

Weighted triangular and circular graph products for configuration processing

Ali Kaveh / Sepehr Beheshti

Received 2011-06-02, revised 2012-02-13, accepted 2012-02-24

Abstract

In this paper new weighted triangular and circular graph products are presented for configuring the new forms of structural models. The graph products are extensively used in graph theory and combinatorial optimization, however, the weighted triangular products defined in this paper are more suitable for the formation of practical space structural models which are impossible to generate by the previous products. Here, the weighted triangular and circular products are employed for the configuration processing of space structures that are of triangular shapes or a combination of triangular and rectangular shapes and also of the solid circular shapes as domes and some space structural models. The covered graph products are represented for selecting or eliminating some parts or panels from the product graph by using the second weights for the nodes of the generators. Cut out products are other new types of graph products which are defined to eliminate all of the connected elements to a specified node, to configure the model or grid with some vacant panels inside of the model. This application can easily be extended to the formation of finite element models.

Keywords

space structures; weighted triangular graph products; circular graph product; configuration processing · generators · cut-outs

Acknowledgement

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

Ali Kaveh

Sepehr Beheshti

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

1 Introduction

For a large system, configuration processing is one of the most tedious and time-consuming parts of the analysis. Different methods have been proposed for configuration processing and data generation, among which one can refer to the formex algebra of Nooshin [1,2], Nooshin et al. [3] and Nooshin & Disney [4]. Behravesht et al. [5] employed set theory and showed that some concepts of set algebra can be used to obtain a general method for describing the interconnection patterns of structural systems. Graph theoretical methods for the formation of structural and finite element models are developed by Kaveh [6,7]. In all these methods a submodel is expressed in an algebraic form and then functions are used for the formation of the entire model. The main functions employed consist of translation, rotation, reflection and projection, or combination of these functions. Four undirected graph products and four directed graph products are employed for the formation of structural models by Kaveh and Koohestani [8], and weighted product graphs are employed by Kaveh and Nouri [9].

There are many other references in the field of data generation; however, most of them are prepared for specific classes of problem. For example, many algorithms have been developed and successfully implemented on mesh or grid generation, a complete review of which may be found in a paper by Thacker [10] and in books by Thomson et al. [11], Liseikin [12]. On the other hand many structural models can be viewed as the graph products of two or three subgraphs, known as their generators. Many properties of structural models can be obtained by considering the properties of their generators. This simplifies many complicated calculations, particularly in relation with eigensolution of regular structures, as shown by Kaveh and Rahami [13–15].

In this paper, weights are assigned to nodes and members of the generators to generate configurations which can not be formed using the existing graph products [16]. The new products are especially suitable for the formation of the models of space structures. The present method can be used in the formation of other systems, and can easily be extended to the formation of finite element models.

2 Definitions from graph theory

A graph $S(V, E)$ consists of a set of elements, $V(S)$, called nodes (vertices) and a set of elements, $E(S)$, called members (edges), together with a relation of incidence which associates two distinct nodes with each member, known as its ends. When weights are assigned to the members and nodes of a graph, then it becomes a weighted graph. Two nodes of a graph are called adjacent if these nodes are the end nodes of a member. A member is called incident with a node if that node is an end node of the member. The degree of a node is the number of members incident with the node. A subgraph S_i of a graph S is a graph for which $V(S_i) \subseteq V(S)$ and $E(S_i) \subseteq E(S)$, and each member of S_i has the same ends as in S . A path graph P is a simple connected graph with $V(P) = E(P) + 1$ that can be drawn in a way that all of its nodes and members lie on a single straight line. A path graph with n nodes is denoted as P_n , and a weighted path is shown by PW_n . A cycle graph C is a simple connected graph with $V(C) = E(C)$ that can be drawn such that all of its nodes and members lie on a circle. A cycle graph with n nodes is shown by C_n , and a weighted cycle is denoted by CW_n . For further definitions the reader may refer to Kaveh [7, 16].

3 Extension of classic graph products

In this section, simple graph products are extended to new triangular and circular graph products for efficient use in structural mechanics.

3.1 Cartesian product of graphs

Some of the structures with regular configurations can be expressed as the Cartesian product of two or more subgraphs. After the formation of the nodes of such a graph, according to the nodes of the generators, a member should be added between two typical nodes (U_i, V_j) and (U_k, V_l) , as shown in Fig. 1, if any of the following conditions are satisfied:

$$\text{If } \{[U_i = U_k \text{ and } V_j V_l \in E(K)] \text{ or } [V_j = V_l \text{ and } U_i U_k \in E(H)]\}$$

It should be noted that $E(K)$ is the elements of subgraph K and $E(H)$ is the elements of subgraph H , and U_i, U_k are the nodes of the subgraph H and V_j, V_l are the nodes of subgraph K . Two examples of this graph product are shown in Fig. 1.

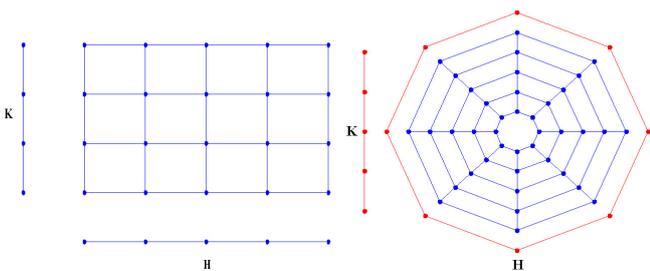


Fig. 1. Cartesian product graphs

3.2 Formulation of weighted Cartesian product

Simple graph products can only generate special configurations and in order to generate configurations encountered in practice some elements should be removed with an additional rule. In order to avoid such a complication, weighted graphs can be employed. Here, zero weights are assigned to the nodes and members which are supposed to be removed.

