

# Identification of Isomorphism in Kinematic Chains by Using the Reduced Graph Matrix

Mohamed Aly Abdel Kader \* and Abdeslam Aannaque

Mohammadia School of Engineering, Mohamed V University in Rabat, Rabat, Morocco;

Email: abdeslam.aannaque@emi.um5.ac.ma (A.A).

\*Correspondence: mohamedaly\_abdelkader@um5.ac.ma (M.A.A.K.)

**Abstract**—Kinematic chain synthesis normally begins with the generation of a comprehensive list of candidate solutions followed by a time-consuming procedure for isomorphism elimination. As a result, the search for isomorphisms in kinematic chains has long attracted the attention of many researchers. Several methods and algorithms have been proposed in the past. Nonetheless, the field still needs fast, efficient and reliable means to prevent duplications across kinematic chains (KC) (i.e., isomorphisms), particularly for configurations with a significant number of bars. Mechanical designers are resorting to kinematic chains and mechanisms with multiple bars to accomplish more complex operations and movements. This complicates the procedure of determining isomorphism. In this paper, we present a simple and efficient method for identifying isomorphisms in kinematic chains by employing a reduced graph matrix, which reduces the adjacency matrix into a compact matrix corresponding to linkages between non-binary bars while implicitly accounting for binary bars. The algorithm's efficiency and computing complexity are assessed for a number of published situations, including single-joint kinematic chains with 8, 10, 12 bars, and three-complex 13, 15, 28 bars, and lastly 42-bar kinematic chains. This comparison demonstrates the validity and effectiveness of the proposed method.

**Keywords**—adjacency matrix, isomorphism, reduced graph matrix, structure synthesis

## I. INTRODUCTION

Many mechanical engineers create complicated machinery and devices that perform difficult jobs. The number of bars required grows in proportion to the complexity of the solution. As a result, the likelihood of discovering isomorphic KCs increases. Two isomorphic KCs correspond to the same physical KC with different vertices numbering. As a result, an efficient, precise, and systematic approach of discovering isomorphic solutions is required early in the design process. The designers must construct as many independent KCs as possible for a certain number of bars and degree of freedom while avoiding isomorphic solutions. The designer will then have a solid foundation from which to investigate and discover the best solution for the problem at hand. To

overcome the challenge of isomorphism identification, several researchers have offered numerous approaches and concepts. When the KC grows too complicated, including a large number of bars, determining all viable, non-isomorphic solutions becomes difficult, if not impossible. The papers listed below address the problem of isomorphism identification.

Woo [1], a pioneer in the field of isomorphism identification, proposed a rule to assess isomorphism based on the number of vertices in two graphs of the same degree. He suggested a vertex-to-vertex incidence matrix-based approach for finding isomorphism in KCs. Sohn and Freudenstein [2] used the vertex-to-vertex incidence matrices' characteristic polynomials. Certain counterexamples were later identified, demonstrating that the developed approach was a required but not sufficient condition. Rao and Varada Raju [3] used the Hamming number, which is employed in digital communication theory. Nonetheless, their approach, too, failed to accurately differentiate between isomorphic KCs in some circumstances. Yadav *et al.* [4] suggested utilizing the idea of distance to determine the length between every two pairs of joints in the KC to discover isomorphisms. Meanwhile, Yadav *et al.* [5] improved on Yadav *et al.* [4]'s prior method for removing non-isomorphic planar KCs with simple joints. Hwang and Hwang [6] synthesized KCs using the contiguity matrix of contracted links as a representation matrix. Chu and Cao [7] introduced an identification approach based on the notion of the link adjacent chain table. This approach produced an invariant quantity, which made identifying isomorphisms in KCs easier. Using an artificial neural network technique, Kong [8] suggested a novel method for finding isomorphic KCs. Rao [9, 10] synthesized F-DOF (N-2)-linked chains as the foundation chains for synthesizing F-DOF and N-linked chains using the Hamming number approach. Chang *et al.* [11] proved that some KC configuration eigenvalues have features that enable the eigenvectors and eigenvalues to identify KC isomorphisms. Ding and Huang [12] developed a strong computer program that automatically drew the KCs of basic joints by employing the KC's characteristic representation code as well as the notions of synthetic degree sequence and characteristic adjacency

matrix. They also produced a database of kinematic chain atlases. For detecting unique solutions in kinematic chains, Lohumi *et al.* [13] suggested a hierarchical clustering approach. Rizvi *et al.* [14] proposed using the adjacency matrix to generate a unique chain identification number. Isomorphisms were identified using this identification number. Rajneesh and Sunil [15] proposed the notions of entropy and connection number to detect isomorphisms. Rai and Punjabi [16] used the binary codes generated by the link labeling approach to identify isomorphisms. While Deng *et al.* [17] used a molecular topological index to identify isomorphisms in KCs. Ding *et al.* [18] used the perimeter loops of kinematic chains to relabel the links and obtain a unique identification number. Using a similar strategy, Yang *et al.* [19] created planar basic joint KCs. Rai *et al.* [20] have recommended employing binary coding to relabel the connections in the KCs in order to produce a unique identification code. Tian [21] developed a numbering technique for finding isomorphism. Sun *et al.* [22] suggested a technique for determining isomorphism based on a Compound Topological Invariant (CTI) composed of the fourth power of the adjacency matrix and eigenvalues. Sun *et al.* [23] suggested a method for removing isomorphism in contract graphs after the vertex has been placed. Varadaraju *et al.* [24] presented a technique on the neighboring matrix that is comparable to the hamming number. The latter is divided into three parts. To determine the isomorphism, Vinjiamuri *et al.* [25] employed the distance between distinct vertices to compute the called net distance. Rajneesh *et al.* [26] proposed an entropy-based technique for finding isomorphisms in multiple joint kinematic chains while ignoring link tolerance and joints clearance. The perimeter loop approach was utilized by Ding *et al.* [27] to identify isomorphism between planetary gear trains. Rongjiang *et al.* [28] enhanced the similarity recognition technique and used it to create Planar KCs using P-pairs. Hollerbach [29] developed a successful lagrangian formulation of manipulator dynamics in which the number of additions and manipulations is proportional to the number of joints. Anderson *et al.* [30] presents for systems with  $n$  generalized coordinates and  $m$  independent algebraic constraints, the technique may accommodate the spatial motion of general multi-rigid-body systems with arbitrarily many closed loops in  $O(n+m)$  operations overall. Mohan *et al.* [31] propose a method to determine the independent equations of motion by using the Euler-Lagrange equations of motion based on the system's kinetic and potential energies and the decoupled natural orthogonal complement matrices. In comparison to current approaches, this study proposes changing the neighboring matrix to the reduced graph matrix to determine isomorphism across kinematic chains in a relatively short amount of time.

