Computer Assisted Mechanics and Engineering Sciences, **16**: 101–115, 2009.

Formation of graph models for regular finite element meshes

Ali Kaveh
Centre of Excellence for Fundamental Studies in Structural Engineering
Iran University of Science and Technology, Tehran-16, Iran

Kambiz Koohestani
Department of Civil Engineering, University of Tabriz, Tabriz, Iran

(Received in the final form September 21, 2009)

Graph theory has many applications in structural mechanics and there are also numerous topological transformations which make the related problems simpler. The skeleton graph and natural associate graph of finite element models are among such transformations. These transformations can efficiently be used for nodal and element ordering of regular finite element models. Natural associate graph and its mesh basis play a key role in optimal finite element analysis by combinatorial force method. In this paper, an efficient method is presented for generation of skeleton graph, natural associate graph as well as their mesh bases for finite elements models, using graph and digraph products.

Keywords: finite elements, graph products, digraph products, associate graph, mesh bases

1. Introduction

In structural mechanics, it is useful to differentiate the geometrical and topological properties of structural models. Such a distinction usually reduces the complexity of the related problems. For example, the bandwidth minimization of structural stiffness and flexibility matrices is completely dependent on the connectivity properties (topology) and is independent of the geometry of the model. Graph theory has a leading role in the study of topological properties of structures and there are useful correspondences between elements of this theory and different structural problems.

Different applications of graph theory in the structural mechanics are well documented in [1]. For some of these applications one can refer to ordering algorithms developed by Cuthill-McKee [2], Kaveh [3] and Gibbs et al. [4] which are closely related to the selection of special spanning trees of graphs. Also mesh basis of a graph can be used for efficient analysis of skeletal structures by the force method [3]. The application of natural associate graph (NAG) of finite element (FE) models has been suggested and used by Kaveh [5], and Feneves and Law [6] as a useful topological transformation for nodal and element ordering of FE models. This graph was also used by Kaveh et al. [7] and Kaveh and Koohestani [8] for efficient FE analysis by combinatorial force method. The skeleton graph (SG) of FE models was also used for nodal ordering [1].

In this paper applications of these two graphs, namely the SG and the NAG, are presented for ordering, partitioning and efficient FE analysis. A simple method is presented for the generation of these graphs using graph products concepts. A model which can be expressed as the product of two or three subgraphs is called a regular model. A simple method is also presented for the formation of mesh bases of the associate graphs of regular FE meshes.

2. Skeleton and natural associate graphs

A natural associate graph (NAG) of a finite element (FE) model is a graph whose nodes are in one to one correspondence with the FEs and two nodes are connected if the corresponding elements have a common boundary. As an example, the NAG of the FE model shown in Fig. 1a is illustrated in Fig. 1b.

a) b)

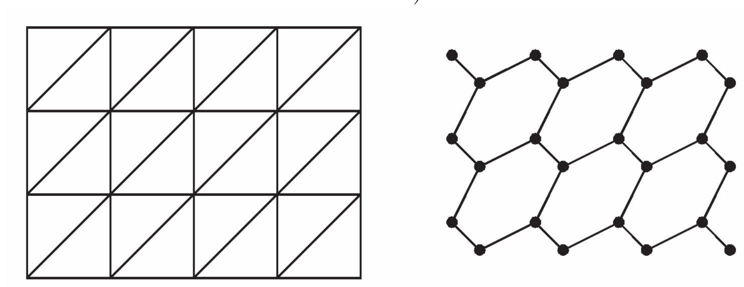


Fig. 1. a) a triangular FE mesh, b) corresponding NAG

This graph can efficiently be used for nodal and element ordering of finite elements [6]. Also, generation of localized self-equilibrating systems for analysis of FE models by combinatorial force method are highly dependent on this graph [7, 8]. However, for generation of this graph, a test should be performed for each element having possible common boundary with the other elements. For large models this process requires a considerable computational time.

The 1-skeleton graph (SG) of a FE model is a graph whose vertices are the same as nodes of the FEM, and its members are the edges of the FE model. This graph is shown in Fig. 2 for FE model of Fig. 1a.

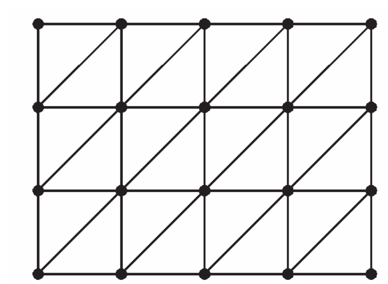


Fig. 2. The skeleton graph (SG) of the FEM in Fig. 1a

3. Graph products

Graph products were developed in the past 50 years (see e.g. Berge [9], Sabidussi [10], Harary and Wilcox [11] and Imrich and Klavzar [12]) and employed in nodal ordering and domain decomposition of regular structures by Kaveh and Rahami [13, 14]. Special graph and digraph products are introduced by Kaveh and Koohestani [15] for efficient configuration processing of different structural models.

Graph products are generally created by employing Boolean or logical operators on two graphs. These operators are usually established based on the equality and adjacency conditions. By changing these conditions, it is possible to make different products. However, among the different graph products, Cartesian, Strong Cartesian, Direct and Lexicographic products are well known. Spectral properties of these graphs as well as their relations with ordering and partitioning of regular structures were previously studied [13, 14]. However, new products which were introduced in [15] have capability of generating complex topologies. For example SG of regular triangular FE models can simply be generated using digraph products. The operator of this product is similar to Strong Cartesian product but their generators are digraphs and by changing the direction of edges of the generators different topologies can be achieved.

In the following a review of the classic graph products and three products which are suitable for the formation of SG of FEs are presented. Also three new digraph products are introduced where one of them is designed for SG of six-node triangular FEs, and the other two are suitable for the formation of the NAG of triangular meshes.

3.1. Classic products of graphs

In general the product of two graphs is a new graph which is completely generated using different combination of equality and adjacency conditions. There are different types of graph products; however, among them the most commonly used products are Cartesian [10], strong Cartesian, Lexicographic [11] and Direct products [16]. A complete description of these can be found in Kaveh [1].