After the formation of the nodes according to the nodes of the generators, a member is added between two typical nodes (U_k, V_l) and (U_i, V_j) , with the weights W_{ik}, W_{ki}, W_{jl} , and W_{lj} for elements and W_i , and W_j for the nodes, if the following conditions are fulfilled:

$$\begin{aligned} &\text{If } \{[U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)] \\ &\text{or } [U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } \\ &\quad W_{lj} \neq 0))] \\ &\text{or } [V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)] \\ &\text{or } [V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or } \\ &\quad W_{ki} \neq 0))]\} \end{aligned}$$

Fig. 2 illustrates the constructed configurations by the above rules. Though the formulae for both configurations in Fig. 2 are identical, however, due to the use of different weighted generators, dissimilar configurations are obtained.

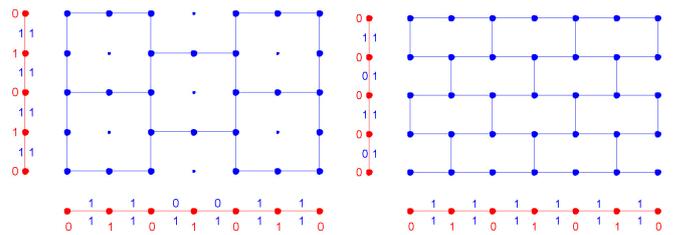


Fig. 2. Weighted Cartesian graph product

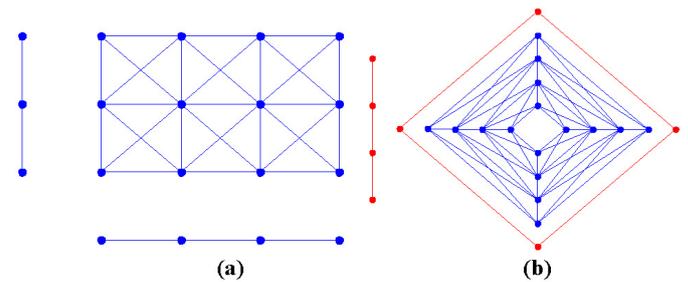


Fig. 3. Strong Cartesian graph products

3.3 Strong Cartesian product of graphs

Some other structures with regular configurations can be expressed as the strong Cartesian product of two or more graphs. After the formation of the nodes of such a graph a member should be added between two typical nodes (U_k, V_l) and (U_i, V_j) if any of the following conditions are satisfied:

$$\begin{aligned} &\text{If } \{[U_i = U_k \text{ and } V_j V_l \in E(K)] \\ &\text{or } [U_i U_k \in E(H) \text{ and } V_j = V_l] \\ &\text{or } [U_i U_k \in E(K) \text{ and } V_j V_l \in E(H)]\} \end{aligned}$$

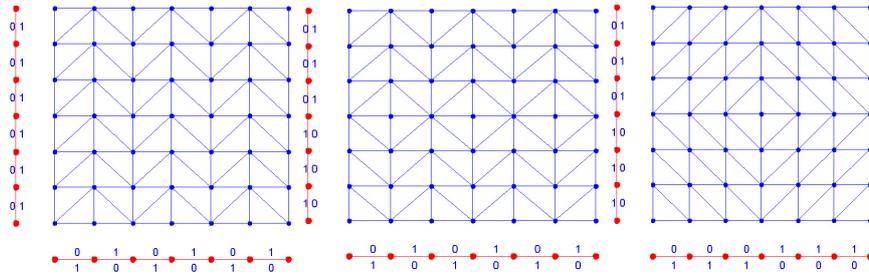


Fig. 4. Weighted strong Cartesian graph product

Two examples of strong Cartesian product are shown in Fig. 3. The configuration in Fig. 3(a) is the product of two paths, and the one in Fig. 3(b) is the product of a cycle and a path.

3.4 Formulation of weighted strong Cartesian product

Simple graph products can only generate panels with two crossing members and in order to generate configurations with panels having single bracing elements, weighted strong Cartesian products can not be used. In order to produce configurations with triangular panels, weighted generators should be employed. Here, zero weights are assigned to the nodes and members which should be removed. After the formation of the nodes according to the nodes of the generator, a member is added between two typical nodes (U_k, V_l) and (U_i, V_j) , with the weights W_{ik}, W_{ki}, W_{jl} , and W_{lj} for elements, if the following conditions are fulfilled:

$$\begin{aligned} & \text{if } \{ [U_i = U_k \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)] \\ & \text{or } [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \\ & \text{or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \} \end{aligned}$$

Three examples of grids generated by these relationships are shown in Fig. 4. For each configuration the corresponding weighted generator are also depicted.

3.5 Direct product of graphs

Some other structures with regular configurations can be expressed as the direct product of two or more graphs. After the formation of the nodes of such a graph, a member should be added between two typical nodes (U_k, V_l) and (U_i, V_j) if the following conditions are satisfied:

$$\text{if } \{ U_i U_k \in E(K) \text{ and } V_j V_l \in E(H) \}$$

The direct product of two paths P_3 and P_4 , and that of P_4 and C_4 are shown in Fig. 5.