This approach works well for six- and eight-bar kinematic chains, as well as certain ten- and twelve-bar kinematic chains; nevertheless, the square of this matrix must be utilized to find the isomorphism of all kinematic chains.

This paper is organized as follows: Section II introduces the reduced graph matrix and how it is derived from the adjacency matrix and vice versa. Section III provides three examples that demonstrate the usefulness of the suggested method. Section IV compares the suggested technique to other isomorphism identification methods. Finally, section V provides a summary of this paper.

## II. GRAPH REPRESENTATION

### A. Graph

In this work, a graph is defined as a set of vertices connected by edges that form a kinematic chain. Vertices represent bars in the KC, while edges indicate linkages between bars, i.e., joints.

The KC can be represented by a square adjacency matrix, as shown below:

$$a_{ij} = \begin{cases} 1 & \text{if vertex } i \text{ is adjacent to vertex } j \\ 0 & \text{otherwise} \end{cases}$$

### B. Reduced Graph Matrix

As illustrated in the picture below, the  $n$ -by- $n$  adjacency matrix is reduced to the  $m$ -by- $m$  reduced graph matrix (RGM) by eliminating all rows and columns relating to the binary bars and modifying the values of  $a_{ij}$  corresponding to the  $m$  non-binary bars ( $i$  and  $j$  in the example below are non-binaries vertices). Decimals are utilized to make each circumstance distinct to minimize any misinterpretation (Fig. 1(c), Fig. 1(f) and Fig.2).

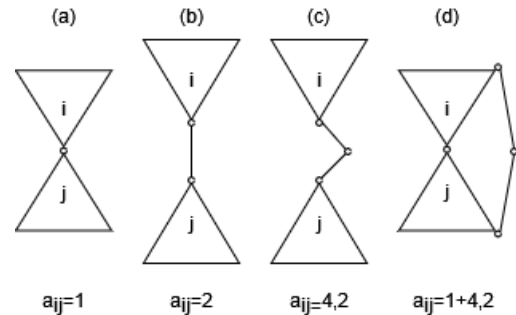


Figure 1. Elements of the adjacency matrix for different types of links between two non-binaries vertices  $i$  and  $j$ .

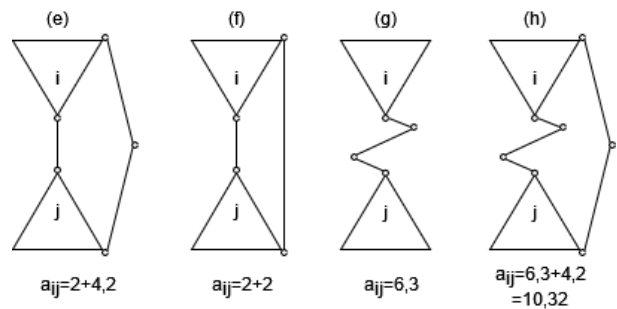


Figure 2. Values of  $a_{ij}$  for different situations.

The major purpose of this technique is to build a non-binary vertices connection matrix that holds information about the topological features of the original kinematic chain. The  $a_{ij}$  is made up of two parts: a value before the

comma that identifies the kind of connection between i and j (direct or via binary bars), and a value after the comma that denotes the number of binary bars in each series between i and j. For example,  $a_{ij} = 5.2$  denotes a direct connection  $a_{ij} = 1$  as well as a connection via two binary bars  $a_{ij} = 1 + 2 + 2$ . Following the comma, the value  $a_{ij} = 5.2$  shows that the two binary vertices are linked in series. The expression  $a_{ij} = 10.32$  denotes that the i-vertex and j-vertex are connected by five binary vertices  $a_{ij} = 2 + 2 + 2 + 2 + 2$  (non-direct connection)  $a_{ij} = 10$ , three of which are in series ( $a_{ij} = 10.3$ ) and one in parallel ( $a_{ij} = 10.32$ ).

The number of binary bars is specified by one or more "0" to make the procedure universal for any number of binary bars linking two non-binary bars. If the number is between 10 and 99, a leading "0" is used. A double "0" is used when the number is between 100 and 999, and so on.

For example,

(1)  $a_{ij} = k.22$  (k is either 8 or 9). The number 22 denotes that the i- and j-vertices are connected by four parallel bars (2+2).  $a_{ij} = k.022$  (k=44 or 45) would imply that the i-vertex and j-vertex are linked in series by twenty-two binary vertices.

(2) If three parallel series of ten, five, and two bars link two non-binary vertices, the element  $a_{ij}$  is given by  $a_{ij} = 34.01052$ . The number 34 is equivalent to  $2(10+5+2)$ . Take note of the leading "0," which plainly indicates the number 10.

(3) When two non-binary vertices are connected by two parallel series of ten and one hundred bars, the element  $a_{ij}$  is 220.00100010.  $a_{ij} = 220.01000100$  may also be used to describe the same scenario.

In summary, the presence of a leading zero in the fractional component of  $a_{ij}$  indicates that the following two consecutive numbers represent the number of binary bars linking vertices i and j. Two consecutive zeros denote that the next three digits represent the number of binary bars in series, and so on. The numbers of bars in a series following the decimal point are arranged in ascending or decreasing order. In this article, we choose ascending order (of the number of binary bars in each series) as indicated in the preceding example (2) ( $a_{ij} = 34.01052$ ).

C. Example

The reduced graph matrix is derived from the adjacency matrix of the following 8 bars KC.

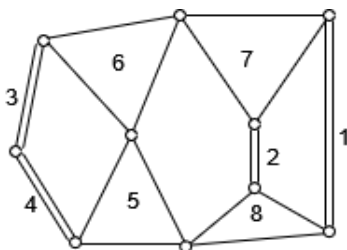


Figure 3. Structure representation of eight bar kinematic chain.

The adjacency matrix of the 8 bars kinematic chain in Fig. 3 is as follows:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (1)$$

The 7 and 8 bars in KC of Fig. 3 are connected by two binary bars 1 and 2, the components  $a_{78} = 2 + 2$ , and the reduced graph matrix is formed by removing the rows and columns corresponding to binary bars 1 and 2 from the adjacent matrix.

$$RGM = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 4 \\ 0 & 0 & 1 & 0 & 4 & 0 \end{bmatrix} \quad (2)$$

The corresponding kinematic chain is:

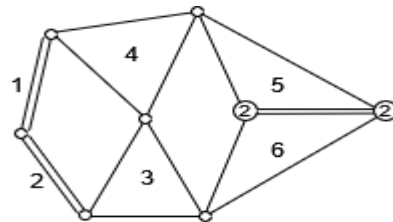


Figure 4. Kinematic chain after eliminating two binary bars.

