CERTIFICATE

This is to certify that the work which is being presented in this dissertation entitled as **"A COMPARATIVE STUDY OF VARIOUS METHODS FOR IDENTIFICATION OF ISOMORPHISM IN KINEMATIC CHAINS"** submitted by ***Miss Rashmi Arora*** in partial fulfillment of the requirement for the award of degree of **MASTER OF ENGINEERING** in **CAD/CAM AND ROBOTICS** in the Mechanical Department, **THAPAR UNIVERSITY, PATIALA** is an authentic record of candidate's own work carried by her from January-2008 to June-2008 under the supervision and guidance of **Dr. S.P. Nigam,** Visiting Professor, Mechanical Engineering Department, Thapar University, Patiala. The matter embodied in this dissertation has not been submitted anywhere else for the award of any other degree.

19/6/08

(Dr. S.P. NIGAM)
Visiting Professor,
Mechanical Engg. Deptt.,
Thapar University,
Patiala, 147004.

Countersigned by:

9/7/2008

(Dr. S.K. MOHAPATRA)
Professor and Head,
Mechanical Engineering Department,
Thapar University,
Patiala, 147004.

(Dr. R.K.SHARMA)
Dean of Academic Affairs,
Thapar University,
Patiala, 147004.

The M.E. (Thesis) Viva Voce Examination of RASHMI ARORA Roll No. 80681018, M.E. (CAD/CAM &Robotics), Thapar University, Patiala has been held on ………………

Supervisor **External Examiner**

ACKNOWLEDGEMENT

First of all I wish to thank lord for having given me an opportunity to work under my guide who always made every tough work interesting. It is my proud privilege to express regards and sincere thanks to **Dr. S.K. Mohapatra, Professor and Head, Mechanical Engg. Deptt., Thapar University, Patiala** for giving me the opportunity of being a member of this project for the partial fulfillment of two years M.E. Course. I wish to express my sincere and heartfelt gratitude to **Dr. S.P. Nigam, Visiting Professor, Mechanical Engg. Deptt., Thapar University** for his unfailing inspiration, whole-hearted cooperation and painstaking supervision, through provoking discussions, criticism and suggestions given by him during the entire period of this dissertation. Without his timely and untiring help, it would have not been possible to present this dissertation in its present form.

I also take this opportunity to thank the entire faculty and staff of **MED, Thapar University, Patiala,** for their help, inspiration and moral support throughout the dissertation period which went a long way in bringing out this dissertation from conception to completion.

In the end, I wish to express my deep sense of gratitude to my family, for supporting and encouraging me at every step of my work. It is the power of their blessings, which has given me the courage, confidence and zeal for hard work.

Date : June 2008

Place: PATIALA **(RASHMI ARORA)**

ABSTRACT

Structural synthesis and analysis of mechanism are very important for the invention and innovation of mechanisms. Structural synthesis of kinematic chains usually involves the creation of a complete list of kinematic chains, followed by an Isomorphism test to discard duplicate chains. A significant unsolved problem in structural synthesis is the guaranteed precise elimination of all Isomorphs. Undetected Isomorphism result in duplicate solutions and an unnecessary effort and falsely identified Isomorphism eliminates possible candidates for new mechanisms. Thus, it is absolutely essential to identify isomorphism from the point of view of time saving and correct synthesis of mechanism kinematic chain. Many methods are available to the kinematicians to detect isomorphism among chains and inversions but each has its own shortcomings.

The work presented in this dissertation deals with an approach to critically study some of the existing methods for identification of Isomorphism among kinematic chains and among inversions of given kinematic chain which are then compared with some common examples to select the best method to detect isomorphism in kinematic chain and mechanisms.

The presented work has been divided into two parts. In the first part, all the methods are applied one-by-one for identification of Isomorphism among kinematic chains and among inversions of a given kinematic chain with problems related to one, two and three degree-of-freedom kinematic chains.

In the second part, all the methods are compared with their strengths and weaknesses with common examples of six links one degree of freedom Watt and Stephenson Chains to select the best method for identification of Isomorphism among kinematic chains and among inversions of a given kinematic chain. Also, the methods are applied for identification of various structural properties as

Degeneration identification, Identification of Type of freedom i.e. Total, Partial and Fractionated degree-of-freedom, in planar kinematic chains of one, two and three degree-of-freedom.

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

A	Adjacency Matrix
ASMTDRL	Arranged Sequence of Modified Total Distance Ranks of all the links
C	Connectivity Matrix
C.H.N.	Chain Hamming Number
C.H.S.	Chain Hamming String
C.L.S.	Chain Loop String
DOF, F	Degree of Freedom
H	Hamming Matrix
I.L.	Independent Loop
j	Number of Joints of given Kinematic Chain
[JJ]	The Joint – Joint Matrix
J.V.C.	Joint Value of a Chain
KC	Kinematic Chain
L.A.S.L.	Loop Adjacency String of a Link
L.F.S.C.	Loop Frequency String of a Chain
L.F.S.L.	Loop Frequency String of a Link
L.H.N.	Link Hamming Number
L.H.S.	Link Hamming String
L.S.	Loop Size
L.S.L.	Loop String of a Link
L.V.	Link Value
L.V.C.	Loop Value of a Chain
L.V.L.	Loop Value of a Link
MD	Modified Link-Link Distance Matrix
MJJ	The Maximum Absolute Value of the Characteristic Polynomial Coefficients
MR	Modified Link-Link Relation Matrix
n	Number of Links of given KC
S.L.	Sub Loops
Σ JJ	The sum of the Absolute Value of the Characteristic Polynomial Coefficients

CONTENTS

CHAPTER – 1
INTRODUCTION

In a mechanism design problem, systematic steps are type synthesis, structural/number synthesis and dimensional synthesis. Structural analysis and synthesis of the Kinematic Chain (KC) and mechanism has been the subject of a number of studies in recent years. One important aspect of structural synthesis is to develop the all-possible arrangements of KC and their derived mechanisms for a given number of links, joints and degree of freedom, so that the designer has the liberty to select the best or optimum mechanisms according to his requirements. In the course of development of KC and mechanisms, duplication may be possible. One very important problem encountered during structural synthesis of chains is the detection of possible isomorphism among planar chains with simple as well as multiple joints.

Investigation of structural aspects of kinematic chains has been mostly limited to generation of distinct chains, the process of which involves detection of isomorphism. Generation of distinct chains alone will have not much significance unless the designer is in a position to compare all of them for the specified task at the conceptual stage of design without having the actual design. This comparison of all the chains from structural dissimilarity point of view is termed as isomorphism identification among chains and among inversions of a given kinematic chain while generating new mechanisms.

The method to detect isomorphism between kinematic chains and between inversions of a given chain, so that unnecessary duplication and omission of a potentially useful chain may be avoided, has been the hot bed of investigations. Therefore, there exists a wealth of literature dealing with this topic in the area of graph theory as well as kinematics. However, scope exists for an efficient solution to the graph isomorphism problem.

A lot of time and effort has been devoted to develop a reliable and computationally efficient technique for isomorphism identification in mechanism kinematic chain and from a long time work is continued in this field. Moreover, recent studies on intelligent CAD and intelligent manufacturing systems have revealed a need for tools to maintain a large knowledgebase, by identifying isomorphic pieces of knowledge.

A number of methods for testing isomorphism have been developed and with the belief that kinematic analysis and synthesis should not end up only with the generation of kinematic chains, attempts were made to compare the kinematic chains and inversions for their anticipated behavior such as Mechanical Advantage, Dynamics, Degeneracy in chains, Types of freedom, Symmetry, Work Space, Rigidity and Compactness or Parallelism of closed kinematic chain etc.

.

CHAPTER - 2
LITERATURE REVIEW

A wealth of literature exists in the area of isomorphism and a lot of time and effort has been devoted to develop different methods for isomorphism identification in kinematic chains. From a long time, work is continued in this field. Some important literatures are as below:

Manolescu, N.I. [1] introduced a method based on the transforming of Baranov Trusses into planar kinematic chains using an idea called "Graphization", denoted by the symbol (G). The chains obtained are kinematic chains with multiple joints and simple links, denoted as **KCmjsl**. Next, using "Dyad Amplification" **(DA)**, one can obtain other KCmjsl with a greater number of links though they have still the same number of degrees of freedom. By another operation-the simplifying of joints **(JS)** one then obtains planar Kinematic Chains with simple joints, (called **KCsj)**, and subsequently planar jointed mechanisms (PJM), and also planar jointed Driving Mechanisms **(DM)**, as they exist in the technology of mechanism design.

An algorithm for determining whether two triconnected planar graphs are isomorphic is presented by **Hopcroft, J.E. and Tarjan, R.E. [2].** The asymptotic growth rate of the algorithm is bounded by a constant times |V| log |V| where |V| is the number of vertices in the graphs.

Uicker, J.J. and Raicu, A. [3] presented a method for determining a set of identification numbers for each possible kinematic chain. Here a theorem was also proved which shows that kinematically equivalent chains may be detected by the equality of these identifying numbers. An extension of the method allows testing the equivalence of chains including different types of kinematic pairs. Another variation was also discussed which explicitly determines the kinematic loops and paths of different lengths.

Mruthyunjaya, T.S. and Raghavan, M.R. [4] worked over a method based on Bocher's formulae for determining the characteristic coefficients (which have been as suggested as index of Isomorphism) of the matrix associated with the kinematic chain. The method provides an insight into the physical meaning of these coefficient and leads to a possible way of arriving at the coefficients by an inspection of the chain. A modification to the matrix notation is proposed with a view to permit derivation of all possible mechanism from a kinematic chain and distinguishing the structurally distinct ones. Finally a generalized matrix notation is proposed to facilitate representation and analysis of multiple-jointed chains.

A Linkage Characteristics Polynomial was defined by **Yan, H.S. and Hall, A.S. [5]**, which is the characteristics polynomial of the adjacency matrix of the kinematic graph of the kinematic chain. A rule which all the coefficients of the characteristic polynomial of a kinematic chain can be identified by inspection, based on the interpretation of a graph determinant, was derived and presented. This inspection rule interprets the topological meaning behind each characteristics coefficient, and might have some interesting possible uses in studies of the structural analysis and synthesis of kinematic chains.

