Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2012
August 12-15, 2012, Chicago, IL, USA

DETC2012-70904

APPLYING RIGIDITY THEORY METHODS FOR TOPOLOGICAL DECOMPOSITION AND SYNTHESIS OF GEAR TRAIN SYSTEMS

Terushkin Maria
School of Mechanical Engineering
Tel Aviv University
Tel Aviv, Israel

Shai Offer
School of Mechanical Engineering
Tel Aviv University
Tel Aviv, Israel

ABSTRACT

This paper introduces a novel way to augment the knowledge and methods of rigidity theory to the topological decomposition and synthesis of gear train systems. A graph of gear trains, widely reported in the literature of machine theory, is treated as a graph representation from rigidity theory—the Body-Bar graph. Once we have this Body-Bar graph, methods and theorems from rigidity theory can be employed for analysis and synthesis. In this paper we employ the pebble-game algorithm, a computational method which allows determination of the topological mobility of mechanisms and the decomposition of gear trains into basic building blocks—Body-Bar Assur Graphs. Once we gain the ability to decompose any gear train into standalone components (Body-Bar Assur Graphs), this paper suggests inverting the process and applying the same method for synthesis. Relying on rigidity theory operations (Body-Bar extension, in this case), it is possible to construct all of the Body-Bar Assur Graphs, meaning the building blocks of gear trains. Once we have these building blocks at hand, it is possible to recombine them in various ways, providing us with a topological synthesis method for constructing gear trains. This paper also introduces a transformation between the Body-Bar graph and other graph representations used in mechanisms, thus leaving room for the application of the proposed synthesis and decomposition method directly to known graph representations already used in machine theory.

Keywords: Topological synthesis, gear trains, rigidity theory, Body-Bar graph, Assur graph, pebble game

1. INTRODUCTION

The topological synthesis of gear trains is widely reported in the literature, its outset dating back to the works of Buchsbaum and Freudenstein in 1970 [3]. Since then, many works have been developed in this topic, many of them rely on graph theory [7], [6], and in robotics [4]. Most of the examples checked in this paper are taken and compared to those that appear in Tsai's book [19]. The authors of this paper are not aware of any work on topological synthesis of gear trains that employs rigidity theory.

This paper introduces a different approach to topological synthesis which relies on rigidity theory. Rigidity theory is a combinatorial theory that encompasses topological theorems and methods used to prove whether graphs, or the corresponding physical systems represented by the graphs, are rigid or mobile [16], [17] [20]. Gear trains are represented in this paper by the Body-Bar graph widely used in rigidity theory; thus, theorems and methods developed for this graph are available for application to topological synthesis. Furthermore, section 5 of this paper demonstrates the relationships between the Body-Bar graph and other well-known graph representations of gear trains [3], [11]. We expect these relationships to make it possible, in the near future, to apply the methods and theorems reported in this paper directly to existent gear train graphs. This paper is organized as follows:

In section 2 we introduce the Body-Bar graph from rigidity theory and explain how we use it to represent mechanisms. In principal, any mechanism, including spatial ones, can be represented by the Body-Bar graph: the bars define the constraints on the bodies, and as result they define the motions of the mechanisms bodies.

In section 3 we use the well-known concept of Assur Groups [1] as reformulated according to the terminology of rigidity theory in 2010 [13]. In this paper, we will refer to them as Assur Graphs. In rigidity theory there are two types of Assur Graphs: Bar-Joint and Body-Bar. In this paper we use only the latter. Thus, for the sake of brevity from now on when we write Assur Graph we relate to Body-Bar Assur Graph. We firstly demonstrate that Assur Graphs are the building blocks of gear trains. We then employ one of the known operations of rigidity theory—the extension operation by which all the Assur Graphs can be constructed. Once we have all the components of the gear trains (the Assur Graphs), we will introduce a method of composition which will allow us to derive various gear trains.

In section 4 one of the most well-known combinatorial algorithms, the pebble game, is introduced and explained briefly. It is interesting to note that, although this algorithm was developed for the purpose of determining the topological mobility of physical systems, in 2010 it was found to be applicable in decomposing mechanisms into Assur Graphs [12], as it is used in this paper.

After establishing the mathematical foundations in the above sections, in sections 5 and 6 we introduce the proposed combinatorial method for topological decomposition and synthesis of gear trains. In section 5 we introduce the representation of gear trains by the Body-Bar graph. Furthermore, the transformation between existent graph representations of gear trains and the Body-Bar graph (and vice versa) is explained in detail. This is important because it will enable us to apply the proposed method in the future to the existent gear-train representations. In section 6.1 we explain how to use the pebble-game algorithm, introduced in section4, to decompose any gear train into components (Assur Graphs). In section 6.2 we rely on section 3, where we introduced the derivation of all the Assur Graphs, to demonstrate the recombination of gear-train components, Assur Graphs, into various gear trains.

2. THE BODY-BAR GRAPH IN RIGIDITY THEORY

This type of graph is the cornerstone representation of this paper, and a variant of it appears in many works in the machine theory, as explained in sections 5.1 and 5.2.

Definition: The Body-Bar graph G=(B,E) consists of |B| bodies and |E| constraints/bars. When dealing with this type of graph, the main focus is on the bodies and the constraints between them. In mechanisms, each type of kinematic pair corresponds to a set of constraints/bars. Because a higher/lower pair imposes one/two constraint(s) between the bodies, there is(are) one/two bar(s), respectively. The order of the body is defined according to the number of bars incident to it. As an example, the mechanism in Figure 1a has five links (including the ground); thus, in its graph (Figure 1b) there are five bodies. Link 1 is connected to link 2 through a revolute joint: there are two bars between body 1 and body 2, while there is one bar between body 2 and body 4 since the corresponding links are connected through a higher pair.

