on multi-objective optimization". *Journal of Computer Application*, 37(2), 2017, pp. 457-462.
37. Cao, T. D., Li, J. W., Wen, F., "Algorithm for proving a conjecture of vertex distinguishing total coloring of graphs". *Acta Scientiarum Naturalium Universitatis Nankaiensis*, 50(1), 2017, p.103-111.
38. Li, M. C., Wen, F., Zhang, L., "Vertex-distinguishing total coloring of  $K_{2n+1}/E(K_{2,n})$ ". *Acta Scientiarum Naturalium Universitatis Nankaiensis*, 45(6), 2012, pp.59-65.
39. Xia, X. F., Zhang, L. C., Liu, X. Y., "Distributed accessibility index and query strategy for large-scale graph data". *Computer Engineering*, 44(3), 2018, pp.65-72.
40. Qiao, L., Chen, D. G., "Research on incremental algorithm for vertex coverage of large-scale graph". *Journal of Beijing Information Science & Technology University*, 35(05), 2020, pp.54-59.
41. Zhang, R. Q., Wei, W., Zhang, Y. X., "Large-scale graph maximum flow acceleration algorithm based on parallel graph computing framework". *Computer Simulation*, 38(2), 2021, pp.267-272.
42. Karthick, T., Maffray, F., Pastor, L., "Polynomial Cases for the Vertex Coloring Problem". *Algorithmica*, 81(3), 2019, pp.1053-1074.