International Journal of Engineering Sciences Paradigms and Researches (IJESPR)
Vol. 48, Issue 02, Quarter 02 (April-May-June 2019)
(An Indexed, Referred and Impact Factor Journal)
ISSN (Online): 2319-6564
www.ijesonline.com

Identifying Novel Graph Properties and Solving Graph Isomorphism for Hard Instances of Symmetrical Graphs

Suchismita Mahato[1], Saurav Kumar Jindal [1], Vicky Kumar[1], Surinder Singh Khurana[2] and Mahesh Kulharia[1]

[1]Department of Computational Sciences, Central University of Punjab, Bathinda-151001, India
[2]Department of Computer Science and Technologys, Central University of Punjab, Bathinda-151001, India

Publishing Date: May 04, 2019

Abstract

Applications of graph theory are ubiquitous however, many unsolved problems cap its utility. One of these is to say whether two graphs are isomorphic or not by using graph invariants within polynomial time. It is extremely difficult as the quest for a complete graph invariant which can distinguish graphs uniquely and is still easy to calculate is not complete. Here, in this paper, we discuss some of the important existing graph invariants and show how they fail in the case of regular graphs. Also, we have discovered some new graph invariants (First Order Cyclical Shapes, First Order Participation Number, Second Order Cyclical Shapes, Second Order Participation Number and Neighbourhood Cluster Analysis) that can be used as complete graph invariants for highly symmetrical graphs. The complexity to calculate our proposed invariants increases merely in polynomial manner with respect to the input size of graph. These invariants can act as stepping stones to test for graph isomorphism. A method to shift the runtime complexity of graph isomorphism from non-polynomial space to polynomial space would have wide ranging effects. Such methods will eventually develop significant advancements in the field of cryptography, automata theory, compilers, image processing etc. and in future can also be modified to develop a solution for complete Graph Isomorphism problem in polynomial time.

Keywords: graph theory, graph isomorphism, Generalised Johnson graphs, graph invariant, strongly regular graphs.

1. Introduction

A graph is a mathematical structure which is used to model pairwise relations between objects. In simpler terms it is an ordered pair comprising of a set of nodes together with a set of edges. Any structure or pattern (be it any building or location or planetary bodies, even the entire universe) can be represented by a graph, therefore, understanding and comparing graph properties has diverse applications, ranging from computer science, chemistry, linguistics, psychology, sociology, etc[1]. Many research groups have tried to study graph properties like connectivity, regularity, symmetry etc. to understand graphs invariants such as colouring, planarity, chromaticity, etc. Graph properties [2] depend only on the structure of the graph and are independent of any labellings or drawings. Graph properties are generally singular boolean values or function of the graph which determine a single domain, for example, connected graphs, regular graphs, bipartite graphs etc. Graph invariant, on the other hand, is a combination of graph properties which result from the overall structure of the graph (for example; eccentricity, radius, diameter, girth etc). The values for the graph invariants represent a broader class of values such as boolean values, integer values, real numbers, sequence of numbers etc. A complete graph invariant is one which successfully and uniquely identifies isomorphisms for two graphs[3]. However, the solution to seemingly trivial problem of graph isomorphism has forever remained a tantalizing mystery. Despite many efforts, the complexity of solving the graph isomorphism has remained a problem of exponential nature. In simpler terms, this makes it next to impossible to identify whether a given graph is identical to another. Graph invariants are significant in obtaining faster computational speed for determining graph isomorphism/non-isomorphism.

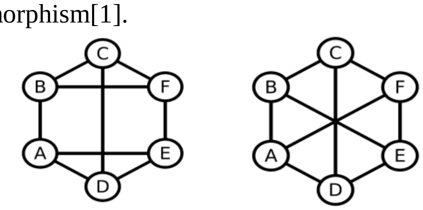
The problem of Graph Isomorphism (GI)[4] is to check whether two given finite graphs are structurally different or one is just a perturbed variant of the other. Two graphs are said to be isomorphic if

- Both have same number of vertices, edges and degree sequence (for an undirected graph is the decreasing order of the degrees of the vertices of the graph); and
- There is a bijective mapping between the vertices of one graph to the vertices of the other graph such that the edge connectivity is maintained.

GI is a classical problem in the theory of computation because of its unresolved complexity status within the polynomial/non-deterministic polynomial (P/NP) space[5]. No efficient method for GI has yet been designed or created with Polynomial (P) complexity[6]. It is the second last of the twelve problems [7] (Graph Isomorphism, Subgraph homeomorphism, Graph genus, Chordal graph completion, Chromatic index, Spanning tree parity problem, Partial order dimension, Precedence constrained 3-processor scheduling, Linear programming, Total unimodularity, Composite number, Minimum length triangulation) whose complexity remains elusive, the last being integer factorization. (Prof. Laszlo Babai is given credit for bringing down the time complexity of GI from sub-exponential to quasi-polynomial time, but it is still not polynomial [6]). The P vs NP problem is one of the biggest problems in theoretical computer science which questions whether the solution to every problem can be verified (NP) as well as solved (P) in polynomial time. It is one of the seven Millenium Prize problems for which Clay Mathematics Institute has announced US $1,000,000 prize for the first solution[8]. Since GI is a NP problem, a polynomial time solution to it would be a vital progression towards solving the P vs NP problem.
The first workable solution of GI was given by McKay[5] in the form of a tool called nautY whose complexity was exponential ($O(2^{\sqrt{n}})$). nautY worked on the principles of colouring vertices of graphs. Initially, nautY colours the vertices of a graph with a single colour and then refines this colouring using a particular vertex invariant. nautY could determine if two graphs are isomorphic by checking whether their canonical labelling were identical after the colouring refinement. Babai & Luks [9] then adopted a better approach by declaring a sub-exponential time algorithm for the graph isomorphism by following an algebraic approach. The time complexity for their algorithm was $2^{O\sqrt{n\log n}}$. Saucy algorithm [10] is an updated solution for graph isomorphism but it does not deal with approximate symmetries. Bliss[11] was designed for computing automorphic groups and canonical labelling of graphs. Recently, GI has been reduced to quasi-polynomial time complexity by Prof. Laszlo Babai[6]. However, his solution is not extendible to extremely symmetric Johnson graphs. All these algorithms use the approach of differential colouring of nodes. Most of these groups have focused on partitioning of the vertices[10] on the basis of colouring[5] according to their connections for solving GI. These methods, though suited for most classes of graphs, fail in the case of highly symmetric graphs[6]. In these types of graphs, the vertices cannot be distinguished on the basis of connectivity because all of them are topographically equal. Hence, a method is required to find a new perspective to solve these kinds of graphs by making use of their symmetry rather than attempting to look for elusive differences. Since the quest to identify a solution to GI with polynomial time and space complexity is still on, we have identified novel properties of graph and have extended them to obtain a highly deterministic invariant.

1.1 Conventional Invariants

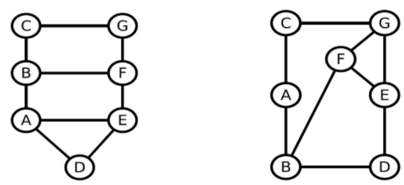
There are numerous graph invariants out of which a few can solve graph isomorphism. A graph invariant is strong only if it can, identify two non-isomorphic graphs as non-isomorphic. For instance, vertex count (number of vertices of a graph) and degree sequence (for an undirected graph is the decreasing order of the degrees of the vertices of the graph) are weak invariants as they are not efficient discriminators for graph isomorphism[1].



(a) Graph I (b) Graph II

Fig 1: Regular graphs. The degree sequence of Fig 1a and 1b are identical (A_3, B_3, C_3, D_3, E_3, F_3, G_3) even though these graphs are not isomorphic to each other.

Eccentricity[12] (maximum geodesic distance between a vertex to all other vertices), **girth[**13] (the number of edges in the shortest cycle of a graph) and **chromatic number**[14] (the smallest number of colors needed to color the vertices of G so that no two adjacent vertices share the same color) can be considered as moderate graph invariants. Eccentricity and girth are easy to compute. However, there are many instances of graphs in which these invariants alone cannot determine whether two graphs are isomorphic or not (See Fig 2a, 2b and Table 1a and 1b).



Graph III **(b) Graph IV**

Fig 2: Irregular graphs. The degree sequence of Fig 2a and Fig 2b are the same (A_3, B_3, E_3, F_3, C_2, D_2, G_2). The girth of graphs in Fig 2a and Fig 2b is 3. But these graphs are not isomorphic to each other.

The eccentricity of all the vertices of the graphs in Fig 1(a) and Fig 1(b) are mentioned in Table 1(a) and 1(b).

Table 1: Eccentricity of Graph I in (a) and Graph II in (b). Even though the eccentricity is same for both graphs I and II, these graphs are not isomorphic to each other. They are regular graphs (vertices having same degrees) having equal degree sequence.