In Table 1 brief definitions of these products are presented for product of two graphs $\mathbf{G}$ with node set denoted by $\{g\}$ and $\mathbf{H}$ with node set denoted by $\{h\}$. Note that the number of nodes of the constructed graph product is $N_G \times N_H$, in which N_G and N_H are total number of nodes of the

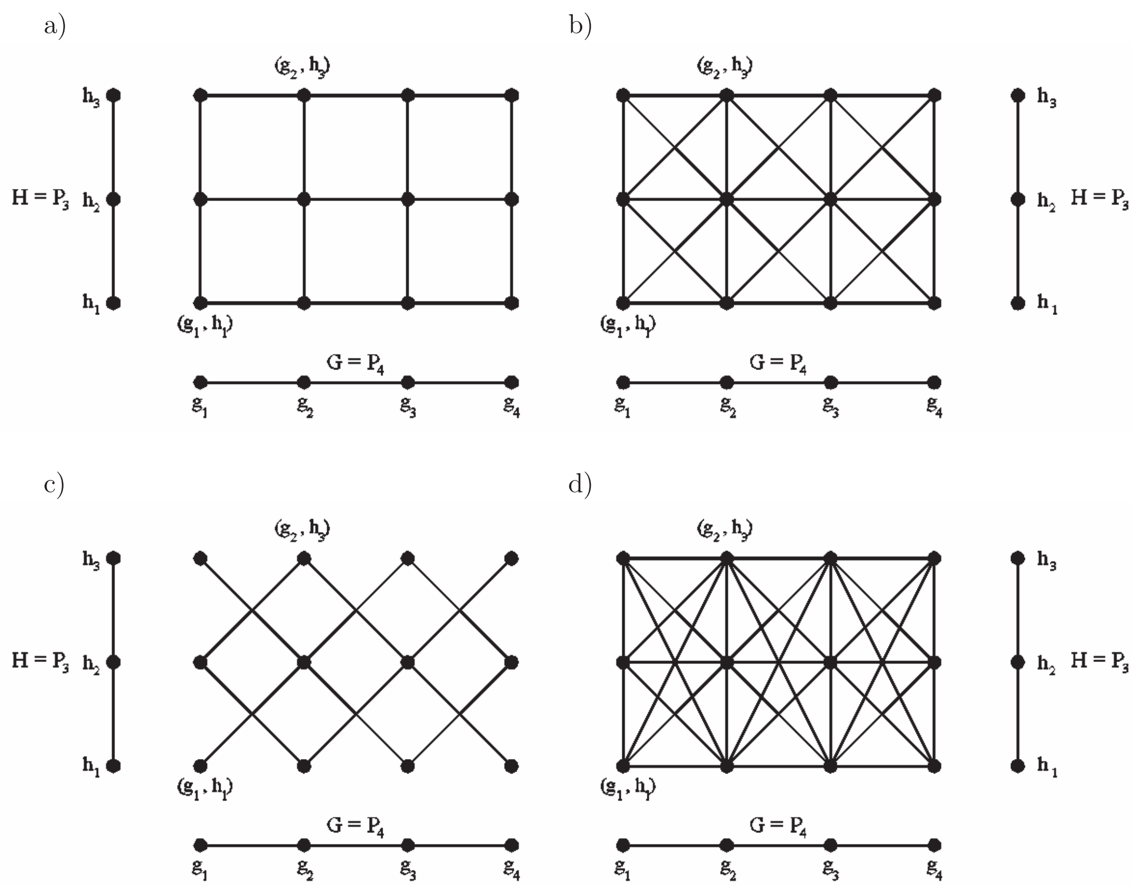


Fig. 3. Classic products of two graphs: a) $\mathbf{G(X)_C H}$, b) $\mathbf{G(X)_{SC} H}$, c) $\mathbf{G(X)_D H}$, d) $\mathbf{G(X)_{LE} H}$

Table 1. Definitions of classic graph products

Graph product name	Symbol	Definition
Cartesian	$(\mathbf{X})_{\mathbf{C}}$	$(g = g'\ \&\ h\,.\text{adj.}\,h')$ or $(h = h'\ \&\ g\,.\text{adj.}\,g')$
Strong Cartesian	$(\mathbf{X})_{\mathbf{SC}}$	$(g = g'\ \&\ h\,.\text{adj.}\,h')$ or $(h = h'\ \&\ g\,.\text{adj.}\,g')$ or $(g\,.\text{adj.}\,g'\ \&\ h\,.\text{adj.}\,h')$
Direct	$(\mathbf{X})_{\mathbf{D}}$	$(g\,.\text{adj.}\,g'\ \&\ h\,.\text{adj.}\,h')$
Lexicographic	$(\mathbf{X})_{\mathbf{LE}}$	$(g\,.\text{adj.}\,g')$ or $(g = g'\ \&\ h\,.\text{adj.}\,h')$

generators $\mathbf{G}$ and $\mathbf{H}$, respectively. In summary, for each product the adjacency of resulted nodes (g, h) and (g', h') are entirely determined through true or false evaluation of conditions presented in Table 1. On the other hand only a true value leads to the generation of an edge between (g, h) and (g', h') in the resulted product.

In Table 1 'adj' and '&' means 'adjacent' and 'and' respectively and the symbols used here are in agreement with those introduced in [15]. Also, in Fig. 3 these products are shown for product of two path graphs $\mathbf{G} = \mathbf{P}_4$ (a path with four nodes) and $\mathbf{H} = \mathbf{P}_3$ (a path with three nodes).

3.2. Graph product of Type 1

Type 1 graph product is in fact a digraph product because its generators $(\mathbf{G}, \mathbf{H})$ are directed graphs. For this product which is denoted by $(\mathbf{X})_1$ the conditions for the connectedness of two nodes (g, h) and (g', h') are as follows:

$$\begin{aligned}&(g = g'\ \&\ (h\,.\text{adj.}\,h' \ \text{or}\ h'.\text{adj.}\,h))\ \text{or}\\&(h = h'\ \&\ (g\,.\text{adj.}\,g' \ \text{or}\ g'.\text{adj.}\,g))\ \text{or}\\&((g\,.\text{adj.}\,g'\ \&\ h\,.\text{adj.}\,h')\ \text{or}\ (g'.\text{adj.}\,g\ \&\ h'.\text{adj.}\,h))\end{aligned} \tag{1}$$

The constructed graphs using this product have also $N_G \times N_H$ nodes. In this product, by changing the direction of the edges of the generators, models with different triangulations can be generated. In Fig. 4 four different topologies are presented which are constructed using this product. This product has also been previously introduced in [15] and used for configuration processing of different structural models.

3.3. Graph product of Type 2

This product which is denoted by $(\mathbf{X})_2$ is applied to undirected graphs. However, the generators of this product ($\mathbf{G}$ and $\mathbf{H}$ graphs) has loop in some nodes. For this product, there is an edge between nodes (g, h) and (g', h') if at least one of the following conditions holds ($\sim$ means not):

$$\begin{aligned}&(g = g'\ \&\ g.\sim\text{adj.}\,g(\text{no loop})\ \&\ h\,.\text{adj.}\,h')\ \text{or}\\&(h = h'\ \&\ h.\sim\text{adj.}\,h(\text{no loop})\ \&\ g\,.\text{adj.}\,g')\end{aligned} \tag{2}$$

The constructed graph by these rules has $N_G \times N_H$ nodes. However, some of nodes are isolated and have no incident edges. Such nodes can be removed from the resulted graph. In Fig. 5 an example of this product is illustarted.

