Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2013, Article ID 176598, 6 pages
http://dx.doi.org/10.1155/2013/176598

Research Article

Classification Formula and Generation Algorithm of Cycle Decomposition Expression for Dihedral Groups

Dakun Zhang,[1] Yonggang Lin,[2] and Guozhi Song[1]

[1] *School of Computer Science and Software Engineering, Tianjin Polytechnic University, Tianjin 300160, China*
[2] *School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China*

Correspondence should be addressed to Guozhi Song; guozhi.song@gmail.com

Received 17 September 2012; Accepted 5 December 2012

Academic Editor: Zengqin Zhao

The necessary of classification research on common formula of D_n group (dihedral group) cycle decomposition expression is illustrated. It includes the reflection and rotation conversion, which derived six common formulae on cycle decomposition expressions of D_n group; it designed the generation algorithm on the cycle decomposition expressions of D_n group, which is based on the method of replacement conversion and the classification formula; algorithm analysis and the results of the process show that the generation algorithm which is based on the classification formula is outperformed by the general algorithm which is based on replacement conversion; it has great significance to solve the enumeration of the necklace combinational scheme, especially the structural problems of combinational scheme, by using group theory and computer.

1. Introduction

D_n group (Dihedral group) is a kind of important group which plays an important role in the research of the properties of group [1, 2]. It can be used to solve a variety of factual problems, such as the classical necklace problem, the enumeration problem of molecule structure [3], modelling of communication networks [4], construction of visual cryptography scheme [5], and analysis of satellite status in the orbit of LEO/MEO satellite network, which all adopt the D_n group [6–8] and use Burnside lemma and Pólya theorem to compute the number of combination necklace schemes, which all depend on the cycle decomposition expressions of D_n group. There are lots of research on the enumeration problem of necklace problem based on D_n group [9–11], but very few concentrates on the structural problems of combinational scheme. In the study of researching satellite status in the orbit of LEO/MEO satellite network and network route simulation, we should get each satellite status and take these as input, that is to say, the structural problems of combinational scheme, which is closely related to the D_n group cycle decomposition expression [12]. For the D_n group of low order, we can compute cycle decomposition expression of every group element manually, but for D_n group of high order, it is not only time consuming but also error by hand, so it is essential to adapt computer and the fast generation algorithm is the key to fulfill this problem. Fu and Wang [13] presented the common formula of D_n group permutation expression and then convert the permutation expression into cycle decomposition expressions of D_n group; however, it is inefficient. If we can get the cycle decomposition expressions of D_n group, it is much easier to solve the problem. There are two types of D_n group's elements: one is derived from reflection conversion, and the other from rotation conversion; the cycle decomposition expressions of these two kinds of elements adhere to different rules, so we can research cycle decomposition expressions of D_n group respectively, and then can get each common formula to fulfill fast generation algorithm on the cycle decomposition expressions of D_n group.

2. Necklace Problem and Permutation Expression of D_n Group

2.1. Necklace Problem and Permutation Expression of D_n Group. The necklace problem is defined as follows. Let us suppose that a necklace can be made from beads of m colors;

then how many different necklaces with n beads can be made?

When n and m are both small, we can work out all the different necklaces using exhaustive algorithm. But with the increasing size of n and m, it gets more and more difficult by exhaustive algorithm, so the group method must be used, and simpler and more efficient method has not been found up to the present [3].

Pólya Theorem. Let G be a group of permutations of the set of n objects; then the number $P(G; m, m, \ldots, m)$ of nonequivalent colorings is given by

$$P(G; m, m, \ldots, m) = \frac{1}{|G|}\left(\sum m^{\lambda_1(g)} + m^{\lambda_2(g)} + \cdots + m^{\lambda_n(g)}\right), \tag{1}$$

where $\lambda_k(g)$ is the number of k-cycle in the permutation g [11].

When we analyze the satellite status in the orbit by Pólya theorem, n satellites in the orbit are the coloring objects and m status is the m colors. The key for the problem is to solve the group of permutations of the set of n objects, that is, the D_n group.

2.2. The Permutation Expressions of D_n Group. Suppose that $X = \{0, 1, 2, \ldots, n-1\}$ (without loss of generality, we adapt $0 \sim n-1$ as sequence number) is the vertex set of the regular n ($n \geq 3$) quadrate and arranged counterclockwise, as shown in Figure 1.

