THE SYNTHESIS OF MECHANISM SYSTEMS USING A MECHANISM CONCEPT LIBRARY

Feng-Ming Ou[1], Hong-Sen Yan[2], Ming-Feng Tang[3]
[1] *Mechanical and System Research Laboratories, Industrial Technology Research Institute, Taiwan, R.O.C*
[2] *Department of Mechanical Engineering, National Cheng Kung University, Taiwan, R.O.C*
[3] *Department of Peripherals & Components, Liteon Technology Corporation, Taiwan, R.O.C.*
E-mail: ofm@itri.org.tw

Received October 2009, Accepted January 2010
No. 09-CSME-63, E.I.C. Accession 3149

ABSTRACT

This paper presents an approach for synthesizing all possible mechanism systems of kinematic building blocks in a mechanism concept library. The kinematic building blocks are defined as SISO primitive mechanisms, and their serial and/or parallel combinations are expressed as corresponding out-trees based on graph representation. By representing the constructive building blocks as labeled vertices and their possible combination relationships as directed edges, the synthesis approach is developed by adopting graph enumeration theorem. An illustrative example of four kinematic building blocks, including two crank-rocker linkages and two slider-crank mechanisms, is provided to validate the presented approach. The result shows that all feasible mechanism systems can be obtained effectively by following the synthesis method and which provides more alternatives in the library during design or re-design of mechanisms.

SYNTHÈSE DE SYSTÈMES MÉCANIQUES À L'AIDE D'UNE LIBRAIRIE DE CONCEPTION DE MÉCANISMES

RÉSUMÉ

On propose une façon de produire une synthèse de tous les systèmes mécaniques possibles d'éléments d'assemblage de construction cinématique dans une librairie de conception de mécanismes. Les éléments d'assemblage cinématiques sont définis comme mécanisme primitif *SISO*, et leurs combinaisons sérielles et/ou parallèles sont exprimées par la représentation graphique correspondante sous forme d'arbre. En représentant les éléments d'assemblage comme sommets étiquetés, et la combinaison de relations possibles comme graphe orienté, la démarche est développée en utilisant le théorème de l'énumération graphique. Pour valider notre approche, nous produisons un exemple démonstratif de quatre éléments d'assemblage cinématique, incluant deux maillons de manivelles oscillantes et deux mécanismes de manivelles glissantes. Les résultats démontrent que tous les systèmes mécaniques réalisables peuvent être obtenus en suivant la méthode de synthèse, et peut offrir plus d'options dans la librairie pendant la conception ou la modification des mécanismes.

1. INTRODUCTION

Mechanism design is concerned mainly with the generation or selection of an appropriate type of mechanism, the determination of the required numbers and types of members and joints, i.e., structural synthesis, and the derivation of geometric dimensions of members between joints to achieve the desired constrained motions, i.e., dimensional synthesis. Generally the structural synthesis phase is the most important stage in the design of mechanisms, and its goal is to find out possible topological structures of mechanisms. Since 1960s, some pioneer studies [1,2] applied graph theory to represent and synthesize kinematic structures of mechanisms, and followed by many publications regarding the conceptual design based on graph theory. For example, the design method derived from separating structure and function [3] and the method based on the ideas of generalization and specialization [4] were successfully applied to the design of various mechanisms, such as window regulating mechanisms [5], engine mechanisms [6], wheel damping mechanisms [7], breaker mechanisms [8], machining centers [9], and mechanical locks [10].

In view of numerous mechanisms that had been invented and designed over the past decades, based on the modular idea, some studies dealt with the conceptual design by making use of the past designs, i.e., combine the existing mechanisms or called kinematic building blocks to generate various alternatives. The merit of modular idea is easier in analysis and design due to the smaller sub-design tasks, and it leads to the reduction of cost, time, and effort. Such an idea could be found in some early designs, for example, Fig. 1 shows a slider-crank lever-cam mechanism, No. 1698 mechanism in reference [11]. It is composed of three constructive elements: two slider-crank mechanisms and a wedge cam-follower. This system is driven by a rotary power source to the input link 1 of the first slider-crank mechanism ABC through the wedge cam-follower and another slider-crank linkage DEF to generate the output reciprocation of slider 4.

In recent years, the studies that used existing mechanisms to map functions to structures in conceptual design were suggested by means of different representations and reasoning methods. With focusing on the function generation issue, Kota and Chiou [12,13] proposed a qualitative matrix representation scheme for conceptual design of mechanisms. By using motion transformation matrices to represent the qualitative functions of mechanisms, their approach was developed based on the decomposition of functional requirements in matrix forms and the combination of some identified kinematic building blocks (hereafter simply referred to as ‘building blocks’). In general, the building blocks were defined as single-input and single-output (SISO) mechanisms with fixed axes for both input and output motions, and a set of 43 building blocks are compiled from hundreds of ingenious existing mechanisms. In 2002, Moon and Kota [14] extended such a study to propose an automated synthesis approach by means of dual-vector algebra. In addition, Li et al. [15] presented a qualitative and heuristic approach by

