ASIAN JOURNAL OF CIVIL ENGINEERING (BUILDING AND HOUSING) VOL. 8, NO. 6 (2007)
PAGES 659-676

SYMMETRY DETECTION FOR STRUCTURAL GRAPH MODELS

A. Kaveh*, K. Laknegadi and M. Zahedi
Department of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran-16, Iran

ABSTRACT

In recent years special methods have been presented for eigensolution of symmetric structures. Therefore, an automatic approach for detecting the symmetry axis and/or planes was inevitable. In this paper a simple automatic approach is developed for symmetry detection in structural graph models. This is achieved by a special coloring of graph utilizing the dominant eigenvector of a special matrix known as the inner-product matrix, and imposing some constraints. Combining the present method and some canonical forms, an efficient tool is obtained for graph factorization. The efficiency of the present method is illustrated through some examples.

Keywords: Symmetry detection, coloring, axis of symmetry, canonical forms, graph theory

1. INTRODUCTION

Symmetry has been widely used in science and engineering [1-5]. Eigenvalue problems arise in many scientific and engineering problems [6-8]. While the basic mathematical ideas are independent of the size of matrices, the numerical determination of eigenvalues and eigenvectors becomes more complicated as the dimensions of matrices increase. Special methods are beneficial for efficient solution of such problems, especially when their corresponding matrices are highly sparse.

Methods are developed for decomposing and healing the graph models of structures, in order to calculate the eigenvalues of matrices and graph matrices with special patterns. The eigenvectors corresponding to such patterns for the symmetry of Form I, Form II and Form III are studied in references [9-10], and the applications to vibrating mass-spring systems and frame structures are developed in [11] and [12], respectively. These forms are also applied to calculating the buckling load of symmetric frames [13-14]. Thought the decomposition and healing approaches are attractive, however, automating the detection of symmetry increases the power of these methods to a great extent. This is the main issue studied in this paper.

Graphs are commonly used in computer science to model relational structures such as

* E-mail address of the corresponding author: alikaveh@iust.ac.ir

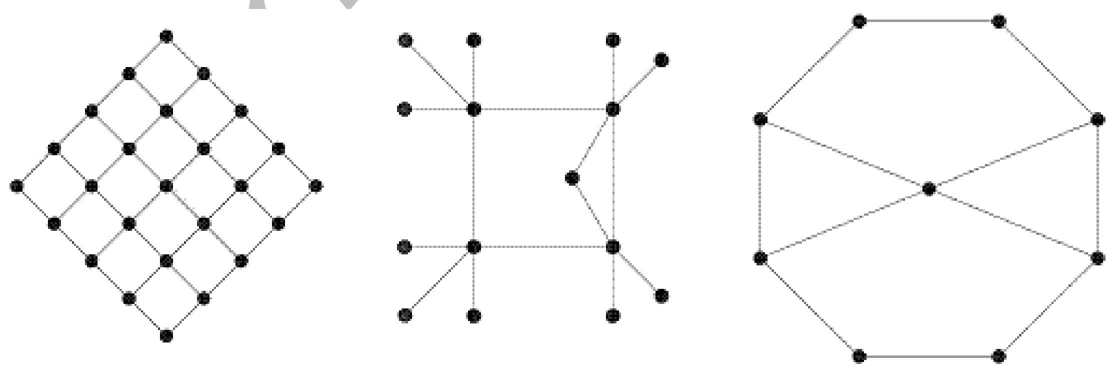
programs, databases, and data structures. Symmetry of graphs has been extensively studied over the past 50 years by using automorphisms of graphs and group theory which have played an important role in graph, and promising and interesting results have been obtained [15-18]. Symmetry is one of the most important aesthetics for delivering an easily understandable graph layout. A graph with a symmetric structure requires to concentrate on its “core” part, while the other parts can be symmetrically mirrored from the symmetric structure.

The problem of determining whether a graph has a non-trivial strict geometric automorphism in two and three dimensions is NP-complete [15]. Testing whether a graph has an axial (central or rotational) symmetry is also an NP-complete problem [19]. The construction of a symmetric drawing of a graph involves two steps. In the first step, known as the *symmetry finding step,* all automorphisms of the graph that can be displayed as symmetries of a drawing are found. The second step known as *the drawing step*, is to construct drawings that display these automorphisms. In this article only the first step is needed.

Several methods for detecting symmetries have been introduced in recent years, such as a group-theoretic method for drawing graphs symmetrically of David Abelson and Soek-Hee Hong [15], branch and cut methods of Buchheim and Junger [16], and the heuristic of de Fraysseix for finding symmetries [19].

In this paper, we combine the above two methods and introduce a simple approach for finding the automorphisms corresponding to axial (central or rotational) symmetries in structural graph models and the core of a symmetric graph, which can easily be used in structural analysis.

First by defining Euclidian distance, a suitable coloring of nodes is obtained and by imposing some constraints similar to those of the branch and cut method [16] the core of the graph and the permutation corresponding to optimal symmetry are specified. Examples of such symmetries are the rotational with maximum order and minimum fixed points or axial (central) symmetry with minimum fixed points, Figure 1.



(a) (b) (c)

Figure 1. (a) 4-Rotational symmetry (b) Axial symmetry (c) Central symmetry

2. PRELIMINARIES

A graph is considered as a pair *G=(V,c)* where *V* is a finite set of *nodes* and $c: V^2 \rightarrow \mathrm{N}$ is an arbitrary coloring of node-pairs. For simplicity, we assume $V = \{1,...,n\}$. In an undirected graph, we have $c(i, j) = c(j,i)$ for all $i, j \in V$. The *adjacency matrix A* of a graph $G = (V,c)$ is an $n \times n$-matrix with entry $c(i, j)$ at position (i, j).