3.6 Formulation of weighted direct new product

Here, zero weights are assigned to the nodes and also to the members which should be removed. After the formation of the nodes according to the nodes of the generator, a member is added between two typical nodes (U_i, V_j) and (U_k, V_l) , with the weights W_{ik}, W_{ki}, W_{jl} , and W_{lj} for elements, if the following conditions are fulfilled:

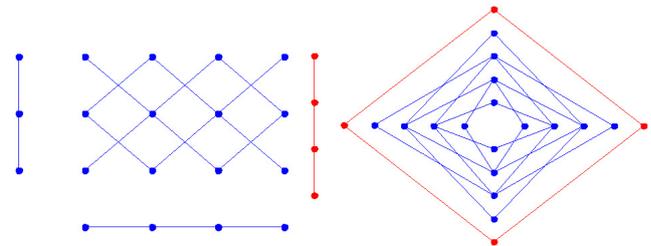


Fig. 5. Direct product graphs

$$\text{if } \{ W_{ki} \neq 0 \text{ or } W_{lj} \neq 0 \}$$

Two examples of weighted direct products of $PW7*PW7$ are illustrated in Fig. 6. The considered weights for the paths are different, thus resulting in different configurations.

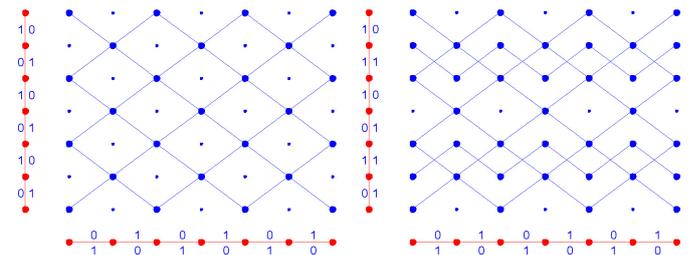


Fig. 6. Weighted direct graph product

3.7 Weighted Cartesian direct graph products

By combining the weighted Cartesian and direct graph products, weighted Cartesian direct graph product is defined. In the weighted Cartesian product both of elements and nodes are weighted and the conditions are implemented by using these weights. But in the weighted direct product only the nodes of the generators are weighted, so the conditions are defined based on the weights of the elements. In the weighted Cartesian direct product both of the elements and nodes are weighted. For configuring vertical and horizontal elements both weights are implemented and the conditions of weighted cartesian product must be fulfilled. But for configuring diagonal elements only nodal weights and weighted direct product must be fulfilled. By simultaneous use of the product's conditions, a new graph product can be defined for configuring some special models with the horizontal, vertical and diagonal elements. After the formation of the nodes according to the nodes of the generators, a member is added between two typical nodes (U_k, V_l) and (U_i, V_j) , with

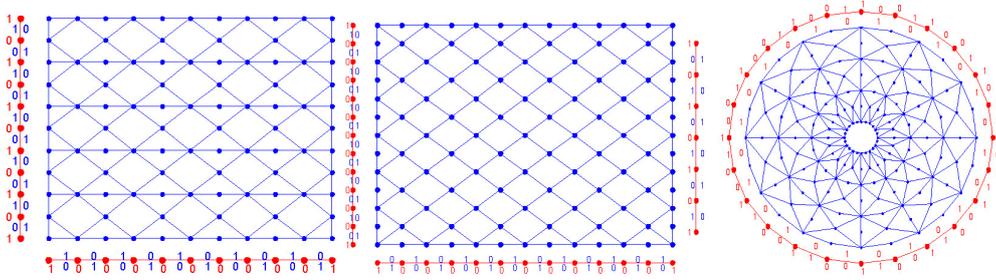


Fig. 7. Weighted Cartesian direct graph product

the weights W_{ik} , W_{ki} , W_{jl} , and W_{lj} for elements and W_i , and W_j for the nodes, if one of the following conditions is fulfilled:

- if $\{[U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)]$
- or $[U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0$
- or $W_{lj} \neq 0))]$
- or $[V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)]$
- or $[V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or}$
- $W_{ki} \neq 0)))]$
- or $[W_{ki} \neq 0 \text{ or } W_{lj} \neq 0]$

Some examples of this product are demonstrated in Fig. 7.

4 Definition of weighted triangular graph products

Since the classic graph products are only use adjacency conditions in the generators, therefore these products are plain and can form only simple shapes and models. For configuring other groups of shapes and models, it is necessary to assign other conditions and definitions to the previously mentioned generators and also to the conditions of the products. Assigning weights to the nodes and elements can be considered as new additions to the conditions and definitions. For creating the new graph products and new shapes, Kaveh and Nouri [9] generated new models in the forms of the classic products with assigning weights to the nodes and elements of the generators.

With a more profound insight it can be conceived that the concept of assigning weights to the nodes and defining new conditions according to these weights, can have a wide range of applications and can make it possible to generate many other different shapes which are impossible to be generated by using classic products. In other words, the new definitions obtained by assigning nodal weights, new variables become available to consider as the adjacency conditions of the nodes in a product graph.

Using the existing graph products, only models in the form of rectangular or circular shapes can be generated with no center node. For generating other types of the shapes, either one should use a trans-figure of an existing product graph or should define new graph products. Now we define a new graph product for generating the triangular and a wide range of polygon shapes. By using this product it becomes possible to generate triangular, trapezoid, parallelogram and many other combinations of triangular and rectangular shapes.

4.1 Assigning weights to the nodes of the generators and product graphs

The weights of product graph nodes are equal to the summation of the weights of the corresponding subgraphs nodes, Fig. 8. It is clear that assigning the weights to the nodes of a product graph according to the weights of the generators can have different types such as sum, difference, or product.)