Then, in the new RGM connection between the two non-binary bars 3 and 4, remove the rows and columns corresponding to the binary bars 1 and 2 (in Fig. 4).  $a_{34} = 4.2 + 1 = 5.2$ .

Because this KC includes four non-binary links, the reduced graph matrix is a 4 by 4 matrix, and it is given by:

$$RGM = \begin{bmatrix} 0 & 5.2 & 0 & 1 \\ 5.2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 4 \\ 1 & 0 & 4 & 0 \end{bmatrix} \quad (3)$$

And the corresponding kinematic chain becomes (see Fig. 5):

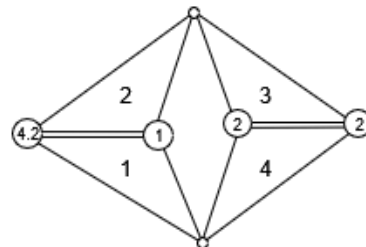


Figure 5. Kinematic chain after eliminating all binary bars.

III. PROPOSED METHOD FOR ISOMORPHISM IDENTIFICATION

Every configuration has an invariant that is independent of the numbering of the vertices. The new matrix derived from the adjacency matrix allows for the identification of

KC isomorphisms. This matrix is the Square of the Reduced Graph Matrix (SRGM), resulting in what will be referred to as the Matrix for Isomorphism Identification (MII) later.

A bijection connects the adjacency matrix to the RGM (for any adjacency matrix there one and only one RGM and vice versa). As a result, the RGM squared, called SRGM, is constant up to a permutation for any isomorphic graph (it contains the same elements but in a different order). Consider two isomorphic KCs with two adjacency matrices A and B. They are represented by RGMA and RGMB reduced matrices. Because the two KCs are isomorphic, there exists a permutation P1 such that  $B = P1 \times A \times P1^t$ . Given how RGM is constructed, RGMA and RGMB have a similar relationship ( $RGMB = P \times RGMA \times P^t$ ). P is obtained from P1 by eliminating from P1 the rows and columns corresponding to the binary vertices. As a result,  $SRGMB = RGMB^2 = P \times RGMA \times P^t \times P \times RGMA \times P^t = P \times RGMA \times RGMA \times P^t = P \times SRGMA \times P^t$ . SRGMA and SRGMB are equal up to the same permutation.

For the KC of Fig. 3, the square of the reduced graph matrix is given by:

$$SRGM = \begin{bmatrix} 28.04 & 0 & 9.2 & 0 \\ 0 & 28.04 & 0 & 9.2 \\ 9.2 & 0 & 17 & 0 \\ 0 & 9.2 & 0 & 17 \end{bmatrix} \quad (4)$$

The Matrix for Isomorphism Identification (MII) is obtained as follows:

- Sort the row elements of the squared reduced graph matrix in descending or ascending order.
- Sort the column elements of the resulting matrix in descending or ascending order (We will use the ascending order in the examples below).

$$MII = \begin{bmatrix} 0 & 0 & 9.2 & 17 \\ 0 & 0 & 9.2 & 17 \\ 0 & 0 & 9.2 & 28.04 \\ 0 & 0 & 9.2 & 28.04 \end{bmatrix} \quad (5)$$

It should be noted that the configurations with six and eight vertices do not require the square of the contract adjacent matrix. It is sufficient to range the rows and columns of the reduced graph matrix.

A. Conversion of an Adjacent Matrix to a Reduced Graph Matrix

Follow the steps in this example Fig. 6 to generate the reduced graph matrix from the adjacency matrix:

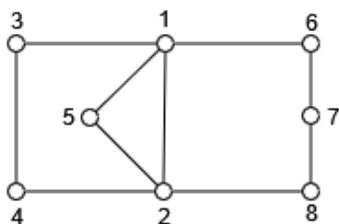


Figure 6. Eight bars KC.

- (1) In fact, node 3 in Fig. 7 is a virtual node (3') between 3 and 4. Between 1 and 3' we have a binary node (3) so  $Ab((1,3')) = 2.1$  and between 2 and 3' we have a binary node (4). So  $Ab(2,3') = 2.1$ . Similarly, node 6 is a virtual node between 6, 7 and 8. Between 1 and 6' we have two binary node 6 and 7 so  $AB(1,6') = 4.2$  and  $Ab(2,6') = 2.1$ . The matrix Ab defined after is the distance between the non-binary nodes (rows) and the nodes 3', 5 and 6'.

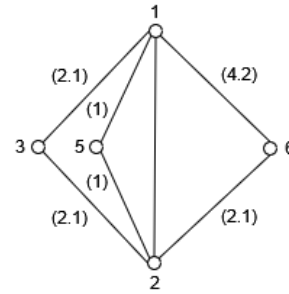


Figure 7. Eight bars KC after reducing the series of binary bars into one binary bar.

$$Ab = \begin{bmatrix} 2.1 & 1 & 4.2 \\ 2.1 & 1 & 2.1 \end{bmatrix} \quad (6)$$

- (2) Finally, add up all the values between the non-binary vertices Fig. 8 then Fig. 9, paying careful attention to the figures after the comma. Before adding up, a shift is necessary.

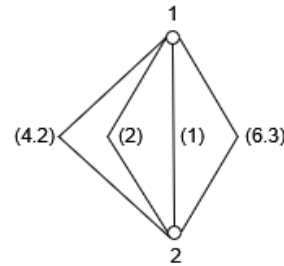


Figure 8. KC after transforming the series of binary bars to direct connections.

$$RGM = \begin{bmatrix} 0 & 13.32 \\ 13.32 & 0 \end{bmatrix} \quad (7)$$

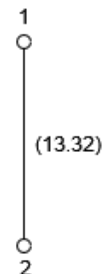


Figure 9. KC transformed to one directed connection.

