International Journal of Hybrid Information Technology
Vol. 9, No.6 (2016), pp. 197-220
http://dx.doi.org/10.14257/ijhit.2016.9.6.18

Using the Eigenvalue Partition to Compute the Automorphism Group

HAO Jian-Qiang[1, 2, a], GONG Yun-Zhan[1, b] and LIU Hong-Zhi[2, c]

[1]*State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[2]*School of Computer Science and Information Engineering, Beijing Technology and Business University, 100048, China*
[a]*bshjq@vip.163.com*, [b]*gongyz@bupt.edu.cn*, and [c]*liuhz@btbu.edu.cn*

Abstract

To solve the automorphism group of a graph is a fundamental problem in graph theory. Moreover, it usually is an essential process for graph isomorphism testing. At present, because existing algorithms ordinarily cannot efficiently compute the automorphism group of a graph, ones cannot entirely resolve the graph isomorphism problem. To calculate the automorphism group of a weighted graph, first, briefly review the history of automorphism. Second, introduce the concept of eigenvalue partition. Third, using algebraic methods, examine not only the relationships between the diagonal form of an adjacency matrix and its eigenvalues and eigenvectors, but also the relationships between its eigenvalues and eigenvectors and the automorphism group. Furthermore, prove Theorem 2 to 8. In addition, propose Conjecture 1 and three open problems. By these theorems, present a novel method to resolve the automorphism group of a weighted graph. If a graph has no duplicate eigenvalues and Conjecture 1 is true, it can determine the automorphism group of a weighted graph in polynomial time by the method. Although this method has certain limitations and needs improvements, it is theoretically a necessary complement to solve the automorphism group. Finally, it shows the close relationships that exist between an orthogonal matrix and a permutation matrix, also an orthogonal matrix and an automorphism.

Keywords: *Automorphism group, Eigenvalue partition, Adjacency matrix, Eigenvalue, Eigenvector*

1. Introduction

1.1. The Automorphism Group of a Graph

To determine whether or not two directed or undirected weighted graphs are isomorphic is a fundamental problem in graph theory. Currently, the main problem encountered in the isomorphism study is: When testing isomorphism of graphs, ones sometimes come across some graphs whose structure is so terribly complicated that it contains a large number of automorphisms [1–6]. However, to compute all automorphisms is a difficult task that has exponential time complexity. Through long-term research, ones already know that computing all automorphisms also belongs to the NP class. Similar to graph isomorphism testing, until now, ones do not know whether there is a polynomial algorithm and whether it is a NP−complete problem [7] such that the efficiency of the algorithms is low. [8] provided the upper bound of the rank of the automorphism group for a simple graph. Nagoya and Toda[9] realized that the time complexities, finding the orbit of even if a node in an automorphism group and finding a non-trivial automorphism group, are exactly the same. Furthermore, Nagoya and Toda [10] examined the time complexity to compute an automorphism group. In

ISSN: 1738-9968 IJHIT

many situations, there may exist many complex automorphisms such that to determine whether two graphs are isomorphic would become difficult.

Although the existences of automorphisms bring about an impediment to research isomorphism problem, it also highlights the importance to research automorphism. Thus, it follows that in order fully to resolve the isomorphism problem, must first carefully examine the automorphism problem.

Automorphism essentially reflects the symmetry properties of a graph. In the early stage of the study, because people were not aware of the symmetrical structure that the graph may contain, or even if realized, also failed to attract enough attention, coupled with the automorphism and the isomorphism problem are the equally difficult problems, leading to the algorithms cannot efficiently solve the automorphism group[11, 12] and running time was long. Finally, the algorithms cannot deal with the graphs with no more than thousands of nodes, or even no more than one hundred.

Finding the automorphism and the symmetric structure of a graph have many practical applications. Turcat and Verdillon[13] used the automorphism to reduce the test cases that people use testing integrated circuit design. [14] used an automorphism algorithm to calculate the maximum symmetric set that a circuit implies. In order to calculate the fastest mixing Markov chain of a graph, [15] also exploited the potential symmetry of agraph. In addition to, ones can use spring algorithms to display symmetric properties of graphs. Eades and Lin[16] provided comprehensive theoretical evidence that many spring algorithms can show graph symmetry. [17] proposed a new algorithm to determine the symmetric structure that a large, sparse graph may contain. Recently, [18] developed a novel approach to canonical labeling where symmetries can be found first and then used to speed up the canonical labeling algorithms. Much open-source software, such as Nauty[1, 2, 4], Bliss[19, 20] and Saucy3[17, 18, 21, 22], this aspect of performance is particularly remarkable. As well, Nauty, bliss, and Saucy3 all explored more deeply the nature of the automorphism.

Perhaps Nauty is the most popular and practical tools for looking an automorphism group and canonical forms of a graph. It has almost become the industry standard used to calculate the canonical label, as well as the automorphism group. Much general-purpose mathematics software has adopted it. Nauty iteratively refines partitioning nodes until place the nodes that have the same properties into an automorphism orbit. Accompanied partition refinement becomes more and smaller. Finally, it will automatically create the canonical label. As a result, only to ascertain whether the canonical label of two graphs are the same, it can decide whether they are isomorphic. Nauty is more effective than Ullmann[11] algorithm. At the same time, Nauty also is one of the earliest software tools that can calculate the canonical label of a graph.

If encountering a sparse graph, the performances of Bliss and Saucy3 are better than Nauty. They also all can compute the canonical form of a graph. For graph automorphism, the current Saucy3 is optimal. In a few seconds, Saucy3 can find the symmetric structure as a graph with millions of nodes.

T.Miyazaki[23] analyzed the algorithm of Nauty how to calculate the canonical forms of a graph and estimates the time complexity of Nauty to calculate the automorphism group. T.Miyazaki found that when faced with a series Miyazaki graph, Nauty also requires the exponential time calculating. Tener and Deo[24] made improvements for solving the problem. To fix the glitch of Nauty, Traces[25] used the strategy of breadth-first search to find the automorphism group and the Canonical Labeling. When detecting an automorphism, Traces adopted the algorithm of Schreier-Sims[26], which integrated the software program developed by Leon[27].

Given a finite group X, many articles theoretically have discussed whether there is a graph G whose automorphism group $\mathrm{Aut}(G)$ is isomorphic to the X. Frucht theorem[28] is a famous theorem in algebraic graph theory, which indicates that given a finite group X, there must exist an undirected graph G whose automorphism group $\mathrm{Aut}(G)$ is isomorphic to the

X. Babai[29] extended Frucht theorem to the digraph and showed that given a finite group X, there must exist a digraph G whose automorphism group $\mathrm{Aut}(G)$ is isomorphous to the X. Finally, Babai[30] offered a supplementation of Frucht theorem. Aszl o Babai[31] carefully considered not only isomorphism, yet automorphism group, but also reconstruction of a graph.

Often there are many symmetric structures in a regular graph. Spielman[32] presented a way to test two strongly regular graph whether isomorphic each other. In order to find out the automorphism group of a graph, [33] discussed the relationship between the automorphism partitioning of all nodes of a graph and the automorphism group. Moreover, based on the automorphism partitioning of the nodes, they constructed an algorithm for computing the automorphism group. Recently, Presa[34] introduced the concept of orbit partition to calculate the automorphism group.

Many literatures, including Godsil[35], Siemons [36], and Bohanec and Perdih [37] also examined the automorphism group. A.Torgasevep[38] researched the automorphism group of an infinite graph. [39] provided a method for computing the rank of the automorphism group of a graph.

To calculate the automorphism group of a graph, experts in this field used a variety of methods. The problem of detecting the automorphism group of a graph G can be converted into the problem of computing the maximum clique (MC) of the association graph of the graph G. Jain and Wysotzki[40] used the neural network approach to solve automorphism partitioning. At the same time, they introduced the concept of the association group. Thus, they changed the problem, seeking the automorphism group, into finding the maximum clique (MC) of the association graph. Buch and Jijnger[41] started from the concept of the branches and the cut to study symmetry of a graph and provided a detailed description of an automorphism algorithm. Gilani and Faghani[42] considered the automorphism group of a weighted graph. Manjunath and Sharma[43] transformed one graph into a simplex, and by means of the simplex studied the isomorphism problem. As a result, they drew the conclusion that two graphs are isomorphic if and only if the two simplex congruent under isometric map. Finally, they represented an exponential time algorithm to compute the automorphisms of a graph.

Babai[44] also examined the close connection between an automorphism and the spectra of a graph. Teranishi[45] studied the algebraic relationship between the automorphism group and the adjacency matrix and derives the lower bound of the order of the automorphism group. Subsequently, Teranishi[46] also demonstrated the relationship between an automorphism and the eigenvalues. Ruecker G and Ruecker C[47] used the method of matrix-power to explore symmetry of a graph.

[48] proposed the concept of cellular algebra to study the isomorphism problem, and came to the conclusion that the time interval needed between determining two graphs whether isomorphism and finding the orbits of the automorphism group of a graph is polynomial.

[49] provided a function that can calculate the upper bound for the automorphism group of a graph of rank n and diameter d. Further they represented the corresponding graph that can capture the upper bound. [50] discussed the relationship between the automorphism group of a graph and the automorphism group of the vertex−deleted subgraphs, yet the edge−deleted subgraphs. The computation of the automorphism group exponential grows as the rank of a graph increases.

1.2. The Problems in the Automorphism Research

To solve an automorphism group of a graph that contains certain symmetric structure, many algorithms assign all nodes with certain common attributes into a same orbit. Finally, so divides all nodes of a graph into a number of different orbits of the automorphism group.

When partitioning the nodes into their respective orbits, a critical step usually is to examine the attributes of all nodes in the k neighborhood of a node. The k neighborhood of a node comprises all nodes, each of which to the distance of the node is $\leq k$. Another critical step usually is to compare the attributes of all nodes in the k neighborhood of two nodes. If all relevant attributes of the k neighborhoods of two nodes are identical, then ones put the two nodes into a same orbit. In view of this, when placing a node into an orbit, ones often have to take advantage of the local properties of the node.

[51] showed that if the multiplicities of eigenvalues of a given graph are bounded by a constant, then the generators of the automorphism group of the graph can be computed in polynomial time. Although they provided the corresponding proof for their claims, they did not detail how to implement their method. Their ideas are less consistent with the result showed by the examples 1 and 2. Based on current knowledge, it also is unknown whether there exists an algorithm that can use the eigenvalue partition to compute the automorphism group of a weighted graph.

In special circumstances, because the structure of a graph can be extraordinarily complicated, it cannot be completed that one only relies on the understanding of all local properties of a node to want to divide the node into which class. As a result, when ones partition nodes, it may not be enough only to compare relevant attributes of the k neighborhoods of two nodes. Ones had to expand the scope of the search from the k neighborhood to the $k+1$ neighborhood. Occasionally, ones need to compare relevant attributes of all nodes of the graph. Accordingly, ones need to understand global properties of the graph. Therefore, better understand global properties of a graph is crucial. In this regard, ones must make further efforts. In the future, how well we use both local and global properties. Furthermore, how effectively we achieve balance between both is the way that makes the automorphism research breakthrough. Taken together, currently the automorphism studies exist following problems:

1. The orbit partition of the automorphism group often fails.
2. Partition usually stays at the theoretical level. At present, such methods have not reached a satisfactory level. Accordingly, ones still lack an effective method.
3. Until now, how to take full advantage of the eigenvalues and eigenvectors of a graph studying the automorphism group has not yet attracted the attention of many experts.
4. The complexity of solving the automorphism group of a graph is equal to the complexity of solving the isomorphic of the graphs.
5. On special occasions, to partition all nodes of a graph into their respective orbits of the automorphism group, ones need to understand and grasp both the local and the global property of the graph. Furthermore, when partitioning, ones need to take advantage of the both flexibly. Ones are not doing enough in this regard.

1.3. The Contribution of the Article

The contributions of this paper are that

1. Survey the brief history of the automorphism and point out the current problems.
2. Introduce the concept of the eigenvalue partition of a graph and employ it to calculate the automorphism group.
3. For the first time, we derive Equations 4 and 8. Furthermore, taking advantage of Equations 4, we propose a novel way in which we make use of eigenvalues and eigenvectors to calculate the automorphism group.
4. Prove Theorem 2 to 8.
5. Provide two examples 1 and 2 to show how to apply our method in practice.
6. Present Conjecture 1.
7. Propose three open problems.