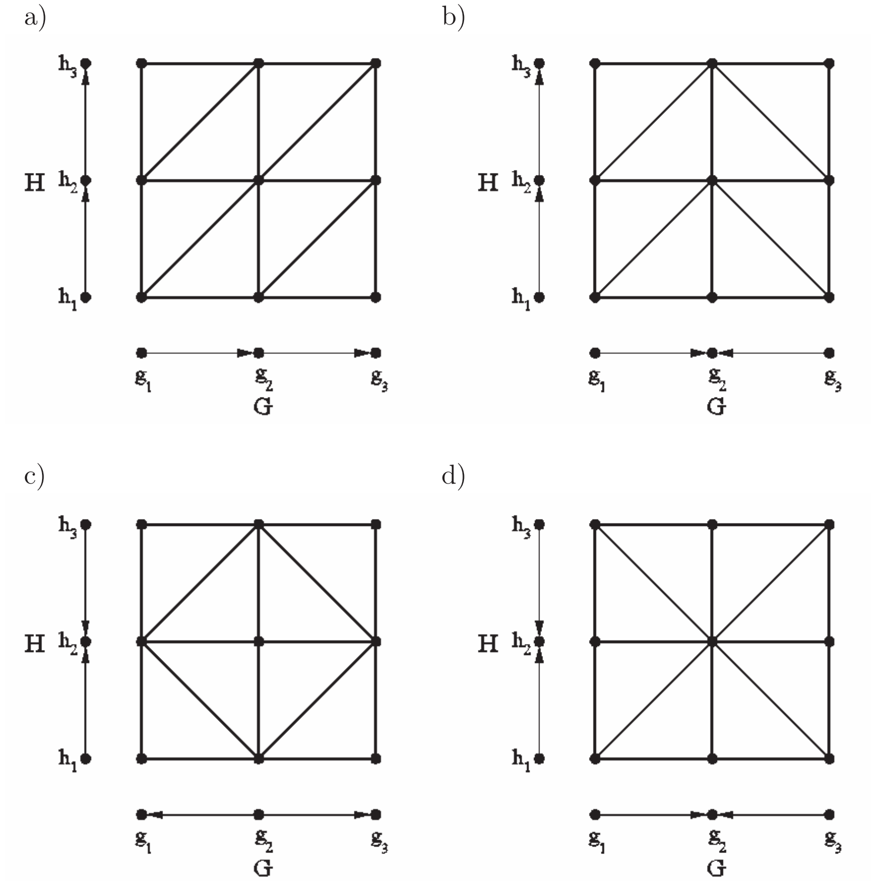


Fig. 4. Type 1 digraph product: $\mathbf{G}(\mathbf{X})_1\,\mathbf{H}$

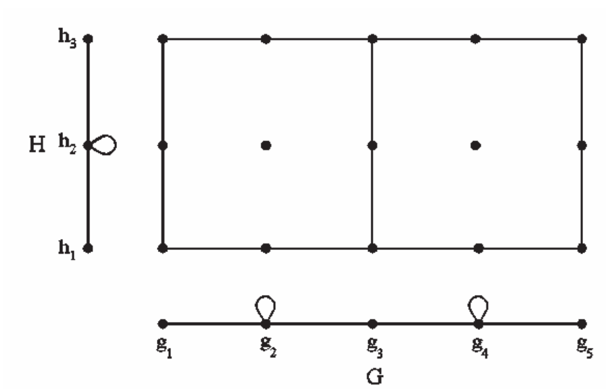


Fig. 5. An example of graph product of type 2: $\mathbf{G}(\mathbf{X})_2\,\mathbf{H}$

3.4. Graph product of Type 5

The graph products of Type 3 and Type 4 which were previously introduced in [15] are not suitable for FE models. However, in this section a new digraph product is presented which is useful for generation of SG of regular six-node triangular FE meshes.

This product which is named as graph product of Type 5 and denoted by $(\mathbf{X})_5$ somehow is a combination of Type 1 and 2 products. Therefore, its generators ($\mathbf{G}$ and $\mathbf{H}$ graphs) are directed and have loops in special nodes. The constructed graph has $N_G \times N_H$ nodes and condition for connectedness of two nodes (g, h) and (g', h') is as follows:

$$
\begin{aligned}
&(g = g' \ \& \ g\,.\sim \text{adj.}\, g(\text{no loop}) \ \& \ (h\,.\text{adj.}\, h' \ \text{ or } \ h'\,.\text{adj.}\, h)) \ \text{ or} \\
&(h = h' \ \& \ h\,.\sim \text{adj.}\, h(\text{no loop}) \ \& \ (g\,.\text{adj.}\, g' \ \text{ or } \ g'\,.\text{adj.}\, g)) \ \text{ or} \\
&(((g\,.\text{adj.}\, g' \ \& \ h\,.\text{adj.}\, h') \ \text{ or } \ (g'\,.\text{adj.}\, g \ \& \ h'\,.\text{adj.}\, h)) \ \& \\
&\qquad ((g, h(\text{no loop}) \ \& \ g', h'(\text{loop}))) \ \text{ or } \ (g, h(\text{loop}) \ \& \ g', h'(\text{no loop})))
\end{aligned} \tag{3}
$$

It is useful to note that, in this product two directed edges are related to each FE and a loop is located between these edges. Thus, any changes on direction of edges should be performed similarly for such two edges. However, the total number of FEs is equal to $2L_G \times L_H$ in which L_G and L_H are the total number of loops of the generators **G** and **H**, respectively. An example of such a product is illustrated in Fig. 6.

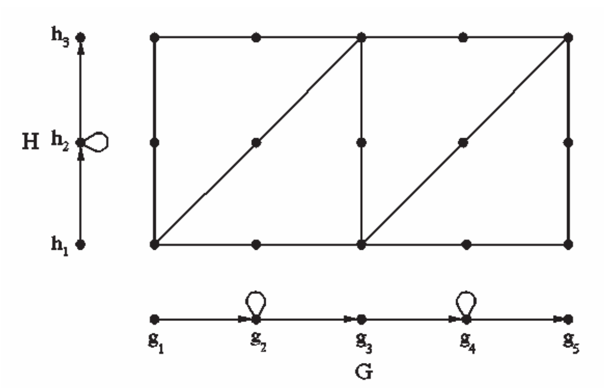


Fig. 6. An example of graph product of type 5: $\mathbf{G}(\mathbf{X})_5\,\mathbf{H}$

The presented graph products up to now are suitable for generation of the SG of regular FE models. In the next section, two new digraph products are presented. Using these products, the NAG of triangular FE meshes can simply be constructed.

4. New digraph products for associate graphs of FE models

In this section, considering useful applications of NAG of FE models, two new digraph products are presented which lead to the simple generation of NAG of the regular triangular FEs.