Several assembly theorems were presented and derived by **Yan, H.S. and Hall, A.S. [6]** for obtaining the Linkage Characteristic Polynomial for a complex chain through a series of steps involving the known polynomials for subunits of the chain, are derived and presented. These theorems give insight into how the topological information concerning the linkage is stored in the polynomial and might contribute to the automated recognition of linkage structure in generalized computerized design programs. Based on graph theory, the characteristics polynomial cannot characterize the graph up to Isomorphism. However, for practical applications in the field of linkage mechanisms, it is extremely likely that the characteristic polynomials are unique for closed connected kinematic chains without any over constrained sub chains.

Yan, Hong-Sen and Hwang, Wen-Miin [7] presented the idea of linkage path code for the identification and recognition of planar kinematic chains with simple joints and turning pairs. For every given chain, this code can be obtained according to the definition directly or by a proposed method which can greatly simplify the procedure for obtaining the code by inspection.

Mruthyunjaya, T.S. [8] made an effort to develop a fully computerized approach for structural synthesis of kinematic chains. The steps involved in the method of structural synthesis based on transformation of binary chains, have been recast in a format suitable for implementation on a digital computer. The methodology thus evolved has been combined with the algebraic procedures for structural synthesis and analysis of simple jointed kinematic chains with a degree of freedom ≥ 0.

The test based on comparison of the Characteristic Coefficients of the adjacency matrices of the corresponding graphs for detection of Isomorphism in kinematic chains has been shown to fail in the case of two pairs of ten links, simple jointed chains, one pair corresponding to single freedom chains and the other pair corresponding to three freedom chains. An assessment of the merits and demerits of available methods for detection of Isomorphism in graphs and kinematic chains was presented by **Mruthyunjaya, T.S. and Balasubramanian, H.R. [9]**, keeping in view the suitability of the methods for use in computerized structural synthesis of kinematic chains. A new test based on the Characteristic Coefficients of the "degree" matrix of the corresponding graph is proposed for detection of isomorphism in kinematic chains. The new test was found to be successful in the case of a number of examples of graphs where the test based on Characteristic Coefficients of adjacency matrix fails. It has also been found to be successful in distinguishing the structures of all known simple-jointed kinematic chains in the categories of (a) single-freedom chains with upto 10 links, (b) two-freedom chains with upto 9 links and (c) three-freedom chains with upto 10 links.

Ambedkar, A.G. and Agrawal, V.P. [10] explained the concept of minimum code and discussed its properties relevant to kinematic chains. An algorithm, based on one method available in the graph theoretic literature in chemistry, is elaborated to demonstrate its applicability in establishing minimum code of kinematic chains with simple joints. Minimum code, being unique for kinematic chains, is suitable for testing isomorphism. The decidability of this code positively indicates its possiblity in cataloguing (storage and retrieval) of kinematic chains and mechanisms.

Min code, as canonical number, was shown to give a unique number for kinematic chains with simple joints. **Ambedkar, A.G. and Agrawal, V.P. [11]** suggested a method to identify mechanisms, path generators and function generators through a set of identification numbers. The concept of min code is also shown to be effective in revealing the topology of kinematic chains and mechanisms consisting of (a) different types of lower pairs, and/or (b) simple and multiple joints.

Loop connectivity properties of multi-loop kinematic chains are used to develop a hierarchical classification scheme of kinematic structures. A loop-loop permanent matrix is defined by **Agrawal, V.P. and Rao, J.S. [12]** leading to mathematical equation (permanent function) and an identification set which is an invariant of a grouping of chains. The scheme reduces computer time and effort in the optimum selection of a kinematic structure from a large family of kinematic chains using a computer-aided design program.

An attempt was made by **Agrawal, V.P. and Rao, J.S. [13]** to develop computationally simple and efficient analytical methods using matrices, Link-Link Variable Characteristic Polynomial **(VCP-L),** Link-Link Variable Permanent Function **(VPF-L)** for identification and Isomorphism of kinematic chains (planar, spatial, simple and multiple-jointed with different types of joints) and their mechanisms such as path generators and function generators. A complete matrix representation of kinematic chains results in the development of unique mathematical expressions called VCP-L and VPF-L which identify kinematic chains up to Isomorphism, and leads to explicit expressions to determine Characteristic Coefficients **(CC-Ls)** analytically or by visual inspection of the chain. Structural invariants contained in VCP-L and VPF-L are identified and a simple method for computing them is developed. A method is developed to derive all possible distinct paths in a kinematic chain analytically.

Many methods were available to the kinematicians to detect isomorphism among chains and among inversions but each has its own shortcomings. **Rao, A.C. [14]** presented a novel approach of Hamming Number Technique which is both reliable and simple. Use is made of the Hamming number, a concept borrowed from digital communication theory. The connectivity matrix of various links, a matrix of zeroes and ones, is first formed and Hamming number matrix is computed. The link Hamming string-which is defined as the string obtained by concatenating the link Hamming number and the frequency of individual Hamming numbers in that row-is then formed. Finally, the chain Hamming string, defined as the string obtained by the concatenation of the chain Hamming number and the link Hamming string, is an excellent test for the isomorphism among chains. Also, the link Hamming String of every link together with those of its neighbours is an excellent test for isomorphism among the inversions of a given chain.

Topology of kinematic chains is useful in comparing them for the structural-error point of view and an attempt was made by **Rao, A.C. and Rao, C.N. [15]** in this direction. The method reported, however, fails to compare the chains which consist of the same number and type of links and joints; ternary-binary, ternary-quaternary, etc. but differ in loop formation only. Further, comparison of the loop hamming values of links and chains is expected to be the simplest and positive test for isomorphism.

Tang, C.S. and Liu, Tyang [16] used a new method of compact mathematical representation for a graph called the Degree Code, to identify graph Isomorphism. Isomorphic graphs have identical Degree Codes; non isomorphic graphs have distinct degree Codes. Therefore by examining the Degree Codes of the graphs, Graph Isomorphism is easily and correctly identified. Degree Codes serve as an effective nomenclature and storage system for graph or mechanism. Although this identification scheme was developed specifically for the structural synthesis of mechanism, it can be applied to any area where Graph Isomorphism is a critical issue.

A completely new concept about the kinematic chain, called the Link's Adjacent Chain Table (shortened as **ACT)**, was originated by **Jin-Kui, Chu and Wei-Qing, Cao [17].** It is an invariant which can be used to describe the topological relationships between links in the kinematic chains (shortened as **KC).** In comparison with the traditional representation of the KC, i.e. with the adjacent matrix, link's ACT is much more audio-visual, simpler and more direct in describing the topological relationship among links. Using the link's ACT the isomorphism of the KC and inversions can be easily determined. Among the existing methods for the identification of the isomorphic graphs, link's ACT takes the least time in calculation. So its introduction of the link's ACT leads to an effectual solution to identify the isomorphism of the KC. It is also felt that the link's ACT reveals at a glance how many inversions are possible out of a given chain.

Shende, S. and Rao, A.C. [18] dealt with the problem of detection of isomorphism which is frequently encountered in structural synthesis of kinematic chains. A new method which incorporates all essential features of graph, easy to compute and reliable, is suggested. Summation Polynomials are used in place of Characteristic Polynomials. The advantage is that they are easy to compute. The coefficient and exponents of Summation Polynomials are very informative and from them valuable information regarding topology of kinematic chains can be predicted. It is capable of detecting isomorphism in all types of planar kinematic chains i.e. chains of single degree or multi degree of freedom and multiple jointed chains.

Tischler, C.R.; Samuel. A.E. and Hunt K.H. [19]. Number Synthesis of kinematic chains usually involves the generation of kinematic chains followed by a time-consuming, computer-intensive procedure for the elimination of isomorphs. A significant unsolved problem in number synthesis is the guaranteed precise elimination of all isomorphs. Since there is no efficient algorithm for always determining whether two kinematic chains are isomorphic, any efficient algorithm has a finite probability of rejecting a unique, potentially useful chain. This method reviews the history of number synthesis and presents a new orderly method for synthesizing kinematic chains. This new method guarantees to produce a complete list of chains, which, only some doubt to the uniqueness of a chain exists, may include an isomorphic chain. As a consequence, this technique produces significantly fewer isomorphs in the output list than do previous techniques; often no isomorphs are produced by the method whatsoever. It is proposed that in many situations where the synthesis of kinematic chains is required, the processing of duplicate chains in the early stage of design is preferable to the omission of a potentially useful chain.

Yadav, J.N.; Agrawal, V.P. and Pratap, C.R. [20] expanded their work of Isomorphism identification through Distance Concept with computer. A new invariant, called the Arranged Sequences of Total Multiplicity Distance Ranks of all the Links **(ASTMDRL)**, has been developed for a binary chain. These invariants are derived from the link-link multiplicity of all the joints in a chain and are taken into consideration, with a view to enhance the discriminating ability of the new invariant. Based on this invariant, a computer aided method has been developed for detecting Isomorphism among planar binary chains.

Type synthesis of rigid-link mechanisms provides a means to determine mechanism topologies before considering link dimensions. Although researchers have provided valuable insight into the enumeration of mechanisms containing flexible members, a mathematical rigorous process for the topological synthesis of compliant mechanism is lacking. One reason for this is the large increase in the number of possible mechanisms for a given number of links when the links are allowed to be flexible. **Murphy, D. Morgan; Midha, Ashok and Howell, L. Larry [21]** addressed this challenge with the development of an enumeration process for compliant mechanisms that builds on the rigid-body type-synthesis techniques and the terminology previously established for compliant elements. Compliant segment and connection information are included in the formulation and methods for determining isomorphism are also introduced.

Tuttle, E.R. [22] presented a completely automated, mathematically rigorous procedure that generates all planar, non-fractionated, pin-jointed kinematic chains having 2-6 independent loops and 1, 2 or 3 degrees of freedom. It also addresses the isomorphism problem and the elimination of rigid sub-chains. The computer programs can be run on any personal computer; over 6 million kinematic chains and 100 million inversions have been generated.

A new concept of Link Path Code for a link of planar kinematic chain with simple joints was introduced by **Yadav, J.N. and Pratap, C.R. [23].** An algorithm has been developed for determining Link Path Codes for all the links of a kinematic chain, leading to the identification of its distinct mechanism. Based on the concept of Link Path Code, an algorithm has also been proposed to quantify the degree of structural similarity between any two simple jointed planar kinematic chains with a given number of links and degree of freedom as well as between any two mechanisms of a planar kinematic chain with simple joints.