(a)

Vertex	Eccentricity
A	2
B	2
C	2
D	2
E	2
F	2
G	2

(b)

Vertex	Eccentricity
A	2
B	2
C	2
D	2
E	2
F	2
G	2

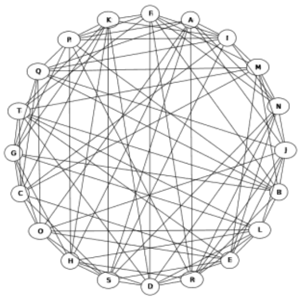
Chromatic number (or Graph vertex colouring), is not computable in polynomial time (across all sets of graphs) as the complexity to calculate it is NP-Complete[15].

1.2 Hard Instances of Symmetrical graphs

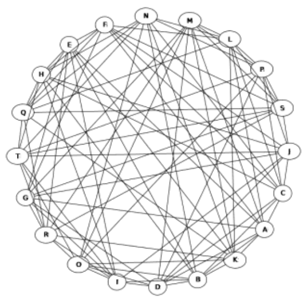
The cyclical shapes (CS) can be classified on the basis of the increasing size. There are some regular graphs for which any combination of graph invariants fail to uniquely identify isomorphic graphs. Two examples of such hard instances of symmetrical graphs are:

1.2.1 Generalised Johnson Graphs

The graphs in Fig 3(a) and Fig 3(b) are called **Generalised Johnson graphs[**17]. These are highly symmetrical graphs characterized by peculiar structural properties which are derived from a set of "discrete and equivalent entities" by the application of a system of "subsets sampling". Generalised Johnson graphs consists of 3 variables n, k and i and is denoted by GJ(n;k; i) where n represents the total number of elements of the set, k represents the size of the subset and i represents the intersection number between any two subsets. Two vertices are said to be connected in a Generalised Johnson graph if their intersection has a size of i set-elements. The value of i ranges from 0 to $k-1$. When the value of i is 0 it results in a special type of graph called Kneser graph[18] (Additional Information Fig 9(a)). When the value of i is $k-1$ it results in Johnson graphs (Additional Information Fig 9(b)). Kneser graphs and Johnson graphs are complementary to each other i.e. union of Kneser and Johnson graph, of same dimensions, result in the formation of complete graph (Additional Information Fig 9(c)). Johnson graph are also highly symmetrical distance transitive graphs[19] (a distance-transitive graph is a graph such that, if a pair of vertices v and w are at any distance d, and any other pair of vertices x and y are also at the same distance, then v and w can be mapped onto x and y). For example, in a Johnson graph (say J(4,2)) as shown in Fig 4, n=4 and k=2. For a graph to be a Johnson graph, the value of i is always equal to $k-1$. Hence, in this example i=1. As n represents the total number of elements of the set, the elements of the set S are 1,2,3,4 and the size of the subset is 2. Therefore, the combinations of subsets taken would be from S whose total size would be 2. Hence, the possible combinations of subset are (1,2), (1,3), (1,4), (2,3), (2,4), (3,4). These are the vertices of the graph. Therefore, the number of vertices for the Johnson graph J(4,2) are 6. Two vertices are said to be connected in a Johnson graph if their intersection size is $k-1$. Therefore, in this example, the vertices which have only 1 common element in the subset are connected to each other. As shown in Fig 4, (1,2) is connected to (2,3) because it shares only 1 common element→2. Likewise, it is connected to (1,3), (1,4) and (2,4). (1,2) is not connected to (3,4) because it does not share any common element with each other.



(a) Graph V: GJ(6,3,1).



(b) Graph VI: GJ(6,3,2).

Fig 3: Generalised Johnson graphs. The degree sequence of both graphs V and VI are (A_9, B_9, C_9, D_9, E_9, F_9, G_9, H_9, I_9, J_9, K_9, L_9, M_9, N_9, O_9, P_9, Q_9, R_9, S_9, T_9).The girth of both graphs is 3 and the chromatic number is 6[16]. The eccentricity of graphs V and VI are also same, i.e. 3 for every vertex. Now, if these graph invariants are combined, it can be concluded that graphs V and VI are isomorphic to each other, which is actually true. But the existing graph invariants alone cannot determine its validity.

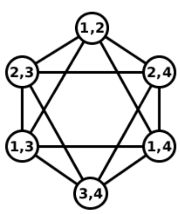


Fig 4: Johnson Graph J(4,2).

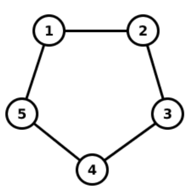
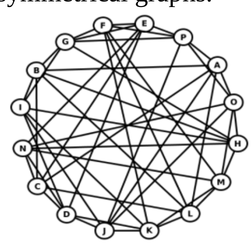


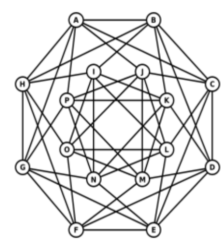
Fig 5: Strongly Regular Graph SRG(5,2,0,1)

1.2.2 Strongly Regular Graphs

A Strongly Regular graph[20] is represented as SRG(v,d,λ,μ) and these parameters denote the total number of vertices (v), degrees of each vertex (d), common neighbours for adjacent vertices (λ) and common neighbours for non-adjacent vertices (μ)[21]. Let us take an example to understand a Strongly Regular Graph SRG(5,2,0,1) as shown in Fig 5. Here, v=5, d=2, λ =0 and μ =1. Vertices 1 and 2 (adjacent) have 0 common neighbours and vertices 1 and 3 (non-adjacent) have 1 common neighbor→2. This is same for every pair of adjacent and non-adjacent vertices in the given graph[22]. It is to be noted that not all Strongly Regular graphs are symmetrical graphs[23] (for example STS). A Steiner Triple System[24], denoted by STS(n), is a pair (S,T) comprising a set S with n elements, and a set T comprising triples of S called blocks such that every pair of elements of S appear together in a unique triple of T. These graphs (Figs 3(a), 3(b), 6(a) and 6(b)) are epitome of symmetry and we have identified novel graph properties on the basis of which novel graph invariants are also discovered that can determine graph isomorphism for such hard instances of symmetrical graphs.



(a) Graph VII.



(b) Graph VIII.

Fig 6: Strongly Regular graphs SRG(16,6,2,2). The degree sequence of both graphs VII (Lattice(4,4) Graph) and VIII (Shrikhande Graph)are (A6, B6, C6, D6, E6, F6, G6, H6, I6, J6, K6, L6, M6, N6, O6, P6). The girth of both graphs is 3 and the chromatic number is 4. The eccentricity of graphs VII and VIII are also same, i.e. 2 for every vertex. Therefore, combining these graph invariants, it may seem that graphs VII and VIII are isomorphic to each other, which is not true. These two graphs are non-isomorphic forms and again, this underlines the complex nature of graph isomorphism.

1.3 Novel Graph Properties on the basis of Cyclical Shapes

In order to identify novel invariants, we examined the relationship between the cyclical shapes and structural organisation of graphs. In the past too some research groups have attempted to study the arrangement of CS constituting the graphs[25].

1.3.1 First Order Cyclical Shape

The First Order Cyclical Shape (FCS) of each vertex is the minimum number of edges involved in forming the shortest cycle for that vertex. It may or may not be equal to the girth of the graph since girth is a global property and FCS is confined to every vertex of the graph. Girth is, therefore, the minimum value of the FCS calculated for all the vertices of the graph. The set of FCS sequence arranged in decreasing order can be a very important graph invariant because for maximum classes of graph, this set comes out to be unique. Let us understand FCS with the help of J(4,2) in Fig 7.

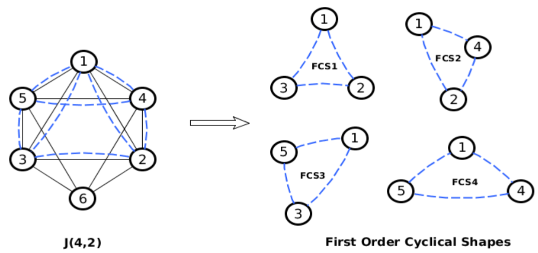


Fig 7: First Order Cyclical Shapes of J(4,2). Vertex 1 has 4 FCS of size 3. They are as follows: 1-5-4, 1-4-2, 1-2-3 and 1-5-3. This way, the FCS is calculated for all the vertices of the graph. The total number of FCS for J(4,2) are 8 and they are: 1-5-4, 1-4-2, 1-2-3, 1-5-3, 2-3-6, 2-4-6, 3-5-6 and 4-5-6.