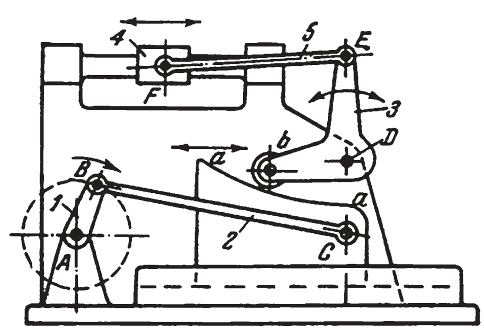


Fig. 1. A slider-crank lever-cam mechanism [11].

searching through and combining proper building blocks according to the design specifications. Murakami and Nakajima [16] further presented a computerized method of retrieving mechanism concepts from a library by specifying a required kinematic behavior and using the qualitative configuration space as a retrieval index.

In these studies, the functional considerations are taken as the starting point and the synthesis strategies are developed by decomposing the functional requirements and matching against the finite SISO building blocks of library to obtain their feasible combinations. However, from another point of view, it is also interesting and worthy to explore the combinatorial possibilities of the building blocks. Such as LEGO toys, the given constructive elements can be combined in many different ways for truly amazing results and which are usually of interest to the players. Therefore, once all possible combinations or called mechanism systems composed of given building blocks can be synthesized, it can activate the usage of mechanism concept library and provide more alternatives during conceptual design.

In 2005, Yan and Ou [17] firstly attempted to present a method to enumerate the combined configurations of a number of given building blocks; however the complicated process for identifying combinations makes the method less effective. Therefore, the purpose of this work is to figure out an easier approach for synthesizing mechanism systems of the building blocks in a mechanism concept library. By adopting graph theory, the approach is proposed based on simple matrix manipulations and the result shows that all possible mechanism systems can be synthesized effectively. In what follows, the combination types and representation scheme are introduced, then the graph-based synthesis approach is properly developed and an illustrative example is provided for verification.

2. TERMINOLOGY AND NOTATION

In this paper, we follow the terminology and notation of graphs in reference [18].

Directed graph (Digraph)

A directed graph G consists of a set of vertices $V = \{v_1, v_2, \cdots\}$, a set of edges $E = \{e_1, e_2, \cdots\}$, and a mapping Ψ that maps every edge onto some ordered pair of vertices (v_i, v_j). Fig. 2(a) shows a digraph with six vertices and ten edges.

Path and cycle

A walk is defined as a finite alternating sequence of vertices and edges. An open walk in which no vertex appears more than once is called a path; and a closed walk that starts and ends

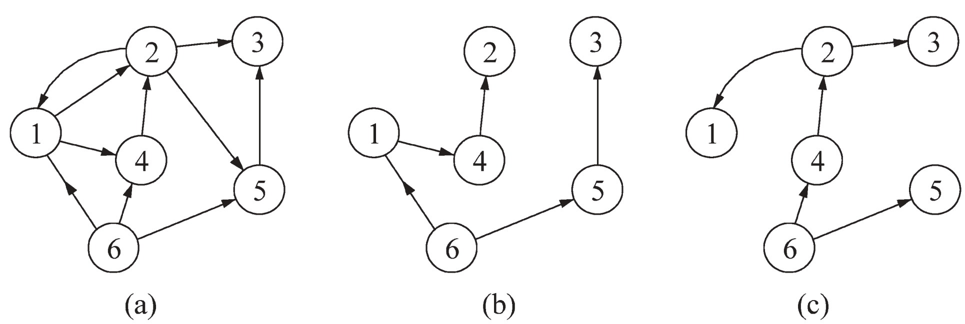


Fig. 2. Graph examples.

at the same vertex but otherwise has no repeated vertices or edges is called a cycle. In the example shown in Fig. 2(a), $v_1 \rightarrow v_4 \rightarrow v_2$ is a path and $v_1 \rightarrow v_4 \rightarrow v_2 \rightarrow v_1$ is a circle.

Connected graph

A graph G is said to be connected if there is at least one path between every pair of vertices in G. For example, Figs. 2(a)–(c) are connected graphs.

Indegree

In a digraph, the number of edges incident into a vertex v_i is called the indegree, $d^-(v_i)$ of v_i. In Fig. 2(a), for example, $d^-(v_1) = 2$.

Out-tree

An out-tree is a directed and connected graph without a cycle, for which the indegree of every vertex, except one (say vertex v_r), is unity; and the indegree of v_r (called the root of the out-tree) being zero. For example, the graph shown in Fig. 2(b) is an out-tree rooted at v_6.

Spanning out-tree

A tree T is said to be a spanning out-tree of a connected digraph G, if T is a subgraph of G and T contains all vertices of G. Figs. 2(b)–(c) are two spanning out-trees rooted at v_6 of the digraph shown in Fig. 2(a).