A *permutation* of *V* is a bijective map $V \rightarrow V$. The set of permutations forms a group under composition, the symmetric group S_V; its natural element is the identity idv. We may denote S_V by S_n, since the symmetric groups of sets of the same cardinality are isomorphic.

An *automorphism* of *G* is a permutation π of *V* with $c(i, j) = c(\pi(i),\pi(j))$ for all $i, j \in V$. The set of automorphisms of *G* forms a group with respect to composition, denoted by Aut(*G*). The *order* of an automorphism π is $\mathrm{ord}(\pi) = \min\{k \in \mathrm{N} \mid \pi^k = idv\}$, where idv denotes the identity permutation of *V*. For a node $i \in V$, the set $\mathrm{orb}_\pi(i) = \{\pi^k(i) \mid k \in \mathrm{N}\}$ is the π-orbit of *i*. Finally, the fixed nodes are those in $\mathrm{Fix}(\pi) = \min\{i \in V \mid \pi(i) = i\}$.

A *reflection (axial symmetry)* of *G* is an autimorphism $\pi \in \mathrm{Aut(G)}$ with $\pi^2 = idv$, i.e., an automorphism of order 1 or 2 . For $k \in \{1,...,n\}$, a *k-rotation* of G is an automorphism $\pi \in \mathrm{Aut}(G)$ such that $|\mathrm{orb}_\pi(i)| \in \{1,k\}$ for all $i \in V$ and $|\mathrm{Fix}(\pi)| \leq 1$ if $k \neq 1$, i.e., an automorphism which cycles are all of the same length (greater than two) and such that the fixed points set is either empty or reduce to a single vertex. Each 2-rotation is a reflection, but not vice versa, and that the identity idv is both a reflection and a 1-rotation.

A *central symmetry* is an involved automorphism $\pi \in \mathrm{Aut}(G)$, such that the fixed points set of π is either included in *E* or reduced to a single vertex.

Now assume that there exist an injective placement $pl: V \rightarrow \Re^2$ and an isomorphism $\phi: \Re^2 \rightarrow \Re^2$ of the Euclidian plane with the following properties: For all $v \in V$ there exists a node $v' \in V$ with $\phi(pl(v)) = pl(v')$, and for $v, w \in V$ we have $c(v,w) = c(v',w')$; for straight line drawings of simple graphs, this means nodes are mapped to nodes and edges are mapped to edges. Then *pl* and ϕ induce an automorphism π of *G* by setting $\pi(v) = v'$. Any automorphism induced like this is called a *geometric automorphism* or a *symmetry* of *G*. An automorphism of a graph *G* is a symmetry if and only if it is a rotation or a reflection [16].

3. CANONICAL FORMS OF MATRICES AND SYMMETRY IN STRUCTURAL MECHANICS

In this section, an N×N symmetric matrix **M** is considered with all entries being real. For three canonical forms, the eigenvalues of [**M**] are obtained using the properties of its submatrices [20].

3.1 Canonical Form I

In this case, matrix M has the following pattern with N=2n :

$$[\mathrm{M}]=\begin{bmatrix} [\mathrm{A}]_{n\times n} & [0]_{n\times n} \\ [0]_{n\times n} & [\mathrm{A}]_{n\times n} \end{bmatrix}_{N\times N}$$

Considering the set of eigenvalues of the submatrix A as $\{\lambda \mathrm{A}\}$, the set of eigenvalues of [M] can be obtained as

$$\{\lambda \mathrm{M}\}=\{\lambda \mathrm{A}\}\,\overline{\cup}\,\{\lambda \mathrm{A}\}$$

Where $\overline{\cup}$ is the sign for the collection of the eigenvalues of the submatrices.

3.2 Canonical Form II

For this case, matrix [M] can be decomposed into the following form:

$$[\mathrm{M}]=\begin{bmatrix} [\mathrm{A}]_{n\times n} & [\mathrm{B}]_{n\times n} \\ [\mathrm{B}]_{n\times n} & [\mathrm{A}]_{n\times n} \end{bmatrix}_{N\times N}$$

The eigenvalues of [M] can be calculated as

$$\{\lambda \mathrm{M}\}=\{\lambda \mathrm{C}\}\,\overline{\cup}\,\{\lambda \mathrm{D}\}$$

where

$$\mathrm{C}=\{\mathrm{A}]+[\mathrm{B}] \text{ and } [\mathrm{D}]=[\mathrm{A}]-[\mathrm{B}]$$

3.3 Canonical Form III

This form has a $N\times N$ Form II submatrix augmented by k rows and columns, as shown in the following:

$$[\mathrm{M}]=\left[\begin{array}{ccc:ccc} & & & L_1 & \cdots & L_k \\ [A] & & [B] & L_1 & \cdots & L_k \\ & & & L_1 & \cdots & L_k \\ [B] & & [A] & L_1 & \cdots & L_k \\ \hdashline C(N+1,1) & . & C(N+1,N) & C(N+1,N+1) & \cdots & C(N+k,N+k) \\ . & . & . & . & . & . \\ . & . & . & . & \cdots & . \\ . & . & . & . & . & . \\ Z(N+1,1) & . & Z(N+1,N) & Z(N+1,N+1) & \cdots & Z(N+k,N+k) \end{array}\right]$$

Where [M] is a $(N+k)\times(N+k)$ matrix, with an $N\times N$ submatrix with the pattern of the Form II, and k augmented columns and rows. The entries of the augmented columns are the same in each column, and all the entries of [M] are real numbers. C(i,j) and Z(i,j) are arbitrary real numbers. The set of eigenvalues for [M] is obtained as

$$\{\lambda M\}=\{\lambda D\}\,\overline{\cup}\,\{\lambda E\}$$

3.4 Symmetry of the Forms I, II, III

Consider a symmetric graph *G* with an axis of symmetry. In the symmetry of Form I, the axis of symmetry does not pass through members and nodes. In this case, *G* is a disjoint graph and its components S_1 and S_2 are isomorphic subgraphs. In the symmetry of Form II, the axis of symmetry passes through members, and the graph G has an even number of nodes. For some graphs the axis of symmetry may pass through an even number of nodes, then one may still consider the graph as Form II by altering the axis of symmetry, while maintaining the topological symmetry of the model.