4.1. Graph product of Type 6

This product which is named "graph product of Type 6" and is denoted by $(\mathbf{X})_6$ is also a Boolean operator which is applied to subgraphs (generators **G** and **H**) having only directed edges. However, the constructed graph has $N_G \times N_H$ nodes and the connectedness for connectivity of nodes (g, h) and (g', h') is as follows:

$$
\begin{aligned}
&(h = h' \ \& \ (g\,.\text{adj.}\, g' \ \text{ or } \ g'.\text{adj.}\, g)) \ \text{ or} \\
&((h\,.\text{adj.}\, h' \ \& \ g\,.\text{adj.}\, g' \ \& \ g'.\sim \text{adj.}\, g) \ \text{ or } \ (h'.\text{adj.}\, h \ \& \ g'.\text{adj.}\, g \ \& \ g\,.\sim \text{adj.}\, g'))
\end{aligned} \tag{4}
$$

It should be noted that, this product is not symmetric and thus for desired results, generators should carefully be selected. As it can be seen from the connectivity conditions, the generator which is chosen as **G** has double directed edges, and edges of graph **H** has always simple directed edges. Such a distinction simplifies the selection of appropriate graphs as generators. An example of this product is shown in Fig. 7.

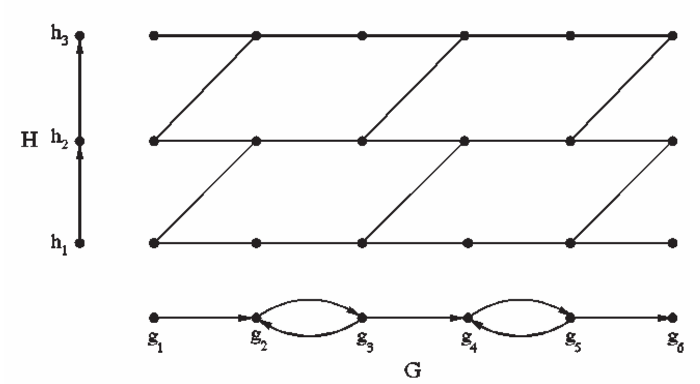


Fig. 7. Type 6 graph product: $\mathbf{G}(\mathbf{X})_6\,\mathbf{H}$

4.2. Graph product of Type 7

This product is another Boolean operator which is named "graph product of type 7" and denoted by $(\mathbf{X})_7$. This product is partially similar to Type 6 product having an additional condition. Also, the generator which is denoted by $\mathbf{G}$ has loop in some nodes as well as directed edges, and graph $\mathbf{H}$ has only simple or double directed edges. This product is also not symmetric and constructed graph has $N_G \times N_H$ nodes. The conditions for connectivity of two nodes (g, h) and (g', h') is as follows:

$$\begin{aligned}&(h = h' \ \ \& \ \ (g\,.\mathrm{adj.}\,g' \ \text{ or } \ g'.\mathrm{adj.}\,g)) \ \text{ or}\\&((h\,.\mathrm{adj.}\,h' \ \ \& \ \ g\,.\mathrm{adj.}\,g' \ \ \& \ \ g'.\sim \mathrm{adj.}\,g) \ \text{ or } \ (h'.\mathrm{adj.}\,h \ \ \& \ \ g'.\mathrm{adj.}\,g \ \ \& \ \ g.\sim \mathrm{adj.}\,g')) \ \text{ or}\\&(g = g' \ \ \& \ \ g.\sim \mathrm{adj.}\,g(\text{no loop}) \ \ \& \ \ (h\,.\mathrm{adj.}\,h' \ \text{ or } \ h'\,.\mathrm{adj.}\,h))\end{aligned} \tag{5}$$

In Fig. 8 an example of this product is shown. Using this product and by changing the direction of edges and location of the loops, NAG of two usual FE models can be constructed. Examples of these models are presented in Sec. 7.

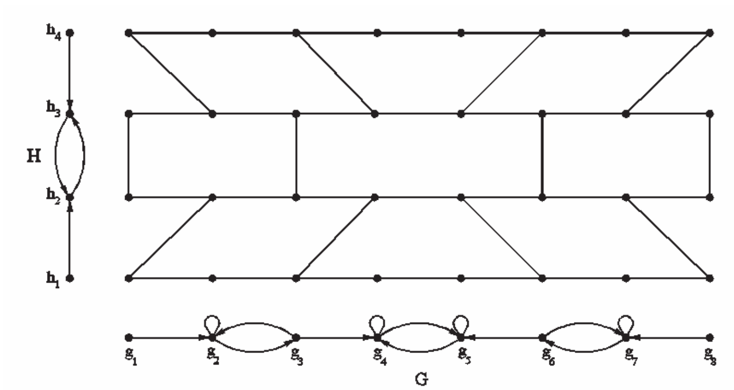


Fig. 8. Type 7 graph product: $\mathbf{G}(\mathbf{X})_7\,\mathbf{H}$

4.3. Identification of different FE models

In this section, different models which discretized by 4- and 8-node rectangular and 3- and 6-node triangular FE are identified by assigning an element number to different directions. Let n_x, n_y be

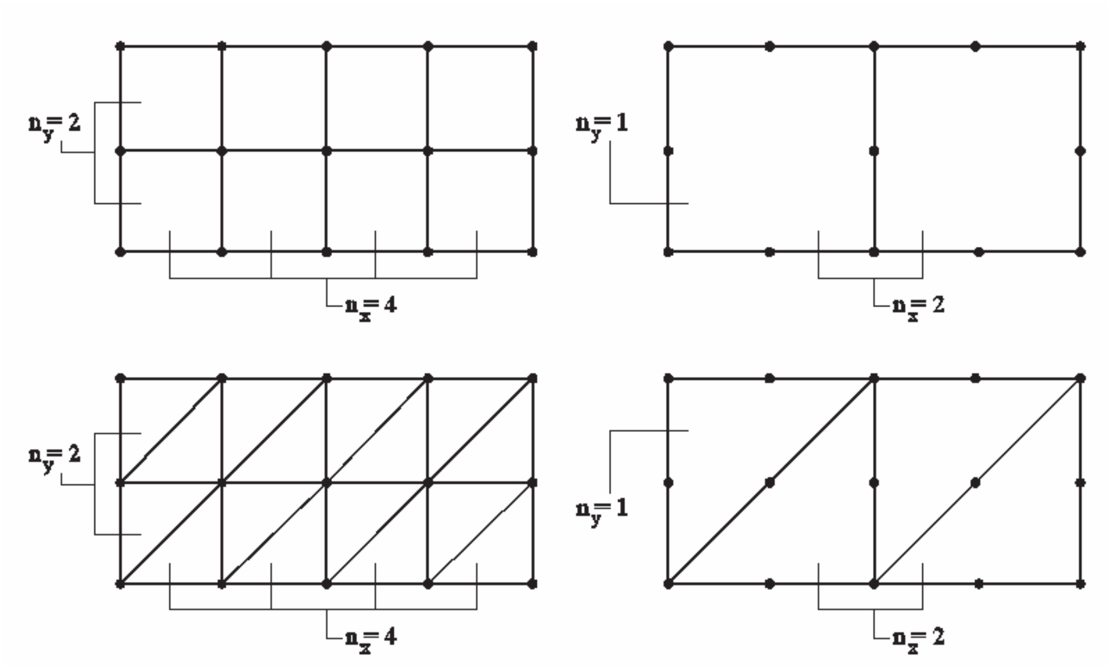


Fig. 9. Definition of n_x, n_y for different models

the number of FEs (triangular or rectangular) in **x** and **y** direction of a rectangular area. However, if discretization is performed using triangular elements, the number of elements which have an edge in a boundary should be considered. For example, in Fig. 9 these numbers are presented for different meshes.