1.3.2 Second Order Cyclical Shape

Second Order Cyclical Shapes (SCS) are vertices connected in closed chain, whose length is atleast one more than the FCS, such that there is no FCS completely encompassed within. A cyclical shape would be considered as SCS if and only if it satisfies the following conditions:

- The length of the cyclical shape should be atleast equal to 1 more than the size of FCS and should not be more than twice the size of FCS.
- The cyclical shape should not encompass any of the FCS.

The concept of shape encompassment is pivotal for the utility of SCS property (See Additional Information Fig 11 SCS example for simple representation of encompassment). A graph may also be considered as a compound structure constructed of the building blocks of small shapes (FCS). The different possible arrangements of these FCSs can lead to emergence of different types of SCS, thus resulting in the formation of different graphs. For making it more clear, let us consider an example. Suppose a vertex (Say, vertex 1 in Fig 7 J(4,2)) is involved in the formation of 4 different FCS (of length 3) and 5 SCS (of length 4). Out of these 5 SCS, since 3 contain the FCS within, these (1-5-6-4; 1-5-3-2 and 1-3-2-4) will not qualify as valid SCS whose number would thus be reduced to 2. These (1-5-6-4; 1-5-3-2 and 1-3-2-4) are rejected as they encompass atleast one FCS within. The shape encompassment can have two different broader definitions:

- A SCS is termed as encompassing a FCS within it, if all of the vertices belonging to any of the FCS are also part of SCS.
- A SCS is termed as encompassing a FCS within it, if "all but one of the edges" belonging to any of the FCS are also part of SCS.

Encompassment is used for selecting SCS because if encompassment is not checked, the set of SCS will have redundant information which is already identified in the set of FCS. This information will not be able to differentiate between vertices as it will contain the total information which has already been counted in the FCS. To avoid such redundancy, encompassment is introduced. Although both the definitions are appropriate, there are better results in the case of latter definition (checking edge involvement in the formation of SCS). This is because, when checking for vertices, the orders in which the vertices appear play an important role in short-listing the SCS. In a case where only the nodes involved in a FCS are part of a SCS and not the edges, the SCS will not be short-listed under the first definition. This property is relaxed in the case of checking for edge involvement (Fig 8) and thus it makes the second definition more permissive and less stringent without compromising the "non-redundant information" aspect of the property. This case is clearly evident in the case of Lattice(4,4) and Shrikhande (Fig 6(a) graph VII and Fig 6(b) graph VIII) as even though both graphs have the same size of FCS, SCS yet the number of SCS per vertex is very different.

Let us understand SCS with the help of J(4,2) in Fig 8.

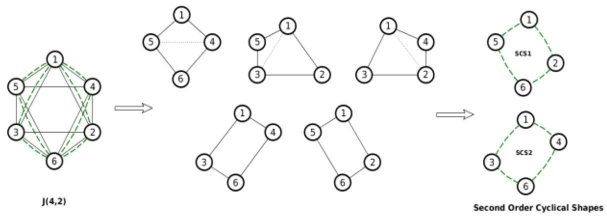


Fig 8: Second Order Cyclical Shapes of J(4,2). Vertex 1 has quadrilaterals 1-2-3-5, 1-3-2-4, 1-3-6-4, 1-2-6-5 and 1-4-6-5. Since 1-2-3-5 contain FCS 1-2-3 and 1-3-5, 1-3-2-4 contain FCS 1-2-3 and 1-2-4, and 1-4-6-5 contains FCS 4-5-6 and 1-4-5, therefore they do not qualify for SCS. Hence, the only SCS for Vertex 1 are 1-3-6-4 and 1-2-6-5 since they do not contain any FCS. This way, the SCS is calculated for all the vertices of the graph. The total number of SCS for J(4,2) are 3 and they are: 1-3-6-4, 2-3-5-4 and 1-2-6-5.

1.4 Graph Invariants on the basis of cyclical shapes

1.4.1 Size of FCS

The size of the FCS for every vertex can act as a graph invariant. Although it is clear that within symmetrical graphs, the size of FCS would always be identical for every vertex, however, it would become very useful for majority of regular graphs where it can vary for every vertex. For instance, in the case of Miyazaki graph[26] M4, the size of the FCS varies from 3 to 6 (Additional Information Fig 10). It is also important to mention here that although the size of FCS is not a complete graph invariant, it can be useful in augmenting principally different other invariants such as eccentricity.

1.4.2 First Order Participation Number (FPN)

The FPN for a vertex is the total count of FCS that it participates in. It is a measure of denseness of the shapes around a given vertex. Interestingly, a regular graph may have multiple zones of varying local structure and the variability in the FPN can become useful in identifying these zones. These differences in FPN arise due to the differential extent of edge sharing as it directly represents local organisation in a graph. FPN can vary for different vertices in a single graph, but in the case of hard instances of symmetrical graphs, they are identical for every vertex. For example the FPN for every vertex of J(4,2) is 4 as mentioned previously in Fig 7.

1.4.3 Size of SCS

The size of the SCS for every vertex is also another graph invariant. Similar to the "size of FCS", it is also a constant among the vertices of symmetrical graphs and can vary for every vertex for some regular graphs (for example, in Miyazaki graph[26] M4 (shown in Additional Information Fig 10), the size of SCS varies from 5 to 10 for the vertices of the graph).
Being a moderate graph invariant, the size of SCS can be combined with other graph invariants (such as size of FCS, FPN etc.) to form a combined complete graph invariant.

1.4.4 Second Order Participation Number (SPN)

The total number of SCS participating in a vertex is the Second Order Participation Number (SPN) of that vertex. For a graph, the SPN of each vertex can be considered as an important property of the graph for determining graph invariants because like FPN, it also helps in identifying multiple zones based on "supra-local structures" which emerge as a result of local organisation of FCS (See Additional Information Fig 11). And, unlike FPN, which may be identical for every vertex in symmetrical graphs, SPN can vary for different vertices. Therefore, it is a relatively strong graph invariant and can act as a complete graph invariant for hard instances of symmetrical graphs. Although, in most of the symmetrical graphs, the SPN is identical for every vertex, but, in some cases (STS[24]) there are variations in the participation of edges involved in SCS resulting in different sets of SPN for different vertices. The difference in SPN arises due to the differential extent of edge sharing not counted for FCS. The SPN for every vertex in J(4,2) is 2 as mentioned previously in Fig 8.

1.4.5 Neighbourhood Cluster Analysis (NCA)

It is not only important to identify the non-isomorphic nature of graphs but it is also required in the case of isomorphic graphs to generate the bijective mapping (vertex correspondence) between two isomorphic ones. It has been observed that NCA solves this correspondence riddle when two graphs are deduced to be isomorphic with the help of the above graph invariants (size of FCS, FPN, size of SCS and SPN). For most of the hard instances of symmetrical graphs it is observed that size of FCS, FPN, size of SCS and SPN are identical throughout the vertex set of graphs. But in the case of Steiner Triple System (STS) graphs[24], it is observed that the vertices involved in the generation of SCS are not equivalent and are partitionable into different sets on the basis of their SCS multiplicity (i.e. the number of SCS in which a given vertex is involved-in in a graph). For such kinds of graphs, when two graphs are isomorphic, to generate a one-to-one correspondence for both graphs neighbourhood cluster analysis can be performed to obtain such correspondence. The steps involved in this process are as follows:

- Identify the vertices (SCS formation set - SFS) involved in the formation of SCS around a given vertex (i).
- For each of these vertices in SFS, calculate their respective SPN.
- Cluster the vertices of SFS on the basis of their SPN.
- For each SCS around a given vertex (i) calculate the frequencies of appearance of each cluster types (previous point) and label it (i) on the basis of these cluster frequencies.

For STS graphs, it is observed that every vertex is unique in the graph, thus making correspondence generation easy and fool-proof. For rest of the symmetrical graphs (Figs 3(a), 3(b), 6(a) and 6(b)), it is observed that there is only a single cluster for all vertices (since the SPN is identical for all vertices). Therefore, correspondence generation in such cases is carried out by Breadth-First Search (BFS) algorithm[27].

2. Research Method

Two graphs G1 and G2 (from a set of Generalised Johnson and/or Strongly Regular graphs) are considered for Graph Isomorphism. If the two graphs have equal vertices and degrees then the girth (length of the smallest cycle contained in a graph) of G1 and G2 is calculated and compared. If the girth for G1 and G2 is equal then structural aspects of graphs are compared. After comparison of girths, the FCS sequence for G1 and G2 is obtained and the FPN for all vertices are calculated. If the vertices of G1 and G2 have identical FPN for all vertices then the SCS sequence for G1 and G2 are obtained and the SPN is calculated for every vertex. The vertices of a graph may be partitioned into various sets where the vertices with identical SPN are placed in the same set. If the vertices of G1 and G2 can be partitioned into sets with identical size and the same corresponding SPN, then the graphs are deduced to be isomorphic and a vertex correspondence is generated. This can be understood with the help of Fig 12 of Additional Information. The vertex correspondence is based on FPN and SPN for a given vertex and traversal through its neighbours using BFS algorithm is carried out.