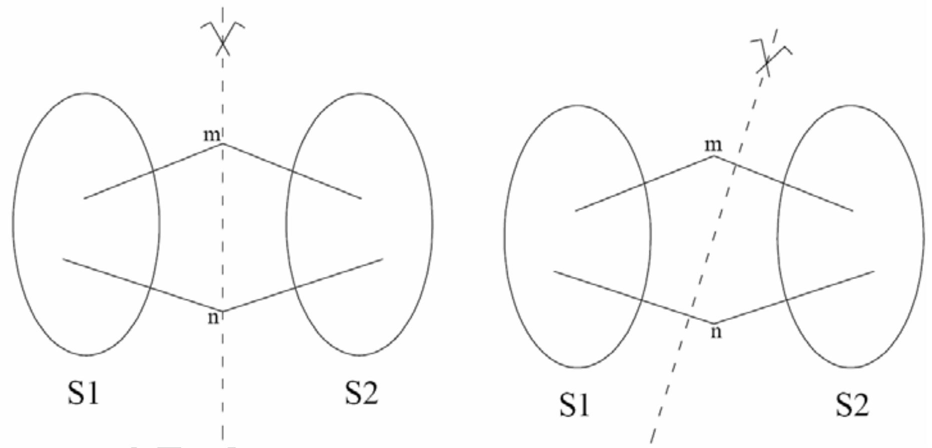


Figure 2. a) The axis of symmetry b) Altering the axis of symmetry

In the symmetry of Form III, the axis of symmetry passes through nodes. A combination of the Form II and III may also be identified in a model.

3.5 Factorization of symmetric graphs

Once the three types of symmetry are identified, the isomorphic subgraphs S_1 and S_2 are modified such that the union of eigenvalues of the Laplacian matrices of the two modified subgraphs becomes the same as the eigenvalues of the entire graph G. The process of the modification made to the subgraphs is called *healing* of the subgraphs, and the entire process may be considered as the *factorization* of a graph. For more detail about *healing* and *factorization* refer to [20].

4. EUCLIDIAN DISTANCES

Let X be a set of n nodes and d a distance among them. The distance d on a set X is *Euclidian*, if there exists an embedding of X in $\Re^{n-1}$ such that the restriction of the usual distance in $\Re^{n-1}$ coincides with the given distance d on X.

Given a distance d on a set X, it is classical to define the corresponding inner-product matrix W defined by

$$W_{i,j} = \frac{1}{2}(d^2(v_i,\cdot)) + d^2(v_j,\cdot) - d^2(\cdot,\cdot))$$

where

$$d^2(v_i,\cdot) = \frac{1}{n}\sum_{j=1}^{n} d^2(v_i,v_j)$$

$$d^2(\cdot,\cdot) = \frac{1}{n}\sum_{i=1}^{n} d^2(v_i,\cdot)$$

It is known that the distance d is Euclidian if and only if the matrix W is diagonalizable. In such a case, W defines a positive semi-definite bilinear form, and all its eigenvalues are positive or null. The maximum of non-null eigenvalues is $n-1$ (bound corresponding to the maximal dimension of the vector space generated by n points).

4.1 Defining a distance in a graph

As one of the first natural distances one can think of defining a distance between two vertices of a graph. This distance can be considered as the length of the shortest paths joining these two vertices. However, for the complete graph on 4 vertices if one edge deleted then the shortest paths do not define a Euclidian distance.

Though many Euclidian distances on abstract sets are defined previously, however, the Czekanovski-Dice distance seems to be the most appropriate to reveal the structure of a graph [24]. With respect to this distance, two vertices are close to each other if they have many common vertices:

For each pair (v_i,v_j) of vertices, we denote N_i the set of the neighbors of v_i:

$$N_i = \{v_k \in V,\, (v_i,v_j) \in E\} \cup \{v_i\}$$

Then, the distance d is defined by:

$$d^2(v_i,v_j) = \frac{|N_i \Delta N_j|}{|N_i| + |N_j|}$$

Where Δ denotes the symmetrical set-difference.

It should be noted that the distances are obviously preserved by any automorphism of the graph: Remark that a pair of non-djacent vertices having no common neighbor are at distance 1, and two adjacent vertices having the same neighbors are at distance 0. Therefore, there are non-isomorphic graphs having the same distances between their vertices.

5. EIGENSOLUTION

In this paper, we need only the last eigenvalue and eigenvector. Therefore the *Arnoldi method* has been used for eigensolution [21]. The Arnoldi method is an efficient procedure for approximating a subset of the eigensystem of a large sparse $n \times n$ matrix *N*. The Arnoldi method is a generalization of the Lancczos process and reduces to that method when the matrix *N* is symmetric [7]. After *z* steps the algorithm produces an upper Hessenberg matrix H_z of order *z*. The eigenvalues of this small matrix H_z are used to approximate a subset of the eigenvalues of the large matrix *N*. The matrix H_z is an orthogonal projection of *N* onto a particular *Krylov* subspace and the eigenvalues of H_z are usually called *Ritz values* or *Ritz approximations*. For more detail one can refer to [21,22].