The mathematical representations of different models which are presented in the next section are completely dependent on these numbers. The complete rules for representation of different products were previously presented in [15] and for the sake of briefness are not repeated in here.

5. Mathematical representation of SG and NAG of different FE models

In this section, the mathematical representation of SG and NAG of different regular FE models are presented. All relations are dependent on n_x, n_y of the models and are presented separately for rectangular and triangular FE models.

4-node rectangular $n_x \times n_y$ mesh:

$$\begin{aligned} \mathbf{SG} &= \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H}, \qquad && \mathbf{G} = \mathbf{P}_{n_x+1}, \qquad && \mathbf{H} = \mathbf{P}_{n_y+1}, \\ \mathbf{NAG} &= \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H}, && \mathbf{G} = \mathbf{P}_{n_x}, && \mathbf{H} = \mathbf{P}_{n_y}. \end{aligned} \tag{6}$$

8-node rectangular $n_x \times n_y$ mesh:

$$\begin{aligned} \mathbf{SG} &= \mathbf{G}(\mathbf{X})_{2}\,\mathbf{H}, \qquad && \mathbf{G} = \mathbf{P}^{e}_{2n_x+1}, \qquad && \mathbf{H} = \mathbf{P}^{e}_{2n_y+1}, \\ \mathbf{NAG} &= \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H}, && \mathbf{G} = \mathbf{P}_{n_x}, && \mathbf{H} = \mathbf{P}_{n_y}. \end{aligned} \tag{7}$$

3-node triangular $n_x \times n_y$ mesh (topologies similar to Fig. 4a or 4b):

$$\begin{aligned}&\mathbf{SG} = \mathbf{G}(\mathbf{X})_1\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}_{n_x+1}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}_{n_y+1},\\ &m1 = (1 : n_x), \quad m2 = (1 : n_y),\end{aligned} \tag{8}$$

$$\begin{aligned}&\mathbf{NAG} = \mathbf{G}(\mathbf{X})_6\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}_{2n_x}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}_{n_y},\\ &m1 = (1 : 2 : 2n_x - 1,\, 2 : 2 : 2n_x - 2_d), \quad m2 = (1 : n_y - 1).\end{aligned} \tag{9}$$

In $m1$ and $m2$ positive and negative signs should be assigned according to topology of the triangulation. Also the symbol $\mathbf{D}$ is used before $\mathbf{P}$ in Eqs. 8 and 9 refering to directed path where its index control the direction and number of edges between nodes (see [15] for further definition and examples).

6-nodes triangular $n_x \times n_y$ mesh (topology similar to Fig. 6):

$$\begin{aligned}&\mathbf{SG} = \mathbf{G}(\mathbf{X})_5\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}^e_{2n_x+1}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}^e_{2n_y+1},\\ &m1 = (1 : 2n_x), \quad m2 = (1 : 2n_y),\end{aligned} \tag{10}$$

$$\begin{aligned}&\mathbf{NAG} = \mathbf{G}(\mathbf{X})_6\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}_{2n_x}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}_{n_y},\\ &m1 = (1 : 2 : 2n_x - 1,\, 2 : 2 : 2n_x - 2_d), \quad m2 = (1 : n_y - 1).\end{aligned} \tag{11}$$

3-node triangular $n_x \times n_y$ mesh (topology similar to Fig. 4c or 4d):
In these forms there is central symmetry and so for its reserve, n_x, n_y should be even numbers and $n_x = n_y$.

For forms similar to Fig. 4c:

$$\begin{aligned}&\mathbf{SG} = \mathbf{G}(\mathbf{X})_1\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}_{n_x+1}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}_{n_y+1},\\ &z = \frac{n_x}{2}, \quad m1 = (1 : z^-,\, z + 1 : n_x^+), \quad m2 = (1 : z^+,\, z + 1 : n_x^-),\end{aligned} \tag{12}$$

$$\begin{aligned}&\mathbf{NAG} = \mathbf{G}(\mathbf{X})_7\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}^{\mathbf{k}}_{2n_x}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}^{\mathbf{k}}_{n_x},\\ &m1 = (1^-, 2_d, \ldots, n_x - 1^-, n_{xd}, n_x + 1^+,\, n_x + 2_d, \ldots, 2n_x - 1^+),\\ &m2 = (1 : z - 1^+, z_d, z + 1 : n_x - 1^-), \quad \mathbf{k} = [1 : 2 : n_x - 1,\, n_x + 2 : 2 : 2n_x].\end{aligned} \tag{13}$$

For the forms similar to Fig. 4d:

$$\begin{aligned}&\mathbf{SG} = \mathbf{G}(\mathbf{X})_1\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}_{n_x+1}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}_{n_y+1},\\ &z = \frac{n_x}{2}, \quad m1 = (1 : z^+,\, z + 1 : n_x^-), \quad m2 = (1 : z^+,\, z + 1 : n_x^-),\end{aligned} \tag{14}$$

$$\begin{aligned}&\mathbf{NAG} = \mathbf{G}(\mathbf{X})_7\,\mathbf{H}, \qquad \mathbf{G} = \mathbf{D}_{m1}\mathbf{P}^{\mathbf{k}}_{2n_x}, \qquad \mathbf{H} = \mathbf{D}_{m2}\mathbf{P}^{\mathbf{k}}_{n_x},\\ &m1 = (1^+, 2_d, \ldots, n_x - 1^+, n_{xd}, n_x + 1^-,\, n_x + 2_d, \ldots, 2n_x - 1^-),\\ &m2 = (1 : z - 1^+, z_d, z + 1 : n_x - 1^-), \quad \mathbf{k} = [2 : 2 : n_x,\, n_x + 1 : 2 : 2n_x - 1].\end{aligned} \tag{15}$$

6. Mesh basis of natural associate graphs

Mesh basis of the NAG of a FE model is an important part of FE analysis using combinatorial force method [7, 8]. A *mesh basis* consists of all the cycles bounding the internal regions of a planar graph except that of the external region. However, there are different cycle selection algorithms for cycle basis of a graph [1]. Though these methods are efficient and more general, however, for large graphs such methods require some computational time. In this section, using the regularity property of FE models, concepts of graph products are employed for the formation of the mesh basis of the NAG of a FE model in a simple and efficient manner.

6.1. Rectangular elements

The NAG of a $n_x \times n_y$ rectangular FE is simply a $n_x \times n_y$ grid. Thus, the NAG can be constructed by the Cartesian product of two simple paths with n_x, n_y. Each cell of such a grid is a mesh which is related to an edge of graph $\mathbf{G}$ and an edge of $\mathbf{H}$. In Fig. 10 a typical cycle and its related edges are shown in bold lines.