For the sake of simplicity, two bars connecting two bodies can be replaced by a revolute joint resulting in a more compact graph as shown in Figure 1c.

In rigidity theory [18] the convention is to draw the bodies as circuits instead of polygons, as shown in Figure 1d. Since the graph is a topological representation, there is, of course, no difference between the variants in Figure 1 since all are topologically equivalent.

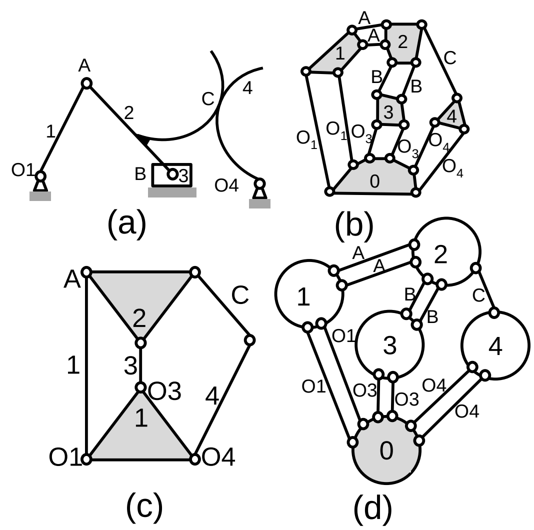

Figure 1 - Example of the Body-Bar graph of a mechanism.
(a) The mechanism; (b) The Body-Bar graph with polygons; (c) The Body-Bar graph with revolute joint; (d) Bodies in the graph circuits, body 0 is grey to indicate ground link.

In the following section we introduce the building blocks of this type of representation and the operations needed to construct these building blocks. In section 4.2 we introduce an efficient algorithm from rigidity theory for decomposing the graphs into building blocks. Since we know how to construct these building blocks, section 6.2 introduces a topological gear train synthesis method in which we construct various building blocks, combine them, and convert them into gear trains. The building blocks rely on ***Assur Groups*** [1] (which were reformulated by the terminology of rigidity theory and are referred to as ***Assur Graphs***[13]) as described in the following section.

3. ASSUR GRAPHS – THE BUILDING BLOCKS OF THE BODY-BAR GRAPH

As was mentioned in the introduction, the paper employs specific types of graphs, called Assur Graphs originated from Assur Groups [1],[2] but were reformulated according to the terminology of rigidity theory as appear in [13]. There are various types of Assur Graphs, such as bar-joint Assur Graphs, Baranov Trusses [9] and more that are not used in the paper, thus are not mentioned. The type of Assur Graph used in the paper is defined below.

3.1. BODY-BAR ASSUR GRAPHS

Definition: Graph G is a ***Body-Bar Assur Graph*** iff G has 3DOF and does not contain any sub-graph (of more than one element) which also has 3DOF.

As an example, the graph in Figure 2a is an Assur graph since it has 3DOF and none of its sub-graphs are rigid. On the other hand, Figure 2b is not an Assur Graph because the three designated bodies with their six connecting bars constitute a rigid body with 3DOF.

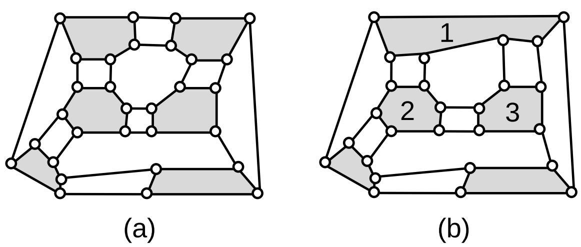

Figure 2 - (a) Body-Bar Assur Graph; (b) Not an Assur Graph.

In the following section we will introduce the construction rule with which it is possible to build all the building blocks (Assur Graphs) of Body-Bar Graph using only a single operation.

3.2. DERIVING ALL THE 2D ASSUR GRAPHS

In this section it is shown that the Assur Graphs can be derived from a single graph, called the ***primary Assur Graph,*** and a series of extension operations each time a body is augmented. The primary Assur graph consists of two bodies connected by three bars as shown in Figure 3a. Note that the primary graph is an exceptional form since exactly three bars connect the two bodies. The mathematical explanation for this exception is beyond the scope of the paper, but the extension rule given below assures that all other Assur Graphs that are constructed can have at most two bars connecting between two bodies.

Another graph used in this section is a body with three bars, (shown in Figure 3b) termed in this paper the ***Body-Bar atom*** and for brevity called the ***atom***. The extension of Assur Graphs is executed by sequentially augmenting the atoms, as described in the extension rule below.

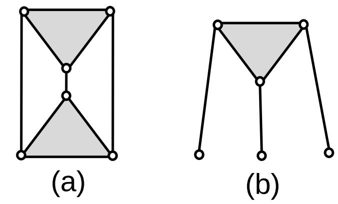

Figure 3 - The primary Assur graph (a) and the body bar atom (b).

The following extension operation relies on the theorem in [18], according to which it has been proved that any Body-Bar Assur Graph can be derived from the primary graph by applying a sequence of extension rules, defined as follows:

Extension operation for Assur Graphs: Suppose we have an Assur Graph (we will refer to it as the 'original graph') and we want to extend it by adding an atom termed *an augmented atom*. The extension process is carried out using the following steps:

1. Add the augmented atom and connect its three bars to the existing elements in such a way that it will be connected to at least two elements, each of which can be either a body or a bar.
2. Disconnect one or two bars in the original Assur Graph and reconnect them to the augmented atom.