6. AUTOMORPHISM

Let $G = (V, c)$ be a graph and assume $V = \{1, ..., n\}$, and let π be a permutation of *V*. Then a real n×n-matrix $M(\pi)$ is defined by

$$M(\pi)_{ij} = \begin{cases} 1 & \text{if } \pi(i) = j \\ 0 & \text{otherwise} \end{cases}$$

yielding a monomorphism (injective homomorphism) *M* of the group S_n of permutations of *V* into the general linear group $GL_n(\Re)$. The matrices in $M(S_n)$ are called *permutation matrices* and can be characterized as the set of n×n-matrices $X = (x_{ij})$ with $x_{ij} \in \{0,1\}$, where a value of 1 for the mapping variables x_{ij} is interpreted as the mapping node *i* to node *j*.

The condition of $\pi \in S_n$ being an automorphism of *G* should be translated into a condition on the corresponding matrix $M(\pi)$. For this purpose, consider the adjacency matrix *A* of *G*. The following lemma is used, which is proved in [16]:

Lemma 6.1. The permutation π of *V* is an automorphism of *G* if and only if the matrix $M(\pi)$ commutes with *A*, i.e., if $M(\pi)A = AM(\pi)$.

6.1 Rotation

Let π be an automorphism of *G* given by variables x_{ij}, i.e., let $M(\pi) = (x_{ij})$. The first condition for π to be a *k*-rotation is $|\text{orb}_\pi(i)| \in \{1, k\}$ for all $i \in V$ and any *k*-rotation has to meet the additional condition $|\text{Fix}(\pi)| \leq 1$.

Let $V = P_1 \oplus ... \oplus P_r$ be the partitioning of *V* according to *c* (fine-labeling). If $k \geq 3$, we know that at most one node may be fixed. Hence the size of every single part must be divisible by *k*, except for at most one part P_l and size one greater than a number divisible by *k.* If *k* does not meet this condition, we can omit the corresponding *k* as a choice. Observe that these feasibility criteria include the condition that *G* can only admit a *k*-rotation if *k* either divides *n* or *n-1*, so that the number of choices for *k* is bounded by number of divisors of either *n* or *n-1*. Let $d(n)$ be the number of divisors of *n*. By Hardy and Wright [23], we know that for each $\varepsilon > 0$ there is an integer n_0 such that

$$d(n) < 2^{(1+\varepsilon)\ln n/\ln\ln n} \quad \text{for all } n \geq n_0$$

For $\varepsilon = \log_2 e - 1$, we get $d(n) + d(n-1) \in O(n^{1/\ln\ln n})$. By Dirichlet, the average number of divisors of all numbers from one to *n* is asymptotic to ln *n* up to a small constant. Thus we have to perform algorithm A *O*(ln *n*) times on average.

In summery, Algorithm A finds an optimal rotational symmetry of *G*.

```
Algorithm A :
  For  k=n,…,3  do
      Let  feasibleorder = 1
      Let  onefixed = 0
      For  l=1,…,r  do
          If  k  does not divide |P_l|  then
              If  k  does not divide (|P_l| −1)  then
                  Let  feasibleorder = 0
              Else
                  If  onefixed = 1  then
                      Let  feasibleorder = 0
                  Else
                      Let  onefixed = 1
      If  feasibleorder = 1 then
          Graph G can has a k-rotation
          If  rotation detected then
              Stop
G does not have any rotational symmetry
```

6.2 Reflection (axial symmetry)

Reflections of *G* are much harder to find than k-rotations for $k \geq 3$, because we have to find fixed points which locate on axis of symmetry. In this type of symmetry, we have to ensure that the automorphism π to be represented satisfies $\pi^2 = \text{idv}$, i.e.,

$$x_{ij} = x_{ji}$$

for all nodes $i, j \in V$. In other words, we only need a single mapping variable for each pair of nodes. Also the number of fixed nodes is not bounded.

6.3 Centralization (central symmetry)

This type of symmetry is the same as axial symmetry, with a difference that the permutations in axial symmetry and central symmetry are different. Figure 3 shows 4 nodes of a graph which are specified as an orbit.

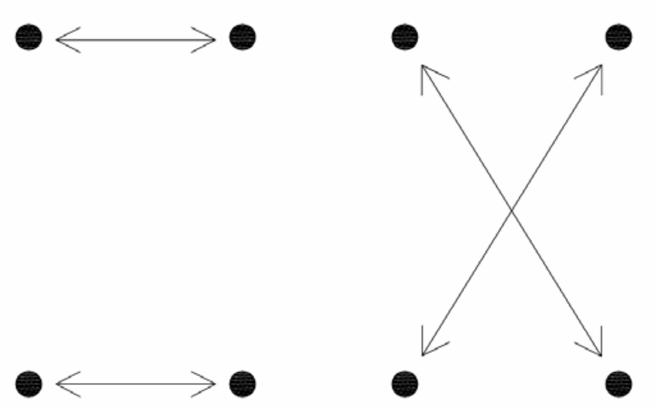


Figure 3. (a) Axial symmetry (b) Central symmetry

7. A SIMPLIFIED ALGORITHM FOR SYMMETRY DETECTION IN STRUCTURAL GRAPHS

The first step of the simplified algorithm is to compute the Czekanovski-Dice distance among the vertices of the graph and corresponding inner-product *W*. As the Czekanovski-Dice distance is Euclidian *W* is diagonalizable and its eiganvalues are all positive or null (for more detail refer to [19]). But for our purpose, just the last eigenvalue, which is always positive, and the corresponding eigenvector of matrix *W* are needed. For computing the last eigenvalue the *Arnoldi method*, which has been briefly introduced in Section 4, is applied. The dominant eigenvector of matrix *W* provide a suitable coloring for nodes of the graph *G* which easily can be used for finding favorite automorphism. In the dominant eigenvector, nodes with the same position have equal values. In order to illustrate this point, consider a graph with 23 nodes as shown in Figure 4.