3. COMBINATION TYPES

In a mechanism concept library, the building blocks are basically defined as SISO mechanisms with fixed axes for both the input and output motions. In this study, 36 building blocks are extracted and numbered in Table 1. Note that the building blocks represent the

Table 1. List of building blocks in the mechanism library.

No.	Name	No.	Name
1	Crank-rocker linkage	19	Gear train
2	Double-crank linkage	20	Friction roller pair
3	Double-rocker linkage	21	Pawl-ratchet wheel
4	Triple-rocker linkage	22	Geneva wheel mechanism
5	Spherical four-bar linkage	23	Disc cam – translating follower
6	Slider-crank mechanism	24	Disc cam – oscillating follower
7	Double-slider mechanism	25	Wedge cam – translating follower
8	Oldham coupling	26	Wedge cam – oscillating follower
9	Scotch yoke mechanism	27	Cylindrical cam – translating follower
10	Geared five-bar linkage	28	Cylindrical cam – oscillating follower
11	Six-bar dwell linkage	29	Face cam
12	Spur-gear pair	30	Roller-gear cam
13	Helical gear pair	31	Indexing cam
14	Bevel-gear mechanism	32	Constant-breadth cam
15	Hypoid-gear mechanism	33	Intermittent motion mechanism
16	Worm and worm gear	34	Chain and sprocket
17	Noncircular gear pair	35	Pulley belt
18	Rack and pinion	36	Screw mechanism

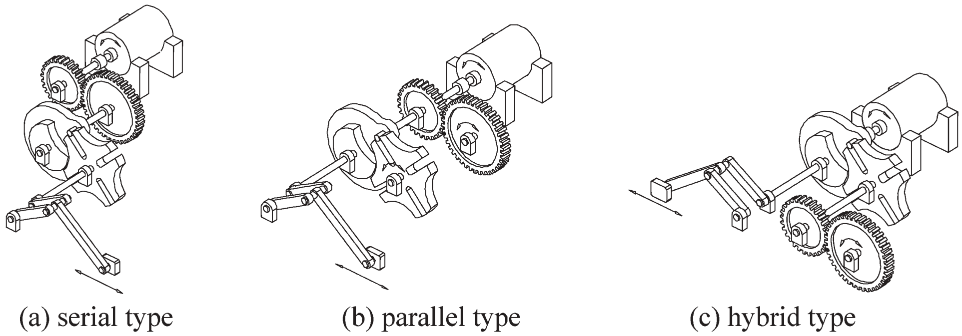


Fig. 3. Combination types.

design concepts rather than the particular structural forms. Considering that only the input or output link could be connected to one another when combining two building blocks, then the two links used for combination are merged as a common link. Thus the combination restriction is required to validate the combined mechanism system and it is presented as:

> *For a building block, the type of joint incident to the merged link and the frame must be the same as the one of another building block to be combined.*

According to the combination restriction, three types of combination for mechanism systems are possible: *serial*, *parallel*, and *hybrid* combinations. For a serial-combined mechanism system, the motion is transmitted from the power source to the ‘last’ building block in sequence to generate the desired output motion. When all the input motions of the constructive building blocks are provided by the same power source, it is called parallel combination. Certainly, for a hybrid combination, the mechanism system is constructed by combining building blocks both in serial and parallel ways. Fig. 3 illustrates the three types of mechanism systems by combining a a slider-crank mechanism (No. 6), a pair of spur gears (No. 12), and a Geneva mechanism (No. 22) from Table 1 in different ways.

For the convenience of describing the combination characteristics of building blocks, based on graph representation, labeled vertices are used to represent the power source and building blocks and directed edges stand for the combination relationships. Then, the mechanism systems can be displayed as labeled out-trees with the power source as the root. For example, the three mechanism systems in Figs. 3(a)–(c) can be represented as three corresponding out-trees shown in Figs. 4(a)–(c) respectively. The root ‘R’ refers to the power source and the labeled vertices ‘12’, ‘22’, and ‘6’ imply the pair of spur gears, the Geneva mechanism, and the slider-crank mechanism, respectively. For each vertex, the pair $\left(J_{input}, J_{output}\right)$ indicates the types of joints incident to the frame and the input/output links. Thus for each directed edges, it is necessary that J_{output} of the vertex which the edge is incident out of must be the same as J_{input} of the vertex that the edge is incident into, so as to form a

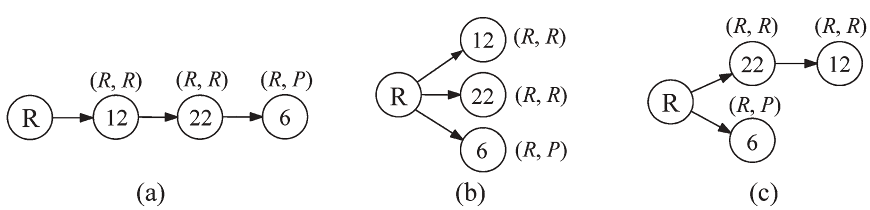


Fig. 4. Graph representations of the mechanism systems in Fig. 3.

feasible combination. Besides, it is noted that every vertex excluding root 'R' must have exactly one directed edge incident into, so that each building blocks can acquire feasible input motions. Therefore, mechanism systems of given building blocks can be expressed explicitly through the graph representation and it is helpful to develop the synthesis approach.