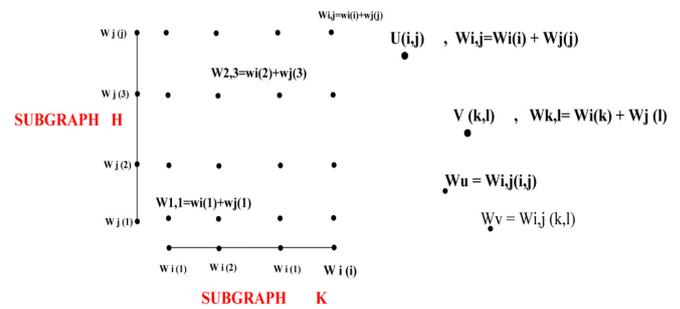


Fig. 8. Assigning weights to the nodes of a product graph

After assigning the weights to the nodes of a product graph, some conditions should be imposed to assess the adjacency of the nodes. First W_{MAX} is defined as the maximum weight of the subgraphs. For example, for two subgraphs with the weights $[1 \ 2 \ 3 \ 4 \ 5]$ and $[2 \ 3 \ 2 \ 3 \ 4]$, W_{MAX} is equal to 5. Considering the significance of W_{MAX} and using $W_{i,j}(i, j)$ and $W_{i,j}(k, l)$, the new conditions are implemented.

4.2 Weighted triangular strong Cartesian graph product

After the formation of weights of the nodes in a product graph, a member should be added between two typical nodes (U_k, V_l) and (U_i, V_j) if the following conditions are satisfied:

Weights of the nodes: $W_{k,l} = W_i(k) + W_j(l)$, $W_{i,j} = W_i(i) + W_j(j)$

Weights of the elements: W_{ik} , W_{ki} , W_{jl} , W_{lj}

- if $\{[U_i U_k \in E(K) \text{ and } V_j V_l \in E(H)] \text{ and } [W_{i,j}, W_{k,l} = W_{max}]]$
- or $[[U_i = U_k \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)] \text{ or } [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)]]$
- or $[[W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \text{ and } [W_{i,j}, W_{k,l} \leq W_{max}]]$

Four examples of grids generated by these relationships are shown in Fig. 9. For each configuration the corresponding weighted generator are also depicted.

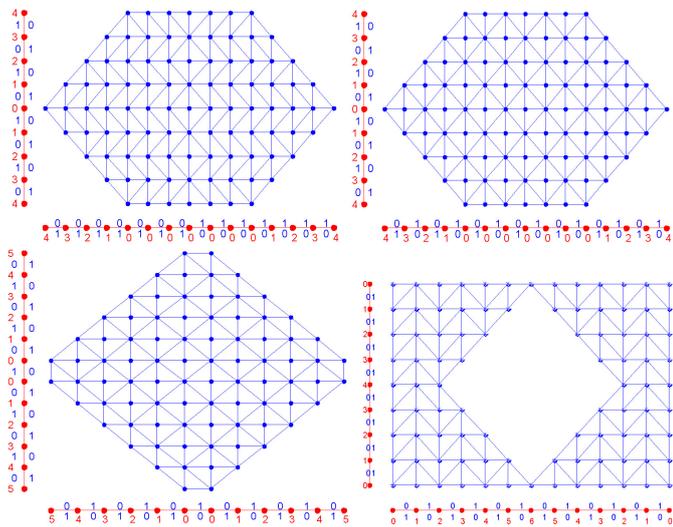


Fig. 9. Weighted triangular strong Cartesian graph product

4.3 Weighted triangular semi strong Cartesian graph product

In this product, both diagonal and horizontal elements can be connected. After the formation of weights of the nodes in the product graph, a member should be added between two typical nodes (U_k, V_l) and (U_i, V_j) if the following conditions are satisfied:

Weights of the nodes: $W_{k,l} = W_i(k) + W_j(l)$, $W_{i,j} = W_i(i) + W_j(j)$

Weights of the elements: $W_{ik}, W_{ki}, W_{jl},$ and W_{lj}

if $\{[[U_i U_k \in E(K) \text{ and } V_j V_l \in E(H)] \text{ and } [W_{i,j}, W_{k,l} = W_{max}]]$

or $[[V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0]] \text{ and } [W_{i,j}, W_{k,l} \leq W_{max}]]$

Two examples of grids generated by these relationships are shown in Fig. 10. For each configuration the corresponding weighted generators are also depicted.

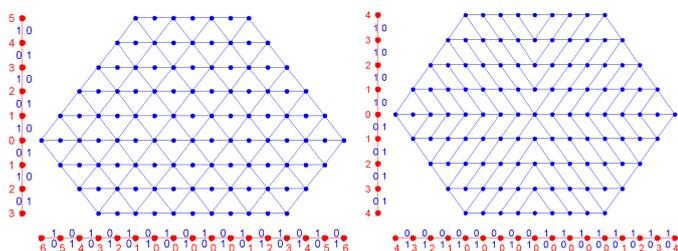


Fig. 10. Weighted triangular semi strong Cartesian graph product

5 Definition of a weighted circular graph product

In the field of space structures, domes are of special importance. From the structural point of view domes are always similar to solid circular shaped models. For configuring solid circles using graph products, one should use the transformation of product graph of two paths or the graph product of a path and a cycle should be employed. Another graph for covering the central part

can be added. In this section, a new graph product is defined to configure solid circular shaped models.

5.1 Weighted Circular Cartesian graph products

In this product weights are assigned to the nodes of the subgraphs, and the conditions and definitions of this product are based on these weights. In this product, the conditions are merely provided for the adjacency of the vertical and horizontal elements. Two nodes (U_i, V_j) and (U_k, V_l) in the product graph, with the weights W_i, W_j, W_k and W_l , and with the weights $W_{ik}, W_{ki}, W_{jl},$ and W_{lj} for elements, are adjacent if the following conditions are fulfilled:

if $\{[[U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)] \text{ or } [U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0))]] \text{ or } [V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)] \text{ or } [V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0))]] \text{ and } [W_j, W_l \leq 1]] \text{ or } [[V_j V_l \in E(K)] \text{ \& } [W_j > 1 \text{ \& } i = 1]] \text{ and } [[W_k = 0 \text{ and } W_{jl} * W_{lj} \neq 0] \text{ or } [W_k \neq 0 \text{ and } W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)]]]$

Some examples of this product are demonstrated in Fig. 11.