The rest of this article is organized as follows:

In the next section 2, we establish some basic terminology and notation. In the section 3, we describe the basic principles. In the section 4, we offer twos examples for using eigenvalue partition to determine the automorphism Group. In the section 5, we analyze the time complexity of the method. In the section 6, we propose three open problems. Finally, in the section 7, we show the summary and conclusion.

2. Terminology and Notation

This paper focus on developing a technique that can solve the automorphism of undirected weighted graphs with neither loops nor multiple edges. An undirected weighted graph consists of a set of vertices, a set of edges, and a set of weight values. For a weighted graph $G=(V(G), E(G), W(G))$, let $V(G)$, $E(G)$, and $W(G)$ denote the set of vertices of G, the set of edges of G, and the set of weight values of edges, respectively. An edge $(u, v) \in E(G)$ connects two vertices $u \in G$ and $v \in G$. Each $w(u, v) \in W(G)$ is the weight values of the edge (u, v). In the following text, when simultaneously involving two graphs, we always assume that their degree sequences are the same except specified.

Definition 1. Let $G=(V(G), E(G), W(G))$ and $H=(V(H), E(H), W(H))$ be two undirected weighted graphs with n nodes. If there exists a bijection function $f: V(G) \rightarrow V(H)$ such that $\forall(u, v) \in E(G)$ and $w(u, v) \in W(G)$ if and only if $(f(u), f(v)) \in E(H)$ and $w(f(u), f(v)) \in W(H)$. Thus, we say f is an isomorphic map of $G \rightarrow H$. Furthermore, we say the graph G and H to be isomorphic, denoted by $G \cong H$. An isomorphic map f of G onto itself is said to be an automorphism of G.

Definition 2. A $n \times n$ permutation matrix is a square matrix whose every row and column contains precisely a single 1 with 0s everywhere else.

A permutation $\gamma \in \mathrm{Sym}(V)$ is a $V(G) \rightarrow V(H)$ map, which is clearly a bijection. $\forall u \in V(G)$, we describe the image of u under the γ with $\gamma(u)$. We describe the u's image of the composite under the $\gamma_1, \gamma_2 \in \mathrm{Sym}(V)$ with $\gamma_1 \circ \gamma_2(u) = \gamma_2(\gamma_1(u))$.

The set of all automorphisms of G is denoted by $\mathrm{Aut}(G)$. It is verified that $\mathrm{Aut}(G)$ forms a group under functional composition, which is called the automorphism group[52] of G. Obviously, $\mathrm{Aut}(G)$ is a subgroup within the symmetry group $\mathrm{Sym}(V)$, $\mathrm{Aut}(G) = \{\gamma | \gamma \in \mathrm{Sym}(V) \wedge \gamma(G) = G\}$.

Theorem 1. Let $G=(V(G), E(G), W(G))$ and $H=(V(H), E(H), W(H))$ be two undirected weighted graphs with n nodes. Moreover, have $|\mathrm{E}(G)| = |\mathrm{E}(H)|$. $\mathbf{M}(G)$ and $\mathbf{M}(H)$ are their adjacency matrices, respectively. Thus, G and H are isomorphic if and only if there exists a permutation matrix P such that $\mathrm{P}\mathbf{M}(G)\mathrm{P}^{\mathrm{T}} = \mathbf{M}(\mathrm{H})$ [53].

Corollary 1. Let $G=(V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathbf{M}(G)$. Thus, G exists an automorphism map if and only if there exists a permutation matrix P such that $\mathrm{P}\mathbf{M}(G)\mathrm{P}^{\mathrm{T}} = \mathbf{M}(G)$.

If a graph G has an automorphism map γ, by Corollary 1 there exists a permutation matrix P to satisfy Equation $\mathrm{P}\mathbf{M}(G)\mathrm{P}^{\mathrm{T}} = \mathbf{M}(G)$. Conversely, if a graph G has a permutation matrix P to satisfy the equation $\mathrm{P}\mathbf{M}(G)\mathrm{P}^{\mathrm{T}} = \mathbf{M}(G)$, one can construct an automorphism map γ corresponding to the permutation matrix P. Clearly, there exists a one-to-one correspondence between an automorphism map γ and a permutation matrix P of a graph. Therefore, we do not strictly distinguish both in this article.

Definition 3. Let $G=(V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathbf{M}(G)$. Suppose that $\mathbf{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots \eta_k$. All the n eigenvalues of $\mathbf{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$

with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities[54] of the eigenvalues are $\alpha_A(\eta_1)$, $\alpha_A(\eta_2)$, $\cdots$, $\alpha_A(\eta_k)$, respectively. We classify all the distinct eigenvalues into three categories as follows:

1. If $\eta_i = 0$, we partition it into the class 0 and call each of the class eigenvalues an eigenvalue of 0 type.

2. If $|\eta_j| = |\eta_i|$ for $i \neq j, i, j = 1, 2, \cdots, n$, we let η_j and η_i belong to the class 1 simultaneously. Accordingly, we call each of the class eigenvalues an eigenvalue of 1 type.

3. We let the remaining eigenvalues belong to the class 2 and call each of the class eigenvalues an eigenvalue of 2 type.

Assuming that $\pi = \{\pi_1, \pi_2, \cdots, \pi_n\}$ is a partition of all eigenvalues of the graph G that satisfies the above conditions, then we call the π an eigenvalues partition of the G, denoted by $\pi(G_e)$.

3. Basic Principle

Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathrm{M}(G)$. Because the matrix $\mathrm{M}(G)$ is symmetric, it has n real eigenvalues. We work out the n real eigenvalues and use $\lambda_1, \lambda_2, \cdots, \lambda_n$ to represent them respectively. Furthermore, we compute the eigenvector X_i corresponding to the eigenvalue λ_i and use $\mathrm{X}_1, \mathrm{X}_2, \cdots, \mathrm{X}_n$ to represent them respectively.

Theorem 2. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathrm{M}(G)$. All the n eigenvalues of $\mathrm{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ respectively. $\mathrm{X}_1, \mathrm{X}_2, \cdots, \mathrm{X}_n$ are eigenvectors corresponding to $\lambda_1, \lambda_2, \cdots, \lambda_n$ respectively.

If the graph G has an automorphism mapping γ and P is the permutation matrix corresponding to the γ, then there must exist an orthogonal matrix $\mathrm{Q} = \mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X}$ to satisfy Equations $\mathrm{Q}\Lambda - \Lambda\mathrm{Q} = 0$, where $\mathrm{X} = (\mathrm{X}_1, \mathrm{X}_2, \bullet\bullet\bullet, \mathrm{X}_n)$, and

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}$$

PROOF. From linear algebra, we know that $\mathrm{X}_1, \mathrm{X}_2, \cdots, \mathrm{X}_n$ are mutually orthogonal vectors satisfying the equations $X_i X_j^T = 0$ for $i \neq j, i, j = 1, 2, \cdots, n$ and $X_i X_j^T = 1$ for $i = 1, 2, \cdots, n$. Furthermore, we can decompose the $\mathrm{M}(G)$ as follows:

$$\mathrm{M}(G) = \mathrm{X}\Lambda\mathrm{X}^{\mathrm{T}}, \tag{1}$$

where the matrix $\mathrm{X} = (\mathrm{X}_1, \mathrm{X}_2, \cdots, \mathrm{X}_n)$ is the orthogonal matrix satisfying the equation $\mathrm{X}\mathrm{X}^{\mathrm{T}} = \mathrm{X}^{\mathrm{T}}\mathrm{X} = \mathrm{I}$ (unit matrix). By Corollary 1, it follows that G exists an automorphism mapping if and only if G exists a permutation matrix P to satisfy $\mathrm{P}\mathrm{M}(G)\mathrm{P}^{\mathrm{T}} = \mathrm{M}(G)$. By substituting (1) into $\mathrm{P}\mathrm{M}(G)\mathrm{P}^{\mathrm{T}} = \mathrm{M}(G)$ we obtain

$$\begin{aligned} \mathrm{P}\mathrm{M}(G)\mathrm{P}^{\mathrm{T}} &= \mathrm{X}\Lambda\mathrm{X}^{\mathrm{T}}. \\ \mathrm{P}\mathrm{X}\Lambda\mathrm{X}^{\mathrm{T}}\mathrm{P}^{\mathrm{T}} &= \mathrm{X}\Lambda\mathrm{X}^{\mathrm{T}}. \\ \mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X}\Lambda &= \Lambda\mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X}. \\ (\mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X})\Lambda &= \Lambda(\mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X}). \end{aligned}$$

Let $\mathbf{Q} = X^{T}PX$. Because $\mathbf{Q}\mathbf{Q}^{T} = X^{T}PX(X^{T}PX)^{T} = X^{T}PXX^{T}P^{T}X = I$, the matrix $\mathbf{Q}$ also is an orthogonal matrix. Therefore, $\mathbf{Q}\Lambda = \Lambda\mathbf{Q}$. $\mathbf{Q}\Lambda - \Lambda\mathbf{Q} = 0$. □

Theorem 3. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $M(G)$. All the n eigenvalues of $M(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ respectively. $X_1, X_2, \cdots, X_n$ are eigenvectors corresponding to $\lambda_1, \lambda_2, \cdots, \lambda_n$ respectively.

If the graph G has an automorphism mapping γ and P is the permutation matrix corresponding to the γ, then there must exist an orthogonal matrix $\mathbf{Q} = X^{T}PX$ such that $\mathbf{Q}\Lambda^{t} - \Lambda^{t}\mathbf{Q} = 0$ where $X = (X_1, X_2, \cdots, X_n)$ for all $t \in N$ (set of natural numbers), and

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix}$$

PROOF. This proof is quite similar to the proof of Theorem 2. We can decompose the $M(G)$ as follows:

$$M(G) = X\Lambda X^{T}, \tag{2}$$

where the matrix $X = (X_1, X_2, \cdots, X_n)$ is the orthogonal matrix satisfying the equation $XX^{T} = X^{T}X = I$ (unit matrix). By Corollary 1, we know that G exists an automorphism mapping if and only if G exists a permutation matrix P to satisfy $PM(G)P^{T} = M(G)$. By substituting (2) into $PM(G)P^{T} = M(G)$, we obtain

$$(PM(G)P^{T})^{t} = M(G)^{t}.$$

$$\overbrace{PM(G)P^{T} \bullet \cdots \bullet PM(G)P^{T}}^{t} = M(G).$$

$$PM(G)^{t}P^{T} = M(G)^{t}.$$

$$P(X\Lambda X^{T})^{t}P^{T} = (X\Lambda X^{T})^{t}.$$

$$P(\overbrace{X\Lambda X^{T} \bullet \cdots \bullet X\Lambda X^{T}}^{t})P^{T} = \overbrace{X\Lambda X^{T} \bullet \cdots \bullet X\Lambda X^{T}}^{t}.$$

$$PX\Lambda^{t}X^{T}P^{T} = X\Lambda^{t}X^{T}.$$

$$X^{T}PX\Lambda^{t} = \Lambda^{t}X^{T}PX.$$

Let $\mathbf{Q} = X^{T}PX$. Because $\mathbf{Q}\mathbf{Q}^{T} = X^{T}PX(X^{T}PX)^{T} = X^{T}PXX^{T}P^{T}X = I$, the matrix $\mathbf{Q}$ also is an orthogonal matrix. Therefore, $\mathbf{Q}\Lambda^{t} = \Lambda^{t}\mathbf{Q}$, $\mathbf{Q}\Lambda^{t} - \Lambda^{t}\mathbf{Q} = 0$. □

By Theorem 2, Let

$$\mathbf{Q} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,n} \end{bmatrix} = (\mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_n),$$

where $\mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_n$, are column vectors with n elements, respectively.

$$\begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,n} \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix} - \begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n \end{bmatrix} \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,n} \end{bmatrix} = 0. \tag{3}$$