4. SYNTHESIS APPROACH

From the fact that different mechanism systems can be represented as corresponding labeled out-trees with the power source as the root, if the constructive elements and their feasible combination relationships are expressed as a connected digraph, the synthesis problem can be transformed into the issue of enumerating all spanning out-trees of the digraph. Based on the standpoint, the synthesis approach with five steps is presented.

Step 1. Select power source and building blocks from the library

The first step is from the mechanism concept library to select the building blocks and to decide the motion type of power source. Meanwhile, according to the operational manners of the chosen building blocks, the I/O joint types should be specified to ensure the feasible combination directions.

Step 2. Represent combination relationships as a connected digraph

By checking the I/O joint types and following combination restriction, this step is to draw a connected digraph, called *combination digraph*, which represents all possible paths of motion transmission among the building blocks and the power source.

For example, Fig. 5 shows the combination digraph formed with a rotary power source and No. 1, No. 6, No. 7, and No. 18 building blocks in Table 1. The numbered building blocks and the power source are represented as five labeled vertices '1', '6', '7', '18', and 'R', and seven directed edges $e_1 \sim e_7$ show the all feasible combination relationships between any two vertices.

Step 3. Enumerate spanning graphs of the digraph

With the aid of the graph enumeration algorithm [18], this step is to obtain all spanning graphs of the combination digraph. Based on the matrix representation of the combination digraph, the spanning graphs can be enumerated through a simple manipulation.

For a given combination digraph with n vertices and m directed edges, the *path-incidence matrix* $\boldsymbol{B} = [b_{ij}]_{n \times m}$ is used for representing the digraph and it is defined as:

$b_{ij} = 1$, if e_j is incident out of v_i;

$b_{ij} = -1$, if e_j is incident into v_i; and

$b_{ij} = 0$, if e_j is not incident to v_i;

where $i = 1, 2, \cdots n$ and $j = 1, 2, \cdots m$.

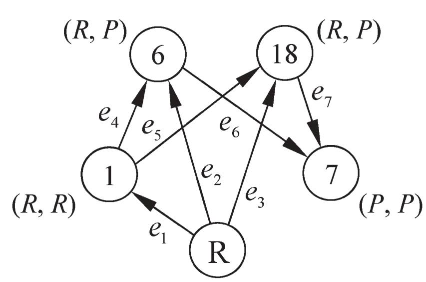


Fig. 5. Combination digraph D formed from four building blocks.

In this study, *non-positive reduced incidence matrix* $\tilde{\boldsymbol{B}}_{\mathrm{R}}$ is introduced as the matrix $\boldsymbol{B}$ with removing the row regarding vertex 'R' and changing all entries of 1 to zero. Thus the matrix $\tilde{\boldsymbol{B}}_{\mathrm{R}}$ reflects all non-root vertices that directed edges incident into, and the product matrix $\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^T$ is a diagonal matrix in which the value of each non-zero entry equals to the number of directed edges incident into the corresponding vertex. Accordingly, the spanning graphs can be generated by the following theorem:

> *For a given combination digraph D, let $\boldsymbol{M}(D)$ be the product matrix $\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^T$ with changing the value of each entry to the addition of the corresponding edges of digraph D. Then, all the spanning graphs can be enumerated by expanding the determinant of $\boldsymbol{M}(D)$.*

Taking the digraph D of Fig. 5 as an example, the matrix $\tilde{\boldsymbol{B}}_{\mathrm{R}}$ is derived in Eq. (1) and then the product matrices $\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^T$ and $\boldsymbol{M}$ can be obtained and shown in Eq. (2). Therefore, eight terms can be determined in Eq. (3) by expanding the determinant of $\boldsymbol{M}$. It is obviously that the eight expanded terms represent the spanning graphs of the combination digraph D in Fig. 5, since the four terms e_1, (e_2+e_4), (e_3+e_5), and (e_6+e_7) refer to all possible input motion sources for the building blocks '1', '6', '18', and '7' respectively. From the combination digraph, the eight corresponding graphs can be drawn and shown in Fig. 6.

$$
\tilde{\boldsymbol{B}}_{\mathrm{R}} = \begin{array}{c} \\ 1 \\ 6 \\ 18 \\ 7 \end{array}
\begin{array}{c} \begin{array}{ccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \end{array} \\
\begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix} \end{array} \tag{1}
$$

$$
\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \Rightarrow \boldsymbol{M} = \begin{bmatrix} e_1 & 0 & 0 & 0 \\ 0 & e_2+e_4 & 0 & 0 \\ 0 & 0 & e_3+e_5 & 0 \\ 0 & 0 & 0 & e_6+e_7 \end{bmatrix} \tag{2}
$$