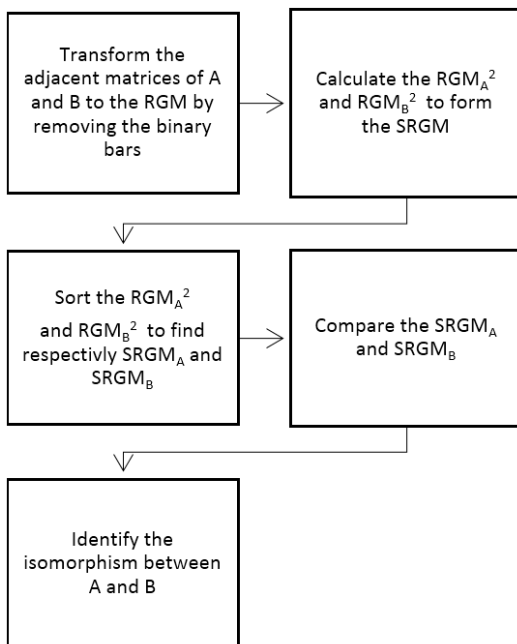
B. Reduced Graph Matrix to Adjacency Matrix Transformation

It is simple to convert the reduced graph matrix to an adjacency matrix, as demonstrated in the steps below:

- (1) Use the RGM to initialize the adjacency matrix A. Begin by looking at the decimal numbers in each element A(i,j), then go through the value following the comma one by one. If the value is 0, the following two values represent one whole number. If the first two consecutive values are 0, the next three values represent one whole number. And so forth.
- (2) The number obtained in step one is the number of binary bars in series between non-binary bars i and j.
- (3) Subtract twice the number found in step 1 from the value of the number preceding the comma. Subtract this number from the value following the comma as well. Create as many new rows and new columns as the value found in step 1. Assign a value of 1 to the elements of the adjacency matrix corresponding to the non-binary bars i and j and the newly created binary bars.
- (4) Repeat Steps 1 through 3 until all values after the comma have been processed.
- (5) While the remaining value before the comma is different from 1 and 0, subtract 2 from this value and add one row and column to the adjacency matrix. Assign a value of 1 to the elements of the adjacency matrix corresponding to the non-binary bars i and j and the newly created binary bars.

Repeat the preceding procedures until you have dealt with all of the decimals in all of the adjacency matrix entries.

The following flowchart summarizes the usage of the proposed method to identify isomorphism between two kinematic chains A and B.



#### IV. APPLICATION OF THE PROPOSED METHOD ON VARIOUS KINEMATIC CHAINS

The following are some examples of how the suggested approach can be used for various KCs with variable numbers of bars.

##### a) Example 1: Eight vertices configuration

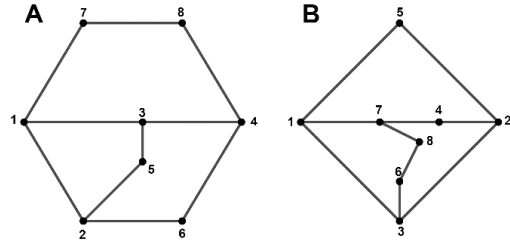


Figure 10. Two isomorphic configurations with eight vertices.

The RGM for the two KCs A and B are:

$$RGM_A = \begin{bmatrix} 0 & 1 & 1 & 4.2 \\ 1 & 0 & 2 & 2 \\ 1 & 2 & 0 & 1 \\ 4.2 & 2 & 1 & 0 \end{bmatrix} \quad (8)$$

$$RGM_B = \begin{bmatrix} 0 & 2 & 1 & 1 \\ 2 & 0 & 1 & 2 \\ 1 & 1 & 0 & 4.2 \\ 1 & 2 & 4.2 & 0 \end{bmatrix} \quad (9)$$

This, in turn, gives the MII matrices:

$$MII_A = \begin{bmatrix} 0 & 1 & 1 & 2 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 1 & 4.2 \\ 0 & 1 & 1 & 4.2 \end{bmatrix} \quad MII_B = \begin{bmatrix} 0 & 1 & 1 & 2 \\ 0 & 1 & 1 & 2 \\ 0 & 1 & 1 & 4.2 \\ 0 & 1 & 1 & 4.2 \end{bmatrix} \quad (10)$$

The MII matrices corresponding to KCs A and B Fig. 10 show that they are isomorphic. In fact, B is obtained from A by permuting vertices 1 and 3, 7 and 4, and subsequently 7 and 6.

##### b) Example 2: Fifteen vertices configuration

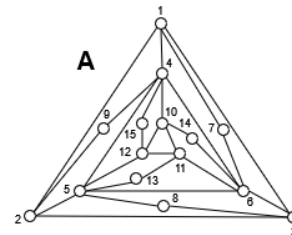


Figure 11. Graph representation of fifteen bars kinematic chains.

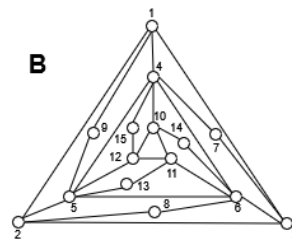


Figure 12. Fifteen KC non-isomorphic with KC of Fig. 10.

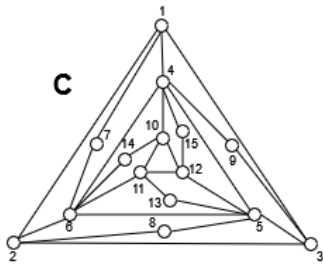


Figure 13. Fifteen KC isomorph with KC of Fig. 10.

The MII of the KCs of Fig. 11, Fig. 12 and Fig. 13 are given by:

$$MII_{A\&C} = \begin{bmatrix} 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 2 & 3 & 3 & 4 & 5 & 6 & 7 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \end{bmatrix} \quad (11)$$

$$MII_B = \begin{bmatrix} 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 1 & 2 & 3 & 3 & 4 & 4 & 4 & 6 & 7 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \\ 2 & 2 & 4 & 4 & 5 & 5 & 6 & 6 & 12 \end{bmatrix} \quad (12)$$

Because the MII for graphs A and C are equal, the two configurations are isomorphic. However, because the MII<sub>B</sub> differ, graph B is not isomorphic to graphs A and C.

c) Example 3: Twenty-eight vertices configuration

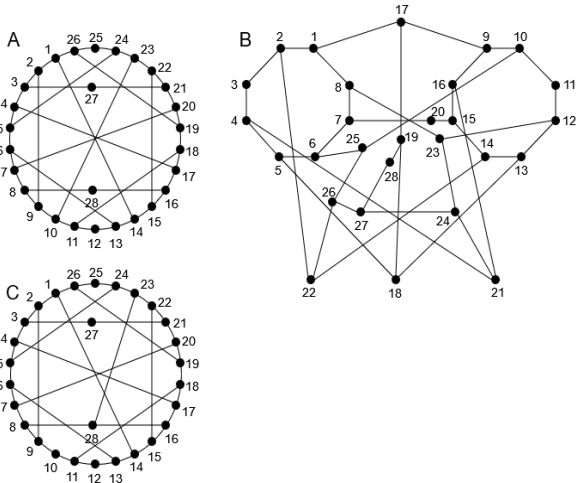


Figure 14. twenty-eight bars kinematic chains.