As we rotate regular n quadrate according to $2\pi/n$ counterclockwise, vertex i has moved to the position originally occupied by vertex $i+1$ (mod n), so this rotation is the conversion on X, marked as R_1:

$$R_1 = \begin{pmatrix} 0 & 1 & 2 & \cdots & n-1 \\ 1 & 2 & 3 & \cdots & 0 \end{pmatrix}. \tag{2}$$

The conversion according to $2k(\pi/n)$ is marked as R_k:

$$R_k = \begin{pmatrix} 0 & 1 & 2 & \cdots & n-1 \\ k & k+1 & k+2 & \cdots & k+n-1 \end{pmatrix}, \tag{3}$$

where the addition and subtraction are the modulo n operation (as the same for the entire paper), R_0 is the identity, and R_k can be shown as

$$R_k(i) = k + i, \quad i = 0, 1, \ldots, n-1. \tag{4}$$

Another conversion is reflection in the symmetric axis according to π, named as reflectivity conversion. Because there is n symmetric axis, we mark the axis through vertex 0 as L_0, the axis through the vertex of the midpoint of edge $[0, 1]$ as $L_1, \ldots$, until L_{n-1}. The corresponding reflectivity conversion is marked as $M_0, M_1, \ldots, M_{n-1}$. For instance,

$$M_0 = \begin{pmatrix} 0 & 1 & \cdots & n-1 \\ 0 & n-1 & \cdots & 1 \end{pmatrix}. \tag{5}$$

We can prove that

$$M_k = k + n - i, \quad k = 0, 1, \ldots, n-1. \tag{6}$$

Let

$$D_n = \{R_k, M_k \mid k = 0, 1, \ldots, n-1\}. \tag{7}$$

Then D_n is closed under the composite operation of the conversion, the identity R_0 exists, and each element has inverse, so D_n forms group, which is dihedral group.

3. The Generation Algorithm on the Cycle Decomposition Expressions of D_n Group Based on Permutation and Complexity Analysis

3.1. The Cycle Decomposition Expressions of D_n Group. The representation of each permutation as a product of disjoint cycles and the decomposition is unique [3]; the product of disjoint cycles is named as cycle decomposition expression of elements of the group. We can devise an algorithm for converting the permutation expression into the cycle decomposition expression of D_n Group.

3.2. The Algorithm Design for Converting the Permutation Expression into the Cycle Decomposition Expression of D_n Group. Let $p_k[i, j]$ ($i = 0, 1$; $j = 0, 1, \ldots, n-1$) express the element (i column j row) in the permutation expression of each element of the D_n group and traverse all the elements of p_k starting from $p_k[0, 0]$; the algorithm is shown as follows.

Algorithm 1. Consider the following.

(1) Start from $p_k[0, 0]$, if $p_k[0, 0]! = p_k[0, 1]$; then go to (3).

(2) $p_k[0, 0]$ is the fixed point which forms an independent cycle; it is denoted by $(p_k[0, 0])$; go to (7).

(3) Search for the element equal to $p[1, 0]$ from $p_k[0, 1]$ to $p_k[0, n-1]$; suppose that the element is $p_k[0, j]$.

(4) Search for $p_k[1, j]$, if $p_k[1, j]! = p_k[0, 0]$; then go to (6).

(5) $p_k[0, 0]$ and $p_k[1, j]$ form a cycle; it is denoted by $p_k[1, j]$; go to (7).

(6) Search for the element equal to $p_k[1, j]$ at $p_k[0, i]$ which has not been written into the cycle; then judge the element whether it is equal to $p_k[0, 0]$ or not; proceed the next searching until you get the element equal to $p_k[0, 0]$ that form a cycle.

(7) During the next searching, delete all the $p_k[i, j]$ that have been written into the cycle from the original data structure; go to (1), until all the $p_k[i, j]$ have been written into the expression of the product of the cycles.

In the analysis of the algorithm, we know that this conversion method is of low efficiency from formula (14), so it cannot be used for problem of great size by group D_n; a fast generation algorithm on the cycle decomposition expressions of D_n group based on permutation must be designed.

3.3. The Time Complexity of Generation Algorithm Based on Permutation. Computational complexity is divided into two kinds: one is time complexity, and the other is space complexity. The analysis of space complexity is similar to that of time complexity, and the analysis of space complexity is more simple [12]; in this paper, the two algorithms' space complexities are the same on the whole, so we limit our study to the time complexity.