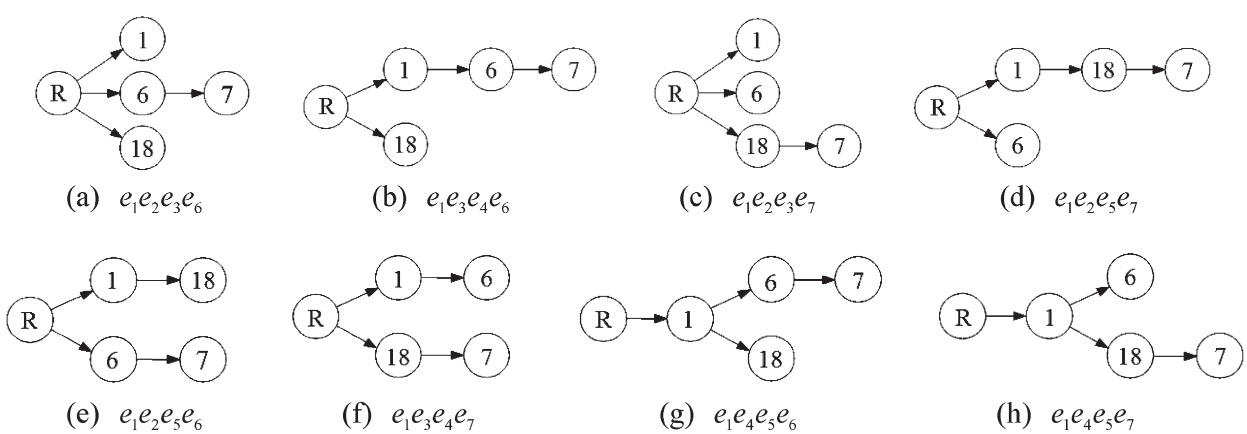


Fig. 6. Spanning graphs of the combination digraph in Fig. 5.

$$\det(\boldsymbol{M}) = \det\begin{vmatrix} e_1 & 0 & 0 & 0 \\ 0 & e_2+e_4 & 0 & 0 \\ 0 & 0 & e_3+e_5 & 0 \\ 0 & 0 & 0 & e_6+e_7 \end{vmatrix} = e_1(e_2+e_4)(e_3+e_5)(e_6+e_7) \tag{3}$$

$$= e_1e_2e_3e_6 + e_1e_3e_4e_6 + e_1e_2e_3e_7 + e_1e_2e_5e_7 + e_1e_2e_5e_6 + e_1e_3e_4e_7 + e_1e_4e_5e_6 + e_1e_4e_5e_7$$

Step 4. Filter out non-out-trees and identify equivalent out-trees

When the chosen building blocks are not all the different ones, this step is necessary to eliminate the unfeasible spanning graphs and identify the equivalent out-trees, to have the atlas of spanning out-trees.

For this case, two conditions are possible to form unfeasible results in the spanning graphs obtained in Step 3. One is the graphs without a power source and the other is the graphs with cycle(s). Therefore, the following rules can be concluded to filter out non-out-trees.

(1) *If an expanded term of det(**M**) does not comprise power-related edge(s), it is invalid due to the combination without a power source.*
(2) *If an expanded term of det(**M**) comprises the edges that form a cycle, it is invalid due to the unreasonable motion transmission.*

On the other hand, since some edges in the combination digraph imply the same combination relationships, the equivalent out-trees can be identified by means of the synonymous-edge set. The identification process is proposed as follows.

From the combination digraph to conclude the synonymous-edge set(s), then the equivalent results can be recognized through comparing the edges of each expanded term with one another based on the synonymous-edge set(s).

Step 5. Transform spanning out-trees to mechanism systems

The final step is to transform the labeled vertices to the corresponding power source and building blocks, and to combine them by following the feasible motion transmission paths in each out-tree, to have all feasible mechanism systems of given constructive elements.

Fig. 7 shows the concrete combination from the spanning out-tree in Fig. 6(a). Therefore, according to the atlas of spanning out-trees, all the mechanism systems composed of given building blocks can be synthesized to perform various functions.

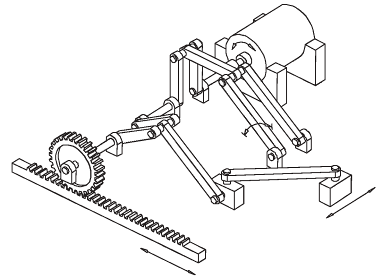


Fig. 7. Mechanism system of the out-tree in Fig. 6(a).

Table 2. Four building blocks and their I/O joint types.

No.	Name	I/O joint type
1_1	Crank-rocker linkage	(R, R)
1_2	Crank-rocker linkage	(R, R)
6_1	Slider-crank mechanism	(R, P)
6_2	Slider-crank mechanism	(R, P)

5. EXAMPLE

Here a general example is provided to demonstrate the proposed approach.