Figure 4 demonstrates the process of the extension operation. The gray region represents the original graph and body q is the augmented atom connected to bodies y and w as shown in Figure 4b. The bar connecting bodies z and w in the original Assur Graph is disconnected and reconnected between the augmented atom and body z.

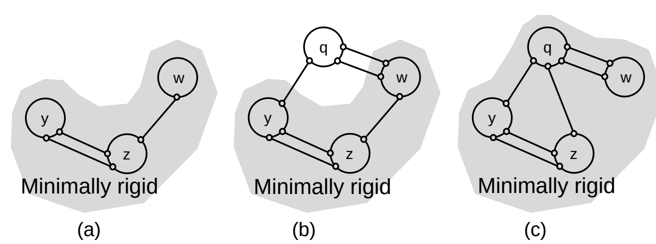

Figure 4 - Extension of Body-Bar Assur graph. (a) The original Assur Graph; (b) The augmented atom q; (c) Disconnecting the bar (z,w) and connecting it to q.

Figure 5 shows the process of deriving Body-Bar Assur graphs from the primary graph by applying the extension operation twice. Note, the graph in Figure 5b appears in [10].

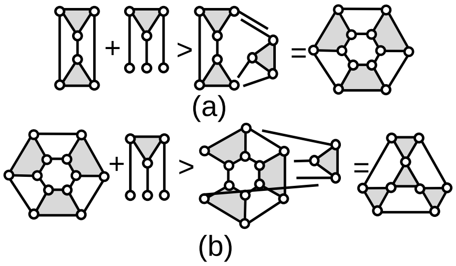

Figure 5 – Example of applying the extension rule. (a) Extending the primary graph; (b) Applying the extension rule to Assur resulted in Figure 5a.

The basic property of Assur Graphs is that they cannot be divided/decomposed into smaller units (otherwise the condition in the definition is not met). Conversely, every mechanism can be decomposed into Assur Graphs. A computational method that was developed for determining the mobility of physical systems turned out to be also useful for decomposing mechanisms into Assur Graphs [12]. This method is known as the pebble game (first reported in 1997 [5]) and is introduced in the following section.

4. THE COMBINATORIAL ALGORITHM FOR DETERMINING MOBILITY—THE PEBBLE GAME

The algorithm introduced in this section was developed in rigidity theory with the aim of determining the topological mobility of physical systems. Although its original intentions, in 2010 it was mathematically proved that applying this algorithm on any mechanism (including spatial mechanisms) decompose it into Assur Graphs. In this section we introduce the algorithm and in section 6.1 we will apply it to the Body-Bar graph of gear-trains resulting in the decomposition of the gear trains into components/Assur Graphs.

4.1. THE COMBINATORIAL ALGORITHM

Let G=(B,E) be a Body-Bar graph with |B| bodies and |E| bars/edges. Each body is given three pebbles, corresponding to the 3DOF possessed by a body in a planar system. A body can use its pebbles to cover any three edges which are incident to that body.

When a bar is directed, one constraint is added; therefore, one degree of freedom (one pebble) is removed from the total system.

Every ungrounded graph, termed in the paper ***floating graph***, must contain at least 3 DOF (of a body in the plane); therefore, three pebbles are always present in the graph. To direct a bar, at least four pebbles should be available on its end bodies. The number of pebbles on each body is not important, the only reservation is that the sum of the pebbles on the two end bodies of a bar should be greater than or equal to four. The bar is being directed by moving one pebble to the edge, defining a constraint between its two end bodies. An example of assigning a pebble to a bar, directing the bar, appears in Figure 6 where there are two pebbles on each body.

Figure 6 - Example of assigning a pebble to a bar (constraint).

Once at least four pebbles are located at the end bodies of the desired edge, we are guaranteed that these bodies are independent in the plane and that adding a bar between them will not result in an over-constrained system. The fact that four pebbles are necessary is used in the algorithm by quadrupling the bar under test before it is being directed. If the independence test is successful, the bar can then be directed: a constraint is added between the bodies, and one degree of freedom is removed from the total system (one pebble is used to mark the bar). In fact, this operation defines a constant distance between the end bodies, and this constraint equals one DOF.

The mains steps of the algorithm are:

1. Arbitrarily pick a body to begin with.

2. Next, there are two possible moves:

2.1 Direct a bar: If i and j are bodies and the sum of pebbles on these two bodies is at least four (i.e., the bodies are independent; there is no constraint between them), assign a pebble from one of the end bodies to the bar, suppose from vertex i (wlog). The bar (i,j) is now directed from i to j (i.e., <i,j>).

In the case that a total sum of at least four pebbles cannot be found at the ends of a constraint after exhaustive search, the bar should be marked as redundant.

2.2 Slide a pebble: If there is a constraint between vertex i and vertex j, then there exists a directed bar <i,j>, and there must be a pebble on j. We should reverse the direction of the bar (so that it is directed from j to i) and move the pebble from j to i.

The algorithm ends when all the bars have been processed (either directed or not directed).

For the sake of demonstration, let us see how the pebble game concludes that the floating Body-Bar graph (the triangle in Figure 7a) is rigid. First, three pebbles are assigned to each vertex (Figure 7b). One of the bars (a,b) can be directed since the sum of pebbles on its end bodies is greater than four (Figure 7c). The same is true for bar (b,c) in Figure 7d, both bars (a,c) in Figure 7e and f, and the remaining bar (a,b) in Figure 7g. In order to direct the remaining bar (b,c), the total sum of at least four pebbles must be moved to its end bodies.

