



A constraint graph representation of metamorphic linkages

Duanling Li^{a,*}, Zhonghai Zhang^a, J. Michael McCarthy^b

^a Automation School, Beijing University of Posts and Telecommunications, Beijing, China

^b Robotics and Automation Laboratory, University of California, Irvine, CA 92697, United States

ARTICLE INFO

Article history:

Received 16 March 2010

Received in revised form 6 September 2010

Accepted 21 September 2010

Available online 16 October 2010

Keywords:

Metamorphic mechanisms

Configuration changes

Constraint graph

Adjacency sub-matrix

ABSTRACT

A metamorphic linkage has the property that the effective number of links and joints can change during the movement of the device. This means the kinematic structural representation of a metamorphic linkage has different forms in which vertices and edges combine depending on the configuration of the device. In this paper, we use the constraint graph of computational geometry rather than the traditional topological graph to characterize a metamorphic linkage in order to simplify the representation of its configuration changes. A constraint graph has geometric elements as vertices and their relationships as edges. We find that the adjacency sub matrix of the constraint graph provides a convenient description of changes in the topology of links and joints in the operation of the metamorphic linkage. Operations on the adjacency submatrix capture topological changes in a metamorphic linkage. We illustrate our results with several examples.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

In this paper, we introduce a constraint graph representation of the topological structure of a linkage. A constraint graph defines the relationships between geometric objects and is used in computer aided design systems. For our purposes, we consider the links of a mechanism to be lines and the revolute joints to be points in structural schematic, and use the constraint graph to represent incidence, distance and angle constraints. This approach provides a mathematical representation for the combinations of the links and joints that occur in metamorphic linkages.

The traditional topological graph of a linkage identifies the links as vertices and the joints as edges. Our constraint graph of the linkage represents both the links and joints as vertices and the geometric relationships between the links and joints define the edges of the graph. We find that the constraint graph for a mechanism yields a matrix with a simplified structure that we call the “adjacency sub-matrix”.

Operations on the adjacency submatrix obtained from the constraint graph of a metamorphic mechanisms define combinations of links and joints. The result is a graph representation for linkages that includes the special characteristics of metamorphic mechanisms. In what follows, we illustrate this theoretical examples of metamorphic mechanisms and the associated transformations of their constraint graphs.

2. Literature review

Dai and Rees Jones (1999)[1] examine a class of mechanisms that are erectable, which means they can move from a folded to a deployed configuration. They note that a primary feature of these devices is that the number of effective links changes as the device moves from one configuration to another, for this reason they call these devices *metamorphic mechanisms*.

* Corresponding author. Tel./fax: +86 10 62282629.

E-mail addresses: liduanling@163.com (D. Li), zhzhonghai@sina.com (Z. Zhang), jmmccart@uci.edu (J.M. McCarthy).

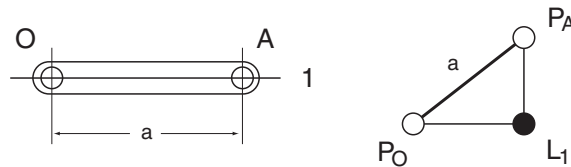


Fig. 1. A binary link and its constraint graph.

Mechanisms can be viewed as assemblies of rigid links connected by joints represented by the notation (n, m) , where n is the number of links and m is the number of joints, see Tsai (2000) [2]. This allows us to group crank-rocker, slider-crank, drag-link and other four-bar linkages into the $(4, 4)$ type of linkage. Using this notation, the metamorphic linkages have the ability to change type, that is an (n_1, m_1) linkage can move to a configuration in which links and joints combine such that it becomes of type (n_2, m_2) .

Freudenstein and Maki (1979) [3] introduced a linkage graph to describe the topological structure of a linkage and to facilitate invention, see also Vucina and Freudenstein (1991) [4]. The linkage graph represents links as vertices and joints as edges. The topological structure of this graph is defined by its vertex adjacency matrix. Dai and Rees Jones (2005) [5] used changes to the adjacency matrix of a topological graph to characterize a metamorphic mechanism. While the combination of both joints and links can occur in metamorphic mechanisms, only changes to links are represented in the adjacency matrix of a linkage graph.

Constraint graphs and geometric constraint solvers evolved from the variational geometry techniques of Light and Gossard (1982) [6]. Bouma et al. (1995) [7] present the principles of geometric constraint solvers based on constraint graphs, see also Hoffman et al. (2004) [8]. Constraint graphs map geometric objects to vertices and geometric relationships to edges and provide a versatile tool for characterizing mechanical assemblies. The adjacency sub-matrix derived from the constraint graph reflect the relationships between both links and joints and can be used to describe the changes that occur in metamorphic mechanisms.

Recent research has expanded the use of linkage graphs to the design of compliant mechanisms [9], parallel mechanisms [10], and linkage synthesis [11,12,13]. Our work contributes to this growing body of research in the systematic design of linkage systems [14,15].

In what follows, we consider only the special cases of points, lines and planes as the geometric elements and the relationships of incidence, distance and angle that we need for the metamorphic linkages that we consider. However, the methodology seems general enough to be used for more complex linkage types.