3. Result and Analysis

The size of FCS, FPN, size of SCS and SPN are mentioned in Table 2 for Generalised Johnson graphs GJ(6,3,1) and GJ(6,3,2) as shown in Fig 3(a) and 3(b).

Table 2: New Graph Invariants for Graph V: GJ(6,3,1) and VI: GJ(6,3,2)

Vertex	GJ(6,3,1)				GJ(6,3,2)			
	Size of FCS	FPN	Size of SCS	SPN	Size of FCS	FPN	Size of SCS	SPN
A	3	18	4	18	3	18	4	18
B	3	18	4	18	3	18	4	18
C	3	18	4	18	3	18	4	18
D	3	18	4	18	3	18	4	18
E	3	18	4	18	3	18	4	18
F	3	18	4	18	3	18	4	18
G	3	18	4	18	3	18	4	18
H	3	18	4	18	3	18	4	18
I	3	18	4	18	3	18	4	18
J	3	18	4	18	3	18	4	18
K	3	18	4	18	3	18	4	18
L	3	18	4	18	3	18	4	18
M	3	18	4	18	3	18	4	18
N	3	18	4	18	3	18	4	18
O	3	18	4	18	3	18	4	18
P	3	18	4	18	3	18	4	18
Q	3	18	4	18	3	18	4	18
R	3	18	4	18	3	18	4	18
S	3	18	4	18	3	18	4	18
T	3	18	4	18	3	18	4	18

The Size of FCS, FPN, Size of SCS and SPN are mentioned in Table 3 for Lattice(4,4) and Shrikhande (both are Strongly Regular graphs SRG(16,6,2,2)) as shown in Fig 6(a) and 6(b).

Table 3: New Graph Invariants of Graph VII: (Lattice(4,4)) and VIII: Shrikhande

Vertex	Lattice(4,4) Graph				Shrikhande Graph			
	Size of FCS	FPN	Size of SCS	SPN	Size of FCS	FPN	Size of SCS	SPN
A	3	6	4	9	3	6	4	3
B	3	6	4	9	3	6	4	3
C	3	6	4	9	3	6	4	3
D	3	6	4	9	3	6	4	3
E	3	6	4	9	3	6	4	3
F	3	6	4	9	3	6	4	3
G	3	6	4	9	3	6	4	3
H	3	6	4	9	3	6	4	3

I	3	6	4	9	3	6	4	3
J	3	6	4	9	3	6	4	3
K	3	6	4	9	3	6	4	3
L	3	6	4	9	3	6	4	3
M	3	6	4	9	3	6	4	3
N	3	6	4	9	3	6	4	3
O	3	6	4	9	3	6	4	3
P	3	6	4	9	3	6	4	3

Combining all the four graph invariants, it can be concluded that these are enough for uniquely distinguishing two symmetrical graphs and can be collectively termed as complete graph invariant for such graphs.

An interesting feature among the Generalised Johnson/Strongly Regular graphs is the possibility that two graphs with identical number of vertices and edges can still be non-isomorphic. We have identified a solution for the problem based on structural symmetry of graphs. The method was validated on multiple pair-wise combinations (J vs SRG; GJ vs J; GJ vs GJ; GJ vs SRG; SRG vs SRG) among these classes to test for GI. In all of these comparisons, the compared graphs had identical number of vertices and degrees. The important results are mentioned in Tables 4, 5, 6, 7 and 8 and discussed below.

3.1 J vs SRG

For such graphs, J(4,2) and SRG(6,4,2,4) were compared. Both graphs have identical values for each parameter (number of vertices, degrees, girth, FPN and SPN). Therefore, they were deduced as isomorphs. The correspondence of vertices was generated using BFS and a one-to-one mapping was found for all vertices in both graphs.

Table 4: J vs SRG

Case	Graphs	Vertices V	Degrees d	Girth g	Size of FCS	FPN [A(B)]*	Size of SCS	SPN [C(D)] ǂ	Results
J vs SRG	J(4,2)	6	4	3	3	4(6)	4	2(6)	Isomorphic
	SRG(6,4,2,4)	6	4	3	3	4(6)	4	2(6)	

* B represents the number of vertices in vertex subset of graph with A as FPN value.
ǂ D represents the number of vertices in vertex subset of graph with C as SPN value.

It is important to note that there is a decrease in the number of SPN formed as compared to FPN for the graphs J(4,2) and SRG(6,4,2,4) because generally number of edges involved in higher shapes always increases. Though the number of shapes formed for size four (FCS + 1) is 5, but since most of the shapes have one or more FCS encompassed, they get rejected for qualifying as a SCS. Therefore, the SPN value is 2 for each vertex. And, since both graphs have identical graph invariants (values for size of FCS, FPN, size of SCS and SPN), it is concluded that both are isomorphic to each other.

3.2 GJ vs J

The graphs GJ(6,3,1) and J(6,3) (where, J(6,3) is synonymous with GJ(6,3,2)) were compared. Both graphs have identical values for each parameter (number of vertices, degrees, girth, FPN and SPN). Hence, GJ(6,3,1) and J(6,3) were deduced to be isomorphic. The correspondence of vertices was generated using BFS and a one-to-one mapping was found for all vertices in both graphs.

As yet another example, GJ(14,7,1) and J(14,7) (where, J(14,7) is synonymous with GJ(14,7,6)) were compared. The respective girths of the two graphs were found to be different. Hence, they were inferred as non-isomorphic. Therefore, there was no need for further calculation of the remaining parameters.

Table 5: GJ vs J

Case		Graphs	Vertices V	Degrees d	Girth g	Size of FCS	FPN [A(B)]*	Size of SCS	SPN [C(D)] ǂ	Results
GJ vs J	1	GJ(6,3,1)	20	9	3	3	18(20)	4	18(20)	Isomorphic
		J(6,3)	20	9	3	3	18(20)	4	18(20)	
	2	GJ(14,7,1)	3432	49	4	-	-	-	-	Not Isomorphic
		J(14,7)	3432	49	3	-	-	-	-	

* B represents the number of vertices in vertex subset of graph with A as FPN value.
ǂ D represents the number of vertices in vertex subset of graph with C as SPN value.

For graphs GJ(6,3,1) and J(6,3), it can be observed that there are huge FPNs formed where vertices of these graphs are just 20. This is because, FPN is directly proportional to the degrees of the vertices, and since there are 9 degrees for each vertex, there are more participating edges involved for FCS. It is also worth mentioning that the SPN set is identical to the FPN set.

This is because every vertex is directly connected with 9 vertices, it is also not connected with 10 other vertices, making the number of directly connected vertices equal to the number of not directly connected vertices. For graphs GJ(14,7,1) and J(14,7), it can be seen that though the number of vertices and degrees of each vertex are same, the girth of both graphs are different.

This is because of the arrangement of edges involved in making the cycles. For GJ(14,7,1) every edge is arranged in such a way that there are cycles with size 4 and in the case of J(14,7) every edge is arranged in a way that there are cycles with size 3 making both graphs structurally different from each other, hence not isomorphic.

3.3 GJ vs GJ

GJ(14,7,2) was compared with GJ(14,7,5). These graphs differed in their respective girths, hence, they were deduced to be non-isomorphic.

In another instance, GJ(14,3,0) (where GJ(14,3,0) is synonymous with K(14,3)) was compared with GJ(14,3,1). The respective girths for these graphs were same. However, as the FPN values for all vertices of one graph did not match with that of the second, therefore, they were inferred to be non-isomorphs.

Table 6: GJ vs GJ

Case		Graphs	Vertices V	Degrees d	Girth g	Size of FCS	FPN [A(B)]*	Size of SCS	SPN [C(D)] ǂ	Results
GJ vs GJ	1	GJ(14,7,2)	3432	441	4	-	-	-	-	Not Isomorphic
		GJ(14,7,5)	3432	441	3	-	-	-	-	
	2	GJ(14,3,0)	364	165	3	3	4620(364)	-	-	Not Isomorphic
		GJ(14,3,1)	364	165	3	3	5940(364)	-	-	

* B represents the number of vertices in vertex subset of graph with A as FPN value.
ǂ D represents the number of vertices in vertex subset of graph with C as SPN value.