Therefore, a pebble from vertex a moves to vertex c, and the direction of (a,c) is reversed (Figure 7h). Next, (b,c) can be directed (Figure 7i). Now that all bars have been processed, all bars are directed, and there are three free pebbles on the graph (3 DOF of a rigid body): the graph is rigid without over-constraints.

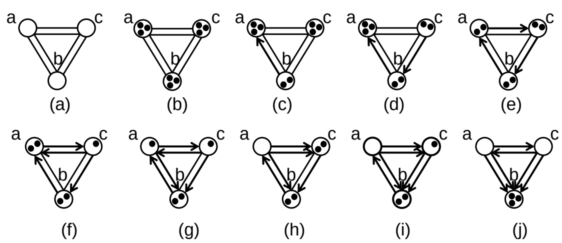

Figure 7– Example of applying the Pebble game for validating the rigidity of a triangle. (a) undirected graph G (b) each vertex is given 3 pebbles (c) ,(d), (f) direct an edge move (e) pebble slide move. Minimally rigid graph, three free pebbles left.

4.2. USING THE PEBBLE GAME TO DECOMPOSE BODY-BAR GRAPHS INTO ASSUR GRAPHS

As was mentioned in the introduction, the pebble game was developed for determining the topological mobility of graphs. In 2010, it was mathematically proved that the pebble game also decomposes any mechanism into Assur graphs [14]. The algorithm is based on Laman [8] count, and later was extended to Body-Bar graphs [15].

The decomposition relies on the fact that the pebble game gives directions to the bars which enables the use of the property of directed cut-sets, defined as follows:

Definition: A directed cut-set is a set of edges whose removal disconnects (i.e., separates) the graph into components and all the edges in this set are directed from one component to the others.

As an example, the three edges going out of body 5 in Figure 8a define a cut-set because, when they are removed, body 5 is separated from the graph and all edges are directed from body 5 to the other graph.

Having this definition, the sequentially directed cut-sets define the order of the decomposition of the Body-Bar Graph into Assur Graphs. For example, in Figure 8a, body 5 was found first to be removed, then body 1 (Figure 8b), resulting in a basic system with one DOF (bodies 2,3,4).

In section 6.1 we will use the pebble game to decompose any gear train into components (Assur Graphs). In section 6.2 we use the pebble game conversely: relying on the extension operation (section 3.2), it is possible to construct all the Assur Graphs and resulting with complex gear trains.

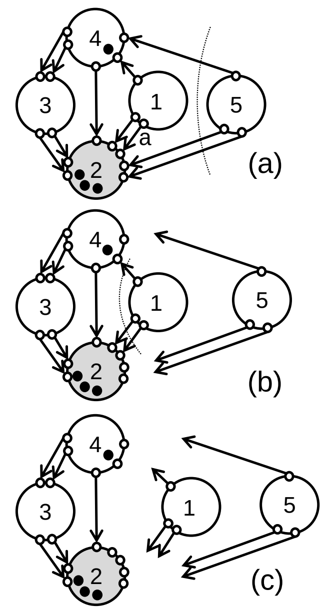

Figure 8 - Example of sequential decomposition of a Body-Bar graph into Assur Graphs by using the pebble game.

5. REPRESENTING GEAR TRAINS AS BODY-BAR GRAPHS

Let us take a Minuteman Cover Drive example from Tsai's textbook [[19], Fig. 7.19] (Figure 9), and represent it as a Body-Bar graph.

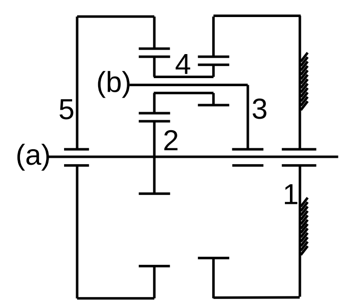

Figure 9 - Minuteman cover drive gear train.

To clarify the relationship between the Body-Bar graph and Tsai's graph, both are given in Figure 10. As can be easily seen, they are equivalent. The bold lines in Tsai's graph relate to gear pairs corresponding to the single bars connecting two bodies in the Body-Bar graph. All other lines relate to lower pairs and therefore are represented by two parallel bars in the Body-Bar graph since they remove two DOF from the total system.

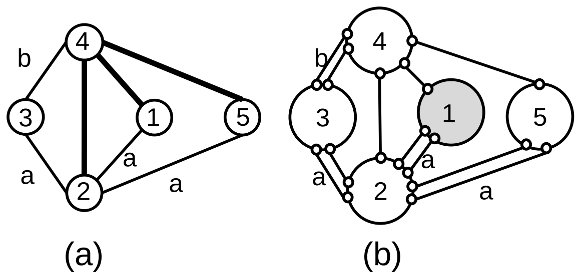

Figure 10 - (a) Tsai's graph; (b) Body-Bar graph.

5.1. THE RELATION BETWEEN THE BODY-BAR GRAPH OF GEAR TRAINS AND OTHER KNOWN GEAR-TRAIN REPRESENTATIONS

As mentioned in the introduction and throughout the paper, the theorems and algorithms used in this paper were proved based on rigidity theory Body-Bar graphs. This is the main motivation for our use of this graph type. In this section we highlight a further application of the work reported in this paper: the application of the methods and algorithms to known and widely used representations in the mechanism literature. This can be achieved by demonstrating the equivalence of the rigidity theory representation and other known representations (in this case, graph and kinematic structural representations [19]). This relationship paves the way for further application. The algorithms and methods reported in this paper can be applied directly to the known representations of mechanisms in general and of gear trains in particular.