First apply formulae (4) and (6) to solve M_k, R_k ($k = 0, 1, \ldots, n-1$); we estimate the time complexity. Formula (4) is corresponding to the second row of M_k; formula (6) is corresponding to the second row of R_k; for each R_k or M_k, we need n additions (modulo n); thus we obtain the second row of the permutation, then express it as the form of formula (3), so we get the expression of permutation of M_k and R_k. There are $2n$ elements in the D_n group, so the time complexity function $T_1(n)$ of the algorithm is

$$T_1(n) = n * (2n) = 2n^2. \tag{8}$$

After obtaining the expression of permutation of all elements in the D_n group, we apply the conversion algorithm in Section 3.2 to every element in the group to get their cycle decomposition expression. The main operation is comparison in this conversion algorithm.

Begin with the first row and the first column $p_k[0, 0]$, comparing $p_k[0, 0]$ with $p_k[1, 0]$, searching the element which is same to $p_k[1, 0]$ in the first row if $p_k[0, 0]$ and $p_k[1, 0]$ are not equal. Comparing $p_k[1, j]$ ($j = 1, \ldots, n-1$) with $p_k[1, 0]$ one by one, at most $(n-1)$ comparisons are made; then comparing $p_k[1, j]$ with $p_k[0, 0]$, searching the element which is same to $p_k[1, j]$ in the first row if $p_k[1, j]$ and $p_k[0, 0]$ are not equal, at most $(n-2)$ comparisons are made, and so forth, the rest may be deduced by analogy and the time complexity function $T_2(n)$ of the algorithm is

$$T_2(n) = n!. \tag{9}$$

As there are $2n$ elements in the D_n group, the time complexity function of the generation algorithm based on permutation is

$$T_3(n) = n^2 + 2n * n! = 2n(n + n!). \tag{10}$$

We can observe from (10) that the complexity of the generation algorithm based on permutation is $Q_1(n * n!)$ with very low efficiency which is unable to fulfill the requirement in the solution of large size problems using D_n group. So a faster generation algorithm needs to be developed.

4. The Derivation of the Common Formula for the Cycle Decomposition Expressions of D_n Group

There are two types of D_n group's elements: one is derived from reflectivity conversion M_k ($k = 0, 1, \ldots, n-1$), and the other from rotation conversion R_k ($k = 0, 1, \ldots, n-1$); the cycle decomposition expressions of these two kinds of elements adhere to different rules, so we can research cycle decomposition expressions of D_n group, respectively; then, we can get each common formula to fulfill fast generation algorithm on the cycle decomposition expressions of D_n group.

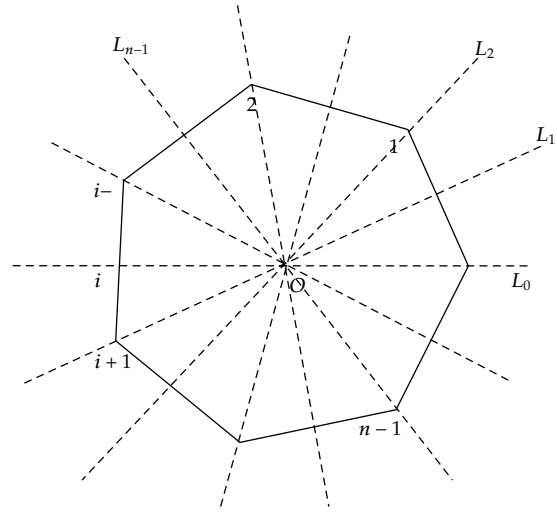

Figure 1: Regular n quadrate.

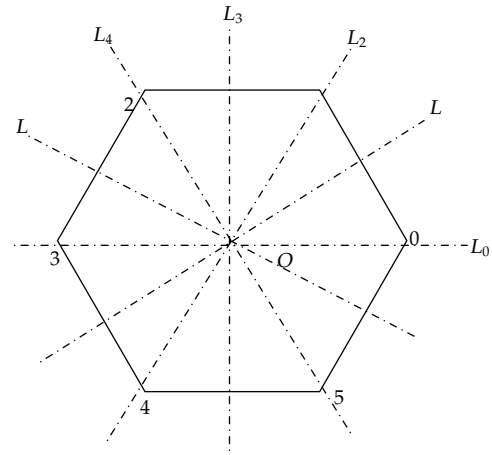

Figure 2: Regular 6 quadrate.