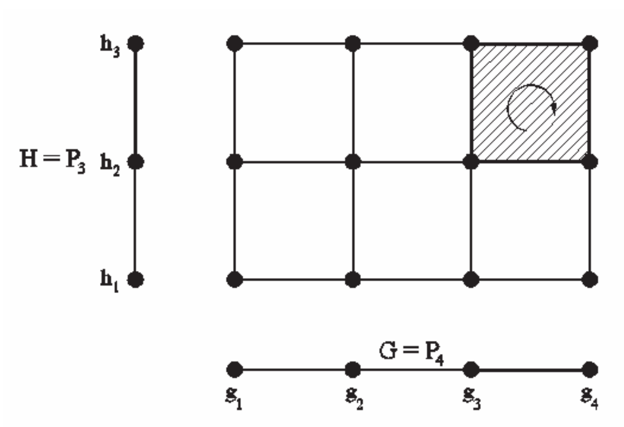


Fig. 10. A cycle and its related edges of two generatorss

Figure 10 shows that all of the edges and nodes of the hatched area which is also a mesh, can be generated by Cartesian product of two edges $\mathbf{P}_2 = \mathbf{G}_{(3,4)} \subset \mathbf{G}$ and $\mathbf{P}_2 = \mathbf{H}_{(2,3)} \subset \mathbf{H}$. Such a relation can be written for all meshes of a grid. For a $n_x \times n_y$ grid, this representation can be written as:

$$\mathbf{C} = \mathbf{G}_{(i,i+1)}(\mathbf{X})_{\mathbf{C}}\, H_{(j,j+1)}, \qquad i = 1, 2, \ldots, n_x - 1, \qquad j = 1, 2, \ldots, n_y - 1, \tag{16}$$

in which $\mathbf{G}$, $\mathbf{H}$ are simple paths with n_x, n_y nodes, respectively, and $\mathbf{C}$ is a cycle which is bounded by i, $i+1$, j, $j+1$ nodes. Obviously, using this relation, $(n_x - 1)(n_y - 1)$ meshes can be generated for the NAG of the regular rectangular FE models.

6.2. Triangular elements

NAG of regular triangular $n_x \times n_y$ FEs can also be generated in a similar manner. However, for such models, three edges of graph $\mathbf{G}$ and one edge of graph $\mathbf{H}$ correspond to a cycle. The mathematical representations of cycles of these models are also related to the products of Type 6 and Type 7 which are introduced in Section 4. For NAGs which are generated using product of Type 6, the mathematical representation of cycles is as follows:

$$\begin{aligned} &\mathbf{C} = \mathbf{G}_4(\mathbf{X})_6\,\mathbf{H}_2, \\ &\mathbf{G}_4 = \mathbf{D}_{(i,i+1_d,i+2)}\mathbf{P}_4 \subset \mathbf{G}, \qquad i = 1,3,5,\ldots,2n_x-3, \\ &\mathbf{H}_2 = \mathbf{D}_j\mathbf{P}_2 \subset \mathbf{H}, \qquad j = 1,2,\ldots,n_y-1. \end{aligned} \tag{17}$$

For two forms which have central symmetry (Figs. 4c and 4d) similar to SG and NAG representations, separate relations should be written.

For the forms similar to Fig. 4c:

$$\begin{aligned} &\mathbf{C} = \mathbf{G}_4(\mathbf{X})_7\,\mathbf{H}_2, \\ &\begin{matrix}\mathbf{G}_4 = \mathbf{D}_{(i,i+1_d,i+2)}\mathbf{P}_4^{\mathbf{k}} \subset \mathbf{G}, \\ \mathbf{H}_2 = \mathbf{D}_j\mathbf{P}_2 \subset \mathbf{H},\end{matrix} \quad \begin{cases} i = 1,3,5,\ldots,n_x-3, & [\mathbf{k}] = [i,i+2], \\ i = n_x-1, & [\mathbf{k}] = [i,i+3], \\ i = n_x+1,n_x+3,\ldots,2n_x-3, & [\mathbf{k}] = [i+1,i+3], \end{cases} \\ &j = 1,2,\ldots,n_y-1. \end{aligned} \tag{18}$$

For the forms similar to Fig. 4d:

$$\begin{aligned} &\mathbf{C} = \mathbf{G}_4(\mathbf{X})_7\,\mathbf{H}_2, \\ &\begin{matrix}\mathbf{G}_4 = \mathbf{D}_{(i,i+1_d,i+2)}\mathbf{P}_4^{\mathbf{k}} \subset \mathbf{G}, \\ \mathbf{H}_2 = \mathbf{D}_j\mathbf{P}_2 \subset \mathbf{H},\end{matrix} \quad \begin{cases} i = 1,3,5,\ldots,n_x-3, & [\mathbf{k}] = [i+1,i+3], \\ i = n_x-1, & [\mathbf{k}] = [i+1,i+2], \\ i = n_x+1,n_x+3,\ldots,2n_x-3, & [\mathbf{k}] = [i,i+2], \end{cases} \\ &j = 1,2,\ldots,n_y-1. \end{aligned} \tag{19}$$

In all of the above relationships, $\mathbf{C}$ is a cycle which is bounded by edges i, $i+1$, $i+2$ of graph $\mathbf{G}$ and edge j of graph $\mathbf{H}$.

It is useful to note that, in each graph generated using Eqs. (17–19) there are some nodes of degree one. Such nodes as well as their incidence edges should be removed from the graph. This leads to a cycle graph with the first Betti number being equal to unity with all its nodes having degree equal to 2.

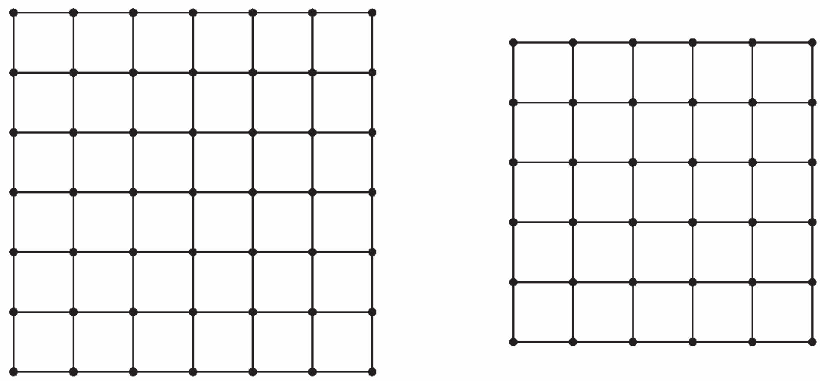
7. Illustrative examples

In this section seven examples are presented for different regular FE models (Fig. 11). Also mathematical representation of skeleton graph (SG) and natural associate graph (NAG) are presented. The representations for mesh bases can easily be obtained by Eqs. (16–19). In each example, the SG of a FE model having n_x ,n_y elements in $\mathbf{x}$ and $\mathbf{y}$ direction respectively, is illustrated in the left and the corresponding NAG and its representation are depicted in the right.