Expanding (3), we obtain

$$\begin{bmatrix} \lambda_1 x_{1,1}\, \lambda_2 x_{1,2} \cdots \lambda_n x_{1,n} \\ \lambda_1 x_{2,1}\, \lambda_2 x_{2,2} \cdots \lambda_n x_{2,n} \\ \vdots \quad \cdots \quad \cdots \quad \vdots \\ \lambda_1 x_{n,1}\, \lambda_2 x_{n,2} \cdots \lambda_n x_{n,n} \end{bmatrix} - \begin{bmatrix} \lambda_1 x_{1,1}\, \lambda_1 x_{1,2} \cdots \lambda_1 x_{1,n} \\ \lambda_2 x_{2,1}\, \lambda_2 x_{2,2} \cdots \lambda_2 x_{2,n} \\ \vdots \quad \cdots \quad \cdots \quad \vdots \\ \lambda_n x_{n,1}\, \lambda_n x_{n,2} \cdots \lambda_n x_{n,n} \end{bmatrix} = 0.$$

$$\begin{bmatrix} (\lambda_1 - \lambda_1) x_{1,1}\, (\lambda_2 - \lambda_1) x_{1,2} \cdots (\lambda_n - \lambda_1) x_{1,n} \\ (\lambda_1 - \lambda_2) x_{2,1}\, (\lambda_2 - \lambda_2) x_{2,2} \cdots (\lambda_n - \lambda_2) x_{2,n} \\ \vdots \quad \cdots \quad \cdots \quad \vdots \\ (\lambda_1 - \lambda_n) x_{n,1}\, (\lambda_2 - \lambda_n) x_{n,2} \cdots (\lambda_n - \lambda_n) x_{n,n} \end{bmatrix} = 0. \tag{4}$$

Where $(\lambda_i - \lambda_i)x_{i,i} = 0$ for $i = 1,2,\cdots,n$. Change the appearance of (4) and express it in the following form:

$$\mathrm{AY} = \mathbf{0}, \tag{5}$$

where

$$\mathrm{A} = \begin{bmatrix} \lambda_1-\lambda_1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & 0 & \lambda_n-\lambda_1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & 0 & \lambda_1-\lambda_2 & 0 & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & 0 & \ddots & 0 & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & 0 & \lambda_n-\lambda_2 & 0 & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & 0 & \ddots & 0 & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \lambda_1-\lambda_n & 0 & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \lambda_n-\lambda_n \end{bmatrix},$$

$$\mathrm{Y} = \left[x_{1,1}\, x_{1,2} \cdots x_{1,n}\, x_{2,1}\, x_{2,2} \cdots x_{2,n} \cdots x_{n,1}\, x_{n,2} \cdots x_{n,n}\right]^{\mathrm{T}}.$$

By analyzing (5), we can conclude that

1. When $i \neq j,\ i, j = 1,2,\cdots,n$,

(a) if $(\lambda_j - \lambda_i) \neq 0$, to make the $\mathrm{AY} = \mathbf{0}$, then $x_{i,j} = 0$.

(b) Otherwise, if $(\lambda_j - \lambda_i) = 0$, to make the $\mathrm{AY} = \mathbf{0}$, then $x_{i,j}$ can take any value. Hence we let $x_{i,j} = 1$.

2. When $i = j,\ i, j = 1, 2,\cdots, n$, if $(\lambda_j - \lambda_i) = 0$, to make the AY = 0, then $x_{i,j}$ can take any value. Hence we let $x_{i,j} = 1$.

Theorem 4. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathrm{M}(G)$. Suppose that $\mathrm{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathrm{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1), \alpha_A(\eta_2), \cdots, \alpha_A(\eta_k)$, respectively. Thus, we have

1. If $\lambda_j - \lambda_i \neq 0$ for $i, j \in \{1, 2, \cdots, n\}$, in Formula 3, then the elements of $\mathbf{Q}$ must satisfy $x_{i,j} = 0$.

2. In Formula 3, $\mathbf{Q}$ has at least $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and at most $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries.

PROOF. 1. For all $i, j \in \{1, 2, \cdots, n\}$, if $\lambda_j - \lambda_i \neq 0$, from (4) it follows that $(\lambda_j - \lambda_i)\, x_{i,j} = 0$ if and only if $x_{i,j} = 0$.

2. From (5), it can be seen that in the matrix *A*, when $i, j = 1,2, \cdots, n$, then $\lambda_j - \lambda_i = 0$ if and only if λ_i and λ_j belong to the same an eigenvalue η_s whose algebraic multiplicity is $\alpha_A(\eta_s)$ with $s = 1,2, \cdots, k$. Therefore, in the diagonal of the matrix *A*, the total number of entries which satisfy condition $\lambda_j - \lambda_i = 0$ equals $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$.

In the diagonal of the matrix *A*, because there is a total of n^2 entries, there is a total of $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and a total of $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries. Therefore, solving (5) exactly, we can conclude that $\mathbf{Q}$ has at least $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and at most $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries. □ Furthermore, by Theorem 3, the equation $\mathbf{Q}\Lambda^{t} - \Lambda^{t}\mathbf{Q} = 0$ holds.

$$\begin{bmatrix} x_{1,1} \cdots x_{1,n} \\ \vdots \ \ddots \ \vdots \\ x_{n,1} \cdots x_{n,n} \end{bmatrix} \begin{bmatrix} \lambda_1^t & 0 & 0 & 0 \\ 0 & \lambda_2^t & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n^t \end{bmatrix} - \begin{bmatrix} \lambda_1^t & 0 & 0 & 0 \\ 0 & \lambda_2^t & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_n^t \end{bmatrix} \begin{bmatrix} x_{1,1} \cdots x_{1,n} \\ \vdots \ \ddots \ \vdots \\ x_{n,1} \cdots x_{n,n} \end{bmatrix} = 0. \quad (6)$$

Expanding (6), we obtain

$$\begin{bmatrix} \lambda_1^t x_{1,1} & \lambda_2^t x_{1,2} \cdots \lambda_n^t x_{1,n} \\ \lambda_1^t x_{2,1} & \lambda_2^t x_{2,2} \cdots \lambda_n^t x_{2,n} \\ \vdots & \cdots \ \cdots \ \vdots \\ \lambda_1^t x_{\mathrm{n},1} & \lambda_2^t x_{\mathrm{n},2} \cdots \lambda_n^t x_{\mathrm{n,n}} \end{bmatrix} - \begin{bmatrix} \lambda_1^t x_{1,1} & \lambda_1^t x_{1,2} \cdots \lambda_1^t x_{1,n} \\ \lambda_2^t x_{2,1} & \lambda_2^t x_{2,2} \cdots \lambda_2^t x_{2,n} \\ \vdots & \cdots \ \cdots \ \vdots \\ \lambda_n^t x_{\mathrm{n},1} & \lambda_n^t x_{\mathrm{n},2} \cdots \lambda_n^t x_{\mathrm{n,n}} \end{bmatrix} = 0. \quad (7)$$

By simplifying and merging (7), we have

$$\begin{bmatrix} (\lambda_1^t - \lambda_1^t)x_{1,1} & (\lambda_2^t - \lambda_1^t)x_{1,2} \cdots (\lambda_n^t - \lambda_1^t)x_{1,n} \\ (\lambda_1^t - \lambda_2^t)x_{2,1} & (\lambda_2^t - \lambda_2^t)x_{2,2} \cdots (\lambda_n^t - \lambda_2^t)x_{2,n} \\ \vdots & \cdots \quad \cdots \quad \vdots \\ (\lambda_1^t - \lambda_n^t)x_{\mathrm{n},1} & (\lambda_2^t - \lambda_n^t)x_{\mathrm{n},2} \cdots (\lambda_n^t - \lambda_n^t)x_{\mathrm{n,n}} \end{bmatrix} = 0. \quad (8)$$

Where $(\lambda_i^t - \lambda_i^t)x_{i,i} = 0$ for $i = 1,2, \cdots, n$. Change the appearance of (8) and express it in the following form:

$$\mathbf{A}\mathbf{Y} = \mathbf{0}, \quad (9)$$

where

$$
\mathtt{A} = \begin{bmatrix}
\lambda_1^t - \lambda_1^t & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\
0 & \ddots & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\
\vdots & 0 & \lambda_n^t - \lambda_1^t & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\
\vdots & \cdots & 0 & \lambda_1^t - \lambda_2^t & 0 & \cdots & \cdots & \cdots & \cdots & \vdots \\
\vdots & \cdots & \cdots & 0 & \ddots & 0 & \cdots & \cdots & \cdots & \vdots \\
\vdots & \cdots & \cdots & \cdots & 0 & \lambda_n^t - \lambda_2^t & 0 & \cdots & \cdots & \vdots \\
\vdots & \cdots & \cdots & \cdots & \cdots & 0 & \ddots & 0 & \cdots & \vdots \\
\vdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \lambda_1^t - \lambda_n^t & 0 & \vdots \\
\vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \ddots & 0 \\
0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \lambda_n^t - \lambda_n^t
\end{bmatrix},
$$

$$
\mathbf{Y} = \left[X_{1,1}\, X_{1,2} \cdots X_{1,n}\, X_{2,1}\, X_{2,2} \cdots X_{2,n} \cdots X_{n,1}\, X_{n,2} \cdots X_{n,n}\right]^T.
$$

By analyzing (9), we can conclude that

1. When $i \neq j,\ i, j = 1,\ 2, \cdots, n$,

(a) if $(\lambda_j^t - \lambda_i^t) \neq 0$, to make the $\mathbf{AY} = \mathbf{0}$, then $x_{i,j} = 0$..

(b) Otherwise, if $(\lambda_j^t - \lambda_i^t) = 0$, to make the $\mathbf{AY} = \mathbf{0}$, then $x_{i,j}$ can take any value. Hence we let $x_{i,j} = 1$.

2. When $i = j,\ i, j = 1,\ 2, \cdots, n$, if $(\lambda_j^t - \lambda_i^t) = 0$, to make the $\mathbf{AY} = \mathbf{0}$, then $x_{i,j}$ can take any value. Hence we let $x_{i,j} = 1$.

Theorem 5. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathbf{M}(G)$. Suppose that $\mathbf{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathbf{M}(G)$ are $\lambda_1, \lambda_2, \cdots \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1,\ 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1), \alpha_A(\eta_2), \cdots,\ \alpha_A(\eta_k)$, respectively. Thus, we have

1. If $\lambda_j^t - \lambda_i^t \neq 0 \text{ for } i, j \in \{1,\ 2, \cdots, n\}$, in Formula 3, then the elements of $\mathbf{Q}$ must satisfy $x_{i,j} = 0$.

2. In Formula 3, $\mathbf{Q}$ has at least $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and at most $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries.

PROOF. 1. If $\lambda_j^t - \lambda_i^t \neq 0 \text{ for } i, j \in \{1,\ 2, \cdots, n\}$, from (8) it follows that $(\lambda_j^t - \lambda_i^t)\ x_{i,j} = 0$ if and only if $x_{i,j} = 0$ with $i, j \in \{1,2, \cdots, n\}$.

2. From (9), it can be seen that in the matrix $\mathbf{A}$, when $i, j = 1,\ 2, \cdots, n$, then $\lambda_j^t - \lambda_i^t = 0$ if and only if λ_i and λ_j belong to the same an eigenvalue η_s whose algebraic multiplicity is $\alpha_A(\eta_s)$ with $s = 1,\ 2, \cdots, k$. Therefore, in the diagonal of the matrix *A*, the total number of entries which satisfy condition $\lambda_j^t - \lambda_i^t = 0$ equal to $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$.

In the diagonal of the matrix *A*, because there is a total of n^2 entries, there is a total of $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and a total of $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries.