The D_n group is corresponding to a regular n quadrate (as shown in Figure 1). For D_n group of low order, we can get the cycle decomposition expressions of each element in the group by eyes, observe the vertex's constituting rule in every cycle with induction. Based on an exhausted series of the cycle decomposition expressions of D_n group, we bring forward the common formula and then prove it by mathematical induction.

(1) *The Cycle Decomposition Expressions for D_n Group of Low Order.* For instance, we enumerate all the cycle decomposition expressions of D_6 group. D_6 group is corresponding to the Regular 6 quadrate, as shown in Figure 2, where R_k ($k = 0, 1, 2, \ldots, 5$) are the cycle decomposition expressions of the reflectivity conversion, and M_k ($k = 0, 1, 2, \ldots, 5$) are the

cycle decomposition expressions of the rotation conversion. The main text paragraph is as follows (see Figure 2):

$$\begin{aligned}
R_0 &= (0)(1)(2)(3)(4)(5), & M_0 &= (0)(3)(1\ \ 5)(2\ \ 4),\\
R_1 &= (0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 5), & M_1 &= (0\ \ 1)(2\ \ 5)(3\ \ 4),\\
R_2 &= (0\ \ 2\ \ 4)(1\ \ 3\ \ 5), & M_2 &= (1)(4)(0\ \ 2)(3\ \ 5),\\
R_3 &= (0\ \ 3)(1\ \ 4)(2\ \ 5), & M_3 &= (1\ \ 2)(0\ \ 3)(4\ \ 5),\\
R_4 &= (0\ \ 4\ \ 2)(1\ \ 5\ \ 3), & M_4 &= (2)(5)(0\ \ 4)(1\ \ 3),\\
R_5 &= (0\ \ 5\ \ 4\ \ 3\ \ 2\ \ 1), & M_5 &= (2\ \ 3)(1\ \ 4)(0\ \ 5).
\end{aligned} \tag{11}$$

(2) *The Common Cycle Decomposition Expressions for D_n Group with Reflectivity Conversion*. The cycle decomposition expressions for M_k are not only related to the parity of n in the D_n group but also to the parity of k in the element M_k, so the formula can be divided into four instances.

(a) n is odd in the D_n group and k is also odd in M_k. While k equals 1, take reflectivity conversion that L^1 is the axis (as shown in Figure 1); 0 and 1 compose the transposition, that is, $(1, 0)$; 2 and $n-1$ compose the transposition, that is, $(2, n-1)$, and so forth.

There are $(n-1)/2$ transpositions and a fixed point $((1-1)/2 + (n-1)/2 + 1)$.

While k is general,

$$\begin{aligned}
&\frac{k-1}{2}+1 \text{ and } \frac{k-1}{2} \text{ compose the transposition,}\\
&\text{that is,} \left(\frac{k-1}{2}+1, \frac{k-1}{2}\right);\\
&\frac{k-1}{2}+2 \text{ and } \frac{k-1}{2}+1 \text{ compose the transposition,}\\
&\text{that is,} \left(\frac{k-1}{2}+2, \frac{k-1}{2}+1\right); \ldots.
\end{aligned} \tag{12}$$

There are $(n-1)/2$ transpositions and a fixed point $((1-1)/2 + (n-1)/2 + 1)$.

Now we obtain the following common formula:

$$\begin{aligned}
M_k = &\left(\frac{k}{2}\right)\left(\frac{k}{2}+1, \frac{k}{2}-1\right)\left(\frac{k}{2}+2, \frac{k}{2}-2\right)\cdots\\
&\left(\frac{k}{2}+\frac{n}{2}-1, \frac{k}{2}-\frac{n}{2}+1\right)\left(\frac{n}{2}+\frac{k}{2}\right).
\end{aligned} \tag{13}$$

We can prove that formula (8) is valid for $k = 1$, $k = 2j+1$ and $k = (2(j=1)+1) = 2j+3$ by mathematical induction. So formula (8) is true for all positive integers k. In the same way we can obtain the following three formulae.