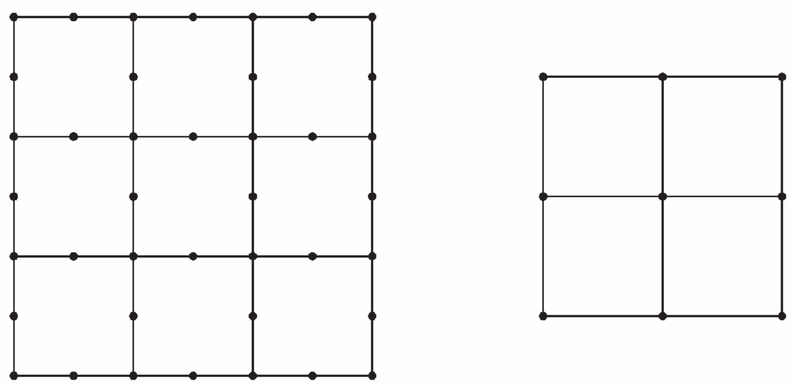
8. Concluding remarks

There are various applications of skeleton graphs, natural associate graphs and their mesh bases for finite element models. These graphs can efficiently be used for nodal and element ordering and combinatorial finite element analysis by the force method. However, generation of these graphs and formation of the mesh bases require considerable computational time.

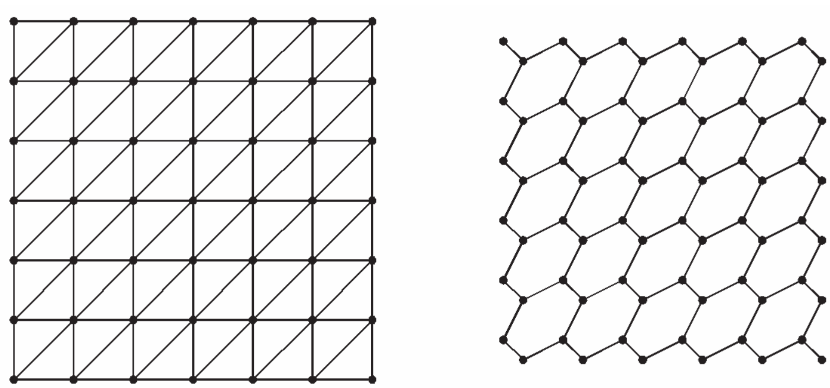
Many FE models are regular and their NAG and SG can be presented as the product of two or three subgraphs. In this paper, using the regularity property of the FE models and employing



a) $n_x = n_y = 6$ $\quad \mathbf{SG} : \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H},\ \mathbf{G} = \mathbf{H} = \mathbf{P}_7,$ $\quad \mathbf{NAG} : \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H},\ \mathbf{G} = \mathbf{H} = \mathbf{P}_6,$

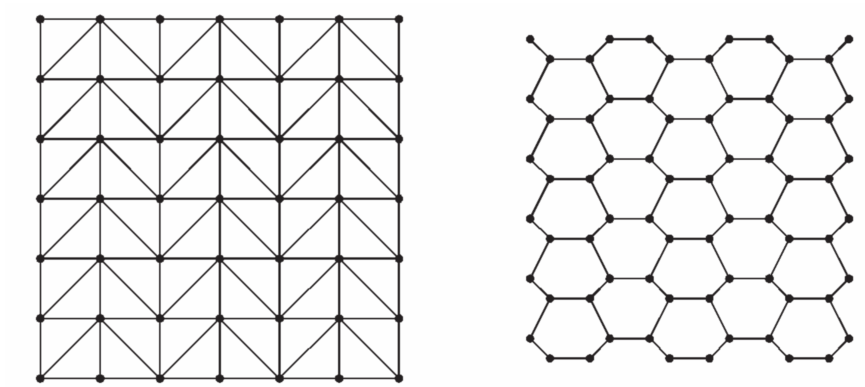


b) $n_x = n_y = 3$ $\quad \mathbf{SG} : \mathbf{G}(\mathbf{X})_2\,\mathbf{H},\ \mathbf{G} = \mathbf{H} = \mathbf{P}_7^{\mathbf{e}},$ $\quad \mathbf{NAG} : \mathbf{G}(\mathbf{X})_{\mathbf{C}}\,\mathbf{H},\ \mathbf{G} = \mathbf{H} = \mathbf{P}_3,$



c) $n_x = n_y = 6$ $\quad \mathbf{SG} : \mathbf{G}(\mathbf{X})_1\,\mathbf{H},\ \mathbf{G} = \mathbf{D}_{(1:6^+)}\mathbf{P}_7,\ \mathbf{H} = \mathbf{D}_{(1:6^+)}\mathbf{P}_7,$ $\quad \mathbf{NAG} : \mathbf{G}(\mathbf{X})_6\,\mathbf{H},\ \mathbf{G} = \mathbf{D}_{(1^+,2_d,3^+,\ldots,11^+)}\mathbf{P}_{12},\ \mathbf{H} = \mathbf{D}_{(1:5^+)}\mathbf{P}_6,$

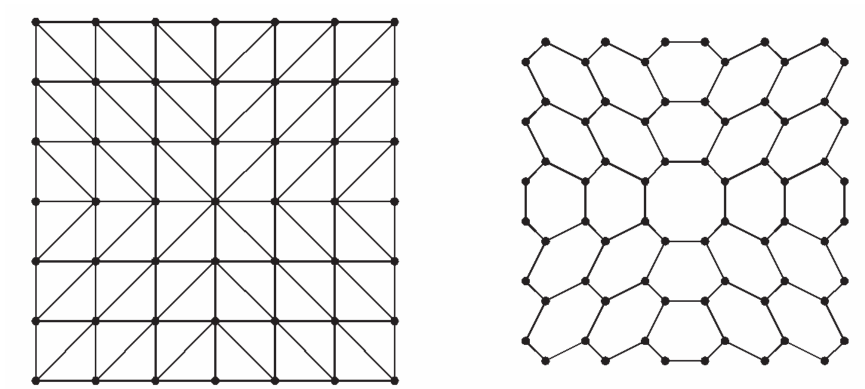
Fig. 11. Seven FE models and the corresponding NAGs *(continued on the next page)*



d) $n_x = n_y = 6$

$\mathbf{SG}: \mathbf{G}(\mathbf{X})_1\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1^+,2^-,\ldots,6^-)}\mathbf{P}_7$,
$\mathbf{H} = \mathbf{D}_{(1:6^+)}\mathbf{P}_7$,