This section will briefly explain the implications of this unified notion by demonstrating the transformation of the Body-Bar graph of the gear trains into gear-train graph representation. To show that this idea is applicable in either direction, section 5.3 demonstrates the transformation of the kinematic structural representation of gear trains into the corresponding Body-Bar graph. A detailed explanation and algorithms for transformation between the rigidity theory representations and mechanism representations will be reported in forthcoming publications by the authors.

The transformation between the Body-Bar graph and known representations in machine theory is demonstrated in Figure 11. A gear-train is shown in Figure 11a, taken from [19]. Figure 11b is the corresponding Body-Bar graph. Note, as was mentioned in section 2, the bodies in the graph can be drawn as circles or polygons and are equivalent since we are referring to the topology of the graph and not its geometry.

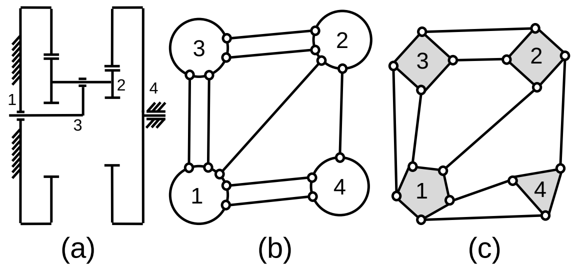

Figure 11 - Example of the body-bar graph of an epicyclic gear train. (a) Function schematic; (b) Body –bar graph; (c) Body-bar graph with polygons.

5.2. TRANSFORMING A BODY-BAR GRAPH INTO ITS CORRESPONDING GEAR-TRAIN GRAPH REPRESENTATION

Let G=(B,E) be a Body-Bar graph with |B| bodies and |E| constraints. The first step is to replace the gear pairs. Since the gear pairs impose one constraint on the two connected bodies, a single bar between the bodies represents them. Thus, all the single bars are replaced by thick edges. For example, there is a single bar/constraint between bodies 2 and 1 and the same between bodies 2 and 4 (Figure 12a); thus, these two bars are replaced by two thick edges as shown in Figure 12b.

The second step is to replace the turning pairs presented in the Body-Bar graph with two bars/constraints between the two bodies. Here, there are two bars between the pairs of the following bodies: (3,2), (3,1) and (1,4); thus, between the latter pairs of bodies, there should be a single edge as shown in Figure 12c.

The last step is to replace all the bodies with vertices. In the gear-train graph all the vertices correspond to links, (i.e. in the terminology of rigidity theory, they correspond to bodies) thus, the four bodies are "shrunk" into the four vertices, as appears in Figure 12d.

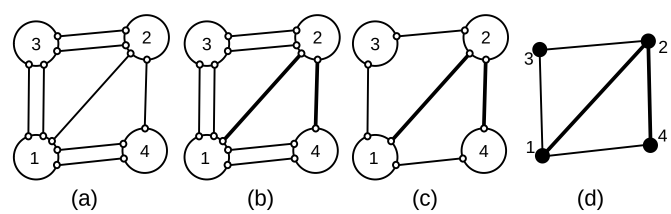

Figure 12 - Transforming the Body-Bar of the gear train in Figure 11 into its corresponding graph representation. (a) The Body-bar graph of the gear trains; (b) Replacing the single constraints with thick edges corresponding to the gear pairs; (c) Replacing two constraints between two bodies with one edge corresponding to a turning pair; (d) Shrinking the bodies into vertices, resulting in the graph representation of the gear train in Figure 11.

In the following section we show that this transformation also works the other way around. This is demonstrated by showing the transformation from one of the known representations used in the mechanism community. We transform a kinematic structural representation into its corresponding representation in rigidity theory – the Body-Bar graph.

5.3. TRANSFORMING A KINEMATIC STRUCTURAL REPRESENTATION INTO THE CORRESPONDING BODY-BAR GRAPH

Suppose you are given a structural representation of a gear train. The process of deriving the Body-Bar graph for the given gear trains is as follows:

1. Replace the solid vertices, corresponding to the gear pairs, with single edges, indicating that there is a single constraint between the corresponding two bodies. In Figure 13b, two edges were added between the two pairs of bodies: (1,2) and (2,4).

2. Replace the revolute joints with two edges, indicating that there are two constraints between the two bodies. For example, the revolute joints A, B, and C were replaced as shown in Figure 13c. Note that at this step the order of the bodies increases. For example, links 4 and 3 were replaced by ternary and quaternary bodies as shown in Figure 13c.

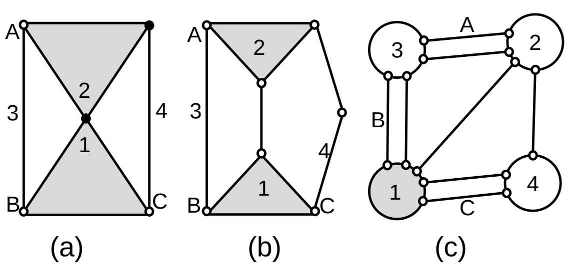

Figure 13 - Transforming the kinematic structural representation of the gear train from Figure 11 into its corresponding Body-Bar graph. (a) The kinematic structural representation of the gear train; (b) Replacing the solid vertices, corresponding to gear pairs, with single constraints/bars; (c) Replacing the binary links, 3 and 4, into bodies with order four and three, respectively.

6. COMBINATORIAL DECOMPOSITION AND SYNTHESIS OF GEAR TRAINS THROUGH BODY-BAR GRAPHS

In this section we first introduce gear trains as Body-Bar graphs, as explained in detail in section 3.1. Once we have the graph, we apply the pebble game which decomposes the graph and its corresponding gear trains into components/Assur graphs, as explained in section 6.1.