A new invariant, called the arranged sequence of total multiplicity distance ranks of all the links (ASTMDRL), had been developed for a binary chain by **Yadav, J.N.; Agrawal, V.P. and Pratap, C.R. [24]**. This invariant is derived from the link-link multiplicity distance matrix of a chain, defined here, in which the multiplicities of all the joints of a chain are taken into consideration, with a view to enhance the discriminating ability of the new invariant. Based on this invariant, a computer aided method has been developed for detecting isomorphism among planar binary chains.

Rao, A.C. [25] introduced further application of Hamming Number Technique to reveal certain properties such as Degeneration and Type of freedom which will be helpful during the synthesis of single and multi degree of freedom kinematic chains. Also the concept of symmetry in kinematic chains is introduced and its utility in detecting Isomorphism among chains and in rating them is illustrated. Writing of the Hamming matrix directly to save time and effort, for a chain, is explained.

Synthesis of planar kinematic chains is in general tedious and time consuming. **Rao, A.C. [26]** reported a very simple and direct method for the generation of n-link Ø degree-of-freedom (d.o.f.) chains from (n-2) links and *f* d.o.f. chains. Hamming number technique reported earlier for testing isomorphism among kinematic chains is extended to reveal identity and symmetry among the links and joints of a chain. This in turn enables generation of distinct n-link chains directly from each distinct (n-2) link without having to test for degeneration and isomorphism. The formulae suggested will give the number of such chains. Chains with (n-1) links and (f+1) d.o.f. are obtained as a by-product. The distinct *n*-link *f* d.o.f. chains obtained from all the basic chains need, however, to be tested for isomorphism among themselves. The advantage of this method can be perceived from the fact that only about 400 10-link chains need be tested to get the already established 230 distinct chains.

Rao, A.C. and Jagdeesh, Anne [27] presented facts as investigation of structural aspects of kinematic chains has been mostly limited to generation of distinct chains, which process involves detection of Isomorphism. Generation of distinct chains alone will not have much significance unless the designer is in a position to compare all of them for the specified task at the conceptual stage of design without having the actual design. Based on the topology of chains, quantitative methods are presented in order to compare all the distinct chains, with the specified number of links and degree of freedom (d.o.f.), (i) for workspace and rigidity, (ii) to select the joint of the input link to introduce motion, and (iii) to test isomorphism, simply and uniquely; in order to save time, efforts and cost.

A new method based on an Artificial Neural Network **(ANN)** Technique was presented by **Zhang, W.J.; Lib, Q. and Kong, F.G. [28]** to identify the Isomorphism of the mechanism kinematic chain. The mechanism kinematic chain is represented by a graph. The Hopfield-Tank ANN model for the Graph Isomorphism Identification is developed, which includes the construction of an Energy Function and a Dynamic Equation. A case study is provided to demonstrate the preliminary success of the approach. An important motivation of this study is based on an observation that existing methods have not provided robust solutions, because of the hard nature of this problem.

Zhange, W.J. and Li, Q. [29] explored the fact that different level of mechanism topology should be considered in order to make the activity or tool of computer comparison of mechanism topology more useful to support a whole design process of mechanisms. In this connection, four abstraction levels are identified to relate the different types of mechanism topology with reference to mechanism design tasks, processes or objectives. A new approach was applied here to compare mechanism topology for all these levels. One of the ideas of this approach is to extend the existing incident Degree Code Approach by identifying more features of mechanism topology and to define them into an extended code. This is further enhanced by an algorithm to perform permutation within a group of vertices that have the same features.

Loop based Isomorphism detection technique was provided by **Rao, A.C. [30].** A new invariant for a chain, called the Chain Loop String is developed for a planar kinematic chain with simple joints to detect isomorphism among chains. Another invariant called the Link Adjacency String is developed, which is a by-product of the same method to detect inversions of a given chain. Using the Loop Concept, method reveals simultaneously chain is isomorphic; link is isomorphic and type of freedom with no extra computational effort.

Rao, A.C. [31] presented a Genetic Algorithm for testing Isomorphism among kinematic chains and to select the best frame and input links. The computational effort involved is minimum and the method is unique as it satisfies both the necessary and sufficient requirements. Fitness of a Binary String corresponding to a link is indicative of its design parameters active in motion generation. Chains are compared for function generation on the basis of the 'fitness' of first generation and second generation 'fitness' etc, in that order.

Rao, A.C. [32] proposed Fuzzy Logic method to investigate Isomorphism among kinematic chains and inversion where in necessary and sufficient conditions are specified for detecting Isomorphism. Numerical measures to compare the numerous distinct chains with the same number of links and degrees of freedom (d.o.f.) for characteristics like Symmetry, Parallelism and Mobility are proposed.

Schmidt, C. Linda and Harshawardhan, Shetty [33] explored the critical fact of Isomorphism identification in structural synthesis of mechanism. A Graph Theory has been applied to the structural analysis to identify graph isomorphism, while generating new mechanisms. Linear time algorithm is presented here for isomorphism identification with its grammar rules.

It is believed that the generation of distinct kinematic chains is not possible without testing for isomorphism. A new method was presented by **Rao, A.C. and Deshmukh, B. Pratap [34]** which does not require the test for isomorphism.

A new method based on Eigen vectors and Eigen values to identify Isomorphism of mechanism kinematic chain was developed by **Chang, Zongyu; Zhang, Ce; Yang, Yuhu and Wang, Yuxin [35].** Kinematic chains are firstly represented by Adjacent Matrices. By comparing the Eigen values and corresponding Eigen vectors of Adjacent Matrices, the Isomorphism of mechanism kinematics chain can easily be identified. The relationships in the mechanisms can also be obtained.

Mruthyunjaya, T.S. [36] made a deep analysis over the kinematic structure of kinematic chains. Along with encompassing isomorphism identification as well as its various methods of identification as characteristic polynomial based approaches, approach based on paths/distances, other approaches to isomorphism, covered characteristics of mechanisms which are determined solely by the pattern of interconnection among the constituent links of the mechanism and hence are independent of matrix properties of the mechanisms.

Mahere, Vishesh; Nigam, S.P. [37] presented a method based on the Adjacency Matrices of kinematic chains to identify Isomorphism, among kinematic chains and among inversions of given kinematic chain and for identification of structural properties of kinematic chains as Degeneration identification, identification of Type of freedom. The advantage of this method is that it covers all planar kinematic chains of one, two and three degree of freedom at once for identification of Isomorphism and for structural properties over all previous methods.

Sarkar, S.C., Khare, A.K. [38] dealt with a theoretical analysis for detecting the flexibility, isomorphism and effect of uncertainty in 10-bar kinematic chains, using the concept of directed graph. In this case, flow of motion between links is evaluated considering all the possible paths for motion transmission instead of only the shortest path, as proposed by earlier investigators.

A number of theories on the relationship between adjacent matrices of isomorphic mechanism kinematic chains have been investigated by **Cubillo, J.P. and Wan, Jinbao [39].** A particular published theory about mechanism kinematic chain isomorphism using adjacent matrices has been revised, after errors in the original theory were discovered. Subsequently, the necessary and sufficient conditions of the eigen values and eigen vectors of adjacent matrices for isomorphic kinematic chains have been proven rigorously. A new procedure to identify isomorphic chains has been developed and presented. With this new procedure, it is only necessary to compare eigen values and several eigen vectors of adjacent matrices of isomorphic kinematic chains to identify the isomorphic chains.

An algorithm to exhaustively enumerate and structurally classify planar simple-jointed kinematic chains using the hierarchical representation of Fang and Freudenstein was proposed by **Butcher, A. Eric; Hartman, Chris [40]** in which all isomorphic chains are automatically eliminated in the enumeration procedure such that isomorphism testing on the final set of chains is eliminated. An efficient rule-based technique for eliminating degenerate kinematic chains is also proposed. This efficient scheme allows for the exhaustive enumeration of complicated cases that have received little or no attention in the past due to the difficult and time-consuming aspect of testing the kinematic chains for isomorphism. For verification, the algorithm is first applied to enumerate the single degree-of-freedom 6, 8 and 10-bar cases, then to the 12-bar case, and finally to the previously intractable 14-bar case.

Some further development on the Eigen system approach for graph Isomorphism detection was done by **P.R., He; W.J., Zhang and Li, Q. [41].** In this method, they proposed a new matrix called adjusted adjacency matrix that meets the requirement of a graph that must contain at least one distinct Eigen value, which was not met by adjacency matrix used in Eigen system approach earlier. This approach is not only effective but also more efficient than that based on the adjacency matrix.

Many methods are reported to test isomorphism among kinematic chains and gear trains but they are limited to the detection of isomorphism and cannot reveal different characteristics of the chains. In the opinion of **Srinath, A. and Rao, A.C. [42],** utility of the any technique used to detect isomorphism will be enhanced if it can reveal the kinematic characteristics such as parallelism, type of freedom, symmetry of the chains, etc. This will help the designer. In this, an attempt is made to develop a technique to fulfill this need using the concept of correlation.

Synthesis of kinematic chains can be viewed as the enumeration of a certain class of graphs. Very efficient algorithms, using group theoretic techniques, exist for exhaustive isomorph-free generation of certain classes of combinatorial objects, which either eliminate or restrict the explicit isomorph detection. An algorithm belonging to one particular class called McKay-type, in combination with an efficient degeneracy testing algorithm, is used for the synthesis of planar mechanisms by **Sunkari, P. Rajesh; Schmidt, C. Linda [43].** The existing results are reexamined and the discrepancies are reconciled.

A new method to identify the distinct mechanisms (DM) from a given kinematic chain was proposed by **Ali, Hasan; Khan, R.A. and Aas, Mohd. [44].** This method is easy to compute and reliable. The KC are represented in the form of Joint-Joint [JJ] matrices. Two structural invariants are the sum of the absolute characteristic polynomial coefficients and maximum absolute value of the characteristic polynomial coefficient. These invariants are used as the composite identification number of a KC and mechanisms. It is capable of detecting isomorphism in all types of planar kinematic chains.