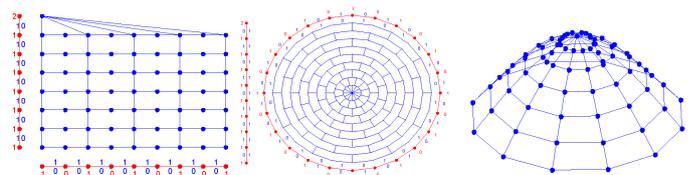


Fig. 11. Weighted circular Cartesian product

5.2 Weighted circular strong Cartesian graph product

In this product the nodes of generators are weighted and conditions are defined based on the weights. Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to the nodes and also the weights $W_{ik}, W_{ki}, W_{jl}, W_{lj}$ assigned to the elements, will be connected if the following conditions are fulfilled:

if $\{[[U_i = U_k \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)] \text{ or } [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0]] \text{ and } [W_j, W_l \leq 1]] \text{ or } [[V_j V_l \in E(K)] \text{ and } [W_j > 1 \text{ \& } i = 1]]]$

Some examples of this product are illustrated in Fig. 12.

5.3 Weighted circular direct graph product

In this product only the diagonal elements will be connected. The nodes of generators are weighted and conditions are defined based on the weights. Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l assigned to them, will be connected if the following conditions are fulfilled:

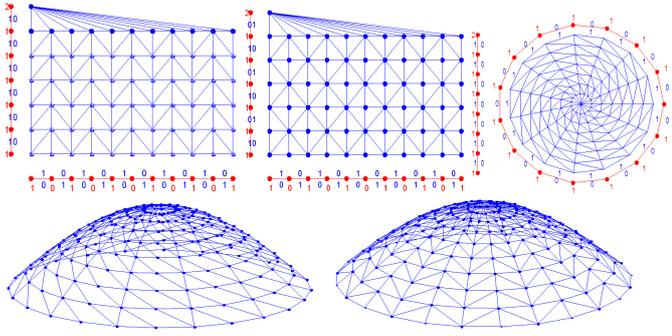


Fig. 12. Weighted circular strong Cartesian product

$$\text{if } \{ [[W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \text{ and } [W_j, W_l \leq 1]] \}$$

$$\text{or } [[V_j V_l \in E(K)] \text{ and } [W_j > 1 \& i = 1]] \}$$

Some examples of this product are demonstrated in Fig. 13.

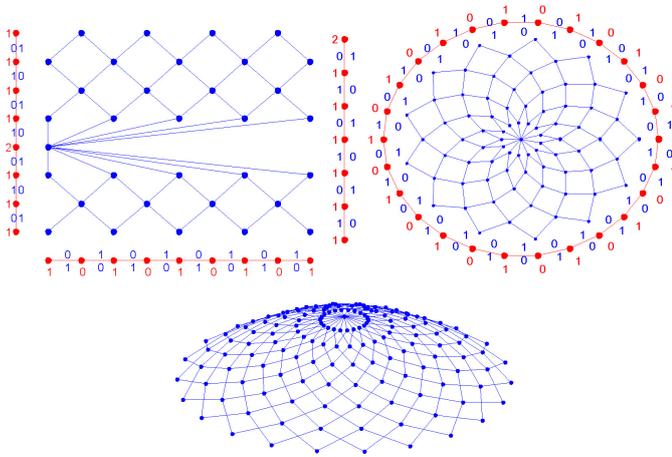


Fig. 13. Weighted circular direct product

5.4 Weighted circular Cartesian direct graph product

In this product only the diagonal elements will be connected. The nodes of generators are weighted and conditions are defined based on the weights. Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

$$\text{if } \{ [[[U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)] \text{ or } [U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0))] \text{ or } [V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)] \text{ or } [V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0))]] \text{ and } [W_j, W_l \leq 1] \}$$

$$\text{or } [[W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \text{ and } [W_j, W_l \leq 1]]$$

$$\text{or } [[V_j V_l \in E(K)] \text{ and } [W_j > 1 \& i = 1]] \}$$

Some examples of this product are demonstrated in Fig. 14.

6 Weighted cut out in graph products

Utilizing all the previous products, the product graphs have continuous and regular shapes and models and all of the nodes in

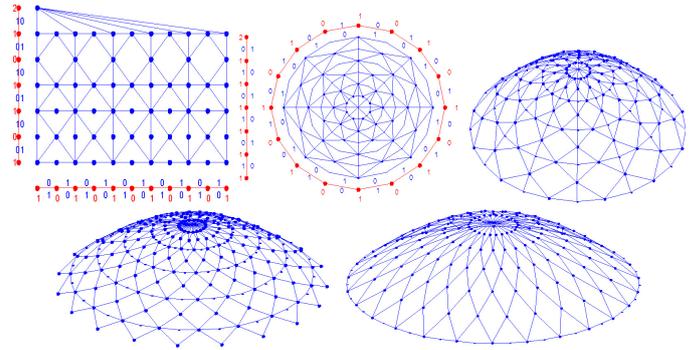


Fig. 14. Weighted circular direct product

the inner panels are connected. However, some models are in forms where there exist some inner panels as hollow and some nodes have no connections with the other nodes. On the other hand, the connected elements to some nodes must be eliminated. In this section a new graph product for eliminating an optional node or nodes with the connected elements (stars) is defined. Using this product, all the nodes in the product graph which have equal or larger weight than an specified value will have no connections with the other nodes and elements.