Therefore, solving (9) exactly, we can conclude that $\mathbf{Q}$ has at least $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ zero entries and at most $\sum_{s=1}^{k} \alpha_A(\eta_s)^2$ non-zero entries. □

In the following, we will examine that when $t \to +\infty$, how (8) change accordingly. Meanwhile, we also will examine how the entries of the matrix $\mathbf{Q}$ change accordingly in the limit process. To achieve this purpose, from (8), we take the limit $t \to +\infty$. It follows that

$$\begin{bmatrix} \lim_{t\to+\infty}(\lambda_1^t - \lambda_1^t)x_{1,1} & \lim_{t\to+\infty}(\lambda_2^t - \lambda_1^t)x_{1,2} \cdots \lim_{t\to+\infty}(\lambda_n^t - \lambda_1^t)x_{1,n} \\ \lim_{t\to+\infty}(\lambda_1^t - \lambda_2^t)x_{2,1} & \lim_{t\to+\infty}(\lambda_2^t - \lambda_2^t)x_{2,2} \cdots \lim_{t\to+\infty}(\lambda_n^t - \lambda_2^t)x_{2,n} \\ \vdots & \cdots \quad \cdots \quad \vdots \\ \lim_{t\to+\infty}(\lambda_1^t - \lambda_n^t)x_{n,1} & \lim_{t\to+\infty}(\lambda_2^t - \lambda_n^t)x_{n,2} \cdots \lim_{t\to+\infty}(\lambda_n^t - \lambda_n^t)x_{n,n} \end{bmatrix} = 0. \quad (10)$$

From (10), it can be seen that when $i \neq j$ and $i, j = 1, 2, \cdots, n$, $\lim_{t\to+\infty}(\lambda_j^t - \lambda_i^t)x_{1,1} = 0$. By carefully analyzing them, we can conclude that

1. If $|\lambda_j| < |\lambda_i| < 1$, then $\lim_{t\to+\infty}(\lambda_j^t - \lambda_i^t) = 0$. Thus $x_{i,j}$ can take any value.

2. If $|\lambda_j| \geq 1 > |\lambda_i| \vee |\lambda_i| \geq 1 > |\lambda_j|$, then $\lim_{t\to+\infty}(\lambda_j^t - \lambda_i^t) = \infty$. Thus $x_{i,j} = 0$.

3. If $1 \leq |\lambda_i| < |\lambda_j|$, then

.
$$\lim_{t\to+\infty}(\lambda_j^t - \lambda_i^t) = \lim_{t\to+\infty} \lambda_j^t(1 - \frac{\lambda_i^t}{\lambda_j^t}) = \lim_{t\to+\infty} \lambda_j^t \bullet \lim_{t\to+\infty}(1 - \frac{\lambda_i^t}{\lambda_j^t}) = \infty.$$

Thus $x_{i,j} = 0$.

4. If $1 \leq |\lambda_j| < |\lambda_i|$, then

$$\lim_{t\to+\infty}(\lambda_j^t - \lambda_i^t) = \lim_{t\to+\infty} \lambda_i^t(\frac{\lambda_j^t}{\lambda_i^t} - 1) = \lim_{t\to+\infty} \lambda_i^t \bullet \lim_{t\to+\infty}(\frac{\lambda_j^t}{\lambda_i^t} - 1) = \infty.$$

Thus $x_{i,j} = 0$.

From the above discussion, we have the following Theorem 6.

Theorem 6. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathbf{M}(G)$. Suppose that $\mathbf{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathbf{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1)$, $\alpha_A(\eta_2)$, $\cdots$, $\alpha_A(\eta_k)$, respectively. If $i \neq j$ and $i, j = 1, 2, \cdots, n$, for the entries of the matrix $\mathbf{Q}$ that satisfy Equations 8, it follows that

1. If $|\lambda_j| < |\lambda_i| < 1$, then $x_{i,j}$ can take any value.

2. If $|\lambda_j| \geq 1 > |\lambda_i| \vee |\lambda_i| \geq 1 > |\lambda_j|$, then $x_{i,j} = 0$.

3. If $1 \leq |\lambda_i| < |\lambda_j|$, then $x_{i,j} = 0$.

4. If $1 \leq |\lambda_j| < |\lambda_i|$, then $x_{i,j} = 0$.

By Theorem 2 and 3, it follows that $\mathrm{Q} = \mathrm{X}^{\mathrm{T}}\mathrm{PX}$ and $\mathrm{P} = \mathrm{XQX}^{\mathrm{T}}$. If one has acquired the matrices Q and X, which all eigenvectors of the G form, by means them, one can compute a permutation matrix P and an automorphism accordingly.

Below we will discuss what graph, as well as the algebraic multiplicities of which graph satisfy what conditions, can make $n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2$ reach the minimum or the maximum values. Let $y_s = \alpha_A(\eta_s)$ for $s = 1, 2, \cdots, k$ and let

$$f(y_1, y_2, \cdots, y_n) = n^2 - \sum_{s=1}^{k} \alpha_A(\eta_s)^2 = n^2 - \sum_{s=1}^{k} y_s^2, \tag{11}$$

$$\varphi(y_1, y_2, \cdots, y_n) = \sum_{s=1}^{k} \alpha_A(\eta_s) - n = \sum_{s=1}^{k} y_s - n,$$

For a known graph, the k most be a constant. The function $f(y_1, y_2, \cdots, y_k)$ is a multivariate function with variables $y_1, y_2, \cdots, y_k$. As we all know that a graph must satisfy the requirement

$$\varphi(y_1, y_2, \cdots, y_k) = \sum_{s=1}^{k} y_s - n,$$

Furthermore, we can construct the function:

$$\begin{aligned} F(y_1, y_2, \cdots, y_k) &= f(y_1, y_2, \cdots, y_k) + \lambda\varphi(y_1, y_2, \cdots, y_k), \\ &= n^2 - \sum_{s=1}^{k} y_s^2 + \lambda(\sum_{s=1}^{k} y_s - n), \end{aligned}$$

We use Lagrange multiplier method to calculate the extremum of the function $f(y_1, y_2, \cdots, y_k)$. We know that to let f achieve its extremum, the function $f(y_1, y_2, \cdots, y_k)$ must meet the following conditions:

$$\frac{\partial F}{\partial y_s} = -2y_s + \lambda = 0, s = 1, 2, \cdots, k$$

$$\frac{\partial F}{\partial \lambda} = \sum_{s=1}^{k} y_s - n = 0.$$

Namely:

$$y_s = \lambda/2, \qquad s = 1, 2, \cdots, k \tag{12}$$

$$\sum_{s=1}^{k} y_s - n = 0. \tag{13}$$

By substituting (12) into (13), we obtain

$$\sum_{s=1}^{k} (\lambda/2) - n = 0, (\lambda/2)\ k = n, \quad \lambda = \frac{2n}{k}. \tag{14}$$

By substituting (14) into (12), we have

$$y_s = \frac{n}{k}, s = 1, 2, \cdots, k, \tag{15}$$

By substituting (15) into (11), we have

$$f(y_1, y_2, \cdots, y_n) = n^2 - \sum_{s=1}^{k} (\frac{n}{k})^2 = n^2 - (\frac{n}{k})^2 k = n^2(1 - 1/k), \tag{16}$$

From (16), it is easy to see that for a graph G, accompanying with the increase in the number of distinct eigenvalues, the maximum of $n^2 - \sum_{s=1}^{k} y_s^2$ equals $n^2(1 - 1/k)$, which is a monotonically decreasing function with respect to the k. By a series of the above derivation steps, we obtain the following Theorem 7:

Theorem 7. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathbf{M}(G)$. Suppose that $\mathbf{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathbf{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1)$, $\alpha_A(\eta_2)$, $\cdots, \alpha_A(\eta_k)$, respectively. Thus, we have

1. For a known graph G, the k is a constant. For $s = 1, 2, \cdots, k$, if $y_s = n/k$, then the function $n^2 - \sum_{s=1}^{k} y_s^2$ gets the maximum value $n^2(1 - 1/k)$. Thus, the orthogonal matrix $\mathbf{Q}$ has at least $n^2(1 - 1/k)$ zero entries and at most n^2/k non-zero entries.

2. For different graph G, if $k = 1$ (G only has a n multiple eigenvalue), the orthogonal matrix $\mathbf{Q}$ satisfying condition $\mathbf{Q} = \mathbf{X}^{\mathrm{T}}\mathbf{P}\mathbf{X}$ has at most n^2 non-zero entries. If $k = n$ (G has n distinct eigenvalues), the orthogonal matrix $\mathbf{Q}$ satisfying condition $\mathbf{Q} = \mathbf{X}^{\mathrm{T}}\mathbf{P}\mathbf{X}$ has at least $n^2 - n$ zero entries and at most n non-zero entries.

From previous studies, it can be seen that the key issue finding the automorphisms of a graph G is how to solve the permutation matrix $\mathbf{P}$. Because $\mathbf{P} = \mathbf{X}\mathbf{Q}\mathbf{X}^{\mathrm{T}}$, computing $\mathbf{P}$ in turn depends on how to solve the orthogonal matrix $\mathbf{Q}$. Suppose that $\mathbf{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathbf{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1), \alpha_A(\eta_2), \cdots, \alpha_A(\eta_k)$, respectively.

Now we show that when establishing (4), how we skillfully arrange the n eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_n$ of the graph G in the diagonal of the matrix such that the orthogonal matrix $\mathbf{Q}$ produced has a simpler structure and is more suitable for the subsequent processing.

When constructing (4), without loss of generality, we can assume that the algebraic multiplicities of the k distinct eigenvalues satisfy $\alpha_A(\eta_1) \le \alpha_A(\eta_2) \le \cdots \le \alpha_A(\eta_k)$. Accordingly, let the k distinct eigenvalues be arranged in ascending order $\eta_1, \eta_2, \cdots, \eta_k$ according to the size of the algebraic multiplicities. If the sizes of two algebraic multiplicities are the same, except the eigenvalues whose algebraic multiplicity is 1, let the eigenvalues be arranged in descending order. For all eigenvalues whose algebraic multiplicity is 1, let them specially be arranged as follows: First of all, if there is a 0-type single eigenvalue, let it be arranged in the front. Second, if there are 1-type eigenvalues, let them be arranged in descending order according to the size of the absolute value of the eigenvalues. Next, if there are 2-type eigenvalues, let them be arranged in descending order according to the size of the eigenvalues. In the following sections, when encountering any a graph, we assume that the sequence arranged in accordance with the above requirements is $\lambda_1, \lambda_2, \cdots, \lambda_n$.

A key advantage of such an arrangement is that it makes (4) closer near the bottom of the main diagonal have more zero entries and ultimately makes the orthogonal matrix $\mathbf{Q}$ a following block diagonal matrix:

$$\mathbf{Q} = \begin{bmatrix} \mathrm{B}_1 & 0 & \cdots & 0 \\ 0 & \mathrm{B}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathrm{B}_l \end{bmatrix}, \tag{17}$$

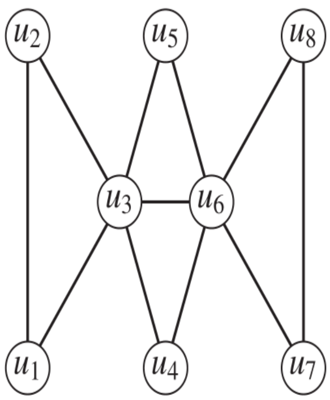
where the B_s is a square matrix of rank r_s with $s = 1, 2, \cdots, l$ and $\sum_{s=1}^{l} r_s = n$.

We know that if an orthogonal matrix $\mathbf{Q}$ is a block diagonal matrix as (17), then each square block matrix B_s of rank r_s ($s = 1, 2, \cdots, l; \sum_{s=1}^{l} r_s = n$) is an orthogonal matrix. We call the block diagonal orthogonal matrix $\mathbf{Q}$ constructed by (17) **the standard block orthogonal matrix**. By the preceding discussion, we establish the following Theorem 8.

Theorem 8. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathrm{M}(G)$. Suppose that $\mathrm{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathrm{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1), \alpha_A(\eta_2), \cdots, \alpha_A(\eta_k)$, respectively. If the graph G exists an automorphism map γ, then there must exist a standard block orthogonal matrix $\mathbf{Q}$ expressed by (17) corresponding to the γ. Moreover, have $\mathbf{Q}\Lambda^t - \Lambda^t\mathbf{Q} = 0$ for $t \in N$.