A systematic procedure to count the number of planar mechanisms subject to design constraints from the kinematic chains was presented by **Hung, Chih – Ching; Yan, Hong – Sen and Pennock, Gordon R. [45].** The procedure is based on well-known principles that can be found in graph theory and combinatorial mathematics. A link-path is employed to include the design constraints that a number of specified joints must correspond to a number of specified links. Also, modified permutation groups, generating function, and Polya's theory are used to count the number of non-isomorphic mechanisms with the required design constraints. Then the pattern inventory is used for the conceptual design of the identified mechanisms. The procedure can identify all of the non-isomorphic mechanisms in a specified kinematic chain. In addition, the procedure can be used to determine the isomorphic mechanisms in a straightforward manner. Three practical examples are included to illustrate the systematic nature of the proposed procedure; namely: the differential-type south pointing chariot, the Watt kinematic chain, and a variable-stroke engine.

Based on the array representation of loops in topological graphs of kinematic chains, **Ding, Huafeng and Huang, Zhen [46]** proposed two basic loop operations, "Θ" and "Θ", for the first time. The existent conditions and properties of "Θ" operation are researched and four laws about the operation are presented. Furthermore, after the important concepts of the independent loop set and its selection theorem are proposed, the loop relationship of kinematic chains is revealed; thus an original theory of loop analysis is established. Finally, some applications are given under the basic theory above, such as the isomorphism identification, the detection of rigid sub-chains, and the freedom-type analysis of kinematic chains.

Therefore, it is seen that various researchers have devoted a lot of time to develop different methods to detect isomorphism. Six important methods with an effort to detect the best method out of them are discussed in detail in further chapters.

CHAPTER -3
MECHANISM SYNTHESIS AND ANALYSIS [1, 33, 46, 47]

Most engineering design practice involves a combination of synthesis and analysis. However, one cannot analyze anything until it has been synthesized into existence.

The process of drawing kinematic diagrams and determining degrees of freedom of more complex mechanisms are the first steps in both the kinematic analysis and synthesis process. In *kinematic analysis,* a particular given mechanism is investigated based on the geometry plus possibly other known characteristics (such as input angular velocity, angular acceleration, etc.). *Kinematic synthesis*, on the other hand, is the process of designing a mechanism to accomplish a desired task. Here, both the type as well as the dimensions of the new mechanism can be part of kinematic synthesis.

Mechanism Synthesis or Kinematic Synthesis is a process of determination of mechanisms that are to fulfill certain motion specifications. It is the very fundamental of design, for it represents the creation of new hardware to meet particular needs in motion – Displacement, Velocity or Acceleration – singly or in combination.

3.1 DIFFERENT PHASES OF MECHANISM SYNTHESIS ---

Thus the overall problem of synthesis may be approached in three interrelated phases –

1. The form and type of mechanism – Type Synthesis
2. The number of links and the nature of connection needed to permit the required movability – Number Synthesis
3. The proportions (length) of the links necessary to accomplish the specified motion transformation – Dimensional Synthesis

3.1.1 TYPE SYNTHESIS ---

The first phase is called Type Synthesis. Type Synthesis refers to the definition of the proper type of mechanism best suited to the problem. Here the choice of the kind of links or constructional units is determined, as linkwork, gears, cams, belts etc. A poor choice at the type synthesis stage can create insoluble problems later on. The design might have to be scrapped after completion, at great expense.

3.1.2 NUMBER SYNTHESIS ---

The term Number Synthesis has been coined to mean the determination of the number of links and joints necessary to produce motion of a particular Degree of Freedom (D.O.F.).

The value of number synthesis is to allow the exhaustive determination of all possible combinations of links which will yield any chosen DOF. This then equips the designer with a definitive catalog of potential linkages to solve a variety of motion control problems.

3.1.2.1 DEGREE OF FREEDOM (F) ---

The concept of degree of freedom (also called the mobility M) is fundamental to both the synthesis and analysis of mechanisms. It can be defined as the number of input parameters (usually pair variables) which must be independently controlled in order to bring the mechanism into a useful engineering purpose. It means that it is the number of independent inputs required to determine the position of all links of the mechanism with respect to ground.

In the case of kinematic chain, it is the number of independent pair variables needed to completely define the relative positions of all links. For example, the truss shown in Fig. 3.1(a) has zero degree of freedom (F=0); here the relative positions of the links result from their lengths and no pair variables can be specified.

The number of degrees of freedom of the four links chain of Fig. 3.1(b) is unity (F=1); here one variable such as Φ is needed to define the relative positions of all links. A five links chain Fig. 3.1(c) has two degrees of freedom (F=2), for two angles such as Φ_1 and Φ_2 are needed to define the relative position of all links.

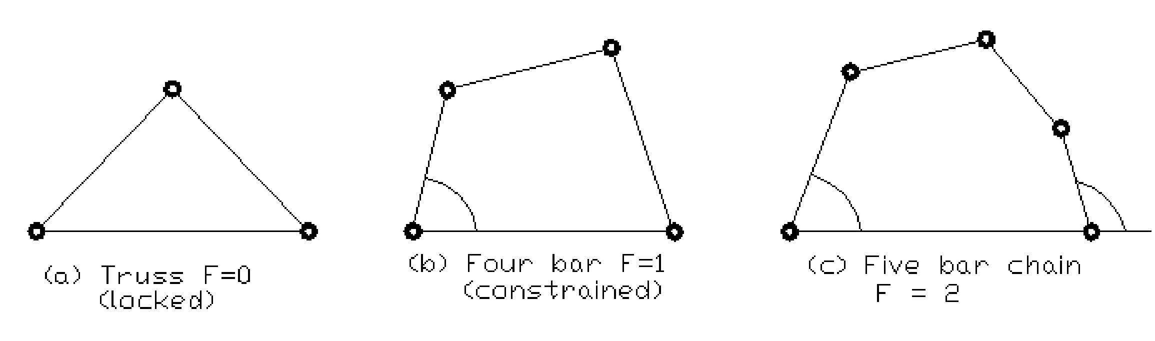


FIG. 3.1 EXAMPLES OF KINEMATIC CHAINS WITH DEGREE OF FREEDOM= 0, 1, 2

A kinematic chain is said to be movable when its number of degrees of freedom is one or greater ($F \geq 1$); it is otherwise locked ($F<1$). If the number of degrees of freedom is equal to unity ($F=1$), the chain is said to be constrained.

Estimation of mobility or degree of freedom can be done by **Gruebler's Equation.** For planar mechanism, Gruebler's Equation for degrees of freedom is:

$F = 3(L-1) - 2J_1$ ------------------------------------(1)

Where

L = Total number of links in a mechanism

F = Degrees of freedom

J_1 = Number of 1 DOF joints

It will be less confusing if **Kutzbach's** modification of Gruebler's equation is used in this form:

$F = 3(L-1) - 2J_1 - J_2$ --------------------------------(2)

Where

J_2 = Number of 2 DOF joints

3.1.3 DIMENSIONAL SYNTHESIS ---

The third phase is called Dimensional Synthesis which deals with the determination of the final dimensions (lengths) of the links necessary to accomplish desired motion characteristics, assuring that the mechanism follows a specified path or moves through certain points and including velocity and acceleration considerations.

For dimensional synthesis, methods are:-

- Graphical Methods – provide the designer with a quick straightforward method but parameters cannot easily be manipulated to create new solutions.
- Analytical Methods – This approach is suitable for automatic computation. Once a mechanism is modeled and coded for computer, parameters are easily manipulated to create new designs.

Thus Mechanism Synthesis is comprised of Number, Type and Dimensional Synthesis. Number and Type Synthesis together are called Structural Synthesis. Structural Synthesis provides the structural characteristics of the mechanism (degree of freedom, number and type of links and joints etc.)

3.2 STRUCTURAL SYNTHESIS [33] ---

Structural analysis and synthesis of mechanism is very important for the invention and innovation of mechanism. With structural synthesis, it is desirable to enumerate the kinematic chains systematically and also to know the inherent characteristics of a chain related to its structure so that all the acceptable chains can be evaluated in depth before making the final selection for the specified task. Many studies have been reported in the past, using different approaches to enumerate all the distinct chains with the given number of links and degrees of freedom.

Mechanism researchers have developed a systematic, algebraic design methodology for structure synthesis. This method, herein-after referred to as the Systematic Method, relies on graph representations of mechanisms. The Systematic Method simplifies the design process by partially separating mechanism function from mechanism shape during design generation. The mechanism structure is determined independently of mechanism dimensions. The six-node seven-edge planar graphs, labeled graphs, and functional schematics shown in Fig. 3.2 illustrate the Systematic Method’s representation scheme.

Many alternative approaches exist which utilize graphical abstractions. They include the contracted graph approach, the systematic procedure for type synthesis; applied to planetary gear train synthesis, the dual graph approach, the Assur group approach, the transformation of binary chain approach and the finite groups approach.