6.1 Weighted cut outs in Cartesian graph product models

This graph product provides the conditions for connecting vertical and horizontal elements according to the weights of the generators. Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph with weights W_i, W_j, W_k and W_l assigned to them, are connected if the following conditions are fulfilled:

$$\text{if } \{ [[U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)] \text{ or } [U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0))] \text{ or } [V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)] \text{ or } [V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0))]] \text{ and } [[W_i, W_j, W_k, W_l \leq 1] \text{ or } [W_i \neq W_j \& W_k \neq W_l]] \}$$

Examples of this product are demonstrated in Fig. 15.

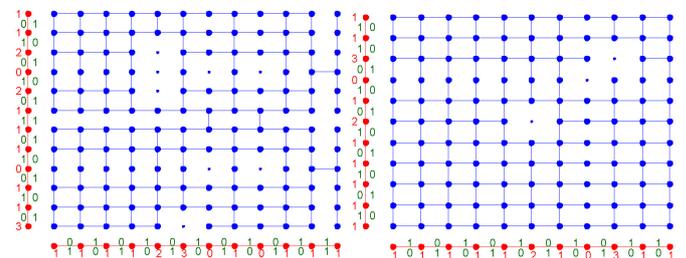


Fig. 15. Weighted circular Cartesian direct product

6.2 Weighted cut out Cartesian direct graph product

This product provides the conditions for connecting vertical, horizontal and diagonal elements according to the weights of the generators. Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights $W_i, W_j, W_k,$ and

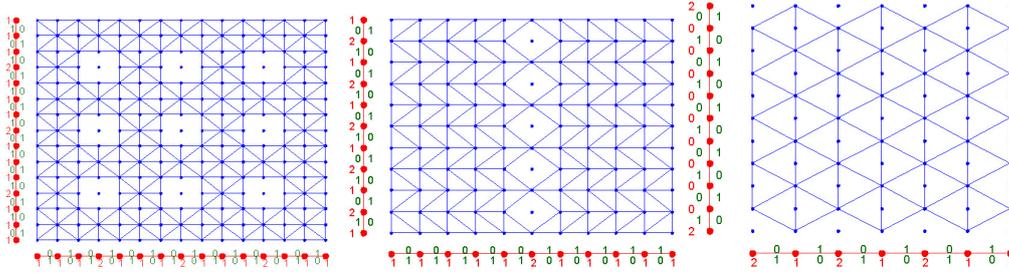


Fig. 16. Weighted cut out Cartesian product

W_l assigned to them and also the weights W_{ik}, W_{ki}, W_{jl} , and W_{lj} assigned to the elements, will be connected if the following conditions are fulfilled:

$$\begin{aligned} & \text{if } \{ [U_i = U_k \text{ and } W_i = 0 \text{ and } (W_{jl} \neq 0 \text{ and } W_{lj} \neq 0)] \\ & \text{or } [U_i = U_k \text{ and } W_i \neq 0 \text{ and } (W_{jl} * W_{lj} = 0 \text{ and } (W_{jl} \neq 0 \text{ or } \\ & \quad W_{lj} \neq 0))] \\ & \text{or } [V_j = V_l \text{ and } W_j = 0 \text{ and } (W_{ik} \neq 0 \text{ and } W_{ki} \neq 0)] \\ & \text{or } [V_j = V_l \text{ and } W_j \neq 0 \text{ and } (W_{ik} * W_{ki} = 0 \text{ and } (W_{ik} \neq 0 \text{ or } \\ & \quad W_{ki} \neq 0))] \\ & \text{or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \text{ and } [[W_i, W_j, W_k, W_l \leq 1] \text{ or } \\ & \quad [W_i \neq W_j \text{ \& } W_k \neq W_l]] \} \end{aligned}$$

Some examples of this product are demonstrated in Fig. 16.

6.3 Weighted cut out strong Cartesian graph product

In this product conditions consist of direct graph product conditions and a new condition for deleting the special nodes with special weights. Thus in this product only the diagonal elements will be connected.

Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

$$\begin{aligned} & \text{if } \{ [U_i = U_k \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)] \\ & \text{or } [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0] \\ & \text{and } [[W_i, W_j, W_k, W_l \leq 1] \text{ or } [W_i \neq W_j \text{ \& } W_k \neq W_l]] \} \end{aligned}$$

Some examples of this product are demonstrated in Fig. 17.

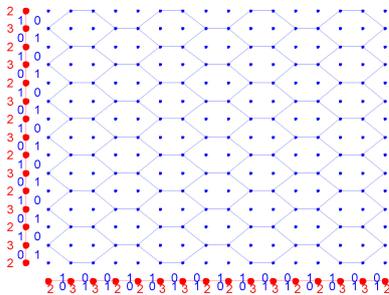


Fig. 17. Weighted cut out Cartesian direct product

6.4 Weighted cut out semi strong Cartesian graph product

In this product conditions consist of direct graph product conditions and a new condition for deleting the special nodes with

special weights. Thus in this product only the diagonal elements will be connected.

Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

$$\begin{aligned} & \text{if } \{ [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } \\ & \quad W_{lj} \neq 0] \\ & \text{and } [[W_i, W_j, W_k, W_l \leq 1] \text{ or } [W_i \neq W_j \text{ \& } W_k \neq W_l]] \} \end{aligned}$$

Some examples of this product are demonstrated in Fig. 18.

