Dean&Francis

ISSN 2959-6157

The Counting of Isomorphism Classes of Mixed Graphs

Shengyu Cai[1], *

Yiqi Zheng[2] ,

Hanwen Feng[3]

[1]Department of Mathematical Physics, Xi'an Jiaotong Liverpool University, China, shengyu.cai23@student.xjtlu.edu.cn
[2]College of Science and Engineering, University of Minnesota Twin Cities, U.S.A., zhen0587@umn.edu
[3]School of Mathematics and Statistics, Central South University, China, f18996868801@126.com

*Corresponding author email: shengyu.cai23@student.xjtlu.edu.cn

Abstract:

Group theory and graph theory have important research value in mathematics today. Counting problems also play a decisive role in combinatorics. This paper introduces the number of isomorphism classes of mixed graphs with *n* vertices. The counting of them which using Burnside's lemma is solved by converting the cases of edges and vertices to some colors.

Keywords:-mixed graph isomorphism; Burnside's lemma; coloring

1. Introduction

Graph theory originated in the 18th century with Leonhard Euler's solution to the Königsberg bridge problem in 1736, laying the foundation for studying graph structures. In the early 20th century, mathematicians developed polynomial representations of graphs, such as graph polynomials, to better understand and classify different graph structures. The formal introduction of the graph isomorphism problem, which involves determining if two graphs are structurally identical, came in the mid-20th century, attracting significant attention. Researchers employed combinatorial counting methods, including Burnside's lemma, to tackle the problem of counting distinct isomorphism classes of graphs. Despite there are lots of advancement, this topic still remains some complex and active areas in research.

This paper aims to provide a detailed and explicit method to solve the number of isomorphic classes for mixed graphs with *n* vertices. In addition, generalizing the existing results which about the counting of the isomorphism classes of simple graphs, and improving the practicability of this conclusion. The significance of this paper is to provide a general conclusion about this topic.

This paper focuses on how to find the number of isomorphism classes of mixed graphs with *n* vertices. In the intersection of group theory and graph theory, it has been a focus on research. However, there are many challenges to solving this problem, including the abstraction of n-vertex graphs, the complexity of

the structure, and how to avoid re-computing isomorphic graphs.
By thinking about the double calculation, this paper draws on the treatment of the coloring method, which constructs a complete graph with *n* vertices and establishes a bijection between edges or vertices and colors. In addition, using the Burnside's lemma to eliminate the duplicate cases.

2. Related Works

With the development of the research, Huahai He and Ambuj K. Singh (2008) explored the query languages and access methods for graph databases, and this work provided a new perspective on the graph isomorphism problem. Meanwhile, Jenny Jin (2018) focused on the analysis and application of Burnside's lemma, giving not only creative thoughts on chemistry and music theory, but also definitions of group theory and so on.
The newest research, such as Learning to Count Isomorphisms with Graph Neural Networks [Xingtong Yu, Zemin Liu, Yuan Fang, Xinming Zhang] (2023), proposes a novel Graph Neural Network (GNN) model, known as Count-GNN, specifically designed for the task of subgraph isomorphism counting. The model employs an edge-centric message passing mechanism, which retains fine-grained structural information by propagating and aggregating messages at the edge level. This has offered new direction of research.
Although there are abundant previous researches, this topic still has some distinct limitations. For instance, it cannot be directly applied in terms of infinite groups or infinite sets. In addition, Burnside's lemma mainly concentrates on the counting of the orbits, it means that may not applicable to some problems which need more detailed analysis in symmetry. Furthermore, calculating the set of fixed points for each group element will be very complicated in some cases, especially when the number of group elements is large or the action of the group is very intricate.

3. Preliminary

3.1 Burnside's lemma

If G is a finite group acting on the set X, for each g in G, let X^g represents the stationary elements, namely the fixed points in X under the action of g. Then the number of orbits (denoted $|X/G|$) is given by the following formula:

$$|X/G| = \frac{1}{|G|}\sum_{g\in G}|X^g|$$

3.2 Mixed Graph

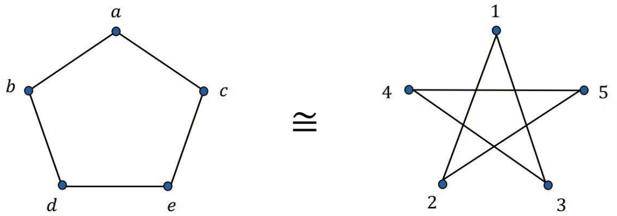
A mixed graph M can be defined as $M=(V,E,A)$ such that:
(a) V is a set of vertices,
(b) E is a set of undirected edges,
(c) A is a set of directed edges (or arcs).
It means that M may contains unordered edges, ordered edges, multiple edges and self loops.