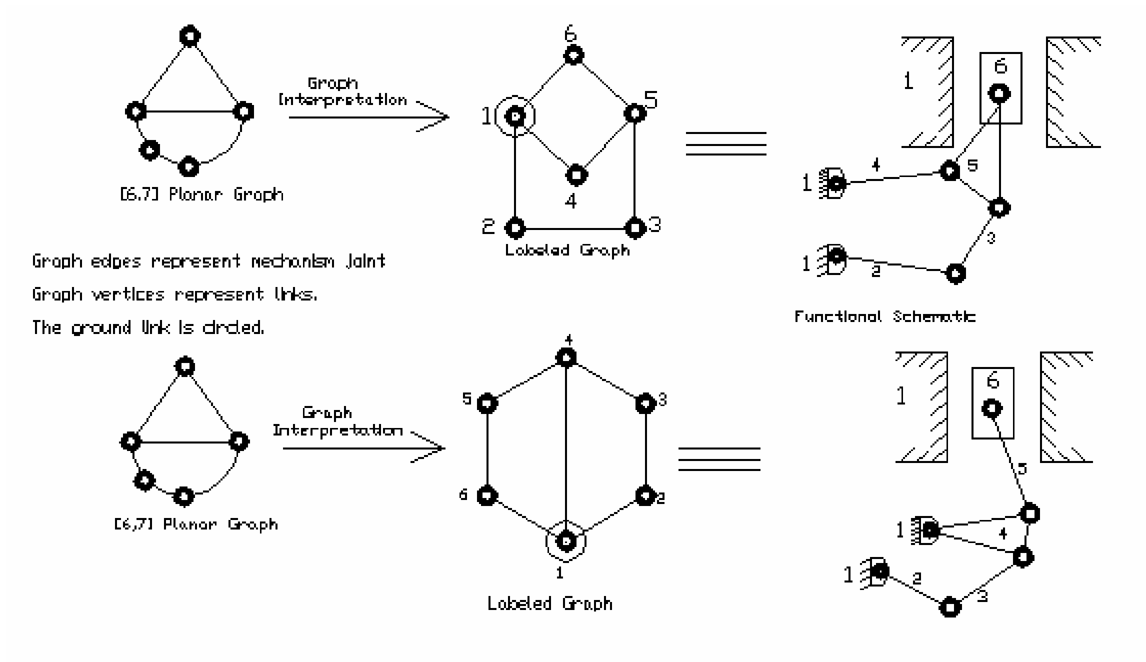


FIG. 3.2 GRAPH AND FUNCTIONAL SCHEMATIC REPRESENTATION OF TWO DIFFERENT 6-LINKS AND 7-JOINT MECHANISM

A Graph Theory has been applied to the structural analysis and synthesis of mechanisms for many years, and has proven to be an effective and systematic approach in the search of new mechanisms. An important step in the kinematic mechanism synthesis process is to identify Graph Isomorphism, while generating new mechanisms.

3.3 A GRAMMAR BASED METHOD FOR STRUCTURE SYNTHESIS [33] ---

A general grammar coupled with an isomorphism detection grammar can replace the Systematic Method's algebraic approach to atlas generation. Any atlas of [n, j] graphs, where n = No. of links and j = No. of joints in the mechanism, can be created with appropriate automation of grammar generation. After eliminating isomorphic graphs (structurally similar graphs), the generated set corresponds to the planar graph atlas. In the proposed graph grammar method, general graph grammar is used to generate all feasible [n, j] planar graphs representing the connectivity between mechanism links. A standard linear algorithm has been converted to a graph grammar and is used for detection of isomorphs.

3.3.1 GENERAL MECHANISM GRAMMAR ---

A grammar is a formal mathematical construct consisting of a set of productions or rules, a set of symbols, and an initial symbol or symbol set. Grammar rules manipulate an initial symbol into a set of symbols, creating a meaningful expression. Graph grammar rules create structures of vertices (also called nodes) linked together by edges. Expressions representing mechanism structures are links and joints represented by the vertices and edges of a graph.

Grammar labeling conventions are designed according to the complexities of the rule base. In mechanism grammar, vertices are labeled by two integers. The first is the label assigned to the vertex and the second is the degree of the vertex. Edges have labels that are shaded in diagrams. Each vertex is labeled "1" and each edge-half is labeled "2". Every vertex and edge half added as a result of the mechanism rules has these labels too. The labels are untouched by the general mechanism grammar rules and are modified only as isomorphism detection grammar rules are applied.

Here grammar's start symbol is the (3, 3) graph shown in Fig. 3.3(a). The graph has the overall label [3, 1, N, L]. The number 3 indicates that the number of vertices is 3 and 1 indicates that the number of loops in the graph is one. Edge labels are shown in graphs of Fig. 3.3(a) and 3.3(b) and suppressed in the rest of Fig.3.3 Vertices and parallel loops are added as a result of the general mechanism rules shown in Fig.3.3. To begin the design process, the designer determines the number of loops and vertices needed in the graph representation of the mechanism. Loop and vertex values are assigned to the initial graph's label. 'L' is the desired number of loops and 'N' is the desired number of vertices. An example of the application of the General Mechanism Grammar rules for (5, 6) graphs is given in Fig. 3.4. For termination, N=5 and L=2. For the sake of legibility, some self evident labels are not shown.