3. Constraint graph of a mechanism

Computer-aided mechanism design seeks to match linkage topologies to design tasks, then formulate and solve the synthesis equations. Graph theory has been a successful tool to search and evaluate linkage topologies, see Vucina and Freudenstein 1991 [4]. However, the structural features that allow a metamorphic mechanism to fold and lock, thus changing its topology, are not captured in this linkage graph and so do not fit into this design process.

In this section, we introduce the constraint graph of a mechanism. We consider the mechanism to be represented in schematic form, which means that revolute pivots are points and links are formed from lines connecting these points. The constraint graph includes vertices that represent the lines that form the links and vertices that form the geometric elements in the mechanism schematic and each edge of the graph is a relationship between two geometric elements. While the geometric elements and relationships can be general, we will focus primarily on geometric elements that are points, lines and planes, and the relationships of incidence, such as a point is incident with a line, and distances between two points and angles between two lines or planes.

Consider the binary link shown in Fig. 1. For our constraint graph let the joints O and A be defined by points P_O and P_A , and let L_1 be the centerline of link connecting these two points. The two points and the line define three vertices in a constraint graph. The incidence of the line L_1 on the points P_O and P_A is represented by edges joining the vertex L_1 to the vertices P_O and P_A . The constraint that the points P_O and P_A lie at a fixed distance a along L_1 is represented by an edge that joins these two vertices.

It is useful to notice that given coordinates for any two of the three vertices P_O , L_1 and P_A , the constraints allow computation of the coordinates of the third vertex. When this occurs, the set of vertices is said to define a *cluster*, Bouma et al. (1995) [7].

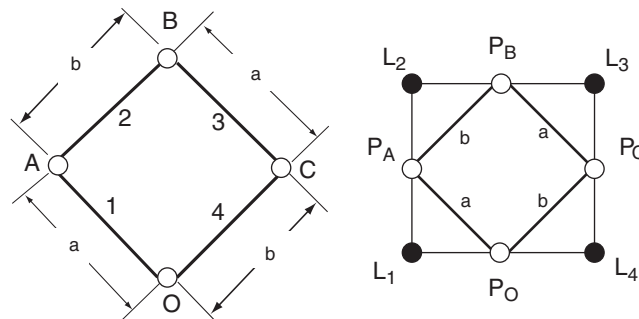


Fig. 2. A parallelogram linkage and its constraint graph.

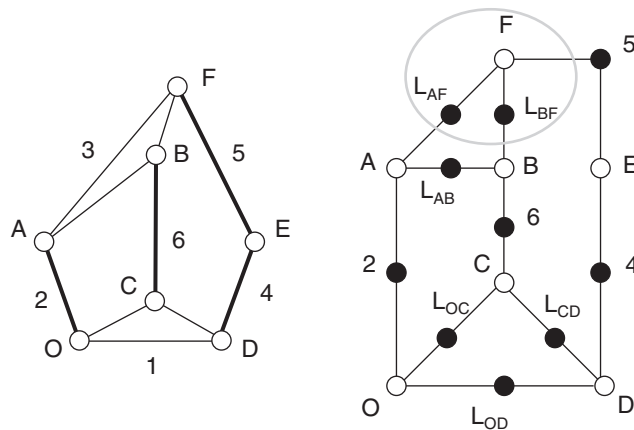


Fig. 3. The schematic of a six-bar linkage and its constraint graph.

The constraint graph for a mechanism is obtained by continuing this process for every link in its schematic. Fig. 2 shows the constraint graph for a four-bar parallelogram linkage. Notice that the vertices of the constraint graph form four clusters connected by the points P_O , P_A , P_B and P_C at the centers of the four joints.

Fig. 3 shows the constraint graph for the schematic of a six-bar linkage. Fig. 4 presents another version of the six-bar constraint graph in which the clusters have been shaded, the distance constraints presented as dark lines, and the ternary links 1 and 3 organized as hexagonal clusters. Notice that in this form the graph has a similar form as the linkage schematic. Once the clusters have been identified, they can be identified as “supervertices” that are incident with geometric elements shared with other clusters. This yields the reduced constraint graph shown in Fig. 4. The reduced constraint graph is a bipartite graph that has each cluster separated by the geometric object that it shares. If we focus only on the clusters and consider their intersection graph, illustrated with the dashed lines, then we obtain Freudenstein's topological graph of a linkage.

Graph theoretic algorithms have been developed to determine decompose constraint graphs to identify clusters, so that the system can be solved to determine the position of each geometric element. From the point of view of geometric constraint solvers, our constraint graphs are underconstrained with a deficit of 1. This means one more constraint is needed so the graph can be solved. This extra constraint is the input crank angle.

This section shows that the topological graph of a linkage, and its constraint graph provides geometric and dimensional content. In what follows, we use the constraint graph to introduce geometric information to the topological graphs of metamorphic linkages.