In section 6.2 it will be shown that it is possible to go the other way around. Also, relying on section 3.2 (where it was shown how to construct Body-Bar Assur Graphs, the corresponding components of gear-trains), it will be shown that the composition of these components results in a combinatorial method for the synthesis of gear trains.

6.1. COMBINATORIAL DECOMPOSITION OF GEAR TRAINS INTO COMPONENTS/ASSUR GRAPHS THROUGH PEBBLE GAME

In section 5 we explained how to represent gear trains with Body-Bar graphs. Once we have the Body-Bar graph, which is widely used in rigidity theory, we can apply the pebble game to it, as introduced in section 4. Executing the pebble-game algorithm on the Body-Bar graph results in a directed graph, i.e., all the bars are directed. Having these directions, the directed cut-sets (sets of edges whose removal separates the graph) define the decomposition into Assur Graphs. The order of the cut-sets defines the decomposition order.

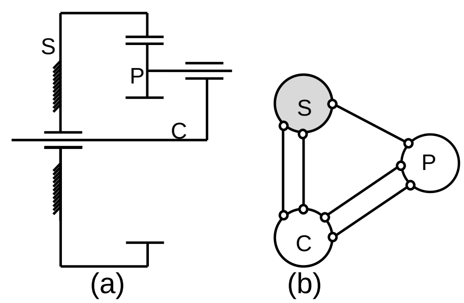

Figure 14 – Example of representing planetary gear systems by a Body-Bar graph. (a) Basic planetary gear system. (b) Corresponding Body-Bar graph.

As an example, we will apply the pebble game to the Body-Bar graph in Figure 14.

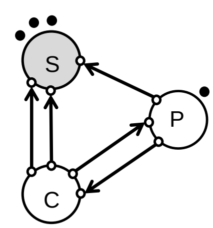

Figure 15 - Pebble game result for planetary system

As seen in Figure 15, the system has one DOF (three pebbles at the ground vertex and one free pebble, indicating one DOF).

Another example appears in Figure 16, in which we apply the pebble game to the corresponding Body-Bar graph. According to the directions of the bars, we derive the sequence of decompositions.

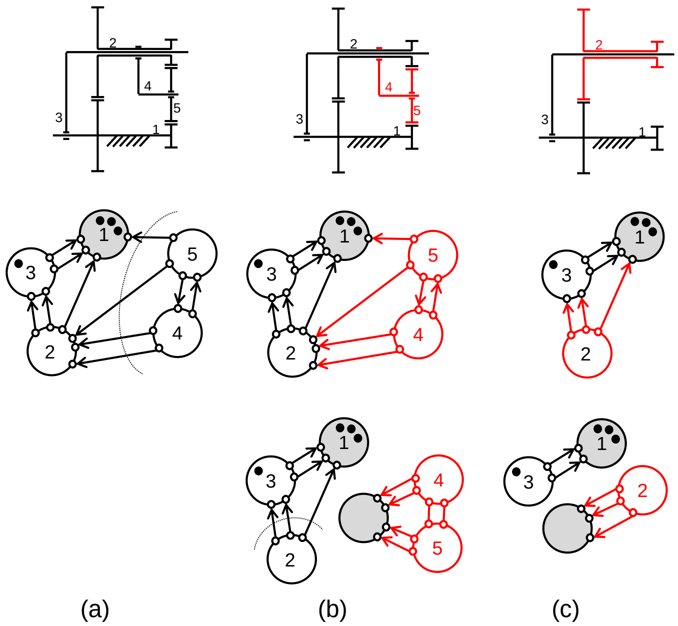

Figure 16 – Example of a sequence decompositions of a gear trains into Assur Graphs.

From Figure 16 it can be seen that after applying the pebble game algorithm, directed cut-sets can be identified (designated by dashed lines). The first cut-set defines the Assur Graph containing links 4 and 5, as can be seen in Figure 16b (components are shown below the original graph). Following this, another directed cut-set can be identified, and link 2 can be removed (Figure 16c). The Body-Bar atom is now derived.

The combinatorial approach can also be employed for topological synthesis as explained in the following section. We refer to topological synthesis methods relying on rigidity theory as a **combinatorial synthesis**.

6.2. COMBINATORIAL SYNTHESIS OF GEAR TRAINS BY COMPOSITIONS OF BODY-BAR ASSUR GRAPHS

In section 3.1 we introduced the extension operation by which it is possible to derive all the Body-Bar Assur Graphs (i.e., all the building blocks/components of gear trains). Note, as was mentioned in section 3.1, it has been mathematically proven that it is possible to derive all the building blocks/Assur Graphs in 2D relying on a well proven theorem in rigidity theory [18]. Now that we are able to systematically derive all the building blocks of gear trains, new methods for combinatorial synthesis of gear trains are available through the composition of different Body-Bar Assur Graphs constructed by applying the extension operation (section 3.1). Moreover, as mentioned above, the mathematical foundation of this work emanate from rigidity theory, a branch in discrete mathematics. Since the latter mathematics is known as the mathematical foundation of computer science, this implies a new and efficient way of computerizing the process of synthesis detailed in this section.

Below we introduce the idea of constructing gear trains through the composition of Assur Graphs. For example, several Assur Graphs appear in Figure 17. We will recombine their corresponding gear train components resulting in complex gear trains, given in Figure 20. Each of the Assur Graphs appearing inf Figure 17 can be derived by applying a sequence of extension operations starting from the primary Assur Graph.