The identity matrices generated by the suggested approach demonstrate that the two graphs A and B in Fig. 14 are isomorphic, as confirmed by the literature [34]. The identity matrices of graphs C and A, however, differ. As a result, graphs A and C are not isomorphic.

$$MII_{A\&B} = \begin{bmatrix} 16 \times 000000000000000001111123 \\ 8 \times 000000000000000001111226 \end{bmatrix} \quad (13)$$

$$MII_C = \begin{bmatrix} 5 \times 000000000000000001111113 \\ 10 \times 000000000000000001111123 \\ 2 \times 000000000000000001111223 \\ 5 \times 000000000000000001111226 \\ 000000000000000001111246 \\ 000000000000000001122249 \end{bmatrix} \quad (14)$$

V. COMPARATIVE ANALYSES

A. Joint Sorting Code (JSC)

The joint sorting code technique [21] starts by listing all of the configuration's independent cycles that form the cycle basis. The basic cycles in Fig. 15 are (1,2,5,3,1), (1,3,4,8,7,1), and (2,6,4,3,5,2). Then, for every two adjacent vertices, joint identification codes (JIC) of the form ABC are generated, where A denotes the joint type (1 for revolute joints and 2 for prismatic joints) and B and C represent the degrees of those two vertices. The fundamental cycles are [133 123 123 133], [133 133 123 122 123], and [123 123 133 123]. By organizing each array in ascending order and appending zeros to it, the kinematic chain structural matrix (KCSM) is formed. The arrays are then combined together. The (KCSM) is configuration-specific and may be used to find isomorphisms in kinematic chains.

$$\begin{bmatrix} 123 & 123 & 133 & 133 & 0 \\ 122 & 123 & 123 & 133 & 133 \\ 123 & 123 & 123 & 123 & 133 \end{bmatrix} \quad (15)$$

This approach is simple and efficient, but it needs determining the graph cycle basis for the kinematic chain first. This is problematic for huge graphs. The technique suggested in this study, on the other hand, is simpler and does not need first identifying the cycle basis.

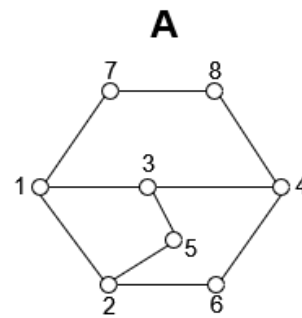


Figure 15. Eight -bars configuration.

B. Similarity Recognition and Isomorphism Identification

The method of similarity recognition is split into two phases [32]. To begin, compute the adjacency matrix's third power and sort each row in descending order (the resultant matrix is known as the power vertex similarity PVS matrix), then rearrange the matrix in descending order (CPVS). Second, we employ the PVS and CPVS to determine the similarity of vertices [32].

Application example:

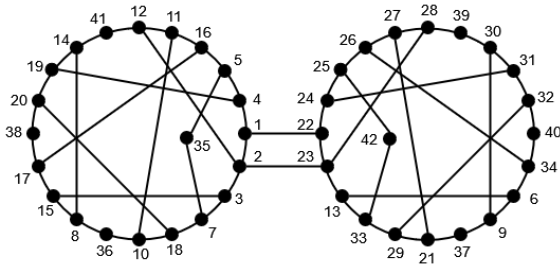


Figure 16. Representation of a forty-two kinematic chain.

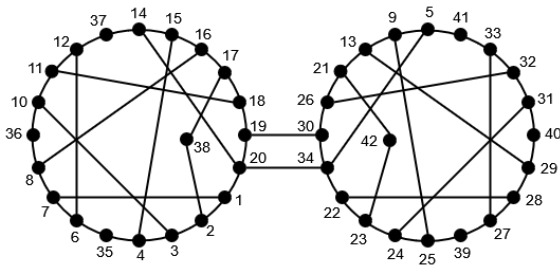


Figure 17. Second representation of a forty-two kinematic chain.

The cubic PVS can distinguish between the two kinematic chains. Similarly, the proposed method in this paper identifies the isomorphism in approximately  $4 \times 10^{-4}s$  as opposed to  $1.5 \times 10^{-3}s$  using the similarity recognition and isomorphism identification method [32]. The difference in runtime is proportional to the number of vertices. The difference in CPU time grows as the number of vertices grows.

**Note:** The integer in front of the parentheses represents a duplicate of the value in MII.

The proposed method

$$\begin{aligned}
 & 00000000000000000000000000000000000001 \\
 & 0000000000000000000000000000000000011111 \\
 & 3(111111111111111111111111111111111111111) \\
 MII_a = MII_b = & 1111111111111111111111111111111222222 \quad (16) \\
 & 1111122222222222222222222222222222222 \\
 & 2222222222222222222222222222222222222 \\
 & 333333333333333333333344666666666666666666
 \end{aligned}$$

**Note:** The MII matrices of both configurations are identical. As a consequence, we can say unequivocally that the two configurations are isomorphic without using any similarity approach and with the shortest feasible computing time.

C. Numbering Method for the Kinematic Chain Isomorphism Recognition of Planar KCs

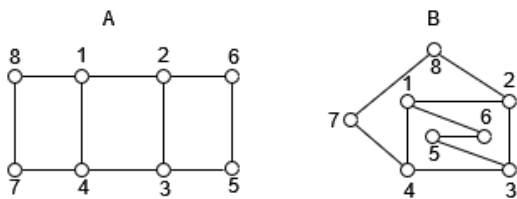


Figure 18. Two non-isomorph eight bars configurations.

Tian et al. [33] published a simple method for numbering the vertices. This method utilizes specific rules based on the degree and weight of vertices, and it will be compared to the above configuration.

The procedure begins by determining the following properties:

$d_i$ : is the vertex degree.  $d_i$  take the values:

- (1)  $d_i = -1$  if vertex  $i$  is binary and is linked to two non-binary vertices,
- (2)  $d_i = -2$  if  $i$  is a binary vertex and is linked to a binary and a non-binary vertex (example from Fig. 18  $d_5 = d_6 = d_7 = d_8 = -2$ )
- (3)  $d_i$  coincides with the proper vertex degree when  $i$  is a non-binary vertex (example,  $d_1 = d_2 = d_3 = d_4 = 3$ ).