4. Graph representations of metamorphic mechanisms

An example of a metamorphic linkage is the folding fan shown in Fig. 5 which consists of a sequence of three dimensional folding elements that fold and lock to form a single member. For our purposes, we model the fold as a planar pair of hinged bars. In this case, the four-bar linkage formed by one folding element must have the dimensions of a kite linkage, that its two pairs of adjacent sides have the same lengths, see Fig. 6. Furthermore, in order to connect these elements together the joints must connect more than two links.

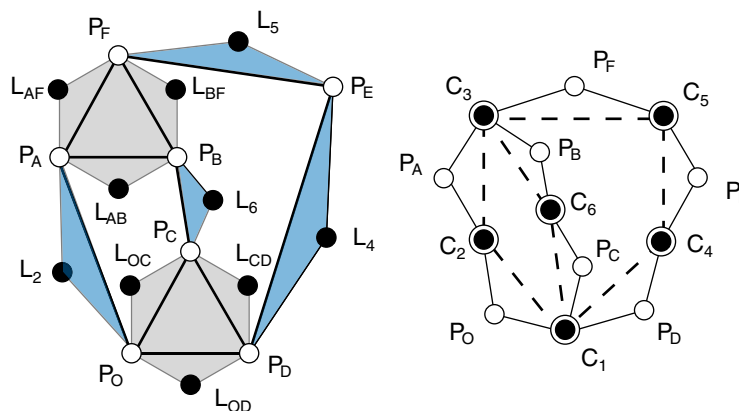


Fig. 4. The clusters of the constraint graph become supervertices in the reduced constraint graph. The intersection graph of the clusters illustrated by dashed lines is topological graph of the mechanism.

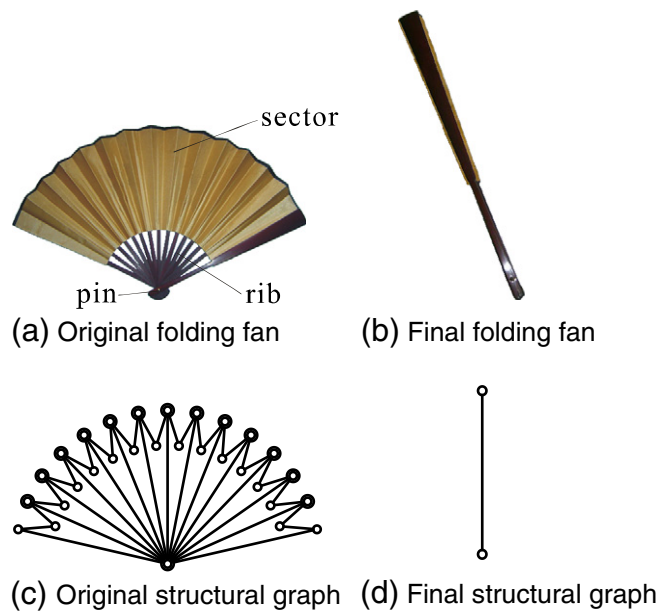


Fig. 5. Folding fan and the corresponding metamorphic mechanism with multiple joints.

4.1. Kite linkage

The constraint graph of the kite linkage in Fig. 6 models the revolute joints as points and the links as lines connecting two joints. Thus, the graph has eight vertices, four points and four lines. The incidence of a point on a line or line on a point defines edges that link these vertices, see Fig. 6.

The topology of the kite mechanism is captured by the vertices and edges of the graph which represent the incidence of the points and lines that form joints and links. Additional information is included as constraints between the geometric objects that form these vertices. In the case of the kite linkage, the conditions $a = |OA| = |OC|$, $b = |AB| = |BC|$, $\alpha_2 = \alpha_4$ and $\alpha_3 = 2\alpha_2 + \alpha_1$ are shown in the graph as distance and angle constraints between the various points and the various edges. For our purposes, we denote a distance constraint by a heavy line and the angle constraint by a pair of thin lines.

4.2. Multi-joint assembly

An important feature of a metamorphic linkage that allows it to change topology is the ability for joints to coincide such that links combine and operate as a single link. The constraint graph provides a way to characterize this change in topology.

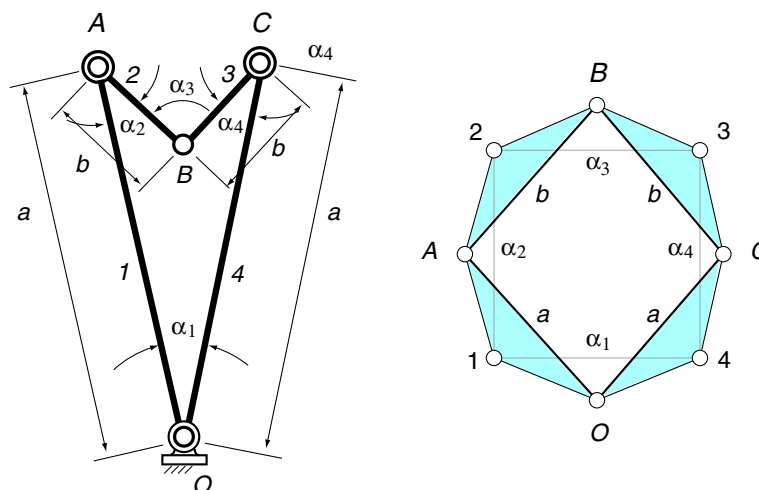


Fig. 6. The four-bar folding element of the folding fan and its constraint graph.