For graphs GJ(14,7,2) and GJ(14,7,5), the girth is different. This is due to the arrangement of edges involved in forming the cycles. For GJ(14,7,2) there is reduced edge involvement in cycle formation making the girth 4. On the other hand, GJ(14,7,5) has more edges involved in cycle formation making the girth 3. Since girths are different, no further calculation was required. For graphs GJ(14,3,0) and GJ(14,3,1), it can be observed that the number of vertices, degrees of each vertex and girth are same. The difference is in the FPN of both graphs. Though FPN is directly proportional to the degrees of a vertex, there is a huge difference in FPN because there is more sharing of edges in graph GJ(14,3,1) resulting in increase in FPN as compared to the edge sharing of GJ(14,3,0). The concept of sharing of edges arises due to the involvement of each edge forming cycles (FCS and SCS). Sometimes, an edge is involved in only one FCS and sometimes it can be involved in more than one FCS. (For example, in the case of Fig 7, it can be observed that edge 1-2 is involved in making 2 FCS(1-2-3 and 1-2-4) for a single vertex whereas for the graph shown in Fig 11 edge 1-2 is involved in making only 1 FCS (1-2-8-7) for vertex 1. Therefore, it can be concluded that edge sharing is more in the case of J(4,2).) Therefore, the FPN for any vertex not only depends on the degree of that vertex but also depends on the number of edges being shared for each FCS.

3.4 GJ vs SRG

The graphs GJ(5,2,0) and SRG(10,3,0,1) have same girth and the FPN and SPN values for the complete vertex set in both graphs were identical. Hence, GJ(5,2,0) and SRG(10,3,0,1) were deduced to be isomorphic. The correspondence of vertices was generated using BFS and a one-to-one mapping was found for all vertices in both graphs.

Table 7: GJ vs SRG

Case	Graphs	Vertices V	Degrees d	Girth g	Size of FCS	FPN [A(B)]*	Size of SCS	SPN [C(D)]ǂ	Results
GJ vs SRG	GJ(5,2,0)	10	3	5	5	6(10)	6	6(10)	Isomorphic
	SRG(10,3,0,1)	10	3	5	5	6(10)	6	6(10)	

* B represents the number of vertices in vertex subset of graph with A as FPN value.
ǂ D represents the number of vertices in vertex subset of graph with C as SPN value.

For graphs GJ(5,2,0) and SRG(10,3,0,1), it is worth mentioning that the number of neighbours and non-neighbours are not equal, but the FPN and SPN are same. This is again because of the sharing of edges. The edges shared in the case of SCS is more than the edges shared in the case of FCS making both FPN and SPN equal in number for every vertex.

3.5 SRG vs SRG

STS-19-x (where x ranges from 1 to 7) were compared amongst themselves out of 11 such graphs that were provided by nautY and Traces[5].These graphs belong to the same family of SRG (57,24,11,9)[28]. There are 11,084,874,829[29] non-isomorphic forms of STS-19. These graphs are difficult to solve for graph isomorphism because they have same number of vertices, edges, girth and FPN. However, the number of vertices with identical SPN values were different for these graphs, thus these were inferred as mutually non-isomorphic.

Table 8: SRG vs SRG

Case	Graphs	Vertices V	Degrees d	Girth g	Size of FCS	FPN [A(B)]*	Size of SCS	SPN [C(D)] ǂ	Results
SRG vs SRG	SRG(57,24,11,9) or STS-19-1	57	24	3	3	132(57)	4	528(20), 31(29), 534(5), 537(3)	Not Isomorphic
	SRG(57,24,11,9) or STS-19-2	57	24	3	3	132(57)	4	528(24),531(16), 534(15), 537(2)	
	SRG(57,24,11,9) or STS-19-3	57	24	3	3	132(57)	4	528(14),531(20), 534(14), 537(6), 540(1), 543(2)	
	SRG(57,24,11,9) or STS-19-4	57	24	3	3	132(57)	4	528(28),531(23), 534(5), 537(1)	
	SRG(57,24,11,9) or STS-19-5	57	24	3	3	132(57)	4	528(13),531(17), 534(19), 537(7), 540(1)	
	SRG(57,24,11,9) or STS-19-6	57	24	3	3	132(57)	4	528(11),531(23), 534(16), 537(7)	
	SRG(57,24,11,9) or STS-19-7	57	24	3	3	132(57)	4	528(21),531(22), 534(12), 537(2)	

* B represents the number of vertices in vertex subset of graph with A as FPN value.
ǂ D represents the number of vertices in vertex subset of graph with C as SPN value.

For all the graphs mentioned in Table 8, it can be noted that they have equal number of vertices, degrees for each vertex, girth and FPN. The only point of difference is in the SPN. The SPN differ in the number of identical sets the vertices are involved in. Comparing STS-19-1 and STS-19-4, it is observed that the number of SPNs are identical, i.e, 528, 531, 534 and 537 but still they are not isomorphic because of the number of vertices involved for every SPN set are different. This is again because of the sharing of edges of each SCS. This makes all these graphs non-isomorphic forms.

The most beautiful observation in the case of STS is the correspondence generation when two isomorphic STSs are compared. For correspondence generation, we perform NCA. For example, in the case of STS-19-1, the 20 vertices (SPN value of 528) are clustered in one set (A), the 29 vertices (SPN value of 531) are clustered in the next set (B), the 5 vertices (SPN value of 534) are clustered in another set (C) and the 3 vertices (SPN value of 537) are clustered in the next set (D). After classification, we pick each SCS and map the vertices involved in the formation of SCS (SCS Formation Set) according to their respective clusters. A count of the clusters involved for each vertex is then maintained. It is observed that this count is unique for every vertex of the graph (Additional Information). Thus, it can be concluded that NCA can uniquely distinguish between every vertex of a STS graph. This entire process is repeated for the second graph and a one-to-one correspondence is generated only if they are isomorphic.

4. Conclusion

Graph properties play a major role in classification of graphs. We have identified two graph properties based on cyclical shapes: FCS and SCS. We have also discovered some graph invariants on the basis of these graph properties: size of FCS, FPN, size of SCS, SPN and NCA. There are many symmetrical graphs having identical graph invariants which may or may not be isomorphic to each other. These graph invariants might be helpful in developing graph invariants for those symmetrical graphs which fail in the case of other pre-defined graph invariants such as degree sequence, girth, chromatic number etc. These graph invariants will prove to be helpful in distinguishing other regular graphs as well and therefore, it can be used for solving graph isomorphism for some section of regular and symmetrical graphs. They can further be used as roots for more graph invariants.

The running time complexity of the entire algorithm depends upon the calculation of FCS and SCS which in this method is $O(n^5)$. The running time complexity for calculation of FCS and SCS in this algorithm are $O(n^2d^2g^2)$ and $O(n^2d^3g^2)$ respectively where n represents the number of vertices, d represents the number of degrees and g represents the girth of the graph. The maximum girth for any connected Strongly Regular graph is 5 and for Generalised Johnson graph is 6[1]. Hence, the upper bound of parameter g can be taken as a constant for finding out the running time complexity of the entire algorithm. Therefore, the overall worst time complexity of the algorithm for determining isomorphs of Generalised Johnson (including Johnson graphs) and Strongly Regular graphs is $O(n^2d^3)$ which will never exceed $O(n^5)$. It is also deduced that if there are two Generalised Johnson Graphs which are tested for isomorphism, FPN values for all the vertices are enough to distinguish whether both graphs are isomorphic or not (testing for SPN is redundant in such case).

The participation number of vertices is a measure for structurally congruent or equivalent vertices. The Achilles' heel of most of the currently available methods is their focus on canonical labelling of the vertices. This fails in extremely symmetric graphs such as Johnson graphs[6]. We, on the other hand, have used the structural symmetry for identification of the isomorphs. It is pertinent to mention here that the same approach can be applied for all regular graphs. This opens up the problem of Graph Isomorphism for a relook and possibly, in future, a polynomial time solution could be identified for "Complete Graph Isomorphism".

Acknowledgments

University Grants Commission for SM's and VK's fellowship and Department of Science & Technology for SKJ's fellowship. We would like to thank our respective families for their tolerance of our obsessive compulsive disorder for graphs.

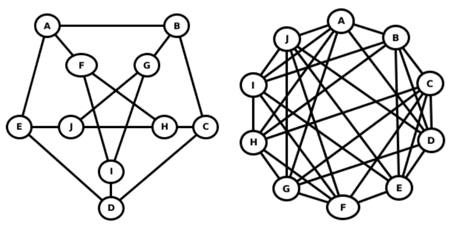
References

[1] West, D. B. (1996). Introduction to graph theory (Vol. 2). Upper Saddle River, NJ: Prentice hall
[2] Ore, O. (1962). Theory of graphs (Vol. 38). American Mathematical Society.
[3] Read, R. C., & Corneil, D. G. (1977). The graph isomorphism disease. Journal of Graph Theory, 1(4), 339-363.
[4] Chartrand, G. (1977). Introductory graph theory. Courier Corporation.
[5] McKay, B. D., & Piperno, A. (2014). Practical graph isomorphism, II. Journal of Symbolic Computation, 60, 94-112.