By solving (8), if we have calculated all elements of the matrix $\mathbf{Q}$ including all zero entries, by Theorem 8 we can obtain the standard block orthogonal matrix $\mathbf{Q}$. Furthermore, we can express the $\mathbf{Q}$ as the following form:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{B}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{B}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{B}_l \end{bmatrix}$$



$\gamma_1 = (1)$, $\gamma_2 = (1, 2)$,
$\gamma_3 = (4, 5)$, $\gamma_4 = (7, 8)$,
$\gamma_5 = (1, 2)(4, 5)$, $\gamma_6 = (1, 2)(7, 8)$,
$\gamma_7 = (4, 5)(7, 8)$, $\gamma_8 = (1, 2)(4, 5)(7, 8)$,
$\gamma_9 = (3, 6)(1, 8)(2, 7)$, $\gamma_{10} = (3, 6)(1, 7)(2, 8)$,
$\gamma_{11} = (3, 6)(1, 8)(2, 7)(4, 5)$, $\gamma_{12} = (3, 6)(1, 7)(2, 8)(4, 5)$,
$\gamma_{13} = (3, 6)(7, 8)(1, 8)(2, 7)(4, 5)$, $\gamma_{14} = (3, 6)(1, 2)(1, 8)(2, 7)(4, 5)$,
$\gamma_{15} = (3, 6)(1, 2)(1, 8)(2, 7)$, $\gamma_{16} = (3, 6)(7, 8)(1, 8)(2, 7)$,

(a) A graph *G*

(b) $\mathrm{Aut}(G)$ of the graph

Figure 1. A Graph *G* and Its Automorphism Group

where B_s with $s = 1, 2, \cdots, l$ are r_s-order orthogonal matrices. When building the matrix $\mathbf{Q}$, if there exist the 0 or 1-type eigenvalues whose algebraic multiplicity is 1, by (17) there must exist a block diagonal orthogonal matrix in the upper left corner of the matrix $\mathbf{Q}$.

We denote it by the orthogonal matrix B_1 whose diagonal entries correspond to the 0 or 1-type eigenvalues. Because the algebraic multiplicities of the 0 or 1-type eigenvalue all is 1, we have

$$\mathbf{B}_1 = \begin{bmatrix} y_1 & \mathrm{O} & \cdots & \mathrm{O} \\ \mathrm{O} & y_2 & \cdots & \mathrm{O} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{O} & \mathrm{O} & \cdots & y_{r_1} \end{bmatrix} = \begin{bmatrix} \pm 1 & \mathrm{O} & \cdots & \mathrm{O} \\ \mathrm{O} & \pm 1 & \cdots & \mathrm{O} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{O} & \mathrm{O} & \cdots & \pm 1 \end{bmatrix} \tag{18}$$

Similar to the B_1, when building the matrix Q, if there are the 2-type eigenvalues whose algebraic multiplicity is 1, then there must exist a block diagonal orthogonal matrix below the B_1 in the upper left corner of the matrix Q. We denote it by the orthogonal matrix B_2 whose diagonal entries correspond to the 2-type eigenvalues. Because the algebraic multiplicities of the eigenvalue all is 1, we have

$$\mathrm{B}_2 = \begin{bmatrix} z_1 & 0 & \cdots & 0 \\ 0 & z_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & z_{r_2} \end{bmatrix} = \begin{bmatrix} \pm 1 & 0 & \cdots & 0 \\ 0 & \pm 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \pm 1 \end{bmatrix} \tag{19}$$

Conjecture 1. Let $G = (V(G), E(G), W(G))$ be an undirected weighted graph with n nodes whose adjacency matrix is $\mathrm{M}(G)$. Suppose that $\mathrm{M}(G)$ has k distinct eigenvalues $\eta_1, \eta_2, \cdots, \eta_k$. All the n eigenvalues of $\mathrm{M}(G)$ are $\lambda_1, \lambda_2, \cdots, \lambda_n$ with $\lambda_i \in \{\eta_1, \eta_2, \cdots, \eta_k\}$ for $i = 1, 2, \cdots, n$. The algebraic multiplicities of the eigenvalues are $\alpha_A(\eta_1)$, $\alpha_A(\eta_2)$, $\cdots$, $\alpha_A(\eta_k)$, respectively.

By Formulas $\mathrm{Q} = \mathrm{X}^{\mathrm{T}}\mathrm{P}\mathrm{X}$ and $\mathrm{P} = \mathrm{X}\mathrm{Q}\mathrm{X}^{\mathrm{T}}$, to compute the permutation matrix P, we have

1. In the B_1 from (18), have $y_i = 1, -1$ when $i = 1, 2, \cdots, r_1$.
2. In the B_2 from (19), have $z_i = 1$ when $i = 1, 2, \cdots, r_2$.

4. For Example Using Eigenvalue Partition

In this section, we will offer two examples that use the eigenvalue partition way to solve all automorphisms of a graph. However, by means of them, we represent the entire process for computing the automorphisms group. First of all, we give an example that use Theorem 2 and Theorem 3 to seek automorphisms.

Example 1. Use the algebraic method to solve the automorphism group of the graph G in Figure. 1(a).

Solution 1. The adjacency matrix of the graph in Figure.1(a) is:

$$\mathrm{M}(G) = \begin{bmatrix} 01100000 \\ 10100000 \\ 11011100 \\ 00100100 \\ 00100100 \\ 00111011 \\ 00000101 \\ 00000110 \end{bmatrix}$$

The characteristic polynomial of $\mathrm{M}(G)$ is as follows:

$$x^8 - 11x^6 - 8x^5 + 27x^4 + 28x^3 - 9x^2 - 12x = 0 \tag{20}$$

First of all, we compute the characteristic polynomial 20. Furthermore, we compute the eigenvalues and the eigenvectors. Next, based on the aforementioned method, let the 8 distinct eigenvalues be arranged in ascending order according to the size of the algebraic multiplicities. Next, in the eigenvalues whose algebraic multiplicity is 1, let $\lambda_1 = 0$ be the

0-type eigenvalues. Let $\lambda_2 = 1.7321$ and $\lambda_3 = -1.7321$ be the 1-type eigenvalues. Let $\lambda_4 = 3.1774$, $\lambda_5 = 0.6784$, and $\lambda_6 = -1.8558$ be 2-type eigenvalues. Next, in the eigenvalues whose algebraic multiplicity is 2, let $\lambda_7 = -1$ and $\lambda_8 = -1$ be the 2-type eigenvalues. Finally, we arrange all eigenvalues and eigenvectors of the graph as follows:

$\lambda_1 = 0$, $X_1 = (0, 0, 0, 0.7071, -0.7071, 0, 0, 0)^T$
$\lambda_2 = 1.7321$, $X_2 = (0.4440, 0.4440, 0.3251, 0, 0, -0.3251, -0.4440, -0.4440)^T$
$\lambda_3 = -1.7321$, $X_3 = (-0.2299, -0.2299, 0.6280, 0, 0, -0.6280, 0.2299, 0.2299)^T$
$\lambda_4 = 3.1774$, $X_4 = (-0.2408, -0.2408, -0.5244, -0.3301, -0.3301, -0.5244, -0.2408, -0.2408)^T$
$\lambda_5 = 0.6784$, $X_5 = (-0.4081, -0.4081, 0.1312, 0.3870, 0.3870, 0.1312, -0.4081, -0.4081)^T$
$\lambda_6 = -1.8558$, $X_6 = (0.1596, 0.1596, -0.4558, 0.4912, 0.4912, -0.4558, 0.1596, 0.1596)^T$
$\lambda_7 = -1$, $X_7 = (0, 0, 0, 0, 0, 0, 0.7071, -0.7071)^T$
$\lambda_8 = -1$, $X_8 = (0.7071, -0.7071, 0, 0, 0, 0, 0, 0)^T$

It is obviously that only the algebraic multiplicities of eigenvalue −1 is 2, and the remaining eigenvalue all are 1. By (4), we have

$$\begin{bmatrix} (\lambda_1-\lambda_1)x_{1,1}(\lambda_2-\lambda_1)x_{1,2}\cdots(\lambda_8-\lambda_1)x_{1,8} \\ (\lambda_1-\lambda_2)x_{2,1}(\lambda_2-\lambda_2)x_{2,2}\cdots(\lambda_8-\lambda_2)x_{2,8} \\ \vdots \quad \cdots \quad \cdots \quad \vdots \\ (\lambda_1-\lambda_8)x_{8,1}(\lambda_2-\lambda_8)x_{8,2}\cdots(\lambda_8-\lambda_8)x_{8,8} \end{bmatrix} = 0. \tag{21}$$

$$\begin{bmatrix}
(\lambda_1-\lambda_1)x_{1,1} & (\lambda_2-\lambda_1)x_{1,2} & (\lambda_3-\lambda_1)x_{1,3} & (\lambda_4-\lambda_1)x_{1,4} & (\lambda_5-\lambda_1)x_{1,5} & (\lambda_6-\lambda_1)x_{1,6} & (\lambda_7-\lambda_1)x_{1,7} & (\lambda_8-\lambda_1)x_{1,8} \\
(\lambda_1-\lambda_2)x_{2,1} & (\lambda_2-\lambda_2)x_{2,2} & (\lambda_3-\lambda_2)x_{2,3} & (\lambda_4-\lambda_2)x_{2,4} & (\lambda_5-\lambda_2)x_{2,5} & (\lambda_6-\lambda_2)x_{2,6} & (\lambda_7-\lambda_2)x_{2,7} & (\lambda_8-\lambda_2)x_{2,8} \\
(\lambda_1-\lambda_3)x_{3,1} & (\lambda_2-\lambda_3)x_{3,2} & (\lambda_3-\lambda_3)x_{3,3} & (\lambda_4-\lambda_3)x_{3,4} & (\lambda_5-\lambda_3)x_{3,5} & (\lambda_6-\lambda_3)x_{3,6} & (\lambda_7-\lambda_3)x_{3,7} & (\lambda_8-\lambda_3)x_{3,8} \\
(\lambda_1-\lambda_4)x_{4,1} & (\lambda_2-\lambda_4)x_{4,2} & (\lambda_3-\lambda_4)x_{4,3} & (\lambda_4-\lambda_4)x_{4,4} & (\lambda_5-\lambda_4)x_{4,5} & (\lambda_6-\lambda_4)x_{4,6} & (\lambda_7-\lambda_4)x_{4,7} & (\lambda_8-\lambda_4)x_{4,8} \\
(\lambda_1-\lambda_5)x_{5,1} & (\lambda_2-\lambda_5)x_{5,2} & (\lambda_3-\lambda_5)x_{5,3} & (\lambda_4-\lambda_5)x_{5,4} & (\lambda_5-\lambda_5)x_{5,5} & (\lambda_6-\lambda_5)x_{5,6} & (\lambda_7-\lambda_5)x_{5,7} & (\lambda_8-\lambda_5)x_{5,8} \\
(\lambda_1-\lambda_6)x_{6,1} & (\lambda_2-\lambda_6)x_{6,2} & (\lambda_3-\lambda_6)x_{6,3} & (\lambda_4-\lambda_6)x_{6,4} & (\lambda_5-\lambda_6)x_{6,5} & (\lambda_6-\lambda_6)x_{6,6} & (\lambda_7-\lambda_6)x_{6,7} & (\lambda_8-\lambda_6)x_{6,8} \\
(\lambda_1-\lambda_7)x_{7,1} & (\lambda_2-\lambda_7)x_{7,2} & (\lambda_3-\lambda_7)x_{7,3} & (\lambda_4-\lambda_7)x_{7,4} & (\lambda_5-\lambda_7)x_{7,5} & (\lambda_6-\lambda_7)x_{7,6} & (\lambda_7-\lambda_7)x_{7,7} & (\lambda_8-\lambda_7)x_{7,8} \\
(\lambda_1-\lambda_8)x_{8,1} & (\lambda_2-\lambda_8)x_{8,2} & (\lambda_3-\lambda_8)x_{8,3} & (\lambda_4-\lambda_8)x_{8,4} & (\lambda_5-\lambda_8)x_{8,5} & (\lambda_6-\lambda_8)x_{8,6} & (\lambda_7-\lambda_8)x_{8,7} & (\lambda_8-\lambda_8)x_{8,8}
\end{bmatrix} = 0.$$