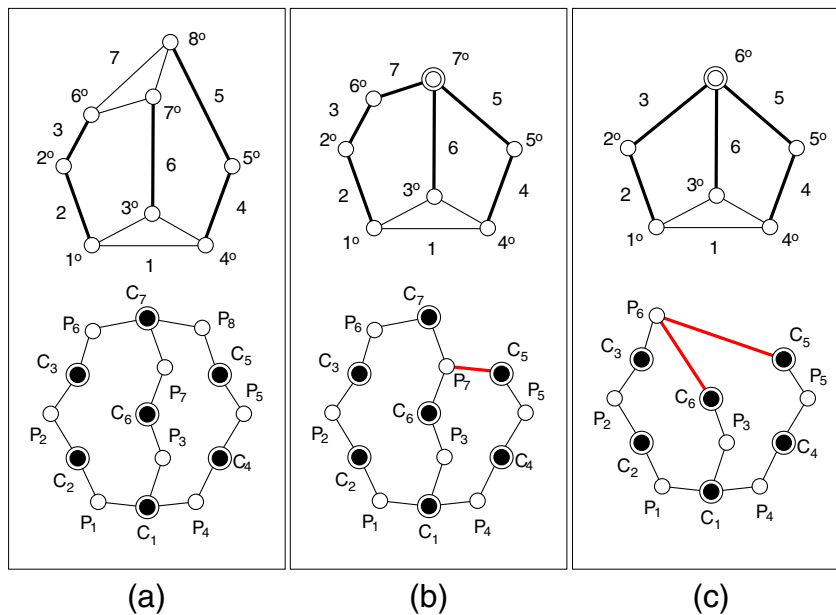


Fig. 7. (a) A general seven-bar linkage schematic and its constraint graph, (b) a seven-bar linkage with a multi-joint assembly obtained by combining joints 7° and 8° and its constraint graph, (c) the six-bar linkage obtained by combining joints 6°, 7° and 8° and its constraint graph.

The structural schematic of a mechanism generally consists of joints that connect only two links, which we term a *binary joint* and Ding et al. (2010) [12] call a “simple joint.” If the joint connects more than two links, we call it a *multi-joint assembly* and Ding et al. (2010) [12] call this a “multiple joint.” A multi-joint assembly that connects k links is counted as $k - 1$ binary joints when determining the mobility of the system.

Fig. 7 shows a sequence in which the joints of a seven-bar linkage are sequentially combined into multi-joint assemblies. In the first case, joints 7° and 8° on the ternary 7 are combined. Notice that this changes the constraint graph. In the second case, the three joints 6°, 7° and 8° are combined which eliminates the ternary link 7, and we see the associated reduction of the topological graph. This shows the ability of the constraint graph to couple dimensional information to the topological graph of the linkage, in this case dimensions that become zero.

4.3. Folding box

The constraint graph formalism can be applied to mechanisms such as the folding box shown in Fig. 8, where we use the faces of the box and their lines of intersection as the geometric elements. Li et al. (2002) [16] constructed an equivalent linkage for this box

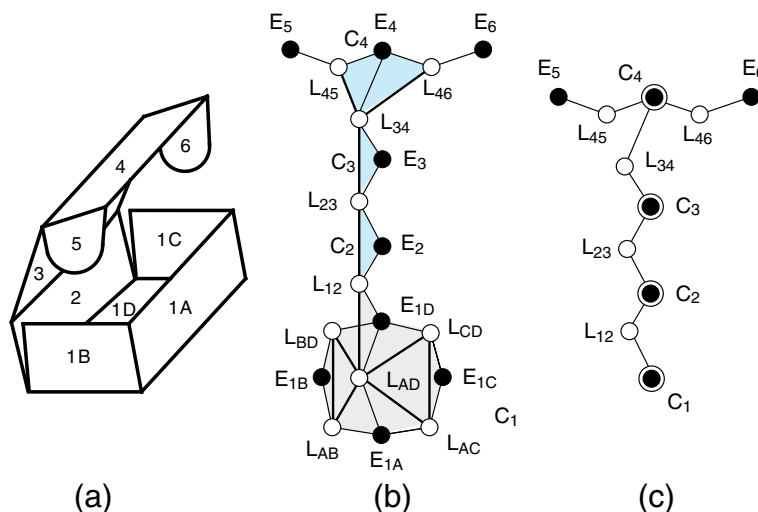


Fig. 8. (a) A folding box defined by planes as links and lines as joints, (b) its constraint graph, and (c) the constraint graph with the cluster as a supervertex.