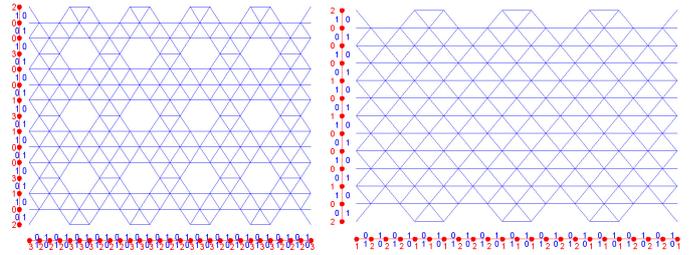


Fig. 18. Weighted cut out strong Cartesian product

7 Covered graph products

By using the new graph products a vast amount of different shapes and models can be generated; however in some product graphs it is necessary to choose or eliminate some parts of the shape of the product graph to generate the desired models. On the other hand, for generating the other shapes, it is often necessary to select some parts of the previous models. In this section by adding a condition to the previous conditions, it becomes possible to select the desired part of the product graph. This condition is implemented by assigning some other weights to the nodes of the subgraphs. This means, the primary elemental and nodal weights generate the full parts of the shapes and the secondary nodal weights select or eliminate the desired part of the original shape or model. Even it is possible to increase the number of selecting process by assigning additional weights to the nodes of the subgraphs. For understanding the importance of this product, three shapes are illustrated where the shape 19(a) can not directly be generated by the prior graph products, but by cutting the shape 19(b) from the shape 19(c), shape 19(a) can be produced. The constitution of the covered product is similar to this cutting process.

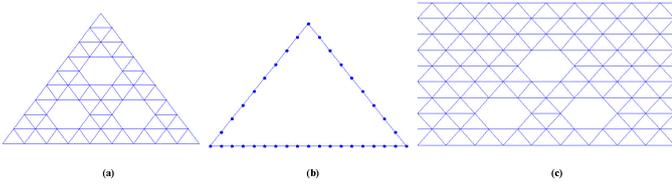


Fig. 19. Cut out direct product

7.1 Covered cut out Cartesian graph product

After the formation of weights of the nodes in the product graph according to the following relations:

Weights of the nodes for product graph: $W_{2i,2j} = W_{2i}(i) + W_{2j}(j)$, $W_{2k,2l} = W_{2i}(k) + W_{2j}(l)$

Weights of the nodes for subgraph: $W_{1i}, W_{1j}, W_{1k}, W_{1l}$ and $W_{2i}, W_{2j}, W_{2k}, W_{2l}$

Weights of the elements: $W_{ik}, W_{ki}, W_{jl}, W_{lj}$

A member should be added between two typical nodes (U_k, V_l) and (U_i, V_j) if the following conditions are satisfied:

If $\{[[U_i = U_k \text{ and } V_j V_l \in E(K)] \text{ or } [U_i U_k \in E(H) \text{ and } V_j = V_l]]$

and $[[W_{1i}, W_{1j}, W_{1k}, W_{1l} \leq 1] \text{ or } [W_{1i} \neq W_{1j} \text{ \& } W_{1k} \neq W_{1l}]]$
and $[W_{2i,2j}, W_{2k,2l} \leq W_{2max}]$

Some examples of this product are demonstrated in Fig. 20.

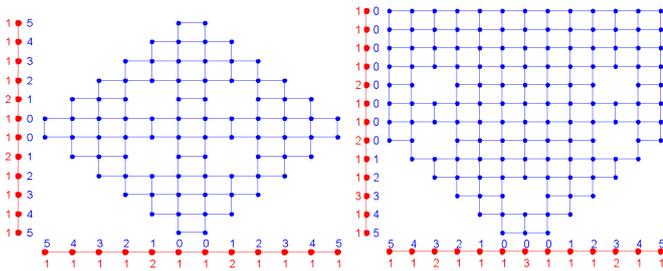


Fig. 20. Covered cut out Cartesian graph product

7.2 Covered cut out strong Cartesian graph product

Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

if $\{[[U_i = U_k \text{ and } V_j V_l \in E(K)] \text{ or } [U_i U_k \in E(H) \text{ and } V_j = V_l] \text{ or } [U_i U_k \in E(K) \text{ and } V_j V_l \in E(H)]]$

and $[[W_{1i}, W_{1j}, W_{1k}, W_{1l} \leq 1] \text{ or } [W_{1i} \neq W_{1j} \text{ \& } W_{1k} \neq W_{1l}]]$
and $[W_{2i,2j}, W_{2k,2l} \leq W_{2max}]$

Some examples of this product are demonstrated in Fig. 21.

7.3 Weighted covered cut out strong Cartesian graph product

Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

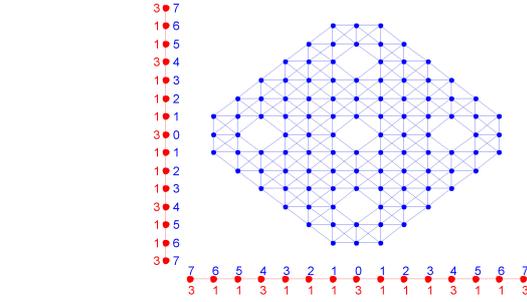


Fig. 21. Covered cut out strong Cartesian graph product

if $\{[[U_i = U_k \text{ and } (W_{jl} \neq 0 \text{ or } W_{lj} \neq 0)] \text{ or } [V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0]]$
and $[[W_{1i}, W_{1j}, W_{1k}, W_{1l} \leq 1] \text{ or } [W_{1i} \neq W_{1j} \text{ \& } W_{1k} \neq W_{1l}]]$
and $[W_{2i,2j}, W_{2k,2l} \leq W_{2max}]$

Some examples of this product are demonstrated in Fig. 22.