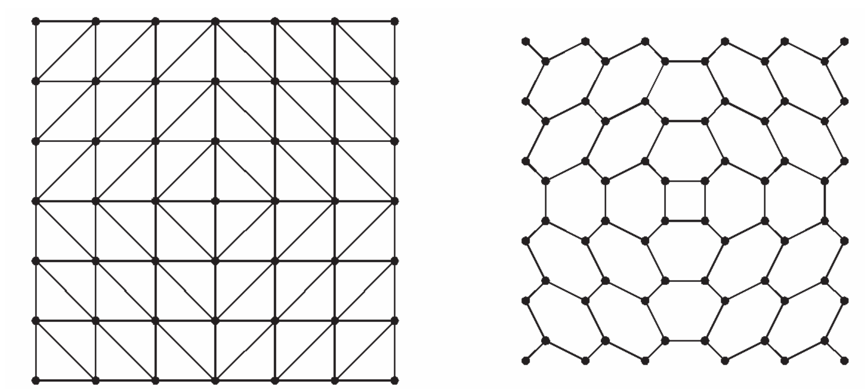
$\mathbf{NAG}: \mathbf{G}(\mathbf{X})_6\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1^+,2_d,3^-,\ldots,11^-)}\mathbf{P}_{12}$,
$\mathbf{H} = \mathbf{D}_{(1:5^+)}\mathbf{P}_6$,



e) $n_x = n_y = 6$

$\mathbf{SG}: \mathbf{G}(\mathbf{X})_1\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1:3^+,4:6^-)}\mathbf{P}_7$,
$\mathbf{H} = \mathbf{D}_{(1:3^+,4:6^-)}\mathbf{P}_7$,

$\mathbf{NAG}: \mathbf{G}(\mathbf{X})_7\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1^+,2_d,\ldots,5^+,6_d,7^-,8_d,\ldots,11^-)}\mathbf{P}_{12}^{[2:2:6,7:2:11]}$,
$\mathbf{H} = \mathbf{D}_{(1:2^+,3_d,4:5^-)}\mathbf{P}_6$,

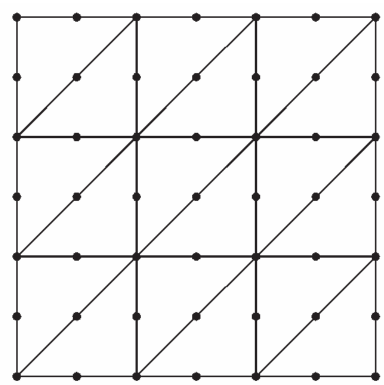


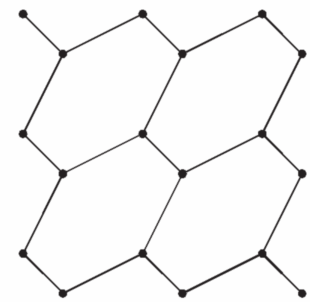
f) $n_x = n_y = 6$

$\mathbf{SG}: \mathbf{G}(\mathbf{X})_1\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1:3^-,4:6^+)}\mathbf{P}_7$,
$\mathbf{H} = \mathbf{D}_{(1:3^+,4:6^-)}\mathbf{P}_7$,

$\mathbf{NAG}: \mathbf{G}(\mathbf{X})_7\,\mathbf{H}$,
$\mathbf{G} = \mathbf{D}_{(1^+,2_d,\ldots,5^-,6_d,7^+,8_d,\ldots,11^+)}\mathbf{P}_{12}^{[1:2:5,8:2:12]}$,
$\mathbf{H} = \mathbf{D}_{(1:2^+,3_d,4:5^-)}\mathbf{P}_6$,

Fig. 11. *(continued)* Seven FE models and the corresponding NAGs





g) $n_x = n_y = 3$

$$\mathbf{SG}: \mathbf{G}(\mathbf{X})_5\,\mathbf{H},\quad \mathbf{G} = \mathbf{D}_{(1:6^+)}\mathbf{P}_7^{\mathbf{e}},\quad \mathbf{H} = \mathbf{D}_{(1:6^+)}\mathbf{P}_7^{\mathbf{e}},$$

$$\mathbf{NAG}: \mathbf{G}(\mathbf{X})_6\,\mathbf{H},\quad \mathbf{G} = \mathbf{D}_{(1^+,2_d,3^+,\ldots,5^+)}\mathbf{P}_6,\quad \mathbf{H} = \mathbf{D}_{(1:2^+)}\mathbf{P}_3,$$

Fig. 11. *(continued)* Seven FE models and the corresponding NAGs

the graph product concepts, efficient and systematic approaches are presented for construction of these graphs and mesh bases of their associate graphs. Though the models studied in this paper are selected from simple ones, however, more complicated models can be generated by combining these regular models.

It should be mentioned that in this paper the NAG and the SG of a finite element model is obtained in one step in a systematic manner, while the traditional methods use different steps, one step of which requires the same computational time and effort as that of the present approach.

Acknowledgements

The first author is grateful to Iran National Science Foundation for support.

References

[1] A. Kaveh. *Optimal Structural Analysis*. 2nd edition. Wiley, Chichester, 2006.
[2] E. Cuthill, J. McKee. Reducing the bandwidth of sparse symmetric matrices. In: *Proc. 24th Nat. Conf.*, 157–172, ACM, New York, 1969.
[3] A. Kaveh. *Application of topology and matroid theory to the analysis of structures*. PhD thesis. London University, Imperial College, March 1974.
[4] N.E. Gibbs, W.G. Poole, P.K. Stockmeyer. An algorithm for reduction the bandwidth and profile of a sparse matrix. *SIAM Journal of Numerical Analysis*, **12**: 236–250, 1976.
[5] A. Kaveh. A note on a two-step approach to finite element ordering. *International Journal of Numerical Methods in Engineering*, **19**: 1753–1754, 1984.
[6] S.J. Feneves, K.H. Law. A two-step approach to finite element ordering. *International Journal of Numerical Methods in Engineering*, **19**: 891–911, 1983.
[7] A. Kaveh, K. Koohestani, N. Taghizadieh. Efficient finite element analysis by graph-theoretical force method, *International Journal of Numerical Methods in Engineering*, **43**: 543–554, 2007.
[8] A. Kaveh, K. Koohestani. An efficient graph-theoretical method for plate bending finite element analysis via force method. *Engineering Computations*, **24**: 679–698, 2007.
[9] C. Berge. *Graphs and Hypergraphs*. North-Holland Publishing Company, Amsterdam, 1973.
[10] G. Sabidussi. Graph multiplication. *Mathematical Zeitschrift*, **72**: 446–457, 1960.
[11] F. Harary, G.W. Wilcox. Boolean operations on graphs. *Math. Scand.*, **20**: 41–51,1967.
[12] W. Imrich and S. Klavzar. *Product Graphs: Structure and Recognition*. Wiley, New York, 2000.
[13] A. Kaveh, H. Rahami. A new spectral method for nodal ordering of regular space structures. *Finite Element in Analysis and Design*, **40**: 1931–1945, 2004.

[14] A. Kaveh, H. Rahami. Block diagonalization of adjacency and Laplacian matrices for graph product; applications in structural mechanics. *International Journal of Numerical Methods in Engineering*, **68**: 33–63, 2006.

[15] A. Kaveh, K. Koohestani. Graph products for configuration processing of space structures. *Computers and Structures*, **86**: 1219–1236, 2008.

[16] P.M. Weichsel. The Kronecker product of graphs. *Proceedings of the American Mathematical Society*, **13**: 47–52, 1962.