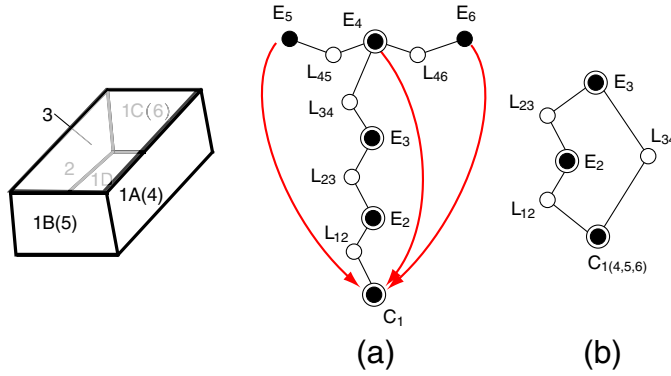


Fig. 9. When the folding box is closed, (a) planes E_4 , E_5 and E_6 become part of the cluster C_1 , and (b) its constraint graph becomes that of a rigid assembly.

and used the topological graph of this mechanism to define the graph representation. The four planes E_{1A} , E_{1B} , E_{1C} , and E_{1D} form the cluster C_1 . When the box closes the planes E_4 , E_5 and E_6 are identified with the planes E_{1A} , E_{1B} , E_{1C} respectively, and therefore become part of the cluster C_1 . See Fig. 9. The result is that this operation transforms an open chain (a) into the rigid assembly (b).

5. Incidence and adjacency matrices

Before considering metamorphic linkages, we examine the relationship between topological graph $G_{(n,m)}$ of a general (n,m) linkage, and its constraint graph $C_{(n,m)}$. In the general case, we see that the constraint graph can be viewed as a topological graph in which the joint edges are transformed into vertices, and in reverse the topological graph is the subgraph of the constraint graph obtained by transforming the joint vertices into edges. This relationship identifies the constraint graph of the general (n,m) linkage as a bipartite graph, for which every edge connects a cluster vertex to a joint vertex, see Bondy and Murty (1976) [17].

To be more precise, we define a graph $G(V,E)$ by a set of vertices V and a set of edges E , together with an incidence function $\phi: E \rightarrow V$ that defines the vertices incident to an edge, see Bondy and Murty (1976) [17]. The incidence function is defined by a $v \times e$ incidence matrix $D(G) = [d_{i,j}]$, where v is the number of vertices and e is the number of edges. For our purposes, the entry $d_{i,j}$ is 1 if vertex i is incident with edge j and 0, otherwise.

As an example, consider the constraint graph of the seven-bar linkage shown in Fig. 7. The subgraph formed by the clusters C_i is the topological graph $G_{(7,8)}$ of the $(7,8)$ linkage [2]. The incidence matrix $D(G_{(7,8)})$ is given by

$$D(G_{(7,8)}) = \begin{matrix} \begin{matrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ C_7 \end{matrix} \end{matrix} \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 & P_8 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}, \quad (1)$$

where P_j denotes the edge defined by the associated joint.

Another matrix associated with a graph is its adjacency matrix $A(G) = [a_{i,j}]$, which is a $v \times v$ matrix with $a_{i,j} = 1$ if vertex i and vertex j are on the same edge and 0 otherwise. Because the constraint graph $C_{(n,m)}$ of the general (n,m) linkage is bipartite, we find that its adjacency matrix is the $(n+m) \times (n+m)$ matrix $A(C_{(n,m)})$,

$$A(C_{(n,m)}) = \begin{bmatrix} 0_{n \times n} & D(G_{(n,m)}) \\ D(G_{(n,m)})^T & 0_{m \times m} \end{bmatrix} \quad (2)$$

where $D(G_{(n,m)})$ is the incidence matrix of the topological graph of the general (n,m) linkage. Thus, the adjacency matrix of the constraint graph for the general $(7,8)$ seven-bar linkage shown in Fig. 7 is given by

$$A(C_{(7,8)}) = \begin{bmatrix} 0_{7 \times 7} & D(G_{(7,8)}) \\ D(G_{(7,8)})^T & 0_{8 \times 8} \end{bmatrix}, \quad (3)$$

where $D(G_{(7,8)})$ is given by Eq. (1) which is also a part of the adjacency matrix of the constraint graph. In the following we called it as $AS(C)$ as the adjacency sub-matrix of the constraint graph.

In what follows, we introduce operations on the adjacency matrices of the constraint graph that represent the changes in type associated with metamorphic linkages.

6. Matrix operations for metamorphic mechanisms

In this section, we present operators that transform the adjacency matrices of the constraint graph for a general (n, m) linkage, into adjacency matrices of constraint graphs of linkages in which joints and/or links have combined to change its type.

We now introduce the combination and elimination matrices $[U_{i,j}]$ and $[E_j]$ which are similar to those described by Dai and Rees Jones (2005) [5]. Because our constraint graphs are bipartite, we can specialize operations on the incidence matrix $D(G)$ to those on the adjacency sub-matrix $AS(C)$.