3.3 Complete graph

A complete graph $K_n=(V,E)$ with n vertices is a graph such that:
(a) V is a set of vertices with $|V|=n$,
(b) E is a set of edges where each edge is an unordered pair of distinct vertices from V,
(c) For every pair of distinct vertices $u,v\in V$, there is one edge $\{u,v\}\in E$.
In other words, a complete graph is a simple undirected graph in which each pair of distinct vertices is connected by exactly one edge.

3.4 Graph isomorphism

An isomorphism of graphs G and H is a bijection f between the vertex sets of G and H, namely $f:V(G)\to V(H)$, such that any two vertices u and v are adjacent in G if and only if $f(u)$ and $f(v)$ are adjacent in H.



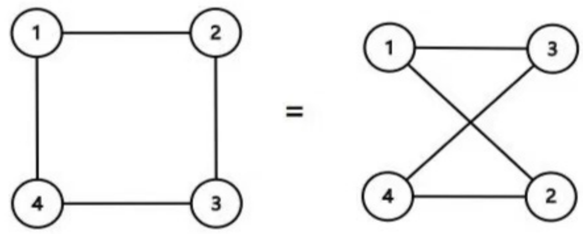
$$f:\ a\mapsto 1, b\mapsto 2, c\mapsto 3, d\mapsto 5, e\mapsto 4$$

4. Methodology

This part will study the isomorphism classes of mixed graphs with *n* vertices, at the same time, using the coloring method to solve this problem.

ISSN 2959-6157

Constructing a set *X* which consists of all mixed graphs with *n* vertices that match the conditions, and labeling these *n* vertices with *1, 2,…,n*. Since for any one graph, it remains isomorphic after only permuting the labels of its vertices, thus choosing the symmetric group *Sn* of order *n*! acts on the set *X*. They lay the groundwork for using Burnside's lemma to eliminate duplicates later.



Listing all possible cases of edges between any two vertices in the mixed graph, if there are *p* different cases listed, selecting *p* different colors for them to form bijection. Listing all possible cases of self loops formed by each vertex in the mixed graph, if there are *q* different cases listed, selecting *q* different colors for them to form bijection as well. Finally, building a complete graph with vertices labeled *1, 2,…,n*, coloring both the edges and vertices of it in order to obtain all elements in the set *X*.

Randomly selecting a permutation *g* from *Sn* for study, and coloring the fixed points corresponding to permutation *g* in the complete graph. It is noted that some edges need to be colored with the same color, as do some vertices, in order to ensure that the complete graph remains unchanged after being acted upon by permutation *g*, considering these edges are equivalent and belong to the same equivalence class, the same as vertices. Therefore, assuming there are *x* equivalence classes of edges and *y* equivalence classes of vertices for the permutation *g*, according to the multiplication principle in combinatorics, there are $p^x \cdot q^y$ fixed points for permutation *g* in the set *X*. This means that:

$$\left|X^g\right| = p^x \cdot q^y$$

Since any permutation can be decomposed into several disjoint cycles multiplied together. Thus, breaking down the studied permutation *g* into *m* disjoint cycles with lengths *l1*, *l2*,…, *lm* respectively.

For vertices, only need to ensure that the vertices which contained in each decomposed permutation

are colored with the same color, because it can make these *n* vertices to remain unchanged as a whole after being acted by permutation *g*. Hence, the number of equivalence classes for vertices can be determined, namely:

$$y = m$$

As for edges, further classification is required. The edges of a complete graph are divided into two categories based on their endpoints: one category consists of edges whose two vertices belong within the same decomposition, and the another category consists of edges whose two vertices do not belong within the same decomposition.

In regards to the first case mentioned above, let's assume one decomposition has a length *l*, since it affects exactly *l* vertices in the process of its action, only $\binom{l}{2}$ edges will undergo transformation as a result. To facilitate analysis, rearranging these *l* vertices into a regular polygon, which satisfies the effect of this decomposed permutation is to rotating this regular polygon once. In such cases, it is evident that two edges are equivalent if and only if they have equal lengths. Furthermore, because there are a total of $\left\lfloor \frac{n}{2} \right\rfloor$ different lengths of edges in the regular n-gon, they correspond to $\left\lfloor \frac{n}{2} \right\rfloor$ equivalence classes. Since this rearrangement make the graph preserves isomorphism, thus the equivalent edges maintain their equivalence relationship before and after it. Therefore, this type of edge corresponds to a total of:

$$\sum_{i=1}^{m} \left\lfloor \frac{l_i}{2} \right\rfloor$$

equivalence classes.