Step 1

From the library in Table 1, two crank-rocker linkages (No. 1) and two slider-crank mechanisms (No. 6) are chosen to illustrate the synthesis process and an electric motor is specified as the input power source of the mechanism systems. The building blocks are numbered and I/O joint types are decided in Table 2, where the number i_m implies that the building block i is selected m times.

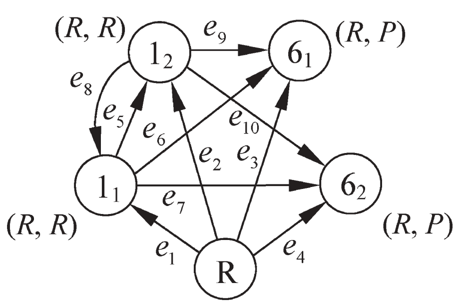
Step 2

According to the combination restriction and graph representation, the combination digraph D and its path-incidence matrix $\boldsymbol{B}$ can be obtained in Fig. 8.

Step 3

From the matrix $\boldsymbol{B}$ shown in Fig. 8, the non-positive reduced incidence matrix $\tilde{\boldsymbol{B}}_\mathrm{R}$ is obtained:

$$\tilde{\boldsymbol{B}}_\mathrm{R} = \begin{array}{c} \\ 1_1 \\ 1_2 \\ 6_1 \\ 6_2 \end{array} \begin{array}{c} \begin{array}{cccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 & e_{10} \end{array} \\ \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & -1 \end{bmatrix} \end{array} \tag{4}$$



$$\boldsymbol{B} = \begin{array}{c} \\ \mathrm{R} \\ 1_1 \\ 1_2 \\ 6_1 \\ 6_2 \end{array} \begin{array}{c} \begin{array}{cccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 & e_8 & e_9 & e_{10} \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & -1 \end{bmatrix} \end{array}$$

Fig. 8. Combination digraph D from four building blocks in Table 2.

Then the determinant of matrix $\boldsymbol{M}$, derived from $\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^{T}$, can be expanded:

$$\det\left(\tilde{\boldsymbol{B}}_{\mathrm{R}} \cdot \tilde{\boldsymbol{B}}_{\mathrm{R}}^{T}\right) = \det \begin{vmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{vmatrix} \Rightarrow \det(\boldsymbol{M}) = \det \begin{vmatrix} e_1+e_8 & 0 & 0 & 0 \\ 0 & e_2+e_5 & 0 & 0 \\ 0 & 0 & e_3+e_6+e_9 & 0 \\ 0 & 0 & 0 & e_4+e_7+e_{10} \end{vmatrix} \quad (5)$$
$$= (e_1+e_8)(e_2+e_5)(e_3+e_6+e_9)(e_4+e_7+e_{10})$$

In Eq. (5), there are $2 \times 2 \times 3 \times 3 = 36$ expanded terms and which represent the corresponding spanning graphs of digraph D in Fig. 8. However, since the vertices in Fig. 8 are not all different and some directed edges represent the same combination relationships, the unfeasible results elimination and equivalent out-trees identification are needed.

Step 4

According to the rules stated earlier, two necessary conditions of feasible expanded terms in this example are:

(1) The expanded terms must have at least one of the power-related edges e_1, e_2, e_3, and e_4.
(2) The expanded terms cannot comprise simultaneously the edges e_5 and e_8, because which form a cycle ‘1_1’→‘1_2’→‘1_1’.

Therefore, after checking all the expanded terms, nine unfeasible results are identified in Table 3, where the suffixes for building block number are ignored.

For identifying the equivalent terms, four synonymous-edge sets $\{e_1, e_2\}$, $\{e_3, e_4\}$, $\{e_5, e_8\}$, and $\{e_6, e_7, e_9, e_{10}\}$ are concluded from Fig. 8. By means of these four sets, the remaining 27 feasible expanded terms of Eq. (5) can be inspected and identified to have the 10 spanning out-trees listed in Table 4.

Step 5

Finally, the 10 spanning out-trees is transformed to the mechanism systems shown in Fig. 9. The result shows that one parallel combination and nine hybrid ones are synthesized in this example, and Table 5 lists all the possible output functions.

Table 3. Unfeasible expanded terms.