$$\begin{bmatrix}
0x_{1,1} & 1.732x_{1,2} & -1.732x_{1,3} & 3.177x_{1,4} & 0.6784x_{1,5} & -1.856x_{1,6} & -1x_{1,7} & -1x_{1,8} \\
-1.732x_{2,1} & 0x_{2,2} & -3.464x_{2,3} & 1.445x_{2,4} & -1.054x_{2,5} & -3.588x_{2,6} & -2.732x_{2,7} & -2.732x_{2,8} \\
1.732x_{3,1} & 3.464x_{3,2} & 0x_{3,3} & 4.910x_{3,4} & 2.410x_{3,5} & -0.1237x_{3,6} & 0.7321x_{3,7} & 0.7321x_{3,8} \\
-3.177x_{4,1} & -1.445x_{4,2} & -4.910x_{4,3} & 0x_{4,4} & -2.499x_{4,5} & -5.033x_{4,6} & -4.177x_{4,7} & -4.177x_{4,8} \\
-0.6784x_{5,1} & 1.054x_{5,2} & -2.410x_{5,3} & 2.499x_{5,4} & 0x_{5,5} & -2.534x_{5,6} & -1.678x_{5,7} & -1.678x_{5,8} \\
1.856x_{6,1} & 3.588x_{6,2} & 0.1237x_{6,3} & 5.033x_{6,4} & 2.534x_{6,5} & (0x_{6,6} & 0.8558x_{6,7} & 0.8558x_{6,8} \\
x_{7,1} & 2.732x_{7,2} & -0.7321x_{7,3} & 4.177x_{7,4} & 1.678x_{7,5} & -0.8558x_{7,6} & 0x_{7,7} & 0x_{7,8} \\
x_{8,1} & 2.732x_{8,2} & -0.7321x_{8,3} & 4.177x_{8,4} & 1.678x_{8,5} & -0.8558x_{8,6} & 0x_{8,7} & 0x_{8,8}
\end{bmatrix} = 0.$$

Equations 22 represented by the matrix is equal to the following Equations 23. In fact, we do not need accurately to calculate the each of elements of the matrix Equations 22. However, based on the eigenvalues and their algebraic multiplicity, we can quickly get the following matrix Equations 23. The reason we describe in detail these intermediate steps of computation is to help people understand the principles of the paper.

$$\begin{bmatrix}
0x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & x_{1,6} & x_{1,7} & x_{1,8} \\
x_{2,1} & 0x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} & x_{2,6} & x_{2,7} & x_{2,8} \\
x_{3,1} & x_{3,2} & 0x_{3,3} & x_{3,4} & x_{3,5} & x_{3,6} & x_{3,7} & x_{3,8} \\
x_{4,1} & x_{4,2} & x_{4,3} & 0x_{4,4} & x_{4,5} & x_{4,6} & x_{4,7} & x_{4,8} \\
x_{5,1} & x_{5,2} & x_{5,3} & x_{5,4} & 0x_{5,5} & x_{5,6} & x_{5,7} & x_{5,8} \\
x_{6,1} & x_{6,2} & x_{6,3} & x_{6,4} & x_{6,5} & 0x_{6,6} & x_{6,7} & x_{6,8} \\
x_{7,1} & x_{7,2} & x_{7,3} & x_{7,4} & x_{7,5} & x_{7,6} & 0x_{7,7} & 0x_{7,8} \\
x_{8,1} & x_{8,2} & x_{8,3} & x_{8,4} & x_{8,5} & x_{8,6} & 0x_{8,7} & 0x_{8,8}
\end{bmatrix} = 0. \tag{23}$$

Obviously, to make the two sides of (23) be equal, all the diagonal entries, $x_{7,8}$, and $x_{8,7}$ can take any value. Moreover, the remaining entries must all be 0.

Because the elements $x_{i,j}$ with $i = 1, 2, \cdots, 8$ compose the orthogonal matrix $\mathbf{Q}$, the orthogonal matrix $\mathbf{Q}$ must be the following form:

$$\mathbf{Q} = \begin{bmatrix} x_{1,1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x_{2,2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x_{3,3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_{4,4} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_{5,5} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x_{6,6} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x_{7,7} & x_{7,8} \\ 0 & 0 & 0 & 0 & 0 & 0 & x_{8,7} & x_{8,8} \end{bmatrix} = 0. \tag{24}$$

Because $x_{i,i}^2 = 1$, $x_{i,i} = \pm 1$ for $i = 1, 2, 3, 4, 5, 6$. Thus, the form of the $\mathbf{Q}$ must be as follows:

$$\mathbf{Q} = \begin{bmatrix} \pm 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \pm 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \pm 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \pm 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \pm 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \pm 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x_{7,7} & x_{7,8} \\ 0 & 0 & 0 & 0 & 0 & 0 & x_{8,7} & x_{8,8} \end{bmatrix} = 0. \tag{25}$$

Furthermore, to construct the orthogonal matrix $\mathbf{Q}$, the $\mathbf{B}$ must are a block orthogonal matrix, where $\mathbf{B} = \begin{bmatrix} x_{7,7} & x_{7,8} \\ x_{8,7} & x_{8,8} \end{bmatrix}$. Let the orthogonal matrix $\mathbf{B}$ be as follows:

$(1): \mathbf{B}_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $(2): \mathbf{B}_2 = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$, $(3): \mathbf{B}_3 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $(4): \mathbf{B}_4 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$,
$(5): \mathbf{B}_7 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ $(6): \mathbf{B}_8 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$, $(7): \mathbf{B}_{11} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, $(8): \mathbf{B}_{12} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$.

By the matrix $\mathbf{B}_i$ with $i = 1, 2, \cdots, 8$, we can correspondingly obtain the orthogonal matrix $\mathbf{Q}_i$ with $i = 1, 2, \cdots, 16$ as follows:

$$\mathbf{Q}_1 = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix}, \quad \mathbf{Q}_2 = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&-1 \end{bmatrix}, \quad \mathbf{Q}_3 = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&-1 \end{bmatrix}, \quad \mathbf{Q}_4 = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix},$$

$$\mathbf{Q}_5 = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix}, \quad \mathbf{Q}_6 = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&-1 \end{bmatrix} \mathbf{Q}_7 = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&-1 \end{bmatrix}, \quad \mathbf{Q}_8 = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&-1&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix},$$

$$\mathbf{Q}_9 = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix}, \quad \mathbf{Q}_{10} = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&-1&0 \end{bmatrix}, \quad \mathbf{Q}_{11} = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&-1 \\ 0&0&0&0&0&0&1&0 \end{bmatrix}, \quad \mathbf{Q}_{12} = \begin{bmatrix} -1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&-1 \\ 0&0&0&0&0&0&-1&0 \end{bmatrix},$$

$$\mathbf{Q}_{13} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&1&0 \end{bmatrix}, \quad \mathbf{Q}_{14} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&-1 \\ 0&0&0&0&0&0&1&0 \end{bmatrix}, \mathbf{Q}_{15} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&1 \\ 0&0&0&0&0&0&-1&0 \end{bmatrix}, \mathbf{Q}_{16} = \begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&-1&0&0&0&0&0&0 \\ 0&0&-1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&-1 \\ 0&0&0&0&0&0&-1&0 \end{bmatrix},$$

By the previously computed eigenvectors, we express the corresponding matrix as follows:

$$\mathbf{X} = [\mathbf{X}_1 \mathbf{X}_2 \mathbf{X}_3 \mathbf{X}_4 \mathbf{X}_5 \mathbf{X}_6 \mathbf{X}_7 \mathbf{X}_8] =$$

$$\begin{bmatrix} 0 & 0.4440 & -0.2299 & -0.2408 & -0.4081 & 0.1596 & 0 & 0.7071 \\ 0 & 0.4440 & -0.2299 & -0.2408 & -0.4081 & 0.1596 & 0 & -0.7071 \\ 0 & 0.3251 & 0.6280 & -0.5244 & 0.1312 & -0.4558 & 0 & 0 \\ 0.7071 & 0 & 0 & -0.3301 & 0.3870 & 0.4912 & 0 & 0 \\ -0.7071 & 0 & 0 & -0.3301 & 0.3870 & 0.4912 & 0 & 0 \\ 0 & -0.3251 & -0.6280 & -0.5244 & 0.1312 & -0.4558 & 0 & 0 \\ 0 & -0.4440 & 0.2299 & -0.2408 & -0.4081 & 0.1596 & 0.7071 & 0 \\ 0 & -0.4440 & 0.2299 & -0.2408 & -0.4081 & 0.1596 & -0.7071 & 0 \end{bmatrix}.$$

Replacing the matrix $\mathbf{Q}$ of $\mathbf{P} = \mathbf{XQX}^{\mathrm{T}}$ by $\mathbf{Q}_i$ with $i = 1, 2, \cdots, 16$, respectively, we obtain $\mathbf{P}_1 = \mathbf{XQ}_1\mathbf{X}^{\mathrm{T}} = \mathbf{I}$. The corresponding automorphism map is: $\gamma_1 = (1)$.

$$\mathbf{P}_2 = \mathbf{XQ}_2\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 01000000 \\ 10000000 \\ 00100000 \\ 00010000 \\ 00001000 \\ 00000100 \\ 00000001 \\ 00000010 \end{bmatrix}. \ \mathbf{P}_3 = \mathbf{XQ}_3\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 01000000 \\ 10000000 \\ 00100000 \\ 00010000 \\ 00001000 \\ 00000100 \\ 00000010 \\ 00000001 \end{bmatrix}. \ \mathbf{P}_4 = \mathbf{XQ}_4\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 10000000 \\ 01000000 \\ 00100000 \\ 00010000 \\ 00001000 \\ 00000100 \\ 00000001 \\ 00000010 \end{bmatrix}.$$

The automorphism maps corresponding to $\mathbf{P}_2$, $\mathbf{P}_3$, and $\mathbf{P}_4$ are $\gamma_6 = (1, 2)(7, 8)$, $\gamma_2 = (1, 2)$, and $\gamma_4 = (7, 8)$, respectively.

$$\mathbf{P}_5 = \mathbf{XQ}_5\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 10000000 \\ 01000000 \\ 00100000 \\ 00001000 \\ 00010000 \\ 00000100 \\ 00000010 \\ 00000001 \end{bmatrix}. \mathbf{P}_6 = \mathbf{XQ}_6\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 01000000 \\ 10000000 \\ 00100000 \\ 00001000 \\ 00010000 \\ 00000100 \\ 00000001 \\ 00000010 \end{bmatrix}. \ \mathbf{P}_7 = \mathbf{XQ}_7\mathbf{X}^{\mathrm{T}} = \begin{bmatrix} 01000000 \\ 10000000 \\ 00100000 \\ 00001000 \\ 00010000 \\ 00000100 \\ 00000010 \\ 00000001 \end{bmatrix}.$$

The automorphism maps corresponding to P_5, P_6, and P_7 are $\gamma_3 = (4,5)$, $\gamma_8 = (1,2)(4,5)(7,8)$, and $\gamma_5 = (1,2)(4,5)$, respectively.

$$P_8 = XQ_8X^T = \begin{bmatrix} 10000000 \\ 01000000 \\ 00100000 \\ 00001000 \\ 00010000 \\ 00000100 \\ 00000001 \\ 00000010 \end{bmatrix}. P_9 = XQ_9X^T = \begin{bmatrix} 00000010 \\ 00000001 \\ 00000100 \\ 00001000 \\ 00010000 \\ 00100000 \\ 10000000 \\ 01000000 \end{bmatrix}. P_{10} = XQ_{10}X^T = \begin{bmatrix} 00000001 \\ 00000010 \\ 00000100 \\ 00001000 \\ 00010000 \\ 00100000 \\ 10000000 \\ 01000000 \end{bmatrix}.$$

The automorphism maps corresponding to P_8, P_9, and P_{10} are $\gamma_7 = (4,5)(7,8)$, $\gamma_{12} = (3,6)(1,7)(2,8)(4,5)$, and $\gamma_{13} = (3,6)(7,8)(1,8)(2,7)(4,5)$, respectively.