7.4 Weighted covered cut out semi strong Cartesian graph product

Using this product the nodes $U(i, j)$ and $V(k, l)$ in the product graph which the weights W_i, W_j, W_k and W_l are assigned to them, will be connected if the following conditions are fulfilled:

if $\{[[V_j = V_l \text{ and } (W_{ik} \neq 0 \text{ or } W_{ki} \neq 0)] \text{ or } [W_{ki} \neq 0 \text{ or } W_{lj} \neq 0]]$

and $[[W_{1i}, W_{1j}, W_{1k}, W_{1l} \leq 1] \text{ or } [W_{1i} \neq W_{1j} \text{ \& } W_{1k} \neq W_{1l}]]$
and $[W_{2i,2j}, W_{2k,2l} \leq W_{2max}]$

Some examples of this product are shown in Fig. 23.

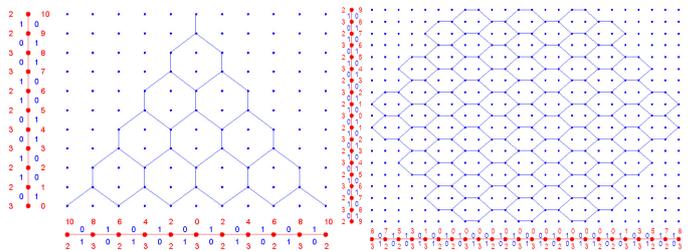


Fig. 22. Weighted covered cut out strong Cartesian graph product

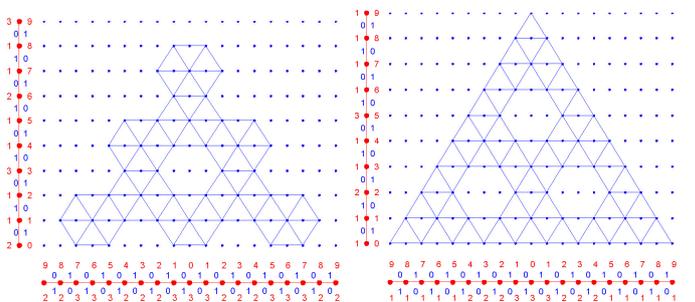


Fig. 23. Weighted covered cut out semi strong Cartesian graph product

8 Concluding remarks

Using the new weighted triangular graph product it becomes possible to generate triangular and a wide range of polygonal

shaped configurations. It is also possible to configure triangle, trapezoid, parallelogram and many combinations of triangular and rectangular shaped models. Employing triangular graph products, the formation of different configurations based on a simple algebra and graph theory becomes feasible. Circular graph products make it possible to generate the solid circles which are the models of some space structures like domes.

The use of graph products reduces the storage requirement for data processing of structures, and it makes it much easier and less costly in the computer. This is obvious since in place of the data for the entire model only the information for two much smaller subgraphs (generators) are needed to be stored.

References

- 1 **Nooshin H**, *Algebraic representation and processing of structural configurations*, Computers and Structures, **5**, (1975), 119–130.
- 2 **Nooshin H**, *Formex Configuration Processing in Structural Engineering*, 1st ed., Elsevier Applied Science Publishers, London, 1984.
- 3 **Nooshin H, Disney P, Yamamoto C**, *Formian*, 2nd ed., Multi-Science Publishers, UK, 1995.
- 4 **Nooshin H, Disney P**, *Formex configuration Processing I, II, III*, International Journal of Space Structures, **15, 16, 17**, (2000, 2001, 2002), 1–52, 1–56, 1–50.
- 5 **Behravesht A, Kaveh A, Nani M, Sabet S**, *A set theoretical approach to configuration processing*, Computers and Structures, **30**, (1988), 1293–1302.
- 6 **Kaveh A**, *A graph theoretical approach to configuration processing*, Computers and Structures, **48**, (1993), 357–363.
- 7 **Kaveh A**, *Structural Mechanics: Graph and Matrix Methods*, 3rd ed., Research Studies Press, UK, 2004.
- 8 **Kaveh A, Koohestani K**, *Graph products for configuration processing of space structures*, Computers and Structures, **86**, (2008), 1219–1236.
- 9 **Kaveh A, Nouri M**, *Weighted graph products for configuration processing of planar and space structures*, International Journal of Space Structures, **24**, (2009), 13–26.
- 10 **Thacker W C**, *A brief review of techniques for generating irregular computational grids*, International Journal for Numerical Methods in Engineering, **15**, (1980), 1335–1341.
- 11 **Thomson J F, Warsi Z U A, Hastin C W**, *Numerical Grid Generation, Foundations and Applications*, 1st ed., Elsevier, Amsterdam, 1985.
- 12 **Liseikin V**, *Grid Generation Methods*, Series: Scientific Computation, Springer Verlag, 1999.
- 13 **Kaveh A, Rahami H**, *Block diagonalization of adjacency and Laplacian matrices for graph products; Applications in structural mechanics*, International Journal for Numerical Methods in Engineering, **68**, (2006), 33–63.
- 14 **Kaveh A, Rahami H**, *A unified method for eigen-decomposition of graph products*, Communications in Numerical Methods in Engineering, **21**, (2005), 377–388.
- 15 **Imrich W, Klavzar S**, *Product Graphs; Structure and Recognition*, 1st ed., John Wiley, NY, USA, 2000.
- 16 **Kaveh A**, *Optimal Structural Analysis*, 2nd ed., John Wiley, Somerset, UK, 2006.
- 17 **Kaveh A, Alinejad B**, *Graph products with specified domains for configuration processing and formation of the adjacency matrices*, Engineering Computations, **27**, (2010), 205–224.