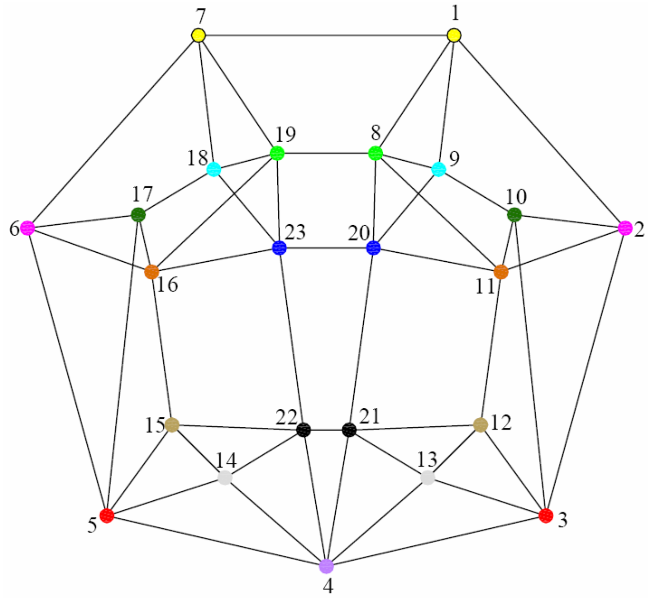


Figure 4. Coloring nodes according to the dominant eigenvector

The dominant eigenvector of matrix W corresponding to graph G is
Last eigenvalue = 8.7323
And the corresponding eigenvector is:

$$[0.2147,0.2125,0.2066,0.1975,0.2066,0.2125,0.2147,0.2043,0.2132,0.2127,0.2065,0.2116,$$
$$0.2098,0.2098,0.2116,0.2065,0.2127,0.2132,0.2043,0.203,0.2036,0.2036,0.203]^T$$

It is clear that nodes with the same position in graph G, have equal values in dominant eigenvector of matrix W.

According to the above described coloring, when we encounter orbits with orders greater than 2, there will be a problem of distinguishing nodes with the same color. This problem, which is described below, can be resolved using a proper initial ordering. Especially since we usually deal with regular graphs in structural analysis, a simple ordering can be utilized. For example, in this article clockwise ordering has been used.

In the next step, according to this coloring and initial ordering, different orbits, $P_1,\ldots,P_r$, can be specified. Now the partitioning of V according to this coloring is available. As an example, for the graph of Figure 2 the partitioning is

$$P_1=(1,7), P_2=(2,6), P_3=(3,5), P_4=(4), P_5=(8,19), P_6=(9,18), P_7=(10,17), P_8=(11,16),$$
$$P_9=(12,15), P_{10}=(13,14), P_{11}=(20,23), P_{12}=(21,22)$$

Now by these orbits, optimal symmetries can be detected. For each graph, first we try to find *k*-rotational symmetry and if no rotational symmetry is found, we try to find central symmetry or axial symmetry.

7.1 Rotational symmetry

According to partitions, $P_1,...,P_r$, which were specified in the previous step, the algorithm-A is used to find maximum order *k*-rotational symmetry. In summery, Algorithm B finds a maximum order *k*-rotational symmetry of *G*.

Algorithm B

```
For k=n,…,3 do
    If  k divide n  or  k divide n-1  then
        Check Algorithm A
If feasibleorder = 1
    If M(π) commutes with A
        G has k-rotational symmetry
    Else
        Use algorithm C to find central symmetry
Else (feasibleorder = 0)
    Use algorithm D to find axial symmetry
```

7.2 Central symmetry

In order to find the symmetry in this method, in two cases we use central symmetry:

1) If in the rotational algorithm *feasibleorder* equals to 1 and $M(\pi)$ does not commute with the A matrix.
2) If neither rotational nor axial symmetry is found in the graph.

It should be noted that in using central symmetry algorithm, correspondences are considered as explained in Section 6.3. In this type of symmetry, there is at most one fixed node.

Algorithm C finds central symmetry of *G*.

Algorithm C

```
Find all correspondences in V according to the pattern which was introduced in Section 6.3
According to these correspondences M(π) is formed.
If M(π) commutes with A
    G has central symmetry
```

7.3 Axial symmetry (reflection)

This type of symmetry is the most complicated problem in symmetry detection because the set of fixed nodes is not bounded and different cases can exist in a graph as it will be specified in this section.

1) The order of orbits in the graph are 1 or 2: In this case the axis of symmetry just passes through orbits of the 1st order or it also passes through some of orbits of the 2nd order.

If the axis passes through orbits of the 2nd order, the method explained in Section 3.3 is used (conservation of symmetrical properties by altering the axis of symmetry).

2) The order of the orbits in the graph are of 1st, 2nd, or a multiplier of 2nd order: When all the orbits of larger than 1st order, are also larger than 2nd order, then the graph will exhibit rotational or central symmetry. But when there is only one orbit of 2nd order, the graph will not exhibit either of the mentioned symmetries.