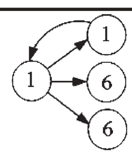
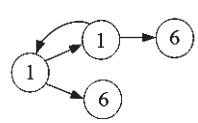
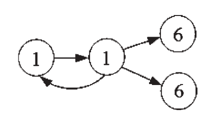
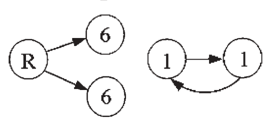
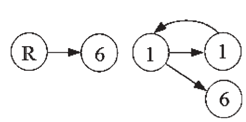
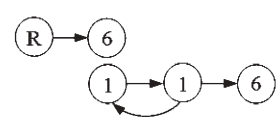
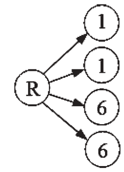
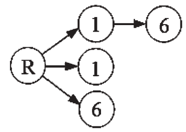
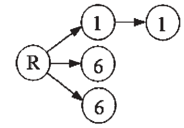
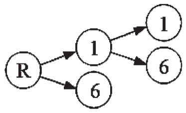
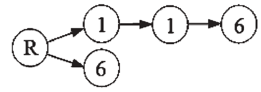
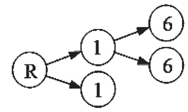
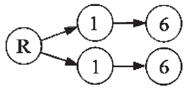
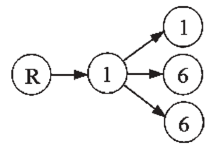
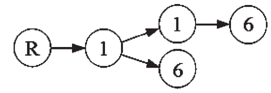
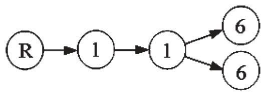
Term(s)	*Graph*	*Term(s)*	*Graph*	*Term(s)*	*Graph*
$e_5e_6e_7e_8$		$e_5e_6e_8e_{10}$ $e_5e_7e_8e_9$		$e_5e_8e_9e_{10}$	
$e_3e_4e_5e_8$		$e_3e_5e_7e_8$ $e_4e_5e_6e_8$		$e_3e_5e_8e_{10}$ $e_4e_5e_8e_9$	

Table 4. Atlas of spanning out-trees.

Term(s)	*Out-tree*	*Term(s)*	*Out-tree*	*Term(s)*	*Out-tree*
$e_1e_2e_3e_4$		$e_1e_2e_3e_7$ $e_1e_2e_3e_{10}$ $e_1e_2e_4e_6$ $e_1e_2e_4e_9$		$e_1e_3e_4e_5$ $e_2e_3e_4e_8$	
$e_1e_3e_5e_7$ $e_1e_4e_5e_6$ $e_2e_3e_8e_{10}$ $e_2e_4e_8e_9$		$e_1e_3e_5e_{10}$ $e_1e_4e_5e_9$ $e_2e_3e_7e_8$ $e_2e_4e_6e_8$		$e_1e_2e_6e_7$ $e_1e_2e_9e_{10}$	
$e_1e_2e_6e_{10}$ $e_1e_2e_7e_9$		$e_1e_5e_6e_7$ $e_2e_8e_9e_{10}$		$e_1e_5e_6e_{10}$ $e_1e_5e_7e_9$ $e_2e_6e_8e_{10}$ $e_2e_7e_8e_9$	
$e_1e_5e_9e_{10}$ $e_2e_6e_7e_8$					

6. CONCLUSION

In order to utilize the mechanism concept library more effectively, the purpose of this work is to synthesize all mechanism systems constructed with given building blocks. First, the serial, parallel, and hybrid combinations of mechanism systems are introduced and described as out-trees. Based on graph theory, a connected digraph, in which the constructive elements are

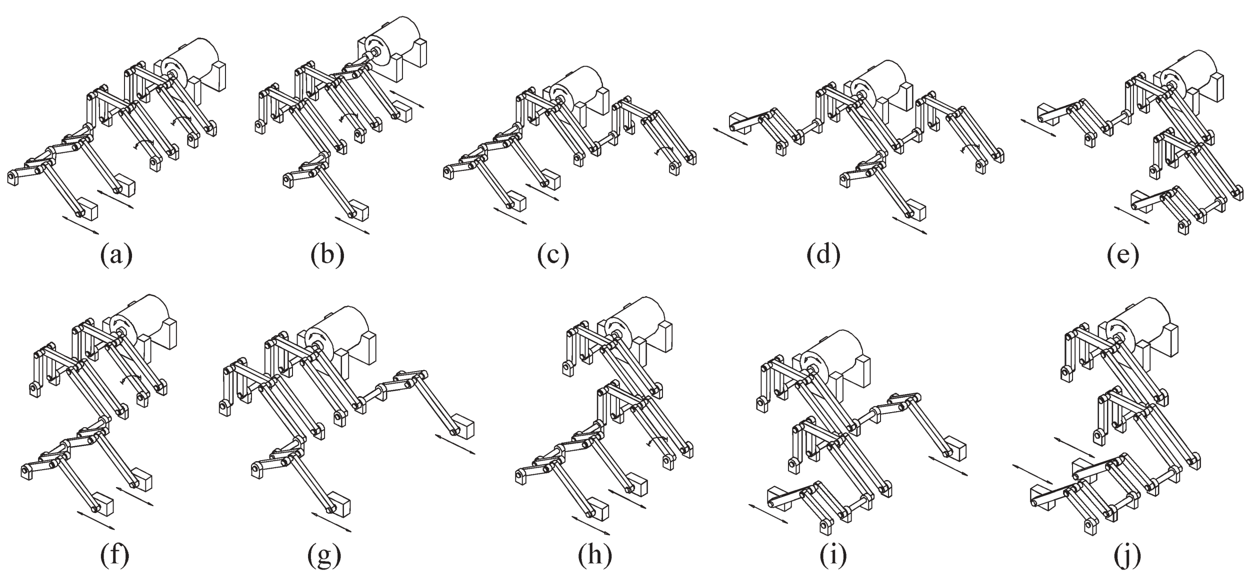


Fig. 9. All feasible mechanism systems synthesized from four building blocks.