Let the column combination matrix $[UC_{i,j}]$ be the $m \times m$ matrix that operates on the right of an $n \times m$ matrix $AS(C)$ such that it replaces column i by the sum of columns i and j , that is

$$[UC_{i,j}] = [I_m] + [K_{j,i}], \quad (4)$$

where $[I_m]$ the $m \times m$ identity matrix and the matrix $[K_{j,i}]$ has 1 in the (j, i) entry and zeros elsewhere—assume $[K_{j,i}]$ has the dimensions necessary to complete the required matrix operations. Similarly, let the row combination matrix $[UR_{i,j}]$ be the $n \times n$ matrix that operates on the left of an $n \times m$ matrix $AS(C)$ such that it replaces row i by the sum of row i and j , that is

$$[UR_{i,j}] = [I_n] + [K_{j,i}], \quad (5)$$

Let the elimination matrix $[EC_j]$ be the $m \times m - 1$ matrix that operates on the right to remove column j from an $n \times m$ matrix $AS(C)$, that is

$$EC_j = \begin{bmatrix} I_{m-j, m-j} & 0_{m-j, j-1} \\ 0_{1, m-j} & 0_{1, j-1} \\ 0_{j-1, m-j} & I_{j-1, j-1} \end{bmatrix}, \quad (6)$$

where $0_{i,j}$ denotes a matrix of zeros.

The elimination matrix $[ER_j]$ will be the $n - 1 \times n$ matrix that operates on the left to remove row j from an $n \times m$ matrix $AS(C)$.

As an example of these combination and elimination matrix operations consider the constraint graph $C_{(7,6,1)}$ of the seven-bar linkage in Fig. 7(b), which has 7 clusters, 6 binary joints and 1 ternary joint. Notice that $C_{(7,6,1)}$ is a bipartite graph, so we have

$$A(C_{(7,6,1)}) = \begin{bmatrix} 0_{7 \times 7} & D(G_{(7,6,1)})(7) \\ D(G_{(7,6,1)})^T & 0_{7 \times 7} \end{bmatrix}, \quad (7)$$

where

$$D(G_{(7,6,1)}) = AS(C_{(7,6,1)}) = \begin{matrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ C_7 \end{matrix} \begin{matrix} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 \\ \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{matrix}. \quad (8)$$

The transformation from $A(C_{(7,8)})$ to $A(C_{(7,6,1)})$ is achieved by the combination of joints P_7 and P_8 into a ternary multi-joint, and can be represented by computing $P_7 = P_7 + P_8$ such that $0 + 0 = 0$, $0 + 1 = 1$ and $1 + 1 = 1$, described by Li (2009) [18], followed by the elimination of the column associated with P_8 . This is achieved by the following operations on the adjacency sub-matrix $AS(C_{(7,8)})$,

$$AS(C_{(7,6,1)}) = AS(C_{(7,8)})[UC_{7,8}][EC_8]. \quad (9)$$

Now, consider the constraint graph $C_{(6,5,1)}$ in Fig. 7(c), which has 6 clusters, 5 binary joints, and 1 ternary joint. This constraint graph is again a bipartite graph that has the adjacency matrix

$$A(C_{(6,5,1)}) = \begin{bmatrix} 0_{6 \times 6} & D(G_{(6,5,1)}) \\ D(G_{(6,5,1)})^T & 0_{6 \times 6} \end{bmatrix}, \quad (10)$$

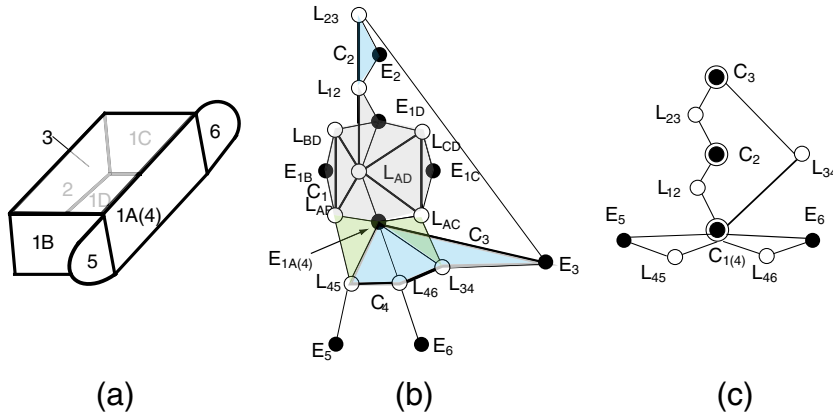


Fig. 10. Constraint graph after link 4 is combined with link 1.

where

$$D(G_{(6,5,1)}) = AS(C_{(6,5,1)}) = \begin{matrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \end{matrix} \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_5 & P_6 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}. \quad (11)$$

The transformation of the incidence matrix $AS(C_{(7,6,1)})$ to $AS(C_{(6,5,1)})$ is obtained by operations that remove link C_7 and joint P_7 .

$$AS(C_{(6,5,1)}) = [ER_7]AS(C_{(7,6,1)})[UC_{6,7}][EC_7]. \quad (12)$$

7. Example: transformation of the folding box

In this example, we show how the constraint graph and the adjacency matrix capture the transformation of the linkage type of a metamorphic mechanism. We choose the example presented in Li et al. (2002) [16], which is a box that is open in the initial configuration and closed in the final configuration.