The second case is slightly different as the two vertices of these edges are distributed among two permutations. Hence, selecting any two permutations from the decomposition of permutation *g* for analysis. Let's assume that these two permutations have lengths *l* and *l'* respectively, which affect *l·l'* edges in total. By selecting an arbitrary edge from them and considering that the order of their product is $lcm(l,l')$, it only needs to be acted upon $lcm(l,l')$ times to return to its original position. Coloring each edge which it passes through during this transformation process with the same color, thus they will remain unchanged as a whole after being acted upon by these two permutations once, belonging to the same equivalence class, then size of this equivalence class is $lcm(l,l')$. Obviously in this case, any two equivalence classes have equal sizes. Therefore, these two permutations correspond to a total of $\frac{l \cdot l'}{lcm(l,l')} = \gcd(l,l')$ equivalent classes. Thus, this type of edge corresponds to:

$$\sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \gcd(l_i, l_j)$$

equivalent classes in total. In particular, if *m* = 1, then this

kind of edge obviously does not exist, so this part is equal to zero.
Consequently, the total number of equivalent classes for edges can be obtained, namely:

$$x = \sum_{i=1}^{m} \left\lfloor \frac{l_i}{2} \right\rfloor + \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \gcd(l_i, l_j)$$

By combining Burnside's lemma with our previous analysis mentioned above. This allows to derive the final result, which is the number of isomorphism classes for mixed graphs with n vertices, namely:

$$|X / S_n| = \frac{1}{n!} \sum_{g \in S_n} p^x \cdot q^y$$

$$x = \sum_{i=1}^{m} \left\lfloor \frac{l_i}{2} \right\rfloor + \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \gcd(l_i, l_j)$$

$$y = m$$

There is a concrete example about calculating the number of isomorphism classes of a mixed graph with three vertices. Suppose the conditions are: three different cases between two vertices (p = 3); two different cases on one vertex (q = 2). Choosing $S3$ acts on the set X which consists of all mixed graphs with three vertices that match the conditions. Using the above formulas, the following table can be obtained, namely:

	x	y
(1)(2)(3)	$0+3\cdot\gcd(1,1)=3$	3
(12)(3) (13)(2) (23)(1)	$(1+0)+\gcd(2,1)=2$	2
(123) (132)	$1+0=1$	1

Therefore, there are:

$$|X / S_3| = \frac{1}{3!} \cdot \left(3^3 \cdot 2^3 + 3 \cdot 3^2 \cdot 2^2 + 2 \cdot 3 \cdot 2\right) = 56$$

isomorphism classes of three vertices mixed graphs.

5. Applications

5.1 Chemical field

In chemistry, molecules can be represented by mixed graphs. Atoms are the vertices of the mixed graph, and chemical bonds are the edges. Different molecules may have the same chemical properties, but their structures may be different. Through mixed graph isomorphism class counting, the number of different structures of molecules with the same chemical properties can be determined. For example, in the study of some isomers, isomers are compounds with the same molecular formula but different structures. Through mixed graph isomorphism class counting, the number of isomers of a given molecular formula can be determined, helping chemists better understand the relationship between the structure and properties of molecules. Taking hexane as an example, its molecular formula is $C6H14$. Through mixed graph isomorphism class counting, it can be determined that there are five isomers of n-hexane, namely n-hexane, 2-methylpentane, 3-methylpentane, 2,2-dimethylbutane, and 2,3-dimethylbutane. These isomers have the same molecular formula, but their molecular structures are different, resulting in different physical and chemical properties.

In drug design, researchers need to understand the interaction between drug molecules and target molecules in the organism (such as proteins, nucleic acids, etc.). These molecules can be represented by mixed graphs. Through mixed graph isomorphism class counting, the number of different drug molecule structures can be determined, helping researchers design more effective drugs. For example, researchers can determine the number of different structural isomers of drug molecules with specific biological activities through mixed graph isomorphism class counting, and then screen out the most promising drug molecules for further research and development. Suppose researchers are looking for a drug molecule that can inhibit the activity of a specific enzyme. They can determine the number of drug molecules with similar structures through mixed graph isomorphism class counting, and then further optimize and screen these molecules to improve the efficacy and safety of drugs.