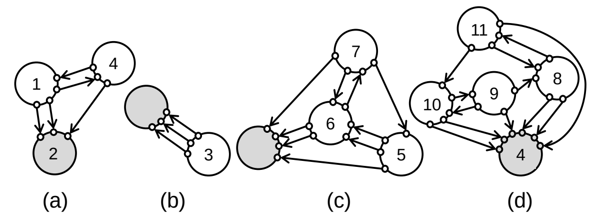

Figure 17 - Building blocks of the gear trains in Figure 19.

In Figure 18 we show the process of deriving the Assur Graph in Figure 17d.

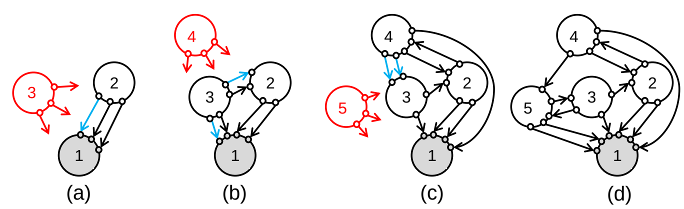

Figure 18 - Assur graph derivation from Body-Bar atom.

In Figure 18a we begin with the primary Assur Graph consisting of two bodies: 1 and 2 connected by three bars, as was explained in section 3.2. To this primary graph we add the atom 3 (colored red), disconnect one bar (colored blue) between bodies 1 and 2, and reconnect it between bodies 2 and 3. The result of this extension is an Assur Graph consisting of three bodies (Figure 18b). The extension operations continue several times. Each time we add a body (red), disconnect one or two bars (blue) and reconnect the blue bar(s) to the augmented atom. By applying this operation repeatedly, the Assur Graph in Figure 18d is derived.

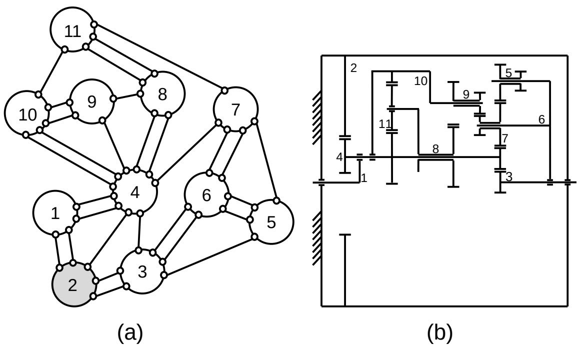

Figure 19 - (a) Body-Bar graph composed of an atom, driving link, atom and two Assur graphs. (b) Corresponding gear train system

In Figure 20 we can see the sequence of Assur Graphs composition resulting in a gear train. In each step an additional Assur Graph is added. In Figure 20b an atom, the simplest Assur Graph, is added (colored red). Note, as was mentioned in section 3.2, each atom must be connected to at least two elements. After augmenting four Assur Graphs, as shown in Figure 20c,d,e the resultant gear train appears in Figure 20e.