I1Di: is the I-level relation code formed by putting the vertex degree ( $d_i$ ) of adjacent vertices to vertex  $i$  in descending order.

**Example 1:**  $I1D_{1,2,3,4} = 33 - 2$ ,  $I1D_{5,6,7,8} = 33$

I2Di: is the II-level relation code formed by putting in descending order the I1Di of the adjacent vertices to vertex  $i$ .

**Example 2:**  $I2D_{1,2,3,4} = 3333 - 233 - 2$ ,  $I2D_{5,6,7,8} = 33 - 233 - 2$

$S_i$ : is the relation code sum. It is the sum of the algebraic relation code of vertex  $i$  in each level (I1Di, I2Di).

**Example 3:**  $S_{1,2,3,4} = 3 + 3 + (-2) = 4$ ,  $S_{5,6,7,8} = 3 + 3 = 6$

This method labels the vertices configuration based on the comparison of  $d_i$ , ImDi, and  $S_i$  for each vertex in the two configurations (see [33]). However, in Fig. 18, the  $d_i$ , ImDi, and  $S_i$  are equal for all types of vertices in both configurations (see examples 1,2, and 3 above), making labeling the vertices in both configurations impossible. As a result, each configuration must be labeled more than  $\frac{n!}{2}$  times. In this example (Fig. 18), 24 different labeling for each configuration implies  $24^2$  comparisons between each configuration's matrices.

By comparing the RGM for both configurations, the proposed method in this paper easily detects the fact that the two configurations are not isomorphs.

$$RGM_A = \begin{bmatrix} 0 & 1 & 0 & 5.2 \\ 1 & 0 & 5.2 & 0 \\ 0 & 5.2 & 0 & 1 \\ 5.2 & 0 & 1 & 0 \end{bmatrix} \quad (17)$$

$$RGM_B = \begin{bmatrix} 0 & 1 & 4.2 & 1 \\ 1 & 0 & 1 & 4.2 \\ 4.2 & 1 & 0 & 1 \\ 1 & 4.2 & 1 & 0 \end{bmatrix} \quad (18)$$

The two configurations in Fig. 18 are not isomorphic. This simple example with a small number of bars demonstrates the superiority and efficiency of the proposed method, which employs the reduced graph matrix to identify isomorphisms in KCs.

**D. A Method for Detecting Isomorphism in Planar Kinematic Chains Based on Compound Topological Invariants**

CTI is produced by the fourth power of the adjacency matrix  $A^4$  (sorting each row element in decreasing order, then comparing the rows and arranging them in decreasing order) and the eigenvalue of the configuration sorted in decreasing order [22]. Despite its simplicity, this approach cannot discover isomorphism between some basic 8-link graphs, as illustrated in Fig 10, since their eigenvalues differ according to the results obtained on MATLAB. The long format of eigenvalues shows the difference.

$$e_A = \begin{bmatrix} -2.281333395043705 \\ -1.944310733422072 \\ -1.140548094310296 \\ -0.099890015366104 \\ 0.598633976166785 \\ 1.000000000000000 \\ 1.275163171393258 \\ 2.592285090582134 \end{bmatrix}, e_B = \begin{bmatrix} -2.281333395043705 \\ -1.944310733422072 \\ -1.140548094310296 \\ -0.099890015366104 \\ 0.598633976166784 \\ 0.999999999999999 \\ 1.275163171393258 \\ 2.592285090582136 \end{bmatrix} \quad (19)$$

The eigenvector was sorted for comparison.

$$V_A = \begin{bmatrix} -0.64 & -0.5 & -0.4 & -0.17 & -0.07 & -0.01 & 0.24 & 0.32 \\ -0.52 & -0.44 & -0.4 & -0.08 & -0.04 & 0 & 0.27 & 0.46 \\ -0.5 & -0.41 & -0.38 & -0.08 & 0 & 0.16 & 0.4 & 0.48 \\ -0.47 & -0.41 & -0.25 & -0.04 & 0.1 & 0.25 & 0.4 & 0.5 \\ -0.46 & -0.4 & -0.17 & 0 & 0.19 & 0.26 & 0.42 & 0.57 \\ -0.45 & -0.21 & -0.11 & 0.09 & 0.24 & 0.3 & 0.44 & 0.59 \\ -0.37 & -0.09 & 0 & 0.18 & 0.25 & 0.37 & 0.44 & 0.63 \\ -0.11 & -0.08 & 0.15 & 0.23 & 0.36 & 0.43 & 0.5 & 0.63 \end{bmatrix} \quad (20)$$

$$V_B = \begin{bmatrix} -0.63 & -0.5 & -0.46 & -0.25 & -0.09 & 0 & 0.25 & 0.4 \\ -0.57 & -0.5 & -0.27 & -0.17 & -0.04 & 0.07 & 0.26 & 0.42 \\ -0.52 & -0.4 & -0.24 & -0.09 & -0.01 & 0.1 & 0.32 & 0.44 \\ -0.44 & -0.4 & -0.19 & -0.08 & 0 & 0.25 & 0.43 & 0.48 \\ -0.44 & -0.37 & -0.08 & 0.04 & 0.23 & 0.3 & 0.45 & 0.5 \\ -0.41 & -0.15 & 0 & 0.11 & 0.24 & 0.37 & 0.46 & 0.59 \\ -0.41 & -0.08 & 0.11 & 0.17 & 0.36 & 0.4 & 0.47 & 0.63 \\ -0.18 & 0 & 0.16 & 0.21 & 0.38 & 0.4 & 0.5 & 0.64 \end{bmatrix} \quad (21)$$

The rows [5, 6, 8] in the two eigenvalue vectors corresponding to the two KCs are not the same in absolute terms, and there is no resemblance in eigenvector values either. This isomorphic kinematic chain will be confused with a non-isomorphic chain, yielding a duplicate solution. As a result, this method based on the CTI falls short of the proposed method in terms of efficiency. If a comparison is used, it should be done to a specific level of precision and not in absolute.

**E. Eliminating Isomorphism Identification Method for Synthesizing Nonfractionated Kinematic Chains Based on Graph Similarity**

The basic idea behind this approach [23] is to add two-degree vertices to a contracted graph and eliminate isomorphism between configurations until the desired configurations are attained. The process for identifying isomorphism consists of the following steps:

- (1) Create the contract graph matrix.
- (2) Add a row and a column to the contract graph matrix for each additional vertex.