Table 5. Output functions of mechanism systems in Fig. 9.

Type	Output functions	Figure(s)
Serial	-	-
Parallel	*Four outputs: 2 oscillations + 2 translations*	(a)
Hybrid	*Two outputs: 2 translations*	(e),(g),(i),(j)
	Three outputs: 1 oscillation + 2 translations	(b),(c),(d),(f),(h)

represented as labeled vertices and all possible combination relationships between any two vertices are expressed as directed edges, is presented. Then, the spanning graphs of the digraph are generated by adopting the algorithm of graph enumeration. Through the unfeasible graphs elimination and equivalent out-trees identification, the atlas of spanning out-trees can be obtained, and finally all feasible mechanism systems are synthesized by recovering the physical meanings of spanning out-trees. Since the synthesis approach is developed from the theoretical basis with matrix forms, it can be computerized and developed as a concept-creation system integrated with the library, so as to facilitate and activate the usage of the building blocks during conceptual design of mechanisms.

ACKNOWLEDGEMENTS

The authors are grateful to the National Science Council (Taiwan, R.O.C.) for supporting this research under grant NSC 93-2212-E006-128.

REFERENCES

1. Dobrjanskyj, L. and Freudenstein, F., "Some applications of graph theory to the structural analysis of mechanisms," *Journal of Engineering for Industry*, pp. 153–158, 1967.
2. Woo, L.S., "Type synthesis of plane linkages," *Journal of Engineering for Industry*, pp. 159–172, 1967.
3. Freudenstein, F. and Maki, E.R., "The creation of mechanisms according to kinematic structure and function," *Journal of Environment and Planning*, Vol. 6, pp. 375–391, 1979.
4. Yan, H.S., *Creative design of mechanical devices*, Springer-Verlag, 1998.
5. Erdman, A.G. and Bowen, J., "Type and dimensional synthesis of casement window mechanisms," *Journal of Mechanical Engineering*, Vol. 103, pp. 46–55, 1981.
6. Freudenstein, F. and Maki, E.R., "Kinematic structure of mechanisms for fixed and variable stroke axial-piston reciprocating machines," *Journal of Mechanisms, Transmission, and Automation in Design*, Vol. 106, pp. 355–364, 1984.
7. Yan, H.S. and Chen, J.J., "Creative design of a wheel damping mechanism," *Mechanism and Machine Theory*, Vol. 20, No. 6, pp. 597–600, 1985.
8. Jobes, C.C., Palmer, G.M., Means, K.H., "Synthesis of a controllable circuit breaker mechanism," *Journal of Mechanical Design*, Vol. 112, pp. 324–330, 1990.
9. Chen, F.C. and Yan, H.S., "A methodology for the configuration synthesis of machining centers with automatic tool changer," *Journal of Mechanical Design*, Vol. 121, pp. 359–367, 1999.
10. Yan, H.S. and Liu, N.T., "Configuration synthesis of one step push-button stopper lock with variable passwords," *Proceedings of 10th World Congress on the Theory of Machine and Mechanisms*, Oulu, Finland, pp 622–629, June 20–24, 1999.

11. Artobolevskii, I.I., *Mechanisms in modern engineering design*, MIR Publishers, 1986.
12. Kota, S., "A qualitative matrix representation scheme for conceptual design of mechanisms," *Proceedings of the 1990 ASME Mechanisms Conference*, Chicago, pp. 217–230, 1990.
13. Chiou, S.J. and Kota, S., "Automated conceptual design of mechanisms," *Mechanism and Machine Theory*, Vol. 34, pp. 467–495, 1999.
14. Moon, Y.M. and Kota, S., "Automated synthesis of mechanisms using dual-vector algebra," *Mechanism and Machine Theory*, Vol. 37, pp. 143–166, 2002.
15. Li, C.L., Tan, S.T., Chan, K.W., "A qualitative and heuristic approach to the conceptual design of mechanisms," *Engineering Applications of Artificial Intelligence*, Vol. 9, pp. 17–31, 1996.
16. Murakami, T. and Nakajima, N., "Mechanism concept retrieval using configuration space," *Research in Engineering Design*, Vol. 9, pp. 99–111, 1997.
17. Yan, H.S. and Ou, F.M., "An approach for the enumeration of combined configurations of kinematic building blocks," *Mechanism and Machine Theory*, Vol. 40, pp. 1240–1257, 2005.
18. Christofides, N., *Graph theory – an algorithm approach*, Academic Press, 1975.