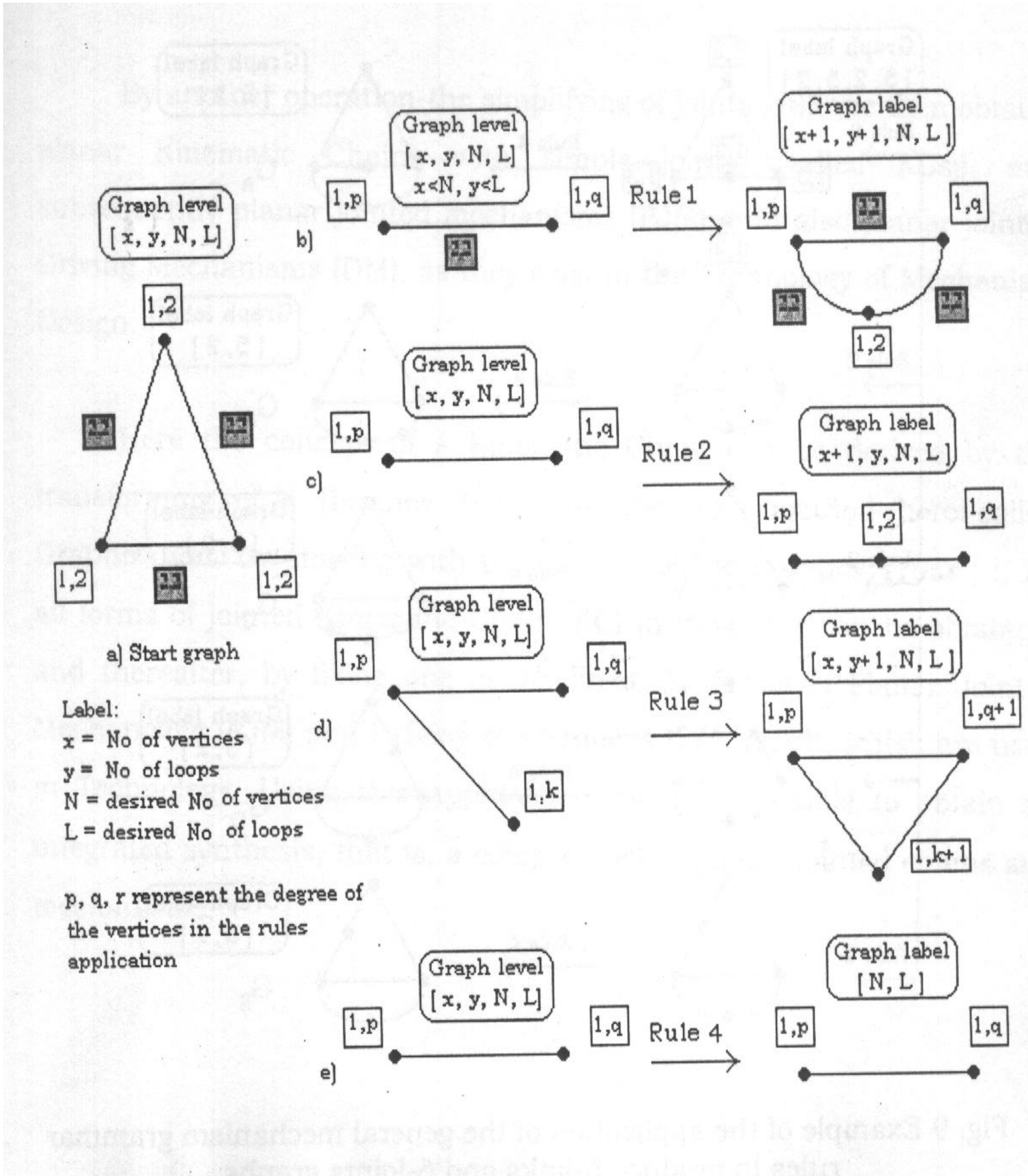


FIG.3.3 GENERAL MECHANISM GRAMMAR

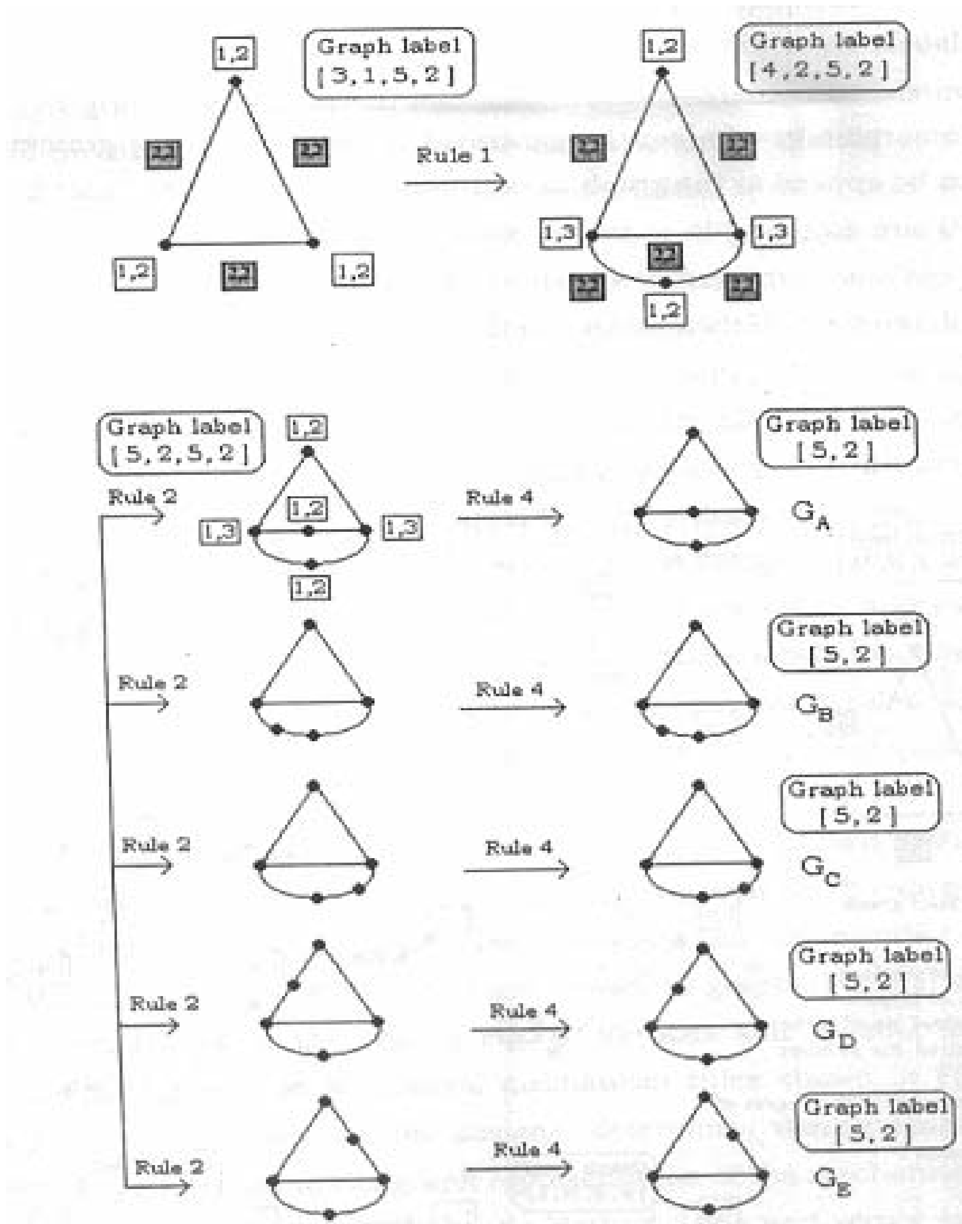


FIG.3.4 EXAMPLE OF THE APPLICATION OF THE GENERAL MECHANISM GRAMMAR RULES TO PRODUCE (5, 6) GRAPHS

Isomorphic graphs can be generated because the same grammar rules can be applied to the graph in non-unique ways. Graphs G_B and G_E of Fig. 3.4 are isomorphic if there exists a one-to-one mapping of the vertices of G_E onto the vertices of G_B such that the two vertices of G_E are adjacent if and only if their images in G_B are adjacent.

3.3.2 ISOMORPHISM DETECTION–AN INHERENT PROBLEM IN GRAPH-BASED STRUCTURE SYNTHESIS ---

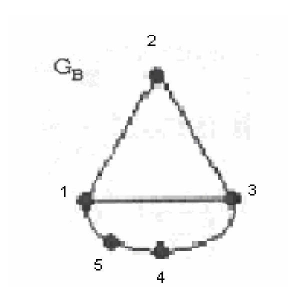
Of the five graphs generated in Fig. 3.4, only graphs G_A and G_B are distinct. Graphs G_C, G_D and G_E are isomorphs of G_A and G_B. The presence of isomorphs in an atlas is undesirable because it would lead designers to the exploration of non-unique options. As the complexity of the desired mechanism increases, the problem of identifying isomorphic graphs increases.

There are many approaches to detecting isomorphic graphs. The isomorphism detection approach taken by the graph theoretical researchers has been largely algorithmic. Triconnected graphs (those in which there are at least three paths connecting any two vertices) have a unique embedding on a sphere. Using this fact, Weinberg presented an O ($|V|^2$) algorithm for isomorphism detection of triconnected planar graphs. Here, V is the set consisting of the vertices of both graphs under examination for isomorphism. The result has been extended to arbitrary planar graphs and improved to O (|V| log |V|) steps by Hopcroft and Tarjan [2]. Then they presented a linear asymptotic growth rate algorithm.

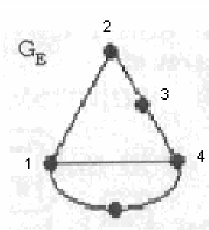
The approach to isomorphism detection adopted by mechanism researchers has largely dealt with the permutation matrix, P. Two graphs with adjacency matrices A and A* are isomorphic if a permutation matrix P can be found such that

$$A^* = P^T A P \qquad \text{---------------------------(3)}$$

The adjacency matrix A = $[a_{i,j}]$ describes the edge connectivity of a graph. Matrix terms $a_{i,j}$ = 1, if vertex i is adjacent to vertex j and $a_{i,j}$ = 0, otherwise. The adjacency matrix for graphs G_B and G_E in Fig.4 are given by:



$$A_B = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \qquad \text{----------------(4)}$$



$$A_E = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{pmatrix} \qquad \text{---------------(5)}$$

For graphs G_B and G_E, the permutation matrix is P_{BE} is:

$$P_{BE} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \qquad \text{-------------------------(6)}$$

Since in general there are *n!* choices for the permutation matrix P, constructing it by trial and error is not a practical approach.

A necessary but not sufficient condition for isomorphism is that both graphs must share the same linkage characteristic polynomial, p(x). The linkage characteristic polynomial is the characteristic polynomial of the adjacency matrix found by solving:

$$p(x) = |xI - A| \qquad \text{-------------------(7)}$$

Uicker and Raicu [3] developed a computational method for the derivation of the linkage characteristic polynomial. Yan and Hall [5, 6] developed a set of rules for determination of the coefficients of the polynomial by inspection. Even in the case of small graphs, finding coefficients for the characteristic polynomial is not a trivial task. The polynomial for the graphs G_B and G_E of the previous isomorphism example is as follows:

$$p_{BE}(x) = x^5 - 6x^3 - 2x^2 + 4x \qquad \text{--------------------(8)}$$

Rather than deriving and then comparing characteristic polynomials, Tsai developed a random number technique to determine if two polynomials are equivalent. However, it is possible for two, non-isomorphic graphs to have the same characteristic polynomial, limiting the effectiveness of these methods.

Additional screening conditions for detection of isomorphic structures can be used. Such conditions include number of joints, number of links, and link assortments. Kinematic structures that differ in these screening conditions cannot be isomorphic to each other.

3.4 GRAMMAR BASED REDUCTION RULES FOR ISMORPHISM DETECTION ---

Grammar rules for the detection of isomorphism have been adapted for application to the set of graphs generated by the mechanism graph grammars presented here. These grammar rules are derived from the linear time algorithm for detecting isomorphism of planar graphs by graph theory researchers. In graph grammar adaptation, the graph reduction rules are applied in a systematic fashion, condensing nodes and edges into labels. Priority ordering of reduction rules also insures a canonical form for the graph at each step in the process. There is a strict decrease in the complexity of the graph as measured by the sum of number of edges and vertices. The rules are applied in the priority in which they are listed in Tables 3.1 and 3.2.

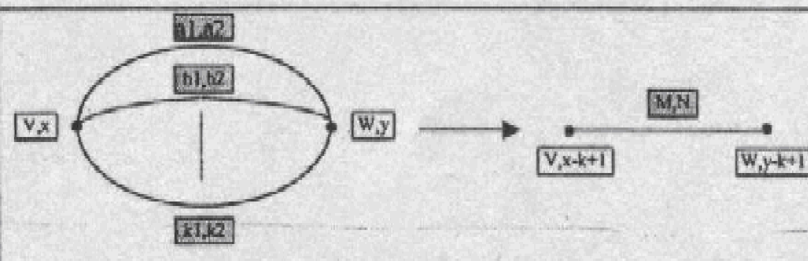
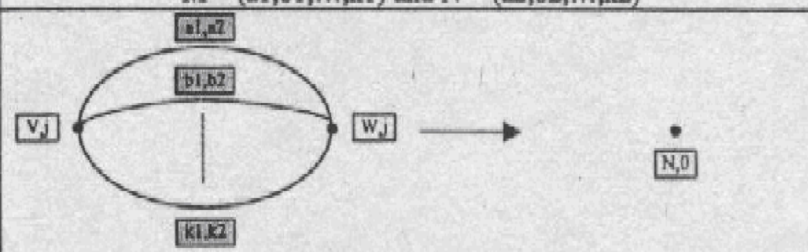
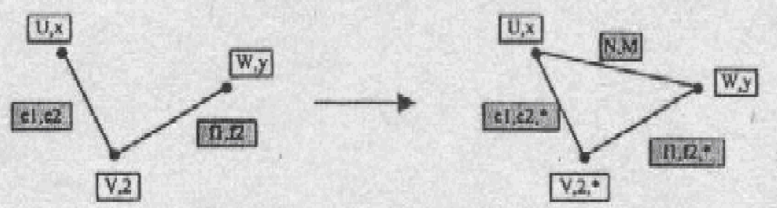
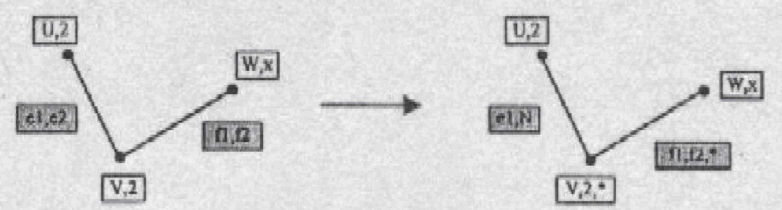
Rule Name	Reduction Rule Schema
Rule A1 Remove Clumps V or W connected to a vertex in addition to each other.	 M = (a1,b1,…,k1) and N = (a2,b2,…,k2)
Rule A2 Remove skeins Where j > 1, V and W not connected to any other nodes.	 N = {(V,a1,a2,W),(V,b1,b2,W),…,(V,k1,k2,W)}
Rule B1 Removing isolated low degree vertices In this case, V has degree 2, x > 2, and y > 2. Similar rules are used when V's degree exceeds 2.	 M = (V,f1,f2) and N = (V,e2,e1)
Rule B2 Removing vertices with non-degree matching neighbors Here x > 2. Similar rules are used when V's degree exceeds 2.	 N = (f2,f1,V,e2)

TABLE 3.1 ISOMORPHISM GRAMMAR REDUCTION RULES

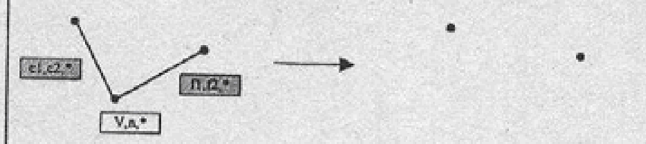
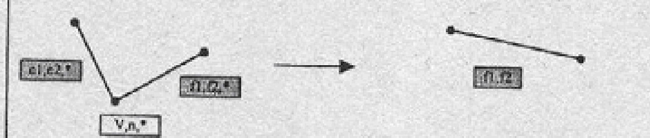
Rule Name	Reduction Rule Schema
Rule X1 Removing vertex and edges	
Rule X2 Deleting vertex and preserving edge	

TABLE 3.2 ISOMORPHISM GRAMMAR REDUCTION RULES

When no further reduction rule is applicable, the graphs that can exist are the five regular polyhedral graphs or a trivial graph consisting of a single vertex and corresponding label sets. The graphs are then tested for isomorphism. First, canonical forms are compared. If the forms are different, the starting graphs are not isomorphic. If the form is the same, isomorphism is tested by matching of label sets. If the sets are identical, the graphs are isomorphic.