3) The order of the orbits in the graph are of 1st, 2nd, 3rd and …: In this case all orbits with odd order, contain one node on the symmetry axis and other points of these orbits will be on either side of the symmetry axis. This is particularly complex case and can cause a number of various cases, i.e., when just one node locates on the symmetry axis or odd number of nudes locate on symmetry axis. In most common case, in order to find the fixed node of each orbit with odd order (and greater than 2):
    a. Orbits of odd order (and greater than 2) are taken out of the graph.
    b. The ith orbit with odd order is added.
    c. A node (on the symmetry axis) whose elimination will result in a symmetrical graph is found.
    d. The next odd orbit with odd order (greater than 2) is dealt as above.
    e. Eventually all fixed nodes of the odd orbits of greater than 2 are distinguished.

According to the above cases, the axial symmetry can be determined using the below algorithm.

Algorithm D

D1. Graph *G* contains orbits with 1 or 2 nodes
  Find all correspondences in *V* and form $M(\pi)$ matrix
  Check whether $M(\pi)$ commutes with *A* or not.

D2. Graph *G* contains orbits with 1,2 or even number of nodes
  Find correspondences in orbits with 2 nodes
  Find correspondences in orbits with even number of nodes (greater than 2)
  (According to the pattern which was introduced in Section 6.3)
  Form $M(\pi)$ matrix and check whether $M(\pi)$ commutes with *A* matrix or not.

D3. Graph *G* contains orbits with 1,2,3,… nodes
  No = number of orbits with odd order (greater than 2) in graph *G*
  Remove all orbits with odd order (greater than 2) from graph *G* and consider as graph *H*
  For *k = 1,…,No* do
    Lo= Order *k*th odd orbit
    Add *k*th orbit with odd order to graph *H*
    For *l = 1,…,Lo* do
      Remove *l*th node of *k*th orbit with odd order
      Use Algorithm D2. and check whether graph *H* is symmetric
      If graph *H* is symmetric then
        Consider *l*th node of *k*th orbit with odd order as a fixed node
        break

8. EXAMPLES

In this section seven examples having different symmetries are investigated. Results are obtained by a Pentium4 1.6 GHz.

Example 1: A graph with 26 nodes is considered as shown in Figure 5.

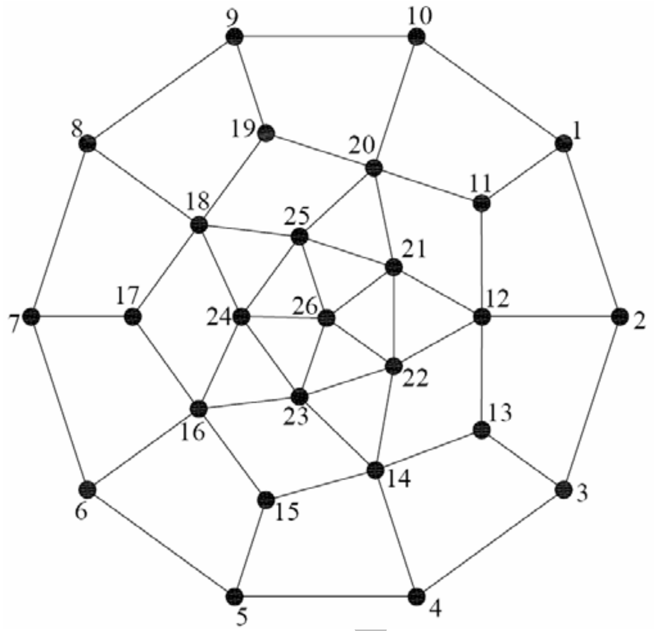


Figure 5. Graph *G* of Example 1

Results:

$Orbit_1 = (1,3,5,7,9)$, $Orbit_2 = (2,4,6,8,10)$, $Orbit_3 = (11,13,15,17,19)$, $Orbit_4 = (12,14,16,18,20)$, $Orbit_5 = (21,22,23,24,25)$, $Orbit_6 = (26)$

Graph G has 5- rotational symmetry with one fixed node, according to the following symmetry permutation:

$$Permutation = (1,3,5,7,9)(2,4,6,8,10)(11,13,15,17,19)(12,14,16,18,20)(21,22,23,24,25)(26)$$

Computational time = 0.19 (sec)

Example 2: A graph with 41 nodes is considered as shown in Figure 6

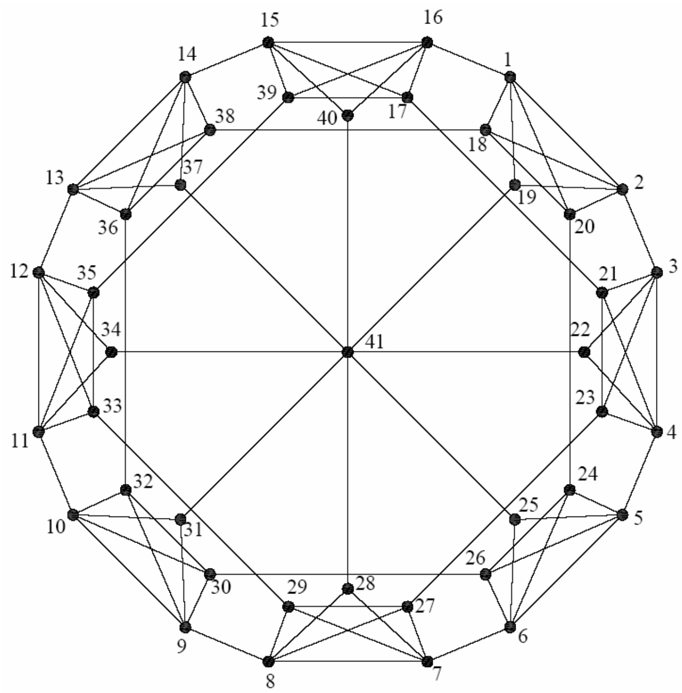


Figure 6. Graph G of Example 2

Results:

Graph G has 8- rotational symmetry with one fixed node, according to the following symmetry permutation:

$$Permutation = (17,18,20,21,23,24,26,27,29,30,32,33,35,36,38,39)(19,22,25,28,31,34,37,40)(41)(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16)$$

Computational time = 0.49 (sec)

Example 3: A graph with 12 nodes is considered as shown in Figure 7**.**

Results:

$$Orbit_1 = (1,3)\ ,\ Orbit_2 = (2,4)\ ,\ Orbit_3 = (5,6,8,9)\ ,\ Orbit_4 = (7,10)\ ,\ Orbit_5 = (11,12)$$

Graph G has a central symmetry, according to the following symmetry permutation:

$$Permutation = (1,3)(2,4)(6,9)(5,8)(7,10)(11,12)$$

Computational time = 0.05 (sec)

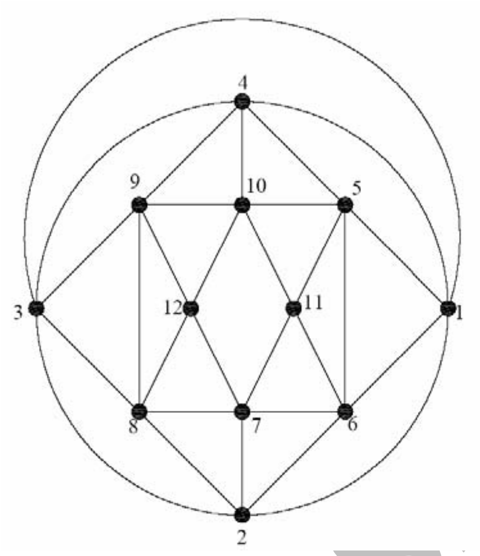


Figure 7. Graph *G* of Example 3

Example 4: A graph with 12 nodes is considered as shown in Figure 8.
Results:

$$Orbit_1 = (1,2,3,4)\ ,\ Orbit_2 = (7,8,11,12)\ ,\ Orbit_3 = (5,6,9,10)$$

Graph G has a central symmetry, according to the following symmetry permutation:

$$Permutation = (1,3)(2,4)(5,9)(6,10)(7,11)(8,12)$$

Computational time = 0.05 (sec)

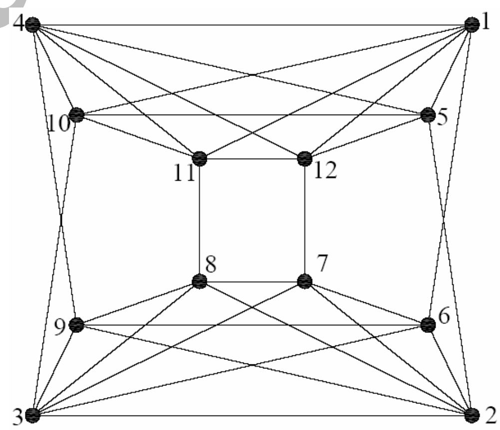


Figure 8. Graph *G* of Example 4

Example 5: A graph with 29 nodes is considered as shown in Figure 9**.**
Results:
Graph G has an axial symmetry, according to the following symmetry permutation:
$Permutation = (1,3)(4,19)(5,18)(2)(6,17)(7,16)(8,15)(9,14)(10,13)(22,29)(20)(23,28)(21)...$
$(22,29)(25,26)(24,27)$

Computational time = 0.28 (sec)

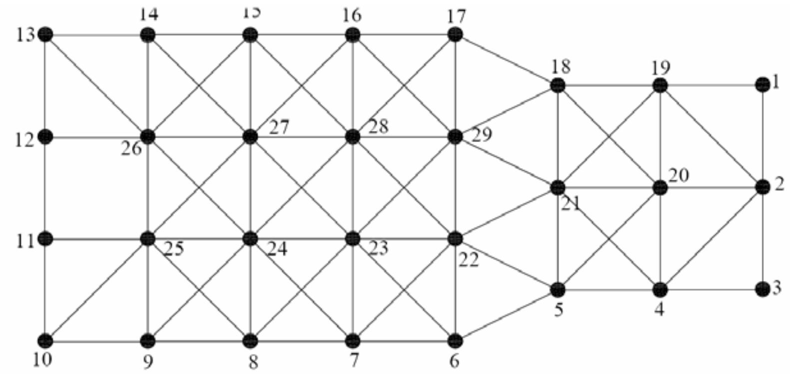


Figure 9. Graph G of Example 5

Example 6: A graph with 17 nodes is considered as shown in Figure 10**.**
Results:
Graph G has axial symmetry with one node on the axis of symmetry, according to the following symmetry permutation:
$Permutation = (1,10)(2,9)(3,8)(4,7)(5,6)(11)(12,17)(13,16)(14,15)$
Computation time = 0.08 (sec)

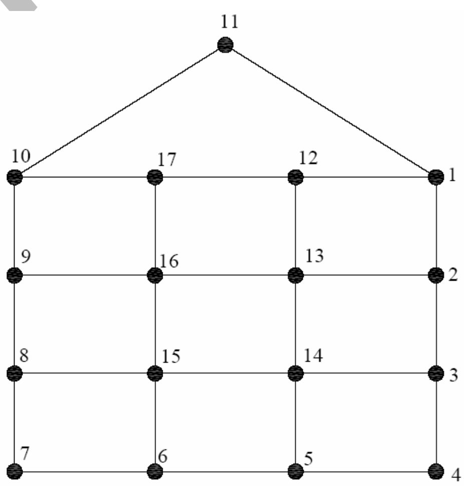


Figure 10. Graph *G* of Example 6

Example 7: A graph with 20 nodes is considered as shown in Figure 11**.**

Results:
$Orbit_1 = (9,14)$, $Orbit_2 = (3,4)$, $Orbit_3 = (1,2,5,6,7)$, $Orbit_4 = (8,10,13,15)$, $Orbit_5 = (11,12)$, $Orbit_6 = (16)$