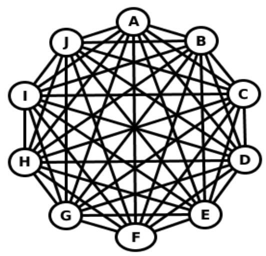
[6] Babai, L. (2016, June). Graph isomorphism in quasipolynomial time. In Proceedings of the forty-eighth annual ACM symposium on Theory of Computing (pp. 684-697). ACM.
[7] Garey, M. R., & Johnson, D. S. (2002). Computers and intractability (Vol. 29). New York: wh freeman.
[8] Fortnow, L. (2009). The status of the P versus NP problem. Communications of the ACM, 52(9), 78-86.
[9] Babai, L., & Luks, E. M. (1983, December). Canonical labeling of graphs. In Proceedings of the fifteenth annual ACM symposium on Theory of computing (pp. 171-183). ACM.
[10] Katebi, H., Sakallah, K. A., & Markov, I. L. (2012). Graph symmetry detection and canonical labeling: Differences and synergies. arXiv preprint arXiv:1208.6271.
[11] Junttila, T., & Kaski, P. (2007, January). Engineering an efficient canonical labeling tool for large and sparse graphs. In 2007 Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments (ALENEX) (pp. 135-149). Society for Industrial and Applied Mathematics.
[12] Dankelmann, P., Goddard, W., & Swart, C. S. (2004). The average eccentricity of a graph and its subgraphs. Utilitas Mathematica, 65, 41-52.
[13] Thomassen, C. (1983). Girth in graphs. Journal of Combinatorial Theory, Series B, 35(2), 129-141.
[14] Li, X., Yao, X., Zhou, W., & Broersma, H. (2009). Complexity of conditional colorability of graphs. Applied mathematics letters, 22(3), 320-324.
[15] Pemmaraju, S., & Skiena, S. (2003). Computational Discrete Mathematics: Combinatorics and Graph Theory with Mathematica®. Cambridge university press.
[16] Etzion, T., & Bitan, S. (1996). On the chromatic number, colorings, and codes of the Johnson graph. Discrete applied mathematics, 70(2), 163-175.
[17] Urrutia, J., Sarmiento, E., Lara, D., González-Aguilar, H., Flores-Peñaloza, D., Fabila-Monroy, R., Cano, J., & Bautista-Santiago, C. (2013). On the connectedness and diameter of a geometric Johnson graph. Discrete Mathematics & Theoretical Computer Science, 15.
[18] Valencia-Pabon, M., & Vera, J. C. (2005). On the diameter of Kneser graphs. Discrete mathematics, 305(1-3), 383-385.
[19] Haemers, W. H. (1996). Distance-regularity and the spectrum of graphs. Linear algebra and its applications, 236, 265-278.
[20] Hubaut, X. L. (1975). Strongly regular graphs. Discrete Mathematics, 13(4), 357-381.
[21] Godsil, C., & Royle, G. (2001). Strongly regular graphs. In Algebraic graph theory (pp. 217-247). Springer, New York, NY.
[22] Aigner, M. (2012). Combinatorial theory. Springer Science & Business Media.
[23] Brouwer, A. E., & Haemers, W. H. (2012). Distance-regular graphs. In Spectra of Graphs (pp. 177-185). Springer, New York, NY.
[24] Treash, C. (1971). The completion of finite incomplete Steiner triple systems with applications to loop theory. Journal of Combinatorial Theory, Series A, 10(3), 259-265.
[25] Ward, G. (2016). Graphs: Isomorphism, Cycles, Reconstruction and Randomness (Doctoral dissertation, Brandeis University).
[26] Miyazaki, T. (1997). The complexity of McKay's canonical labeling algorithm. In Groups and Computation II (Vol. 28, pp. 239-256). Aer. Math. Soc.: Providence, RI.
[27] Bundy, A., & Wallen, L. (1984). Breadth-first search. In Catalogue of artificial intelligence tools (pp. 13-13). Springer, Berlin, Heidelberg.
[28] Kaski, P., & Östergård, P. (2004). The Steiner triple systems of order 19. Mathematics of Computation, 73(248), 2075-2092.
[29] Colbourn, C. J., & Dinitz, J. H. (Eds.). (2006). Handbook of combinatorial designs. CRC press.

Additional Information

4.1 Hard Instances of Symmetrical Graphs



(a) Kneser Graph: K(5,2)   (b) Johnson Graph: J(5,2)



c) K(5,2)+J(5,2)=Complete Graph K_5

Figure 9: Generalised Johnson Graphs

4.2 Variations in size of FCS and SCS

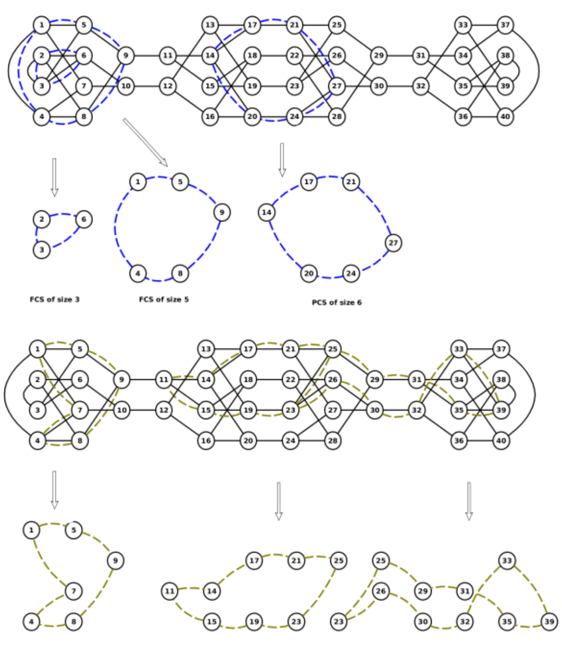


Figure 10: Miyazaki Graph M4. The size of FCS for various vertices varies from 3 to 6 as shown. Similarly, the size of SCS for various vertices also varies from 6 to 10.

4.3 Example of SCS with and without encompassment

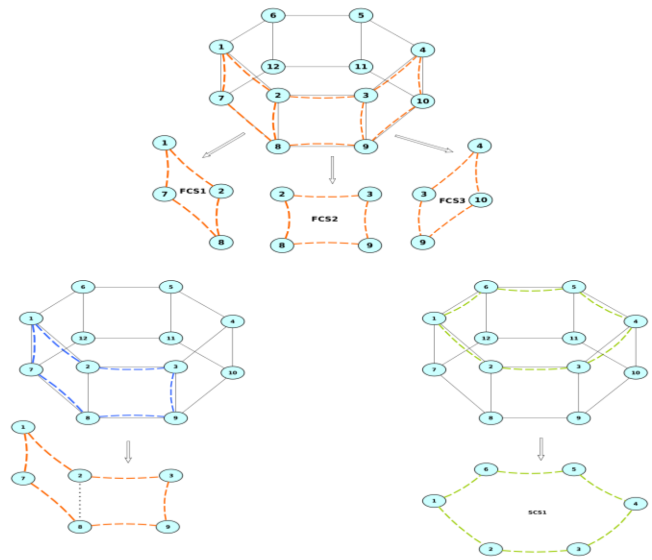


Figure 11: Example to understand SCS Encompassment. In this example, there are 6 FCS of size 4 formed out of which 1-2-8-7, 2-3-9-8 and 3-4-10-9 are highlighted. The remaining FCS are 5-4-10-11, 6-5-11-12 and 1-6-12-7. When calculating SCS (of size 6), encompassment of the edges of all FCS are checked. In the case of hexagon 1-2 3-9-8-7 it is found that 2 FCS are encompassed (1-2-8-7 and 2-3-9-8). Hence, this shape does not qualify for SCS. On the other hand, hexagon 6-5-4-3-2-1 do not have any FCS encompassed and it emerges due to the arrangements of FCS. Therefore, hexagon 6-5-4-3-2-1 qualifies for SCS. Another such SCS for the graph is hexagon 12-11-10-9-8-7 (not highlighted in the figure)