$$P_{11} = XQ_{11}X^T = \begin{bmatrix} 00000010 \\ 00000001 \\ 00000100 \\ 00001000 \\ 00010000 \\ 00100000 \\ 01000000 \\ 10000000 \end{bmatrix}. P_{12} = XQ_{12}X^T = \begin{bmatrix} 00000001 \\ 00000010 \\ 00000100 \\ 00001000 \\ 00010000 \\ 00100000 \\ 01000000 \\ 10000000 \end{bmatrix}. P_{13} = XQ_{13}X^T = \begin{bmatrix} 00000010 \\ 00000001 \\ 00000100 \\ 00010000 \\ 00001000 \\ 00100000 \\ 10000000 \\ 01000000 \end{bmatrix}.$$

The automorphism maps corresponding to P_{11}, P_{12}, and P_{13} are $\gamma_{14} = (3,6)(1,2)(1,8)(2,7)(4,5)$, $\gamma_{11} = (3,6)(1,8)(2,7)(4,5)$, and $\gamma_{10} = (3,6)(1,7)(2,8)$, respectively.

$$P_{14} = XQ_{14}X^T = \begin{bmatrix} 00000010 \\ 00000001 \\ 00000100 \\ 00010000 \\ 00001000 \\ 00100000 \\ 01000000 \\ 10000000 \end{bmatrix}. P_{15} = XQ_{15}X^T = \begin{bmatrix} 00000001 \\ 00000010 \\ 00000100 \\ 00010000 \\ 00001000 \\ 00100000 \\ 10000000 \\ 01000000 \end{bmatrix}. P_{16} = XQ_{16}X^T = \begin{bmatrix} 00000001 \\ 00000010 \\ 00000100 \\ 00010000 \\ 00001000 \\ 00100000 \\ 01000000 \\ 10000000 \end{bmatrix}.$$

The automorphism maps corresponding to P_{14}, P_{15}, and P_{16} are $\gamma_{15} = (3,6)(1,2)(1,8)(2,7)$, $\gamma_{16} = (3,6)(7,8)(1,8)(2,7)$, and $\gamma_9 = (3,6)(1,8)(2,7)$, respectively.

Finally, we list all automorphisms of the graph G in Figure.1(*a*) into Figure.1(*b*). □

Example 2. Use the algebraic method to solve the automorphism group of the weighted graph G in Figure. 2(*a*).

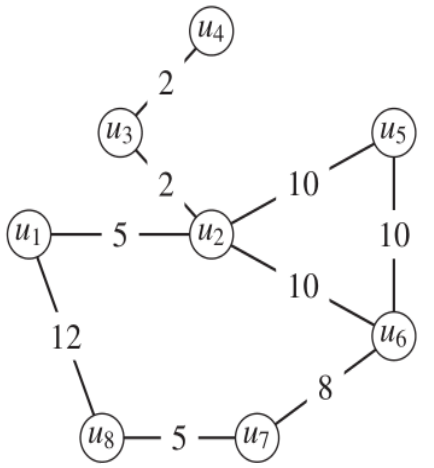
Solution 2. The adjacency matrix of the graph in Figure. 2(*a*) is:

$$\mathbf{M}(G) = \begin{bmatrix} 0 & 5 & 0 & 0 & 0 & 0 & 0 & 12 \\ 5 & 0 & 2 & 0 & 10 & 10 & 0 & 0 \\ 0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 10 & 0 & 0 & 10 & 0 & 8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 8 & 0 & 5 \\ 12 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \end{bmatrix}.$$

The characteristic polynomial of $\mathbf{M}(G)$ is as follows:

$$x^8 - 566x^6 - 2000x^5 + 74605x^4 + 298000x^3 - 1852728x^2 - 1160000x + 5856400 = 0. \tag{26}$$

First of all, we compute the characteristic polynomial 26. Next, compute the eigenvalues and the eigenvectors. Based on the aforementioned method, we let $\lambda_1 = 21.9641$, $\lambda_2 = 12.1667$, $\lambda_3 = 3.3194$, $\lambda_4 = 1.8699$, $\lambda_5 = -1.9455$, $\lambda_6 = -9.0727$, $\lambda_7 = -13.6836$, and $\lambda_8 = -14.6182$ be 2-type eigenvalues. Finally, we arrange all eigenvalues and eigenvectors of the graph G as follows:



$\gamma = (1).$

(a) A graph *G* (b) $\mathrm{Aut}(G)$

Figure 2. A Graph G and Its Automorphism Group

$\lambda_1 = 21.9641$, $\mathbf{X}_1 = (0.2183, 0.5390, 0.0495, 0.0045, 0.5017, 0.5630, 0.2449, 0.1750)^T$
$\lambda_2 = 12.1667$, $\mathbf{X}_2 = (-0.6296, 0.1094, 0.0185, 0.0030, 0.2497, 0.1944, -0.1532, -0.6839)^T$
$\lambda_3 = 3.3194$, $\mathbf{X}_3 = (0.2957, 0.3324, 0.3144, 0.1895, 0.1742, -0.2746, -0.7472, -0.0567)^T$
$\lambda_4 = 1.8699$, $\mathbf{X}_4 = (-0.1214, -0.0856, 0.6354, 0.6796, -0.1415, 0.0591, 0.2976, 0.0167)^T$
$\lambda_5 = -1.9455$, $\mathbf{X}_5 = (-0.0375, 0.0380, 0.6888, -0.7081, -0.1098, -0.0167, 0.0937, -0.0098)^T$
$\lambda_6 = -9.0727$, $\mathbf{X}_6 = (0.2651, 0.4889, -0.1133, 0.0250, -0.6958, 0.1424, 0.0972, -0.4041)^T$
$\lambda_7 = -13.6836$, $\mathbf{X}_7 = (0.2075, -0.4891, 0.0730, -0.0107, -0.1678, 0.7187, -0.4082, -0.0328)^T$
$\lambda_8 = -14.6182$, $\mathbf{X}_8 = (-0.5824, 0.3162, -0.0441, 0.0060, -0.3334, 0.1711, -0.2912, 0.5777)^T$

It can be seen that all the algebraic multiplicities of eigenvalue is 1. By (4), we have

$$\begin{bmatrix} (\lambda_1 - \lambda_1)x_{1,1}(\lambda_2 - \lambda_1)x_{1,2}\cdots(\lambda_8 - \lambda_1)x_{1,8} \\ (\lambda_1 - \lambda_2)x_{2,1}(\lambda_2 - \lambda_2)x_{2,2}\cdots(\lambda_8 - \lambda_2)x_{2,8} \\ \vdots \qquad \cdots \qquad \cdots \qquad \vdots \\ (\lambda_1 - \lambda_8)x_{8,1}(\lambda_2 - \lambda_8)x_{8,2}\cdots(\lambda_8 - \lambda_8)x_{8,8} \end{bmatrix} = 0. \tag{27}$$

By substituting all eigenvalues into (27) and simplifying, we have

$$\begin{bmatrix} 0x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & x_{1,6} & x_{1,7} & x_{1,8} \\ x_{2,1} & 0x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} & x_{2,6} & x_{2,7} & x_{2,8} \\ x_{3,1} & x_{3,2} & 0x_{3,3} & x_{3,4} & x_{3,5} & x_{3,6} & x_{3,7} & x_{3,8} \\ x_{4,1} & x_{4,2} & x_{4,3} & 0x_{4,4} & x_{4,5} & x_{4,6} & x_{4,7} & x_{4,8} \\ x_{5,1} & x_{5,2} & x_{5,3} & x_{5,4} & 0x_{5,5} & x_{5,6} & x_{5,7} & x_{5,8} \\ x_{6,1} & x_{6,2} & x_{6,3} & x_{6,4} & x_{6,5} & 0x_{6,6} & x_{6,7} & x_{6,8} \\ x_{7,1} & x_{7,2} & x_{7,3} & x_{7,4} & x_{7,5} & x_{7,6} & 0x_{7,7} & x_{7,8} \\ x_{8,1} & x_{8,2} & x_{8,3} & x_{8,4} & x_{8,5} & x_{8,6} & x_{8,7} & 0x_{8,8} \end{bmatrix} = 0. \tag{28}$$

Because the elements $x_{i,j}$ with $i = 1, 2, \cdots, 8$ compose the orthogonal matrix $\mathbf{Q}$, the orthogonal matrix $\mathbf{Q}$ must be the following form:

$$\mathbf{Q} = \begin{bmatrix} x_{1,1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & x_{2,2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & x_{3,3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_{4,4} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & x_{5,5} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & x_{6,6} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & x_{7,7} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x_{8,8} \end{bmatrix} = 0. \tag{29}$$

Because the $\mathbf{Q}$ is orthogonal, have $x_{i,i}^2 = 1$. As a result, $x_{i,i} = \pm 1$ for $i = 1, 2, 3, 4, 5, 6, 7, 8$. Thus, the form of the $\mathbf{Q}$ must be as follows:

$$\mathbf{Q} = \begin{bmatrix} \pm 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \pm 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \pm 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \pm 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \pm 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \pm 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \pm 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \pm 1 \end{bmatrix} = 0. \tag{30}$$

By the previously computed eigenvectors, we express the corresponding matrix as follows:

$$\mathbf{X} = [\mathbf{X}_1 \mathbf{X}_2 \mathbf{X}_3 \mathbf{X}_4 \mathbf{X}_5 \mathbf{X}_6 \mathbf{X}_7 \mathbf{X}_8] = \begin{bmatrix} 0.2183 & -0.6296 & 0.2957 & -0.1214 & -0.0375 & 0.2651 & 0.2075 & -0.5824 \\ 0.5390 & 0.1094 & 0.3324 & -0.0856 & 0.0380 & 0.4889 & -0.4891 & 0.3162 \\ 0.0495 & 0.0185 & 0.3144 & 0.6354 & 0.6888 & -0.1133 & 0.0730 & -0.0441 \\ 0.0045 & 0.0030 & 0.1895 & 0.6796 & -0.7081 & 0.0250 & -0.0107 & 0.0060 \\ 0.5017 & 0.2497 & 0.1742 & -0.1415 & -0.1098 & -0.6958 & -0.1678 & -0.3334 \\ 0.5630 & 0.1944 & -0.2746 & 0.0591 & -0.0167 & 0.1424 & 0.7187 & 0.1711 \\ 0.2449 & -0.1532 & -0.7472 & 0.2976 & 0.0937 & 0.0972 & -0.4082 & -0.2912 \\ 0.1750 & -0.6839 & -0.0567 & 0.0167 & -0.0098 & -0.4041 & -0.0328 & 0.5777 \end{bmatrix}.$$

Finally, by Formula $\mathrm{P} = \mathbf{XQX}^{\mathrm{T}}$, we obtain the permutation matrix P corresponding to the $\mathbf{Q}$ as follows: $\mathrm{P}_1 = \mathbf{XQ}_1\mathbf{X}^{\mathrm{T}} = \mathrm{I}$.

This shows that the graph $\mathbf{G}$ only has a trivial automorphism $\gamma = (1)$. We list the automorphism group into Figure. 2(b). □

5. Complexity Analysis

In this section, we analyze the time complexity of the method.

The time complexity of computing eigenvalues and eigenvectors is $O(n^3)$ [55]. Furthermore, Shroff's algorithm[56] can be implemented using $n^2/4$ processors, taking $O(n \log^2 n)$ time for random matrices. The time complexity of two $n \times n$ matrix multiplication usually is $O(n^3)$. If use the method of the paper, in the best case, a graph of order n has n distinct eigenvalues. After we have sorted all eigenvalues into three categories, by Formula $\mathrm{P} = \mathrm{XQX}^{\mathrm{T}}$ and Conjecture 1, we know that the operations for computing the automorphism group need only 0 or 1-type eigenvalues, which lead to the corresponding elements of the matrix Q only be 1 or −1. Meanwhile, the elements of the matrix Q corresponding to 2-type eigenvalues only are 1. When a graph has no duplicate eigenvalues, the matrix Q has a few elements involved in computing. As a result, the number of the matrix Q involved in $\mathrm{XQX}^{\mathrm{T}}$ is less. Thus, there rarely are the elements in the the automorphism group of the graph. Therefore, the total time complexity is $O(n^4)$. In the best case, the method can calculate the automorphism group in polynomial time. However, in the worst case, its time complexity is high. Reducing the time complexity is the future work.