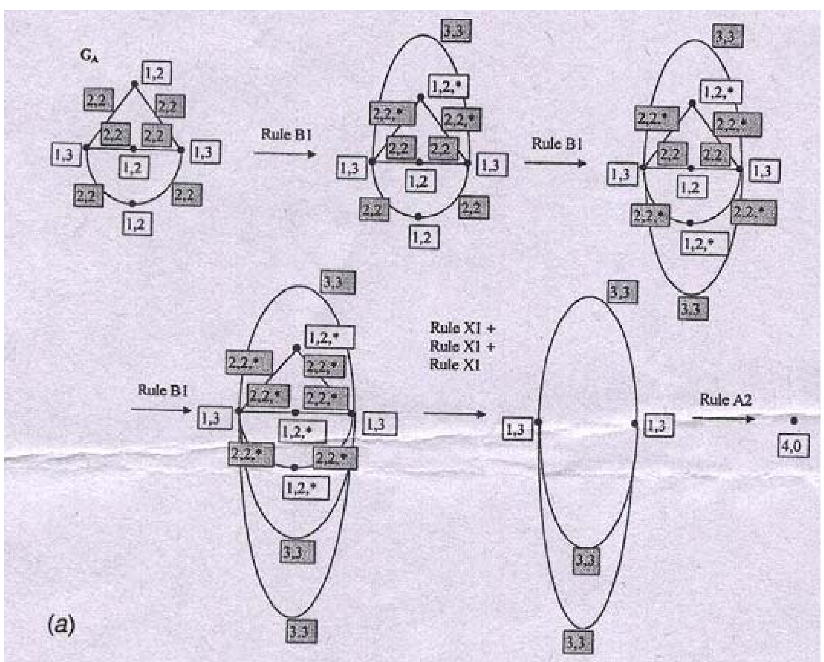


FIG. 3.5 (a) ISOMORPHISM DETECTION GRAMMAR APPLIED TO GRAPH G_A OF FIG. 3.4

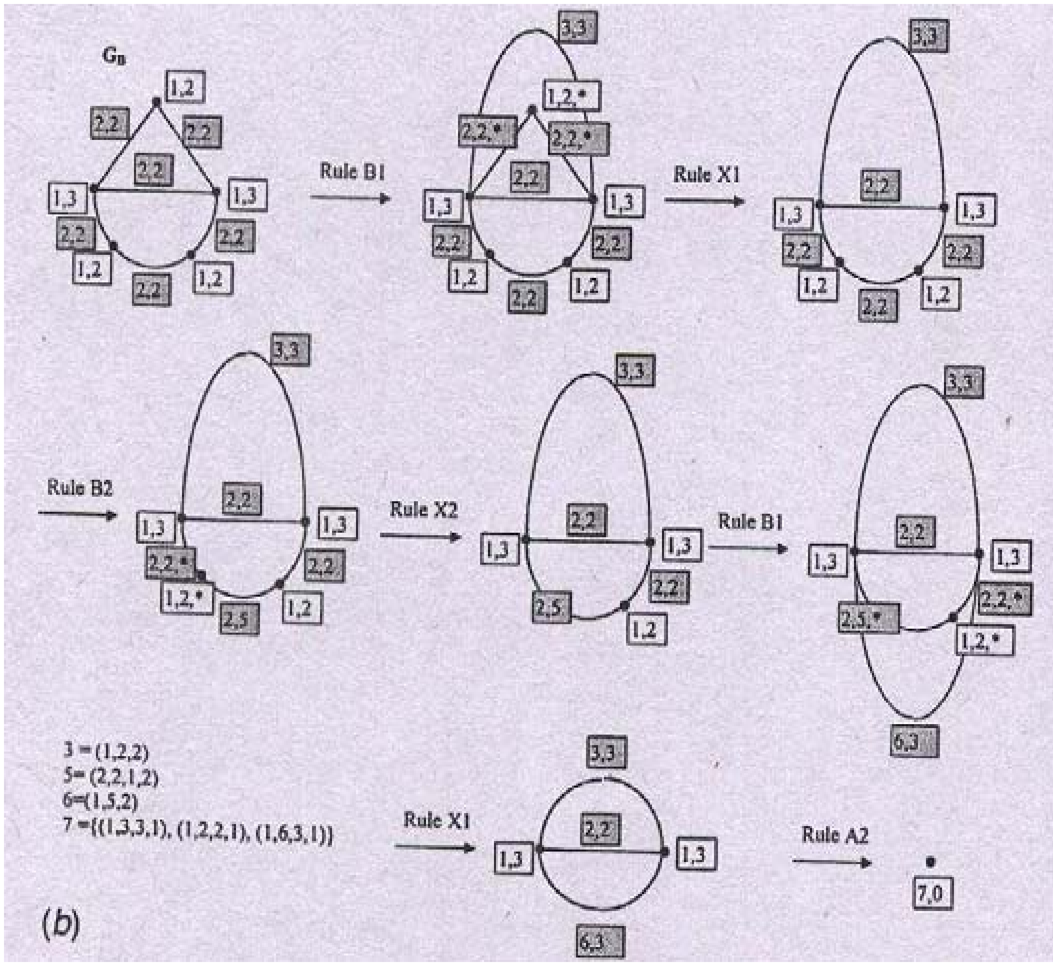


FIG. 3.5 (b) ISOMORPHISM DETECTION GRAMMAR APPLIED TO GRAPH G_B OF FIG. 3.4

Set Integer Label	Set or Vector Assigned to the Integer Level
3	(1,2,2)
4	{(1,3,3,1),(1,3,3,1),(1,3,3,1)}
5	(2,2,1,2)
6	(1,5,2)
7	{(1,3,3,1),(1,2,2,1),(1,6,3,1)}

TABLE 3.3: SET AND VECTOR LABEL DECRIPTIONS USED IN ISOMORPHISM DETECTION GRAMMAR APPLICATION IN FIGURES 5(a) AND 5(b)

Set and vector partitioning algorithms that create graph labels are an integral part of the isomorphism detection method. The set-partitioning algorithm assigns unique integers to a set of integers. Identical sets generated during the repeated application of a single grammar rule are assigned a unique integer. Identical sets generated by different rules are assigned different integers and identical sets generated by the same rule but in different traversals of the prioritized grammar rules are also assigned different integers. The purpose of such an assignment is to ensure a canonical form after every reduction. The vector-partitioning algorithm is very similar to the set-partitioning algorithm. It assigns unique integers to vectors (ordered sets).

Tables 3.1 and 3.2 describe the grammar reduction rules. The notation in the tables is more general than required for the current application. “V”, “W” and “U” represent the first label of the graph vertices. (In this application the first label will be “1”). The second vertex label represents the degree of the vertex and the variables used for this purpose are “x”, “y” and “j”. Edges have two labels, one for each half edge. Here they are designated by “a1, a2”, “b1, b2” and so on for different edges in the same graph. In this application, the edges are generated with the label “2, 2”. The action of the grammar rules of this section will change those labels using integers assigned by the rule. In the tables, “M” and “N” are these integer variables. Lastly, the label “*” added to a vertex or edge is a marker that indicates it is ready for removal by rules X1 or X2 shown in Table 3.2.

The Isomorphism Detection Grammar rules are demonstrated in Fig. 3.5 for graphs G_A and G_B of Fig. 3.4. Before the application of the rules, each vertex has a label of “1” and each half-edge has a label of “2”. A second label is attached to each vertex indicating the degree of the vertex. In Figs. 3.5(a) and 3.5(b), Isomorphism Detection Grammar rules are used to transform graphs G_A and G_B to reduced canonical forms. After the reductions, the integers describe sets and vectors as shown in Table 3.3. It can clearly be seen that G_A and G_B are not isomorphic to each other.

3.5 CONCLUSIONS ---

Thus this graph grammar methodology for structure synthesis of mechanism provides four improvements over the Systematic Method:

- Simplifying the method by elimination of algebraic transformations in favor of direct graph manipulations by grammar rules.
- Adding rigor to the method by applying grammar formalism.
- Providing automation of the key generation steps of the method.

- Enabling the application of the method to mechanism schematics represented by non planar graphs (this task is currently outside the boundaries of the present Systematic Method).

This grammar methodology will automatically meet the algebraic method requirements of the Systematic Method for mechanism synthesis and can generate atlases of higher-order planar graphs than are currently available. Furthermore, there is an initial verification that an isomorphism-detection algorithm is well suited to grammar rewriting and is able to detect isomorphic graphs.

3.6 MECHANISM ANALYSIS ---

Once a tentative mechanism design has been synthesized, it must then be analyzed. A principal goal of kinematic analysis is to determine the accelerations of all the moving parts in the assembly.

3.6.1 POSITION ANALYSIS ---

Dynamic forces are proportional to acceleration, from Newton's second law. The need is to know the dynamic forces in order to calculate the stresses in the components. The design engineer must ensure that the proposed mechanism or machine will not fail under its operating conditions. Thus the stresses in the materials must be kept well below allowable levels. To calculate the stresses, the accelerations need to be known. In order to calculate the accelerations, firstly the positions of all the links or elements in the mechanism for each increment of input motion must be found, and then differentiate the position equations versus time to find velocities, and then differentiate again to obtain the expressions for acceleration. For example, in a simple Grashof four-bar linkage, everyone probably wants to calculate the positions, velocities, and accelerations of the output links (coupler and rocker) for perhaps every two degrees (180 positions) of input crank position for one revolution of the crank.

3.6.2 VELOCITY ANALYSIS ---

Once a position analysis is done, the next step is to determine the velocities of all links and points of interest in the mechanism. It needs to be known the velocities in mechanism or machine, both to calculate the stored kinetic energy from $mV^2/2$, and also as a step on the way to the determination of the link's accelerations which are needed for the dynamic force calculations. Many methods and approaches exist to find velocities in mechanisms.