(b) n is odd in the D_n group and k is even in M_k:

$$\begin{aligned}
M_k = &\left(\frac{k}{2}\right)\left(\frac{k}{2}+1, \frac{k}{2}-1\right)\left(\frac{k}{2}+2, \frac{k}{2}-2\right)\cdots\\
&\left(\frac{k}{2}+\frac{n-1}{2}, \frac{k}{2}-\frac{n-1}{2}\right).
\end{aligned} \tag{14}$$

There are $(n-1)/2$ transpositions.

(c) n is even in the D_n group and k is odd in M_k:

$$\begin{aligned}
M_k = &\left(\frac{k-1}{2}+1, \frac{k-1}{2}\right)\left(\frac{k-1}{2}+2, \frac{k-1}{2}-1\right)\cdots\\
&\left(\frac{k-1}{2}+\frac{n}{2}, \frac{k-1}{2}-\frac{n}{2}-1\right).
\end{aligned} \tag{15}$$

There are $n/2$ transpositions.

(d) n is even in the D_n group and k is even in M_k:

$$\begin{aligned}
M_k = &\left(\frac{k}{2}\right)\left(\frac{k}{2}+1, \frac{k}{2}-1\right)\left(\frac{k}{2}+2, \frac{k}{2}-2\right)\cdots\\
&\left(\frac{k}{2}+\frac{n}{2}-1, \frac{k}{2}-\frac{n}{2}+1\right)\left(\frac{k}{2}+\frac{n}{2}\right).
\end{aligned} \tag{16}$$

There are $(n-2)/2$ transpositions and two fixed points.

(3) *The Common Cycle Decomposition Expressions for the Element R_k in the Group with Rotation Conversion*. The type of R_k is $(n/d)^d$, $d = (k, n)$, and the type of R_k varies along with the value of k and n. There are two instances.

(a) *n is prime*. While k equals 0, we deal with it as follows:

$$R_0 = (0)(1)\cdots(n-1). \tag{17}$$

While $k = 1, 2, \ldots, n-1$, $d = (k, n) = 1$, the type of R_k is $(n/d)^d = n^1$; that is, every element R_k in the group makes up of a cycle, and there are n terms in the cycle, it is denoted by

$$R_k = (0\ \ k\ \ k+k\cdots k+k+\cdots+k). \tag{18}$$

(b) *n is composite*. When k equals 0, R_0 is the same to formula (14).

While $k = 1, 2, \ldots, n-1$, the type of R_k is $(n/d)^d$, $d = (k, n)$, R_k makes up of d cycles, and there are (n/d) terms in each cycle; we can obtain

$$\begin{aligned}
R_k = &(0, k, k+k, \ldots)(1, 1+k, 1+k+k, \ldots)\cdots\\
&((d-1), (d-1)+k, (d-1)+k+k, \ldots).
\end{aligned} \tag{19}$$

5. The Generation Algorithm on the Cycle Decomposition Expressions of D_n Group Based on the Classification Formulae and Complexity Analysis

5.1. The Idea of the Algorithm Designing. The generation algorithm on the cycle decomposition expressions of D_n group based on the classification Formulae is relatively simple. For M_k $(k = 0, 1, \ldots, n-1)$ with reflectivity conversion, first judge the parity of n and k; then substitute them into formulae (13)–(16), thus we can get the cycle decomposition expressions of n elements of the group. For R_k $(k = 0, 1, \ldots, n-1)$ with rotation conversion, substitute $k = 0$ into formula (17) and get R_0; while n is prime, substitute k into formula (18); while n is composite, substitute k into formula (19); thus, we can obtain all the cycle decomposition expressions of R_k $(k = 0, 1, \ldots, n-1)$.

5.2. Algorithm. Consider the following.

(1) Input n, $k = 0$.

(2) If n is even, then go to (6).

(3) If k is even, then go to (5).

(4) Substitute k into formula (8); we obtain M_k, $k = k+1$, $k < n$; go to (3); else go to (9).

(5) Substitute k into formula (9); we obtain M_k, $k = k+1$, $k < n$; go to (3); else go to (9).

(6) If k is even, go to (8).

(7) Substitute k into formula (10); we obtain M_k, $k = k+1$, $k < n$; go to (6); else go to (9).

(8) Substitute k into formula (13); we obtain M_k, $k = k+1$, $k < n$; go to (6); else go to (9).

(9) Substitute $k = 0$ into formula (14); we obtain R_0.

(10) If n is composite, go to (13).