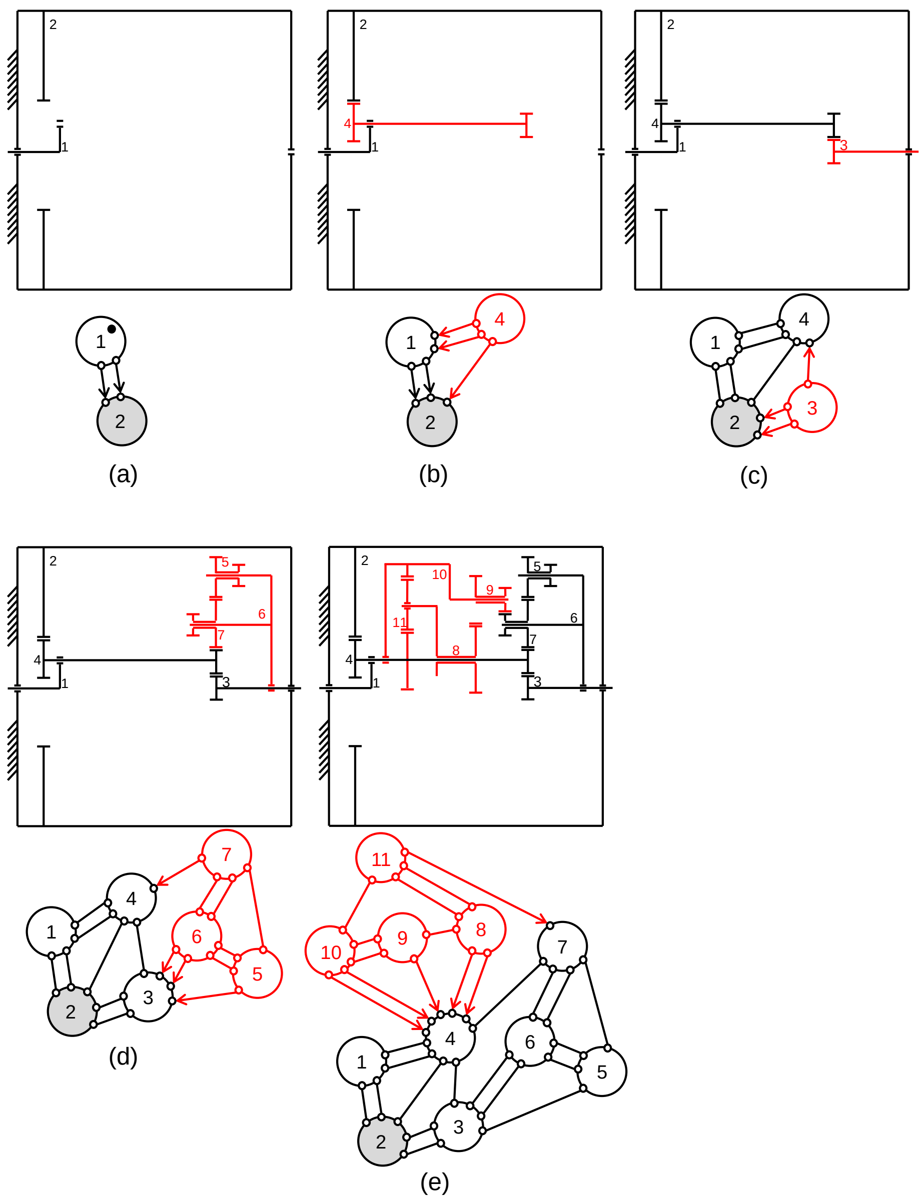

Figure 20 – Example of the combinatorial synthesis of a gear train. (a) Body-Bar atom with 1- DOF (pebble); (b) Planetary system with 1- DOF; (c) Addition of Body-Bar atom (link 3); (d) Addition of another Assur Graph (links 5,6,7); (e) Addition of the last Assur Graph (links 8, 9, 10, 11).

CONCLUSIONS AND FURTHER RESEARCH

This paper introduced a combinatorial approach for the decomposition and synthesis of gear trains. Since this approach is combinatorial, it is easy to implement both the decomposition and the synthesis via computer software. Additionally, because this work relies on body-bar graphs, mathematically proven to be valid also in 3D, this work is expected to give rise to future development in the field of combinatorial synthesis of gear trains in 3D.

ACKNOWLEDGMENTS

The authors would like to thank Dima Mazor for his help.

REFERENCES

[1] L.V. Assur: Investigation of Planar Framed Mechanisms with Lower Pairs with Relation to their Structure and Classification. M.: Publishing Hous of the USSR Academy of Sciences, 1952. pp. 588.

[2] G. G. Baranov, “Classification, construction, kinematics, and kinetostatics of mechanisms with pairs of first kind,” Proc. Seminar on Theory of Machines and Mechanisms [in Russian], Vol. 2, No. 46 (1952), pp. 15–39.

[3] F. Buchsbaum, F. Freudenstein: Synthesis of kinematic structure of geared kinematic chains and other mechanisms, Journal of Mechanisms 5 ,1970, pp. 357–392.

[4] D. Z. Chen, S. C. Shiue,: Topological Synthesis of Geared Robotic Mechanism, ASME J. Mech. Des., 120, 1998, pp. 230–239.

[5] D. J. Jacobs and B. Hendrickson. An algorithm for two-dimensional rigidity percolation: the pebble game. Journal of Computation Physics, 137, pp. 346-365, 1997.

[6] C.H. Hsu, A graph representation for the structural synthesis of geared kinematic chains, J. Franklin Inst., Vol. 330 (1993), pp. 131–143.

[7] C.H. Hsu, K.T. Lam: A new graph representation for the automatic kinematic analysis of planetary spur-gear trains, Mathematical and Computer Modeling ,vol. 19, No. 11, June 1994, pp. 67–81

[8] G. Laman: On Graphs and the Rigidity of Plane Skeletal Structures, J. Engineering Mathematics, 4 (1970), 331-340.

[9] N.I. Manolescu: A method based on Baranov trusses and using graph theory to find the set of planar jointed kinematic chains and mechanisms, Mechanism and Machine Theory vol. 8, No. 1, 1973, pp. 3–22.

[10] E. E. Peisakh: An algorithmic description of the structural synthesis of planar Assur groups, Journal of Machinery Manufacture and Reliability , vol. 36, No. 6, 505-514

[11] R. Ravisankar, T.S. Mruthyunjaya: Computerized synthesis of the structure of geared kinematics chains, Mechanism and Machine Theory 20 (1985) 367–387.

[12] O. Shai, A. Sljoka, W. Whiteley: Directed Graphs, Decompositions, and Spatial Rigidity submitted to Discrete Applied Mathematics, 2010.

[13] B. Servatius, O. Shai, W. Whiteley: 2008, Geometric Properties of Assur Graphs, Computational Geometry, Volume: 39, Issue: 1, Pages: 55-64

[14] A.Sljoka, O. Shai, W. Whiteley: Checking mobility and decomposition of linkages via Pebble Game Algorithm, ASME Design Engineering Technical Conferences, August 28-31, 2011, Washington.

[15] A. Lee, I. Streinu: Pebble Game Algorithms and (k, l)-Sparse Graphs, proc. EuroComb 2005, DMTCS proc. A.E. (2005), pp. 181-186.

[16] T.-S. Tay: Rigidity of multigraphs I: linking rigid bodies in n-space. Journal of Combinatorial Theory Series, B, vol. 26, pp. 95–112, 1984.

[17] Tiong-Seng Tay: Linking (n - 2)-dimensional panels in n-space II: (n- 2; 2)-frameworks and body and hinge structures. Graphs and Combinatorics, 5, pp. 245-273, 1989.

[18] T-S. Tay: Henneberg’s Method for Bar and Body Frameworks. Structural Topology 17, 1991, 53-58.

[19] Lung-Wen Tsai: Enumeration of Kinematic Structures According to Function, CRC Press.

[20] W. Whiteley, T.-S. Tay, N. White: Skeletal rigidity of simplicial complexes I, European J. of Combinatorics 16, 1995, pp. 381-403.