6. Three Open Problems

In previous studying, we assume that a graph has no duplicate eigenvalues. Meanwhile, we already discuss how to calculate the automorphism group of a graph. However, this is not enough. In general occasions, a graph has many duplicate eigenvalues. In light of this, we ask the following questions:

1. How do ones use (4) to solve the corresponding elements of the orthogonal matrix Q and form an orthogonal matrix Q to solve the permutation matrix P by Formula $\mathrm{P} = \mathrm{XQX}^{\mathrm{T}}$?

2. How do ones compute each square block matrix B_s of order r_s in (17) where $s = 1, 2, \cdots, l$ and $\sum_{s=1}^{l} r_s = n$?

3. How do ones use (8) to compute the corresponding element in the orthogonal matrix Q?

7. Conclusions

In summary, we have obtained the following conclusions: Both from the theoretical and a practical point of view, the way is a novel method for solving the automorphisms of a weighted graph. When faced with a graph that has no duplicate eigenvalues, if Conjecture 1 is true, it can look for all automorphisms of a graph in polynomial time. Otherwise, this method has certain limitations and needs be improved. However, it has both theoretical and practical usefulness. In future studies, we will explore how to improve it.

Acknowledgements

The work described in this paper was supported by Key Project of the National Natural Science Foundation of China (No.91318301), National Natural Science Foundation of China (No.61202080), China Postdoctoral Science Foundation (No.2015M581032).

References

[1] B. McKay, “Computing automorphisms and canonical labellings of graphs Combinatorial Mathematics”, Lecture Notes in Mathematics, Springer Berlin / Heidelberg, vol. 686, **(1978)**, pp. 223–232.
[2] B. McKay, Practical graph isomorphism.
[3] J. Gela, “On the automorphism groups of mapping graphs Graph Theory”, Lecture Notes in Mathematics, Springer Berlin / Heidelberg, vol. 1018, **(1983)**, pp. 46–54.
[4] B. D. McKay, “Isomorph-free exhaustive generation”, Journal of Algorithms, vol. 26, no.2. **(1998)**, pp. 306–324.
[5] P. Cameron and Q. Mary, “Automorphisms of graphs”, Topics in algebraic graph theory, vol.102, **(2004)**, pp. 137.
[6] P. Kaski and P. J. Stergrd, “Representations and Isomorphism”, Algorithms and Computation in Mathematics, Springer Berlin Heidelberg, vol. 15, **(2006)**, Ch. 3, pp. 47–104.

[7] A. Lubiw, “Some np-complete problems similar to graph isomorphism”, Siam Journal on Computing, vol.10, **(1981)**, pp. 11.
[8] I. Krasikov, A. Lev and B. D. Thatte, “Upper bounds on the automorphism group of a graph”, Discrete Mathematics, vol. 256, no.12, **(2002)**, pp. 489–493.
[9] T. Nagoya and S. Toda, “Relating Complete and Partial Solution for Problems Similar to Graph Automorphism Mathematical Foundations of Computer Science 2007”, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, vol. 4708, **(2007)**, pp. 584–595.
[10] T. Nagoya and S. Toda, “Computational complexity of computing a partial solution for the graph automorphism problems”, Theoretical Computer Science, vol.410, no.2123, **(2009)**, pp. 2064–2071.
[11] J. R. Ullmann, “An algorithm for subgraph isomorphism”, J. ACM, vol.23, no.1, **(1976)**, pp. 31–42.
[12] L. P. Cordella, P. Foggia, C. Sansone and M. Vento, “An improved algorithm for matching large graphs, **(2001)**.
[13] C. Turcat and A. Verdillon, “Symmetry, automorphism, and test”, Computers, IEEE Transactions on C, vol.28, no. 4, **(1979)**, pp. 319–325.
[14] C. L. Chou, C. Y. Wang, G. W. Lee and J. Y. Jou, “S. IEEE computer, Graph automorphism-based algorithm for determining symmetric inputs”, Proceedings - Ieee International Conference on Computer Design, **(2004)**, pp. 417–419.
[15] S. Boyd, P. Diaconis, P. Parrilo and L. Xiao, “Fastest mixing markov chain on graphs with symmetries”.
[16] P. Eades and X. Lin, “Spring algorithms and symmetry”, Theoretical Computer Science, vol.240, no. 2, **(2000)**, pp. 379–405.
[17] P.T. Darga, K. A. Sakallah and I. L. Markov, “Faster symmetry discovery using sparsity of symmetries”, **(2008)**.
[18] H. Katebi, K. Sakallah, I. Markov, Graph symmetry detection and canonical labeling: Differences and synergies, Proc. Turing-100, Manchester, UK.
[19] T. Junttila and P. Kaski, “Engineering an Efficient Canonical Labeling Tool for Large and Sparse Graphs”, Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments and the Fourth Workshop on Analytic Algorithmics and Combinatorics, Siam, Philadelphia, **(2007)**.
[20] T.Junttila and P.Kaski, “Conflict Propagation and Component Recursion for Canonical Labeling Theory and Practice of Algorithms in(Computer) Systems”, Lecture Notes in Computer Science, Springer Berlin / Heidelberg, vol. 6595, **(2011)**, pp. 151–162.
[21] H. Katebi, K. Sakallah and I. Markov, “Symmetry and satisfiability: An update”, Theory and Applications of Satisfiability TestingSAT 2010, **(2010)**, pp. 113–127.
[22] H. Katebi, K. A. Sakallah and I. L. Markov, “Conflict anticipation in the search for graph automorphisms”, **(2012)**.
[23] T. Miyazaki, “The complexity of mckays canonical labeling algorithm”.
[24] G. Tener and N. Deo, “Efficient isomorphism of miyazaki graphs, algorithms”, vol.5, **(2008)**, pp. 7.
[25] A. Piperno, “Search space contraction in canonical labeling of graphs”, Arxiv preprint arXiv:0804, vol. 4881.
[26] C. C. Sims, “Computation with permutation groups”, **(1971)**.
[27] J. Leon, “Partition backtrack programs”, available at ftp://ftp.math.uic.edu/pub/leon/partn, **(2012)**.
[28] R. Frucht, “Herstellung von graphen mit vorgegebener abstrakter gruppe”, Compos. Math, vol.6, **(1938)**, pp. 239–250.
[29] L. Babai, “Finite digraphs with given regular automorphism groups”, Periodica Mathematica Hungarica, vol. 11, no.4, **(1980)**, pp. 257–270.
[30] L. Babai, “Automorphism groups of graphs and edge-contraction”, Discrete Mathematics, vol.306, no.1011, **(2006)**. pp. 918–922.
[31] L. aszl o Babai, “Automorphism groups, isomorphism, reconstruction (chapter 27 of the handbook of combinatorics)”.
[32] D. A. Spielman, “Faster isomorphism testing of strongly regular graphs” **(1996)**, pp. 2.
[33] M. Razinger, K. Balasubramanian and M. E. Munk, “Graph automorphism perception algorithms in computer-enhanced structure elucidation”, Journal of Chemical Information and Computer Sciences, vol. 33, no.2, **(1993)**, pp. 197–201.
[34] J. Presa, “Efficient algorithms for graph isomorphism testing”, Ph.D. thesis, **(2009)**, pp. 2.
[35] C. Godsil, “On the full automorphism group of a graph”, Combinatorica, vol.1, no.3, **(1981)**, pp. 243–256.
[36] J. Siemons, “Automorphism groups of graphs”, Archiv der Mathematik, vol.41, no. 4, **(1983)**, pp. 379–384.
[37] S. Bohanec, M. Perdih, “Symmetry of chemical structures: A novel method of graph automorphism group determination”, Journal of Chemical Information and Computer Sciences, vol.33, no.5, **(1993)**, pp. 719–726.
[38] A. Torgasev, “On the automorphism group of an infinite graph”, PUBLICATIONS DE L’INSTITUT MATHMATIQUE (BEOGRAD)(NS), vol.34, no.48, **(1983)**, pp. 233–237.
[39] R. Chang, W. Gasarch and J. Toran, “On finding the number of graph automorphisms”, **(1995)**.
[40] B. J. Jain and F. Wysotzki, “Automorphism partitioning with neural networks”, Neural Processing Letters, vol. 17, no.2, **(2003)**, pp. 205–215.
[41] C. Buchheim and M. Jnger, “Detecting symmetries by branch & cut”, Mathematical Programming, vol.98, no.1, **(2003)**, pp.369–384.
[42] A. R. Gilani and M. Faghani, “An algorithm for computing symmetry of fullerenes”, Optoelectronics and Advanced Materials-Rapid Communications, vol. 4, no.4, **(2010)**, pp. 556–559.

[43] M. Manjunath and V. Sharma, "Applications of dimensionality reduction and exponential sums to graph automorphism", Theoretical Computer Science, vol.412, no.29, **(2011)**, pp. 3639–3649.
[44] L. Babai, "Automorphism group and category of cospectral graphs", Acta Mathematica Hungarica, vol. 31, no.3, **(1978)**, pp. 295–306.
[45] Y. Teranishi, "Main eigenvalues and automorphisms of a graph", Linear and Multilinear Algebra, vol. 54, no.3, **(2006)**, pp. 211–217.
[46] Y. Teranishi, "Eigenvalues and automorphisms of a graph", Linear and Multilinear Algebra, vol.57, no.6, **(2009)**, pp. 577–585.
[47] G. Ruecker and C. Ruecker, "On using the adjacency matrix power method for perception of symmetry and for isomorphism testing of highly intricate graphs", Journal of Chemical Information and Computer Sciences, vo.31, no. 1, **(1991)**, pp. 123–126.
[48] S. Evdokimov, M. Karpinski and I. Ponomarenko, "On a new high dimensional weisfeiler-lehman algorithm", Journal of Algebraic Combinatorics, vol.10, no.1, **(1999)**, pp. 29–45.
[49] P. Dankelmann, D. Erwin, S. Mukwembi, B. G. Rodrigues, E. Mwambene and G. Sabidussi, "Automorphism group and diameter of a graph", Journal of Graph Theory, vol.70, no.1, **(2012)**, pp. 80–91.
[50] S. G. Hartke, H. Kolb, J. Nishikawa and D. Stolee, "Automorphism groups of a graph and a vertex-deleted subgraph", Electronic Journal of Combinatorics, vol. 17, no.1.
[51] L. Babai, D. Y. Grigoryev and D. M. Mount, "Isomorphism of graphs with bounded eigenvalue multiplicity", in: Proceedings of the fourteenth annual ACM symposium on Theory of computing, STOC '82, ACM, New York, NY, USA, **(1982)**, pp. 310–324. doi:10.1145/800070. 802206. URL http://doi.acm.org/10.1145/800070.802206
[52] A. Mowshowitz and V. Mitsou, "Entropy, Orbits, and Spectra of Graphs", Wiley-VCH Verlag GmbH & Co. KGaA, **(2009)**, pp. 1–22.
[53] C.-Y. Chao, "On groups and graphs", available at http://www.jstor.org/stable/1993974 **(1965)**.
[54] R. Beezer, A first course in linear algebra., available at http://linear.ups.edu/ **(2005)**.
[55] J. Demmel, O. Marques, B. Parlett and C. Vmel, "Performance and accuracy of lapack's symmetric tridiagonal eigensolvers", SIAM Journal on Scientific Computing, vol.30, no.3, **(2008)**, pp. 1508–1526.
[56] G. M. Shroff, "A parallel algorithm for the eigenvalues and eigenvectors of a general complex matrix", Numerische Mathematik, vol.58, no.1, **(1990)**, pp. 779–805.

Authors

HAO Jian-Qiang is a PhD student at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, China. His research interests include Graph Theory, Computational Geometry, Computer Graphics, and Software Testing.

GONG Yun-Zhan is a Professor and Doctoral Tutor at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, China. His research interests include Network and Service Test.

LIU Hong-Zhi is a Professor at the School of Computer Science and Information Engineering, Beijing Technology and Business University, China. His research interests include Software Engineering and Software Project Management.