Graph G has axial symmetry with one orbit of odd order (greater than 2), according to the following symmetry permutation:

$Permutation = (1,6)(2,5)(3,4)(7)(8,15)(9,14)(10,13)(11,12)(16)$
Computational time = 0.12 (sec)

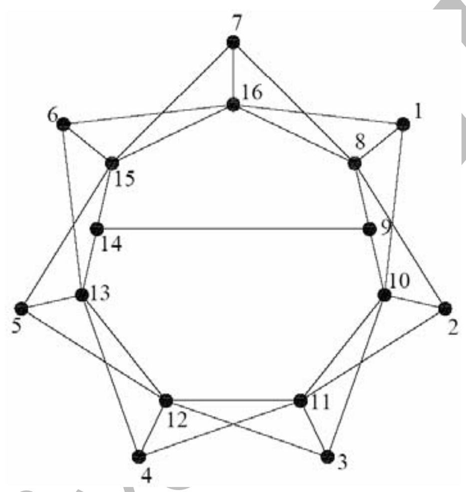


Figure 11. Graph *G* of Example 7

9. CONCLUDING REMARKS

The problem of detecting symmetries in structural graph models in an automatic manner is an important issue, since symmetry has been utilized as a common tool for decreasing the computational time and storage in static, dynamic and stability analyses. Using the present method, symmetries of a graph can be detected, however, for a proper drawing additional considerations are needed. The latter is not an important restriction for our problem since in structural analysis the graph models are often neither complicated nor irregular. A proper initial nodal ordering of graphs plays an important role which can be performed by graph theoretical algorithms similar to those of [20].

REFERENCES

1. Hargittai, I., *Symmetry; Unifying Human Understanding*, Pergamon Press Ltd, UK, 1986.
2. Gruber, B., *Symmetries in Science VIII*, Plenum Press, NY, 1995.

3. Glockner, P.G., Symmetry in structural mechanics, *ASCE, Journal of Structural Division*, **99**(1973) 71-89.
4. Zingoni, A., Group-theoretical applications in solid and structural mechanics: a review, Chapter 12 in *Computational Structures Technology*, Edited by BHV Topping and Z. Bittnar, Saxe-Coburg Publication, UK, 2002.
5. Zingoni, A., Pavlovic MN, and Zlokovic GM. A symmetry-adapted flexibility approach for multi-storey space frames: General outline and symmetry-adapted redundants, *Structural Engineering Review*, **7**(1995) 107-119.
6. Livesley, R.K., *Mathematical Methods for Engineers*, Ellis Horwood Series in Mathematics and its Applications, UK, 1989.
7. Jennings, A., and McKeown, JJ. *Matrix Computation*, John Wiley and Sons, UK, 1992.
8. Bathe, K.J., and Wilson, EL. *Numerical Methods for Finite Element Analysis*, Englewood Cliffs, Printice Hall, USA, 1976.
9. Kaveh, A., and Sayarinejad MA. Eigen solutions for matrices of special patterns, *Communications in Numerical Methods in Engineering*, **19**(2003)125-136.
10. Kaveh, A., and Sayarinejad MA. Eigen solutions for factorable matrices of special patterns, *Communications in Numerical Methods in Engineering,* **20**(2004)133-146.
11. Kaveh, A., and Sayarinejad MA. Graph symmetry in dynamic systems, *Computers and Structures*, **82**(2004)2229-2240.
12. Kaveh, A., and Salimbahrami B. Eigen solutions of symmetric frames using graph factorization, *Communications in Numerical Methods in Engineering*, **20**(2004)889-910.
13. Kaveh, A., and Rahami H. New canonical forms for analytical solution of problems in structural mechanics, *Communications in Numerical Methods in Engineering*, No. 9, **21**(2005) 499-513.
14. Kaveh, A., Shahryari L., Buckling load of planar frames with semi-rigid joints using weighted symmetric graphs, *Computers and Structures*, **85**(2007)1704-1728.
15. Abelson, D., Hong, S-H., and Taylor, D.E., A group theoretic method for drawing graphs symmetrically. 2001.
16. Buchheim, C., Junger, M., Detecting symmetry by branch and cut, University of Koln, Germany, 2001.
17. Kutnara, K., Malniˇc, A., Marušiˇc, D. and Miklaviˇc Š. Distance-balanced graphs: Symmetry conditions, *Discrete Mathematics*, **306**(2006) 1881-1894.
18. Buchheim, C., Hong, S-H., Crossing Minimization for Symmetries.
19. Fraysseix, H., A heuristic for graph symmetry detection, *Lecture Notes in Computer Science 1731*, Springer Verlag, 1999.
20. Kaveh, A., *Structural Mechanics: Graph and Matrix Methods*, 3rd edition, *RSP*, Somerset, UK, 2004.
21. Lehoucq, R.B., Sorensen, D.C., Deflation techniques for an implicitly re-started arnoldi iteration, *SIAM J. Matrix Analysis and Applications*, **17**(1996) 789-821.
22. Lehoucq, R.B., Sorensen D.C. and Yang C., Solution of large-scale Eigen value problems with implicitly restarted Arnoldi methods, SIAM Publications, Philadelphia, 1998.
23. Hardy, G.H., Wright EM. *An Introduction to the Theory of Numbers*, Clarendon Press, 1938.
24. De Fraysseix, H., Kuntz, P., *Pagination of large scale networks*, Algorithms Review 2, No. 3, (1992) 105-112.