4.4 Overall Approach

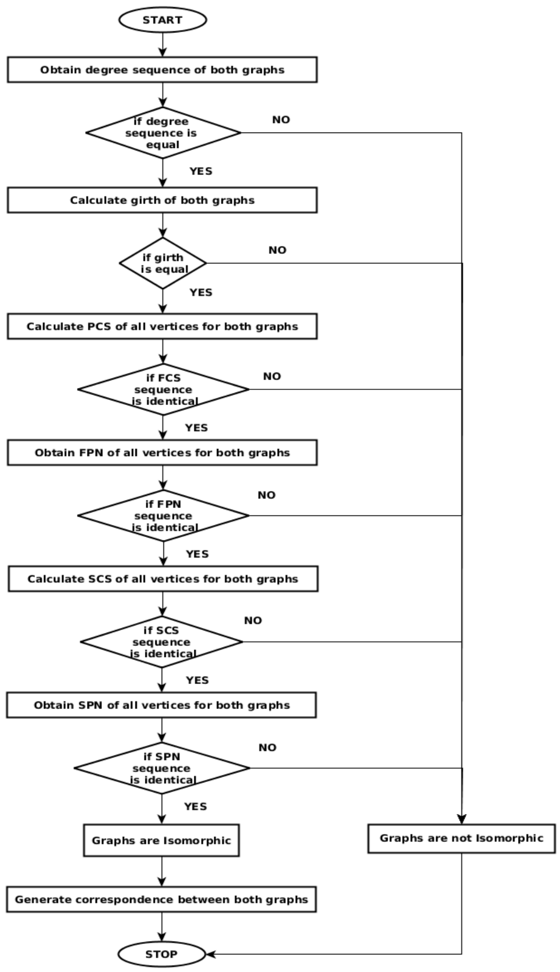


Figure 12: Flowchart for Method to solve Graph Isomorphism for Generalised Johnson and Strongly Regular graphs

4.5 Neighbourhood Cluster Analysis

To understand the NCA let us consider a STS-19-1. It has 57 vertices and each vertex has 24 edges. The girth for this graph is 3 and FPN value for each vertex is 132.

Vertex 1 is involved in 528 SCS. These are 1-2-18-11, 1-2-18-13, 1-2-18-38, 1-3-18-10, 1-3-18-13, 1-3-18-22, 1-3-18-38, 1-5-18-10, 1-5-18-11, 1-5-18-22, 1-5-18-28, 1-10-18-28, 1-10-18-38, 1-11-18-22, 1-11-18-38, 1-13-18-22, 1-13-18-28, 1-2-19-12, 1-2-19-13, 1-2-19-34, 1-2-19-49, 1-4-19-13, 1-4-19-22, 1-7-19-10, 1-7-19-12, 1-7-19-22, 1-7-19-34, 1-10-19-34, 1-12-19-22, 1-12-19-49, 1-13-19-22, 1-13-19-34, 1-2-20-11, 1-2-20-12, 1-2-20-38, 1-2-20-43, 1-6-20-10, 1-6-20-12, 1-6-20-22, 1-6-20-38, 1-9-20-10, 1-9-20-11, 1-9-20-22, 1-9-20-43, 1-10-20-38, 1-10-20-43, 1-11-20-22, 1-11-20-38, 1-12-20-22, 1-12-20-43, 1-2-21-14, 1-2-21-17, 1-2-21-46, 1-2-21-53, 1-5-21-10, 1-5-21-17, 1-5-21-22, 1-5-21-53, 1-7-21-10, 1-7-21-14, 1-7-21-22, 1-7-21-46, 1-10-21-46, 1-10-21-53, 1-14-21-22, 1-14-21-53, 1-17-21-22, 1-2-23-17, 1-2-23-46, 1-2-23-53, 1-3-23-10, 1-3-23-16, 1-3-23-22, 1-3-23-53, 1-9-23-10, 1-9-23-17, 1-9-23-22, 1-9-23-46, 1-10-23-46, 1-10-23-53, 1-16-23-22, 1-16-23-46, 1-17-23-22, 1-2-24-14, 1-2-24-15, 1-2-24-34, 1-2-24-43, 1-6-24-10, 1-6-24-14, 1-6-24-22, 1-8-24-10, 1-8-24-34, 1-10-24-34, 1-10-24-43, 1-14-24-22, 1-14-24-34, 1-15-24-43, 1-3-25-12, 1-3-25-15, 1-3-25-22, 1-3-25-34, 1-4-25-11, 1-4-25-15, 1-4-25-22, 1-4-25-28, 1-8-25-11, 1-8-25-12, 1-8-25-28, 1-8-25-34, 1-11-25-22, 1-11-25-34, 1-12-25-22, 1-12-25-28, 1-15-25-28, 1-3-26-12, 1-3-26-14, 1-3-26-38, 1-5-26-11, 1-5-26-12, 1-5-26-28, 1-9-26-11, 1-9-26-14, 1-9-26-28, 1-9-26-46, 1-11-26-38, 1-11-26-46, 1-12-26-28, 1-12-26-46, 1-14-26-28, 1-14-26-38, 1-3-27-13, 1-3-27-14, 1-3-27-43, 1-3-27-49, 1-7-27-11, 1-7-27-14, 1-7-27-28, 1-7-27-43, 1-8-27-11, 1-8-27-13, 1-8-27-28, 1-8-27-49, 1-11-27-49, 1-13-27-28, 1-13-27-43, 1-14-27-28, 1-14-27-49, 1-3-29-15, 1-3-29-16, 1-3-29-34, 1-3-29-53, 1-6-29-16, 1-6-29-28, 1-6-29-53, 1-9-29-11, 1-9-29-15, 1-9-29-28, 1-9-29-34, 1-11-29-34, 1-11-29-53, 1-15-29-28, 1-15-29-53, 1-16-29-34, 1-3-30-10, 1-3-30-49, 1-3-30-53, 1-4-30-11, 1-4-30-17, 1-4-30-28, 1-4-30-53, 1-7-30-10, 1-7-30-11, 1-7-30-28, 1-10-30-28, 1-10-30-53, 1-11-30-49, 1-11-30-53, 1-17-30-28, 1-17-30-49, 1-2-31-12, 1-2-31-13, 1-2-31-34, 1-2-31-38, 1-4-31-13, 1-4-31-16, 1-4-31-28, 1-4-31-38, 1-5-31-12, 1-5-31-16, 1-5-31-28, 1-5-31-34, 1-12-31-28, 1-13-31-28, 1-13-31-34, 1-16-31-34, 1-16-31-38, 1-4-32-11, 1-4-32-17, 1-4-32-43, 1-4-32-53, 1-6-32-12, 1-6-32-17, 1-6-32-53, 1-7-32-11, 1-7-32-12, 1-7-32-34, 1-7-32-43, 1-11-32-34, 1-11-32-53, 1-12-32-43, 1-12-32-53, 1-17-32-34, 1-17-32-43, 1-4-33-14, 1-4-33-16, 1-4-33-46, 1-4-33-53, 1-5-33-12, 1-5-33-16, 1-5-33-34, 1-5-33-53, 1-9-33-14, 1-9-33-34, 1-9-33-46, 1-12-33-46, 1-12-33-53, 1-14-33-34, 1-14-33-53, 1-16-33-34, 1-16-33-46, 1-3-35-12, 1-3-35-14, 1-3-35-34, 1-3-35-43, 1-4-35-14, 1-4-35-17, 1-4-35-43, 1-4-35-46, 1-8-35-12, 1-8-35-17, 1-8-35-34, 1-8-35-46, 1-12-35-43, 1-12-35-46, 1-14-35-34, 1-17-35-34, 1-17-35-43, 1-3-