(11) Substitute k into formula (15).

(12) $k = k + 1$, $k < n$; go to (11); else go to (15).

(13) Substitute k into formula (16).

(14) $k = k + 1$, $k < n$; go to (13).

(15) Stop.

5.3. The Time Complexity of the Generation Algorithm Based on Classification Formulae. The generation algorithm on the cycle decomposition expressions of D_n group based on the classification Formulae is relatively simple. Judge the parity of n and each k $(k = 1, 2, \ldots, n)$; substitute k into Formulae (8)~(13); thus get M_k the cycle expression. For R_k, the only thing is to judge the fraction of n. For every M_k or R_k, at most $2n$ additions are made. As there are $2n$ elements in the D_n group, so the time complexity function is

$$T_4 = 2n + n * (2n) = 2n^2 + 2n = 2\left(n^2 + n\right). \quad (20)$$

As a result, the time complexity of generation algorithm based on permutation is $O_1(n) = O(n * n!)$, while the time complexity of generation algorithm based on classification Formulae is $O_2(n) = O(n^2)$, so the time complexity of generation algorithm based on permutation is much more greater than that of generation algorithm based on classification Formulae. The results of the process show that the generation algorithm which is based on the classification formula is of superiority.

6. Conclusions

This paper includes the reflectivity and rotation conversion, which derived six common Formulae on cycle decomposition expressions of D_n group; it designed the generation algorithm on the cycle decomposition expressions of D_n group, which is based on the method of replacement conversion and the classification formula; algorithm analysis and the results of the process show that the generation algorithm which is based on the classification formula is outperformed by the general algorithm which is based on replacement conversion, it has great significance to solve the necklace problem and the combinational scheme of the satellite status in the orbit of LEO/MEO satellite network, especially the structural problems of combinational scheme, by using group theory and computer.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (NSFC) under Grant no. 61272006.

References

[1] G. Z. Hu, *The Application of Modern Algebra*, Tsinghua University Press, Beijing, China, 1992.

[2] K. Shinoda and M. Yamada, "A family of Hadamard matrices of dihedral group type," *Discrete Applied Mathematics*, vol. 102, no. 1-2, pp. 141–150, 2000.

[3] G. Renault, "Computation of the splitting field of a dihedral polynomial," in *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC '06)*, pp. 290–297, July 2006.

[4] S. Song and D. Wang, "Reliability analysis of the Cayley graphs of dihedral groups," *Tsinghua Science and Technology*, vol. 16, no. 1, pp. 36–40, 2011.

[5] Uno Miyuki and M. Kano, "Visual cryptography schemes with dihedral group access structure for many images," in *Proceedings of the 3rd International Conference on Information Security Practice and Experience (ISPEC '07)*, vol. 4464, pp. 344–359, 2007.

[6] H. Lange and S. Recillas, "Abelian varieties with group action," *Journal fur die Reine und Angewandte Mathematik*, no. 575, pp. 135–155, 2004.

[7] S. Bucikiewicz, L. Dębski, and W. Florek, "Application of algebraic combinatorics to finite spin systems with dihedral symmetry," *Acta Physica Polonica A*, vol. 100, no. 4, pp. 453–475, 2001.

[8] J. Ecker, "Affine completeness of generalised dihedral groups," *Canadian Mathematical Bulletin*, vol. 49, no. 3, pp. 347–357, 2006.

[9] K. H. Leung and B. Schmidt, "Asymptotic nonexistence of difference sets in dihedral groups," *Journal of Combinatorial Theory A*, vol. 99, no. 2, pp. 261–280, 2002.

[10] B. G. Xu, "On the formulas of enumeration in necklace problem," *Journal of East China Jiaotong University*, vol. 20, no. 5, pp. 113–114, 2003 (Chinese).

[11] Z. M. Wang, "The application of group index in combinatorial calculating," *Journal of Tangshang Teachers College*, vol. 23, no. 1, pp. 9–10, 2001 (Chinese).

[12] D. K. Zhang and G. X. Wang, "Construction of three-dimensional model of platonic solid coloring mode based on group theory," *Ruan Jian Xue Bao/Journal of Software*, vol. 15, no. 2, pp. 292–299, 2004 (Chinese).

[13] X. Q. Fu and X. D. Wang, *Algorithm and Datastruct*, Publishing House of Electronics Industry, Beijing, China, 2000.