- (3) Determine the fourth power of the kinematic chain matrix, then sort the row elements, then the rows themselves.

- (4) Compute the kinematic chain's distance matrix, then sort the row elements and rows themselves.

If two configurations have the same sorted fourth power matrix and sorted distance matrix, the initial edges are similar.

In contrast, the method presented in this study is as follows:

- (1) Convert the adjacency matrix to the reduced graph matrix.
- (2) Determine the reduced graph matrix's second power, then sort the elements of each row and column.

When two configurations have the same sorted second power of the reduced graph matrix, they are isomorphic.

When the detailed steps of the two methods are compared, the proposed method in this paper significantly reduces computation time (the time required to compare those two 42-bar kinematic chains (Fig. 16, Fig. 17) is about  $5 \times 10^{-4}s$  for the proposed method versus  $15 \times 10^{-3}s$  for the method proposed in Ref. [23]), because it calculates the second power matrix rather than the fourth power matrix and does not require the distance matrix). These two criteria reduce execution time dramatically, especially when dealing with a high number of vertices.

**VI. AUTOMATIC COMPARATIVE**

In this section, we compare the previous three techniques to the proposed method on a set of configurations with n bars (n ranging from 8 to 110) to see how they compare in terms of execution time.

The x-axis corresponds to the value k verifying  $n = k + k \times k$  where n is the number of bars. The logarithm of the computer time required to build the invariant used for the isomorphism identification is shown on the y-axis. For the proposed method, the invariant used is the matrix for isomorphism identity MII.

An example of the configurations used in the comparison

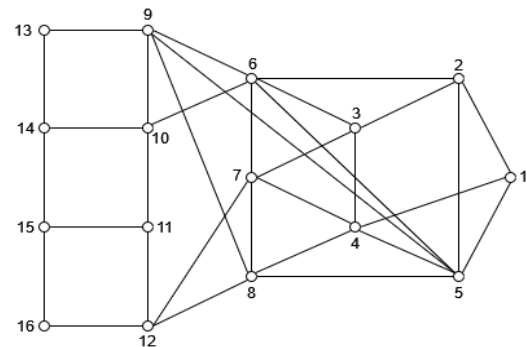


Figure 18. Sixteen bars KC.



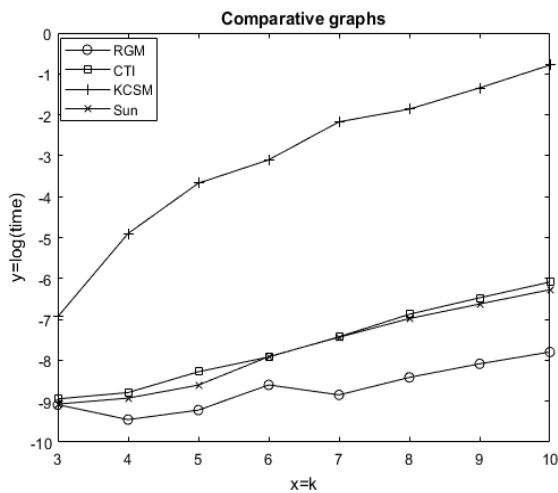


Figure 19. Comparative graphs of the proposed method and the above methods.

The first approach in the comparison section is shown by the (+) graph. It takes more time to produce the matrix identity known as KCSM, as seen in Fig. 19, and there is a significant variation in execution time between this approach and the other methods, ranging between 102 and 106, implying that this technique is less efficient.

The square ( $\square$ ) graph represents the third technique in the comparative section. As shown in Fig 19, when the number of bars is more than 42 ( $k = 6$ ) ( $n = 6 + 6 \times 6 = 42$ ), the time taken by this technique to generate the matrix identity CTI is equivalent to that of the Sun's method, but it is still slower than the suggested method, especially as the number of bars grows.

The (x) sign graph corresponds to the second approach in the comparison section. This method takes less time than the square graph for KCs with up to 35 bars ( $k = 6$ ) and almost the same time otherwise. The latter approach is more efficient for any number of bars than the proposed method.

The graph in Fig.19 clearly demonstrates the suggested method's superiority in terms of speed (short time of execution).

## VII. CONCLUSION

This paper presented a simple and efficient method for quickly identifying isomorphisms in kinematic chains. The suggested technique employs a reduced graph matrix RGM to greatly minimize the calculation time necessary for identifying isomorphisms. The RGM is calculated by deleting the components that correspond to the binary bars in the KC while accounting for them with an appropriate code. A comparison with various published publications confirms the superiority of the suggested technique.

## NOMENCLATURE

**CPVS:** Cubic descending power adjacency matrix  
**CTI:** Compound Topological Invariant  
**DOF:** Degree of Freedom  
**KC:** Kinematic Chain

**KCSM:** kinematic chain structural matrix  
**MII:** Matrix of isomorphism identification  
**RGM:** Reduced graph matrix  
**SRGM:** Square of reduced graph matrix  
**PVS:** Power adjacency matrix

## CONFLICT OF INTEREST

The authors declare that they have not competing interests.

## AUTHOR CONTRIBUTIONS

Mohamed Aly Abdel Kader carried out the research, conceived of the presented idea, conducted the research, analyzed the data and wrote the manuscript. Prof Abdeslam Aannaque supervised the project, provided feedback and guidance, and proofread the manuscript; all authors had approved the final version.