36-13, 1-3-36-38, 1-3-36-43, 1-5-36-11, 1-5-36-17, 1-5-36-43, 1-6-36-13, 1-6-36-17, 1-6-36-38, 1-6-36-46, 1-11-36-38, 1-11-36-46, 1-13-36-43, 1-13-36-46, 1-17-36-38, 1-17-36-43, 1-5-37-15, 1-5-37-17, 1-5-37-22, 1-5-37-53, 1-7-37-15, 1-7-37-22, 1-7-37-38, 1-8-37-13, 1-8-37-17, 1-8-37-38, 1-8-37-53, 1-13-37-22, 1-13-37-53, 1-15-37-38, 1-15-37-53, 1-17-37-22, 1-17-37-38, 1-5-39-16, 1-5-39-43, 1-5-39-53, 1-8-39-13, 1-8-39-16, 1-8-39-38, 1-8-39-53, 1-9-39-13, 1-9-39-14, 1-9-39-43, 1-13-39-43, 1-13-39-53, 1-14-39-38, 1-14-39-53, 1-16-39-38, 1-16-39-43, 1-4-40-13, 1-4-40-15, 1-4-40-38, 1-5-40-10, 1-5-40-15, 1-5-40-34, 1-5-40-49, 1-6-40-10, 1-6-40-13, 1-6-40-38, 1-6-40-49, 1-10-40-34, 1-10-40-38, 1-13-40-34, 1-15-40-38, 1-15-40-49, 1-5-41-11, 1-5-41-43, 1-5-41-49, 1-6-41-13, 1-6-41-14, 1-6-41-46, 1-6-41-49, 1-7-41-11, 1-7-41-14, 1-7-41-43, 1-7-41-46, 1-11-41-46, 1-11-41-49, 1-13-41-43, 1-13-41-46, 1-14-41-49, 1-2-42-11, 1-2-42-15, 1-2-42-43, 1-6-42-16, 1-6-42-22, 1-6-42-28, 1-8-42-11, 1-8-42-16, 1-8-42-28, 1-11-42-22, 1-15-42-28, 1-15-42-43, 1-16-42-22, 1-16-42-43, 1-4-44-11, 1-4-44-16, 1-4-44-43, 1-4-44-53, 1-6-44-10, 1-6-44-16, 1-6-44-49, 1-6-44-53, 1-9-44-10, 1-9-44-11, 1-9-44-43, 1-9-44-49, 1-10-44-43, 1-10-44-53, 1-11-44-49, 1-11-44-53, 1-16-44-43, 1-16-44-49, 1-4-45-14, 1-4-45-15, 1-4-45-22, 1-4-45-46, 1-5-45-10, 1-5-45-15, 1-5-45-22, 1-5-45-49, 1-8-45-10, 1-8-45-46, 1-8-45-49, 1-10-45-46, 1-14-45-22, 1-14-45-49, 1-15-45-46, 1-15-45-49, 1-2-47-14, 1-2-47-15, 1-2-47-34, 1-2-47-46, 1-5-47-15, 1-5-47-16, 1-5-47-28, 1-5-47-34, 1-6-47-14, 1-6-47-16, 1-6-47-28, 1-6-47-46, 1-14-47-28, 1-14-47-34, 1-15-47-28, 1-15-47-46, 1-16-47-34, 1-16-47-46, 1-7-48-15, 1-7-48-16, 1-7-48-22, 1-8-48-13, 1-8-48-16, 1-8-48-49, 1-8-48-53, 1-9-48-13, 1-9-48-15, 1-9-48-22, 1-9-48-49, 1-13-48-22, 1-13-48-53, 1-15-48-49, 1-15-48-53, 1-16-48-22, 1-16-48-49, 1-3-50-13, 1-3-50-15, 1-3-50-34, 1-3-50-49, 1-6-50-13, 1-6-50-17, 1-6-50-46, 1-6-50-49, 1-7-50-15, 1-7-50-34, 1-7-50-46, 1-13-50-34, 1-13-50-46, 1-15-50-46, 1-15-50-49, 1-17-50-34, 1-17-50-49, 1-2-51-12, 1-2-51-13, 1-2-51-38, 1-2-51-49, 1-7-51-12, 1-7-51-16, 1-7-51-28, 1-7-51-38, 1-9-51-13, 1-9-51-28, 1-9-51-49, 1-12-51-28, 1-12-51-49, 1-13-51-28, 1-16-51-38, 1-16-51-49, 1-3-52-12, 1-3-52-15, 1-3-52-22, 1-3-52-38, 1-8-52-12, 1-8-52-17, 1-8-52-38, 1-8-52-46, 1-9-52-15, 1-9-52-17, 1-9-52-22, 1-9-52-46, 1-12-52-22, 1-12-52-46, 1-15-52-38, 1-15-52-46, 1-17-52-22, 1-17-52-38, 1-2-54-17, 1-2-54-46, 1-2-54-49, 1-3-54-10, 1-3-54-16, 1-3-54-49, 1-4-54-16, 1-4-54-17, 1-4-54-28, 1-4-54-46, 1-10-54-28, 1-10-54-46, 1-16-54-46, 1-16-54-49, 1-17-54-28, 1-17-54-49, 1-2-55-14, 1-2-55-17, 1-2-55-43, 1-2-55-53, 1-7-55-14, 1-7-55-16, 1-7-55-28, 1-7-55-43, 1-8-55-16, 1-8-55-17, 1-8-55-28, 1-8-55-53, 1-14-55-28, 1-14-55-53, 1-16-55-43, 1-17-55-28, 1-17-55-43, 1-6-56-12, 1-6-56-17, 1-6-56-38, 1-6-56-53, 1-7-56-12, 1-7-56-15, 1-7-56-34, 1-7-56-38, 1-9-56-15, 1-9-56-17, 1-9-56-34, 1-12-56-53, 1-15-56-38, 1-15-56-53, 1-17-56-34, 1-17-56-38, 1-4-57-14, 1-4-57-38, 1-4-57-43, 1-8-57-10, 1-8-57-12, 1-8-57-38, 1-8-57-49, 1-9-57-10, 1-9-57-14, 1-9-57-43, 1-9-57-49, 1-10-57-38, 1-10-57-43, 1-12-57-43, 1-12-57-49, 1-14-57-38, 1-14-57-49.

The 57 vertices can be clustered in 4 groups based on the FPN. Out of 57 vertices, 20 belong to the cluster 'A' and all vertices in this cluster participate in the formation of 528 SCS. Similarly, 29, 5 and 3 vertices belong to B, C and D cluster respectively. Furthermore vertices from B, C and D cluster particitpate in the formation of 531, 534 and 537 SCS respectively. The detailed cluster assignment for 57 vertices are as under:
1=A, 2=A, 3=B, 4=A, 5=A, 6=A, 7=A, 8=B, 9=A, 10=B, 11=B, 12=B, 13=B, 14=B, 15=B, 16=B, 17=B, 18=B, 19=C, 20=A, 21=D, 22=A, 23=C, 24=D, 25=B, 26=A, 27=B, 28=A, 29=B, 30=A, 31=C, 32=C, 33=A, 34=A, 35=C, 36=A, 37=B, 38=A, 39=B, 40=A, 41=D, 42=B, 43=A, 44=B, 45=B, 46=B, 47=B, 48=B, 49=A, 50=B, 51=B, 52=B, 53=B, 54=B, 55=B, 56=B, 57=A.

The vertices of SCS in which vertex 1 participates can be mapped using the above mentioned cluster ids as
1-2-18-11 (where 1, 2, 18, 11 are vertex numbers) : A-A-B-B
....
....
....
for all 528 SCS.

Hence for vertex 1, the cluster A B, C and D appear 1172, 811, 82 and 47 times.

For the overall graph, the frequencies of clusters are as follows:

Table 1: Frequency of clusters of STS-19-1

Vertex ID	Frequency of Cluster ID			
1	A=1172	B=811	C=82	D=47
2	A=1091	B=752	C=166	D=103
3	A=633	B=1303	C=139	D=49
4	A=1170	B=698	C=193	D=51
5	A=1142	B=754	C=111	D=105
6	A=1143	B=756	C=110	D=103
7	A=1060	B=807	C=140	D=105
8	A=604	B=1328	C=114	D=78
9	A=1145	B=805	C=109	D=53
10	A=578	B=1301	C=139	D=106
11	A=552	B=1382	C=110	D=80
12	A=579	B=1299	C=196	D=50
13	A=521	B=1383	C=143	D=77
14	A=492	B=1388	C=110	D=134
15	A=463	B=1499	C=85	D=77
16	A=466	B=1469	C=139	D=50
17	A=438	B=1442	C=166	D=78
18	A=609	B=1244	C=165	D=106
19	A=551	B=733	C=714	D=138
20	A=1087	B=726	C=167	D=132
21	A=554	B=794	C=171	D=629
22	A=1056	B=813	C=138	D=105
23	A=496	B=877	C=656	D=107
24	A=553	B=824	C=170	D=601
25	A=522	B=1358	C=195	D=49
26	A=1060	B=864	C=83	D=105
27	A=524	B=1356	C=136	D=108
28	A=1086	B=863	C=111	D=52
29	A=552	B=1384	C=107	D=81
30	A=1032	B=891	C=111	D=78
31	A=577	B=821	C=685	D=53
32	A=581	B=760	C=687	D=108
33	A=975	B=812	C=220	D=105
34	A=1117	B=726	C=191	D=78
35	A=501	B=844	C=712	D=79
36	A=1004	B=835	C=194	D=79
37	A=523	B=1380	C=142	D=79
38	A=1201	B=752	C=110	D=49
39	A=551	B=1329	C=163	D=81
40	A=1090	B=837	C=109	D=76
41	A=578	B=825	C=144	D=601
42	A=522	B=1385	C=141	D=76
43	A=1112	B=756	C=138	D=106
44	A=631	B=1275	C=139	D=79
45	A=583	B=1355	C=81	D=105
46	A=634	B=1244	C=139	D=107
47	A=548	B=1325	C=115	D=136
48	A=475	B=1433	C=138	D=78
49	A=1144	B=780	C=110	D=78
50	A=496	B=1350	C=170	D=108
51	A=576	B=1332	C=140	D=76
52	A=523	B=1411	C=138	D=52
53	A=634	B=1271	C=139	D=80
54	A=552	B=1355	C=167	D=50
55	A=490	B=1358	C=166	D=110
56	A=604	B=1304	C=113	D=103
57	A=1087	B=837	C=113	D=75

This never gets repeated in the rest of STS-19-x graphs