5.2 Computer science field

In computer networks, networks can be represented by mixed graphs. Nodes are devices in the network, and edges are connections between devices. Through mixed graph isomorphism class counting, the number of networks with the same topological structure can be determined, helping network administrators better understand the structure and performance of the network. For example, for an enterprise's internal network, through mixed graph

ISSN 2959-6157

isomorphism class counting, the number of networks with the same topological structure can be determined, helping network administrators evaluate the reliability and security of the network. If multiple networks have the same topological structure, then when performing network upgrades or maintenance, one of the networks can be selected for testing, and then the test results can be applied to other networks, thereby improving work efficiency. Taking a simple local area network as an example, it consists of several computers and a server. Through mixed graph isomorphism class counting, the number of local area networks with the same topological structure can be determined. If multiple local area networks have the same topological structure, then when troubleshooting the network, one of the networks can be selected for testing, and then the test results can be applied to quickly locate and solve problems.

In graphic recognition, images can be represented by mixed graphs. Pixels are the vertices of the mixed graph, and the relationships between pixels are the edges. Through mixed graph isomorphism class counting, the number of images with the same shape can be determined, helping computers better recognize images. For example, for a handwritten digit recognition system, through mixed graph isomorphism class counting, the number of digital images with the same shape can be determined, and then these images can be classified and recognized. If multiple digital images have the same shape, they can be classified into one category, thereby improving the accuracy of digit recognition. Suppose you want to recognize the handwritten number “5”. The image of the number “5” can be represented by a mixed graph. Then, through mixed graph isomorphism class counting, the number of digital “5” images with the same shape can be determined. If multiple digital “5” images have the same shape, they can be classified into one category, and then feature extraction and classification can be performed on this category of images to improve the accuracy of digit recognition.

6. Conclusion

In this mathematical study of the number of isomorphism classes of mixed graphs, this paper explores in depth how to effectively calculate it. In the process of using coloring method, innovatively converting the different kinds of edges between two vertices and the different kinds of self loops on one vertex to the different colors of edges and vertices in the complete graph respectively, then according to the Burnside’s lemma to draw this important result. It not only enriched the existing mathematical theory system, but also provides a strong support for practical application, these illustrate its significance.

However, there are still some shortcomings in this study. For example, in the process of solving practical problems, the simplicity of this result needs to be improved. Future work can focus on optimizing it so that it can be more efficiently used in relevant application scenarios.

To sum up, this study has made some remarkable breakthroughs in calculating the number of isomorphism classes of the graph, but there are still many spaces for development.

References

[1] Armstrong, M. A. (Mark A. (1988). Groups and symmetry. Springer-Verlag.

[2] Babai, L. (2018). Groups, Graphs, Algorithms: The Graph Isomorphism Problem. http://www.people.cs.uchicago.edu/~laci/papers/icm18-babai.pdf .

[3] Bell, M. V. (n.d.). P´olya’s enumeration theorem and its applications. https://helda.helsinki.fi/bitstream/handle/10138/159032/GraduTiivistelma.pdf?sequence=3

[4] Bogart, K. P. (1991). An obvious proof of Burnside’s lemma. American Mathematical Monthly via American Mathematical Monthly.

[5] Dummit, D. S., & Foote, R. M. (2004). Abstract algebra (Third edition.). John Wiley and

Sons, Inc.

[6] Fripertinger, H. (1993). Enumeration in musical theory (3.Aufl., Vol. 1). Institut für Elektronische Musik an der Hochschule für Musik und Darstellende Kunst.

[7] Hao, T. (2023). Burnside’s Lemma and Its Applications in Combinatorics Problems.

Highlights in Science, Engineering and Technology, 47, 126–130.

https://doi.org/10.54097/hset.v47i.8175

[8] Wright, E. M. (1981). Burnside’s lemma: A historical note. J. Comb. Theory B via J. Comb. Theory B.

[9] Yu, X., Liu, Z., Fang, Y., & Zhang, X. (2023). Learning to count isomorphisms with

Graph Neural Networks. Proceedings of the AAAI Conference on Artificial Intelligence, 37(4), 4845–4853.

https://doi.org/10.1609/aaai.v37i4.25610

[10] Zabrocky, M. (n.d.). CHAPTER 6: POLYA ENUMERATION. Yorku. https://garsia.math.yorku.ca/~zabrocki/math4160f19/notes/ch6_polya.pdf