A six-bar metamorphic mechanism corresponding to a cardboard box is shown in Fig. 8 together with its original constraint graph. The process of closing the cardboard box corresponds the following steps: (i) the open chain formed by link 1, 2, 3 and 4 is closed by combining Link 4 with Link 1, the corresponding constraint graph is shown in Fig. 10; (ii) Link 5 is combined with Link 1

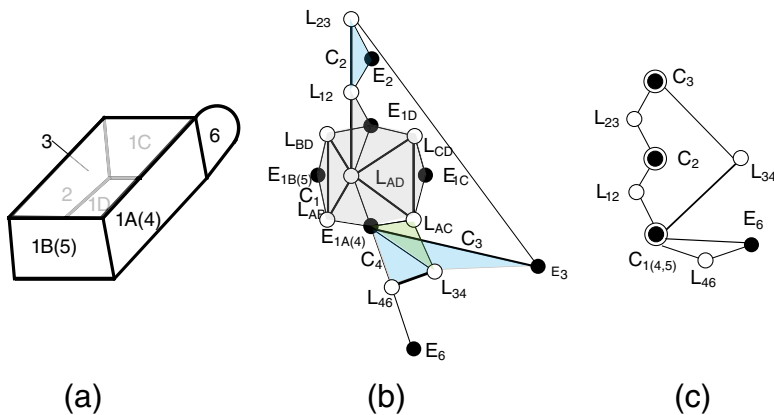


Fig. 11. Constraint graph after link 5 is combined with link 1(4).

(4), resulting in constraint graph shown in Fig. 11; and (iii) finally, Link 6 is combined with link 1(4,5) to obtain the constraint graph shown in Fig. 9.

The transformation of the constraint graphs associated with the metamorphic process of closing the box are calculated using adjacency matrices. As before we focus on the adjacency sub-matrix because our graphs are bi-partite.

7.1. Construct the initial adjacency sub-matrix

According to the constraint graph shown in Fig. 8, black vertices are regarded as the mark of rows and white vertices (kinematic pairs) of the constraint graph as the mark of columns. The corresponding adjacency sub-matrix $AS(C_0)$ is as follows.

$$AS(C_0) = \begin{matrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ E_5 \\ E_6 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} & L_{45} & L_{46} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}. \quad (13)$$

7.2. Row operations to combine Link 4 and Link 1

Add the elements in the fourth row corresponding to link 4 to the elements in the first row corresponding to link 1, then eliminate the elements in the fourth row. Here modulo-2 arithmetic of matrix operation are used, sometimes known as exclusive-or arithmetic, namely $0 + 0 = 0$, $1 + 0 = 1$, $0 + 1 = 1$, $1 + 1 = 0$. The result is the matrix $AS(C_1)$ as follows.

$$\begin{aligned} AS(C_1) &= \begin{matrix} C_{1(4)} \\ C_2 \\ C_3 \\ E_5 \\ E_6 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} & L_{45} & L_{46} \\ \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}, \\ &= [ER_4][UR_{1,4}][AS(C_0)]. \end{aligned} \quad (14)$$

Notice that no column was removed in this computation. Eq. (14) is the adjacency sub-matrix after link 4 is combined with link 1.

7.3. Row and column operations to combine Link 5 and Link 1

In Eq. (14) we eliminate the elements in the fourth row corresponding to Link 5 directly because connectivity relationship does not transform. The matrix is obtained as follows,

$$\begin{matrix} C_{1(4,5)} \\ C_2 \\ C_3 \\ E_6 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} & L_{45} & L_{46} \\ \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}, \quad (15)$$

In this calculation, we find that the fourth column only has a single entry of 1 which means joint L_{46} is not being used, therefore it should be removed. Thus, the adjacency sub-matrix $AS(C_2)$ is obtained as

$$\begin{aligned} AS(C_2) &= \begin{matrix} C_{1(4,5)} \\ C_2 \\ C_3 \\ E_6 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} & L_{45} \\ \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{matrix}, \\ &= [ER_5][AS(C_1)][EC_4] \end{aligned} \quad (16)$$

7.4. Row and column operations to combine Link 6 and Link 1

Similarly, in Eq. (16) we eliminate the elements in the fourth row corresponding to Link 5 directly because connectivity relationship does not transform. The matrix is obtained as follows,

$$\begin{matrix} C_{1(4,5,6)} \\ C_2 \\ C_3 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} & L_{46} \\ \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \end{matrix}. \quad (17)$$

Again because the fourth only has an element 1, it is removed. The final adjacency sub-matrix $AS(C_f)$ is obtained,

$$AS(C_f) = \begin{matrix} C_{1(4,5,6)} \\ C_2 \\ C_3 \end{matrix} \begin{matrix} L_{12} & L_{23} & L_{34} \\ \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \end{matrix}. \quad (18)$$

The final adjacency sub-matrix $AS(C_f)$ can be computed directly by combining these results,

$$AS(C_f) = [ER_6][ER_5][ER_4][UR_{1,4}][AS(C_0)][EC_4][EC_5]. \quad (19)$$

The adjacency matrix obtained in this calculation is identical with the results in Li et al. (2000) [15]. Thus, we find that the adjacency sub-matrix of a constraint graph and its transformation operator provide an effective representation of transformations that occur in metamorphic mechanisms.