## REFERENCES

- [1] L. S. Woo, "Type synthesis of plane linkages," *J. Eng. Ind.*, pp. 89-159, 1967.
- [2] W. J. Sohn and F. Freudenstein, "An application of dual graphs to the automatic generation of the kinematic structures of mechanisms," *J. Mech. Transmissions Autom. Des.*, pp. 108-392, 1986.
- [3] A. C. Rao and D. Varada Raju, "Application of the hamming number technique to detect isomorphism among kinematic chains and inversions," *Mech. Mach. Theory*, vol. 26, pp. 55-75, 1991.
- [4] J. N. Yadav, C. R. Pratap, and V. P. Agrawal, "Detection of isomorphism among kinematic chains using the distance concept," *The Journal of Mechanical Design. Des.*, vol. 117/607, 1995.
- [5] J. N. Yadav, C. R. Pratap, and V. P. Agrawal, "Computer-aided detection of isomorphism among kinematic chains and mechanisms using the concept of modified distance," *Mech. Mach. Theory*, vol. 31, pp. 439-444, 1996.
- [6] H. S. Yan, "A methodology for creative mechanism design," *Mech. Mach. Theory*, vol. 27, no. 3, pp. 235-242, 1992.
- [7] W. M. Hwang and Y. W. Hwang, "Computer-aided structural synthesis of planar kinematic chains with simple joints," *Mech. Mach. Theory*, vol. 27, no. 2, pp. 189-199, 1992.
- [8] F. G. Kong, Q. Li, and W. J. Zhang, "An artificial neural network approach to mechanism kinematic chain isomorphism identification," *Mech. Mach. Theory*, vol. 34, pp. 271-283, 1999.
- [9] E. R. Tuttle, "Generation of planar kinematic chains," *Mech. Mach. Theory*, vol. 31, no. 6, pp. 729-748, 1996.
- [10] A. C. Rao, "Hamming number technique-I: further applications," *Mech. Mach. Theory*, vol. 32, no. 4, pp. 477-488, 1997.
- [11] A. C. Rao, "Hamming number technique-II: generation of planar kinematic chains," *Mech. Mach. Theory*, vol. 32, no. 4, pp. 489-499, 1997.
- [12] Z. Y. Chang, C. Zhang, Y. H. Yang, Y. X. Wang, "A new method to mechanism kinematic chain isomorphism identification," *Mech. Mach. Theory*, vol. 37, pp. 411-417, 2002.
- [13] H. F. Ding, Z. Huang, "A unique representation of the kinematic chain and the atlas database," *Mech. Mach. Theory*, vol. 42, pp. 637-651, 2007.
- [14] M. K. Lohumi, A. Mohammad, and I. A. Khan, "Hierarchical clustering approach for determination of isomorphism among planar kinematic chains and their derived mechanisms," *J. Mech. Sci. Technol.*, vol. 26, no. 12, pp. 4041-4046, 2012.
- [15] S. H. Rizvi, A. Hasan, and R. A. Khan, "An efficient algorithm for distinct inversions and isomorphism detection in kinematic chains," *Perspectives in Science*, vol. 8, pp. 251-253, 2016.
- [16] K. R. Rajneesh and P. Sunil, "Kinematic chains isomorphism identification using link connectivity number and entropy neglecting tolerance and clearance," *Mech. Mach. Theory*, vol. 123, pp. 40-65, 2018.

- [17] R. K. Rai and S. Punjabi, "A new algorithm of links labelling for the isomorphism detection of various kinematic chains using binary code," *Mech. Mach. Theory*, vol. 131, pp. 1–32, 2019.
- [18] T. Deng, H. Xu, P. Tang, P. Liu, and L. Yan, "A novel algorithm for the isomorphism detection of various kinematic chains using topological index," *Mech. Mach. Theory*, vol. 146, pp. 103740, (2020).
- [19] H. F. Ding, P. Huang, W. J. Yang, A. Kecskeméthy, "Automatic generation of the complete set of planar kinematic chains with up to six independent loops and up to 19 links," *Mech. Mach. Theory*, vol. 96, pp. 75–93, 2016.
- [20] W. J. Yang, H. F. Ding, X. Z. Lai, and M. Wu, "Automatic synthesis of planar simple joint mechanisms with up to 19 links," *Mech. Mach. Theory*, vol. 113, pp.193–207, 2017.
- [21] H. Eleashy, "A new atlas for 8-bar kinematic chains with up to 3 prismatic pairs using joint sorting code," *Mech. Mach. Theory*, vol. 124, pp. 118–132, 2018.
- [22] L. Sun, Z. Ye, R. Cui, W. Yang, and C. Wu, "Compound topological invariant based method for detecting isomorphism in planar kinematic chains," *ASME J. Mech. Des.*, vol. 12, pp. 054504-1, 2020.
- [23] L. Sun, Z. Ye, R. Cui, X. Huang, and C. Wu, "Eliminating isomorphism identification method for synthesizing nonfractionated kinematic chains based on graph similarity," *Mech. Mach. Theory*, vol. 167, pp. 104500, 2022.
- [24] V. Dharanipragada and M. Chintada, "Split hamming string as an isomorphism test for one degree-of-freedom planar simple jointed kinematic chains containing sliders," *J. Mech. Des.*, vol. 138, no. 8, pp. 082301, August 2016.
- [25] V. V. Kamesh, K. M. Rao, and A. B. S. Rao, "An innovative approach to detect isomorphism in planar and geared kinematic chains using graph theory," *J. Mech. Des.*, vol. 139, no. 12, pp. 122301, December 2017.
- [26] R. K. Rai and S. Punjabi, "Isomorphism detection of planar kinematic chains with multiple joints using information theory," *J. Mech. Des.*, vol. 141, no. 10, pp. 102303, October 2019.
- [27] W. Yang and H. Ding, "The perimeter loop based method for the automatic isomorphism detection in planetary gear trains," *J. Mech. Des.*, vol. 140, no. 12, pp. 123302, December 2018.
- [28] R. Cui, Z. Ye, S. Xu, C. Wu, and L. Sun, "Synthesis of planar kinematic chains with prismatic pairs based on a similarity recognition algorithm," *J. Mechanisms Robotics*, vol. 13, no. 5, pp. 051001, October 2021.
- [29] J. M. Hollerbach, "A recursive Lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity," *IEEE Trans. On systems, Man and Cyber*, SMC-10 No. 11, 1980, pp. 730-736.
- [30] K. S. Anderson and J. H. Critchley, "Improved 'order-n' performance algorithm for the simulation of constrained multi-rigid-body dynamic systems," *Multibody Syst. Dyn.*, vol. 9, pp. 185–212, 2003.
- [31] Mohan, S. K. Saha, "A recursive, numerically stable and efficient simulation algorithm for serial robots," *Multibody System Dynamics*, vol. 17, pp. 291-319, 2007.
- [32] L. Sun, R. Cui, Z. Ye, Y. Zhou, Y. Xu, and C. Wu, "Similarity recognition and isomorphism identification of planar kinematic chains," *Mech. Mach. Theory*, vol. 145, pp. 103678, 2020.
- [33] Q. Tian, "Numbering method for the kinematic chain isomorphism recognition of a planar mechanism," *Journal of Physics: Conference Series*, 1983 (2021) 012028.
- [34] H. F. Ding and Z. Huang, "Isomorphism identification of graphs: Especially for the graphs of kinematic chains," *Mechanism and Machine Theory*, vol. 44, no. 1, pp. 122–139, 2009.

Copyright © 2023 by the authors. This is an open access article distributed under the Creative Commons Attribution License ([CC BY-NC-ND 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/)), which permits use, distribution and reproduction in any medium, provided that the article is properly cited, the use is non-commercial and no modifications or adaptations are made.