8. Conclusions

This paper uses the constraint graph of a linkage schematic to characterize the topology and dimensional properties of a mechanism. We show that the constraint graph includes dimensional information that is critical to the function of a metamorphic mechanism, while maintaining the graph theoretic operations that are useful for design.

We find that the constraint graphs of linkages are bipartite, which means that each link vertex is between two joint vertices, and the graph adjacency matrix is characterized by its adjacency sub-matrix. Matrix operations on the adjacency sub-matrix are presented that define the structural changes associated with a metamorphic mechanisms. These operations combine joints into multi-joint assemblies in the same way that existing operations combine links. Examples illustrate the effectiveness of this approach in characterizing metamorphic mechanisms.

This constraint graph representation of metamorphic mechanisms provides a foundation for the analysis and synthesis of complex metamorphic mechanisms.

Acknowledgments

The authors gratefully acknowledge the support of the Beijing Natural Science Foundation (Project No. 3053017 and 3082014), the National Natural Science Foundation of China (Project No.50705010, 51075039) and “the Fundamental Research Funds for the Central Universities” (Project No. 2009CZ08).

References

- [1] J.S. Dai, J. Rees-Jones, Mobility in metamorphic mechanisms of foldable/erectable kinds, *ASME Journal of Mechanical Design* 121 (3) (1999) 375–382.
- [2] L.W. Tsai, *Mechanism design: enumeration of kinematic structures according to function*, CRC Press LLC, Florida, 2000.
- [3] F. Freudenstein, E.R. Maki, The Creation of Mechanisms According to Kinematic Structure and Function, *Journal of Environment and Planning B* 6 (1979) 375–391.
- [4] D. Vucina, F. Freudenstein, An application of graph theory and nonlinear programming to the kinematic synthesis of mechanisms, *Mechanism and Machine Theory* 26 (6) (1991) 553–563.
- [5] J.S. Dai, J. Rees-Jones, Matrix representation of topological changes in metamorphic mechanisms, *ASME Journal of Mechanical Design* 127 (4) (2005) 610–619.
- [6] R. Light, D. Gossard, Modification of geometric models through variational geometry, *Computer-aided Design* 14 (4) (1982) 209–214.
- [7] W. Bouma, I. Fudos, C.M. Hoffmann, J. Cai, R. Paige, A geometric constraint solver, *CAD* 27 (6) (1995) 487–501.
- [8] C.M. Hoffmann, M. Sitharam, B. Yuan, Making constraint solvers more usable: overconstraint problem, *CAD* 36 (4) (2004) 377–399.
- [9] M.A. Pucheta, A. Cardona, Design of bistable compliant mechanisms using precision position and rigid-body replacement methods, *Mechanism and Machine Theory* 45 (2) (2010) 304–326.
- [10] Y. Lu, L. Ding, J. Yu, Autoderivation of topological graphs for type synthesis of planar 3DOF parallel mechanisms, *Journal of Mechanisms and Robotics* 2 (1) (2010) 1–8.
- [11] C.C. Hung, H.S. Yan, G.R. Pennock, A procedure to count the number of planar mechanisms subject to design constraints from kinematic chains, *Mechanism and Machine Theory* 43 (6) (2008) 676–694.
- [12] H. Ding, J. Zhao, Z. Huang, Unified structural synthesis of planar simple and multiple joint kinematic chains, *Mechanism and Machine Theory* 45 (4) (2010) 555–568.

- [13] E.C. Kinzel, J.P. Schmiedeler, G.R. Pennock, Kinematic synthesis for finitely separated positions using geometric constraint programming, *ASME Journal of Mechanical Design* 128 (9) (2006) 1071–1079.
- [14] T. Yang, *Basic theory of mechanical system-structure, kinematics and dynamics*, Mechanical Industry Press, Beijing, 1996.
- [15] Z. Zhang, Q. Liao, D. Li, Synthesis of a metamorphic mechanism with random configuration, *Mechanical Science and Technology* 25 (11) (2006) 1276–1280.
- [16] D. Li, J.S. Dai, Q. Zhang, G. Jin, Structure synthesis of metamorphic mechanisms based on the configuration transformations, *Chinese Journal of Mechanical Engineering* 38 (7) (2002) 12–16.
- [17] J.A. Bondy, U.S.R. Murty, *Graph Theory with Applications*, North-Holland Press, New York, 1976.
- [18] D. Li, Z. Zhang, J.S. Dai, J.M. McCarthy, Configuration based improved synthesis of metamorphic mechanisms, *Proceedings of the ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2009)*, 33rd Mechanisms and Robotics Conference (MECH), 2009, pp. 1–10.