3.6.3 ACCELERATION ANALYSIS ---

Once a velocity analysis is done, the next step is to determine the accelerations of all links and points of interest in the mechanism or machine. Acceleration needs to be known to calculate the dynamic forces from $F = ma$. The dynamic forces will contribute to the stresses in the links and other components. Many methods and approaches exist to find accelerations in mechanisms.

CHAPTER -4
WHAT IS ISOMORPHISM?

4.1 ISOMERS ---

The word **Isomer** is from the Greek and means having equal parts. Isomers in chemistry are compounds that have the same number and type of atoms but which are interconnected differently and thus have different physical properties. Linkage isomers are analogous to these chemical compounds in that the links (like atoms) have various nodes (electrons) available to connect to other links' nodes. The assembled linkage is analogous to the chemical compound. Depending on the particular connections of available links, the assembly will have different motion properties. The number of isomers possible from a given collection of links (as in any row of Table 4.1) is far from obvious. In fact the problem of mathematically predicting the number of isomers from all link combinations has been a long-unsolved problem. Many researchers have spent much effort on this problem with some recent success. Table 4.2 shows the number of valid isomers found for one-DOF mechanisms with revolute pairs, up to 12 links.

TOTAL LINKS	LINK SETS				
	BINARY	TERNARY	QUATERNARY	PENTAGONAL	HEXAGONAL
4	4	0	0	0	0
6	4	2	0	0	0
6	5	0	1	0	0
8	7	0	0	0	1
8	4	4	0	0	0
8	5	2	1	0	0
8	6	0	2	0	0
8	6	1	0	1	0

TABLE 4.1: 1 DOF PLANAR MECHANISMS WITH REVOLUTE JOINTS AND UP TO 8 LINKS

LINKS	VALID ISOMERS
4	1
6	2
8	16
10	230
12	6856

TABLE 4.2: NUMBER OF VALID ISOMERS

4.2 ISOMORPHIM AMONG GRAPHS ---

Two graphs, say G_1 and G_2 are isomorphic when there is a one-to-one correspondence between the vertices of G_1 and those of G_2, with the property that the number of edges joining any two vertices of G_1 is equal to the number of edges joining the corresponding vertices of G_2. So the identification of mechanisms is rendered to the identification of isomorphic graphs.

As shown in Fig. 3.4 out of five graphs G_A, G_B, G_C, G_D and G_E; only two graphs G_A and G_B are distinct. Graphs G_C, G_D and G_E are isomorphs of G_B. Graphs G_B and G_E are isomorphic if there exists a one-to-one mapping of the vertices of G_E onto the vertices of G_B such that the two vertices of G_E are adjacent if and only if their images in G_B are adjacent. Isomorphism among graphs G_B, G_C, G_D and G_E is structural similarity between any two graphs of G_B, G_C, G_D and G_E. As shown in Fig. 4.1, graphs G_B and G_D are structurally similar because graph G_D has the same structure as graph G_B and it can be drawn as graph G_B and there is one-to-one correspondence between the vertices of G_B and those of G_D.

The presence of isomorphs in an atlas is undesirable because it would lead designers to the exploration of non-unique options. As the complexity of the desired mechanism increases, the problem of identifying isomorphic graphs increases.

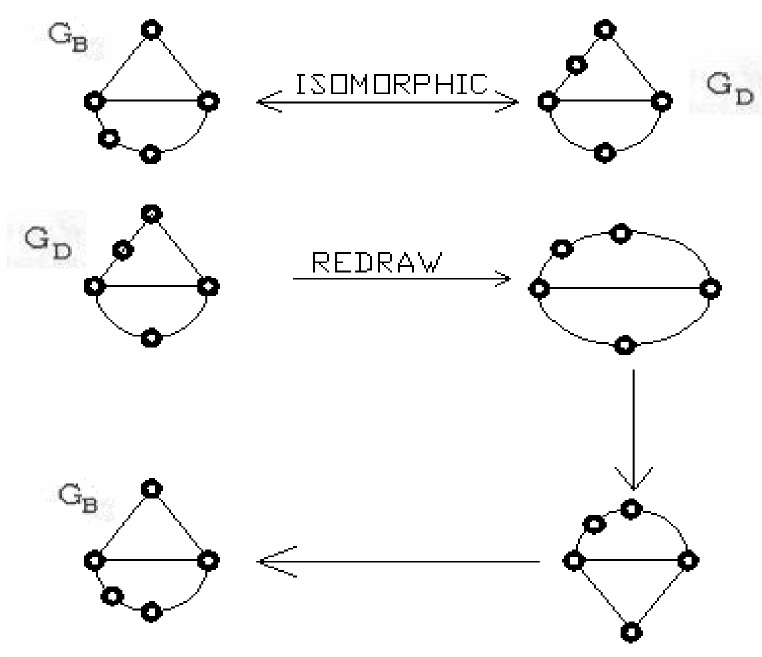


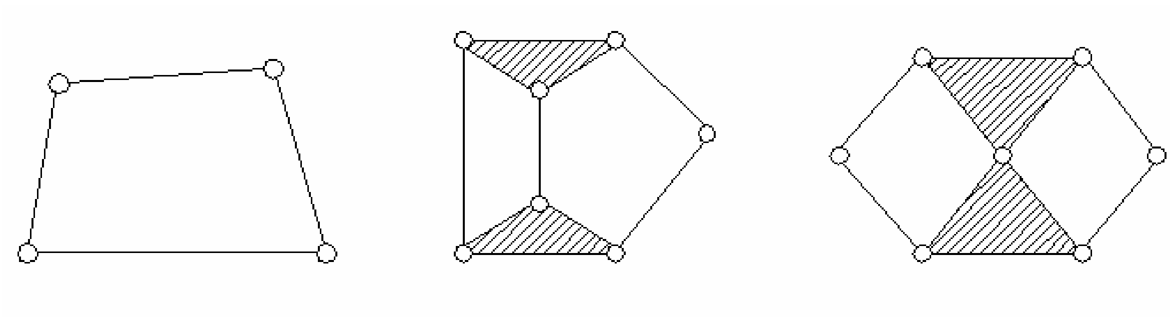
FIG.4.1 ISOMORPHIC GRAPHS OF 5-LINKS AND 6-JOINTS

4.3 ISOMORPHISM AMONG KINEMATIC CHAINS ---

If kinematic chains are isomorphic then it means that these are structurally similar i.e. there is one-to-one correspondence of vertices of two kinematic chains. Kinematic chains too are like graphs as shown in Fig. 3.4 and Fig. 4.1. After a complete process of structural synthesis when a full combination of chains comes then that includes chains of similar structure but different in appearance and these chains are known as isomorphic chains which should be discarded for getting a collection of distinct chains for creation and invention of new mechanism.

Figure 4.2 shows all the isomers for the simple cases of one DOF with 4 and 6 links. It is noted that there is only one isomer for the case of 4 links. An isomer is only unique if the interconnections between its types of links are different. That is, all binary links are considered equal. Link lengths and shapes do not figure into the Gruebler's criterion or the condition of isomorphism. The 6-link case of 4 binaries and 2 ternaries has only two valid isomers. These are known as the **Watt's chain** and the **Stephenson's chain** after their discoverers. The Watt's chain has the two ternaries directly connected, but the Stephenson's chain does not.

There is also a third potential isomer for this case of six links, as shown in Figure 4.3, but it fails the test of distribution of degree of freedom, which requires that the overall DOF (here 1) be uniformly distributed throughout the linkage and not concentrated in a sub chain. So this arrangement (Fig. 4.3) has a structural sub chain of DOF=0 in the triangular formulation of the two ternaries and the single binary connecting them. This creates a truss, or delta triplet. The remaining three binaries in series form a four bar chain (DOF=1) with the structural sub chain of the two ternaries and the single binary effectively reduced to a structure which acts like a single link. Thus this arrangement has been reduced to the simpler case of the four bar linkage despite its six bars. This is an **invalid isomer** and is rejected.



(a) The only four bar isomer (b) Stephenson's six bar isomer (c) Watt's six bar isomer

FIG. 4.2 ALL VALID ISOMERS OF THE FOUR BAR AND SIX BAR LINKAGES

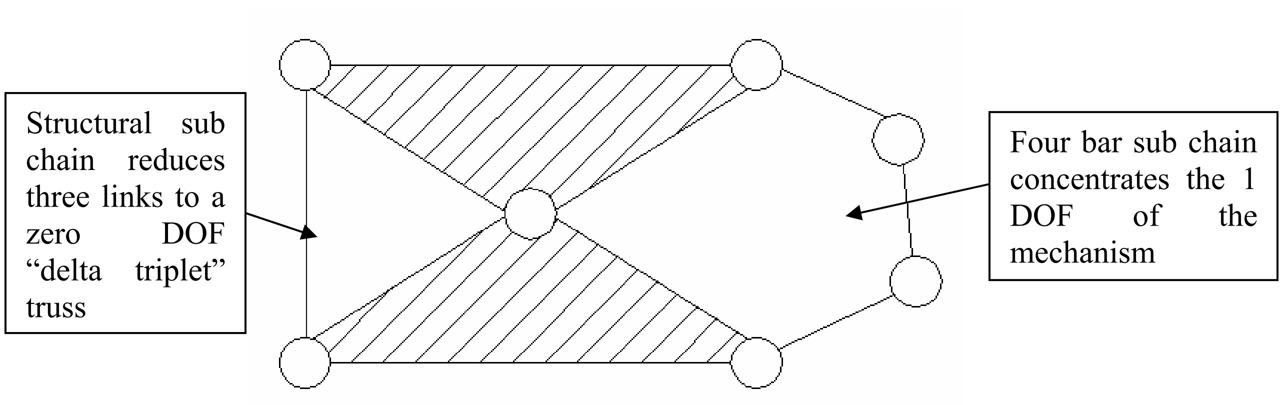


FIG. 4.3 AN INVALID SIX BAR ISOMER WHICH REDUCES TO THE SIMPLER FOUR BAR

As shown in Figure 4.4 all chains are 6 links chain but chains (a) and (b) are structurally similar means they are isomorphic chains.

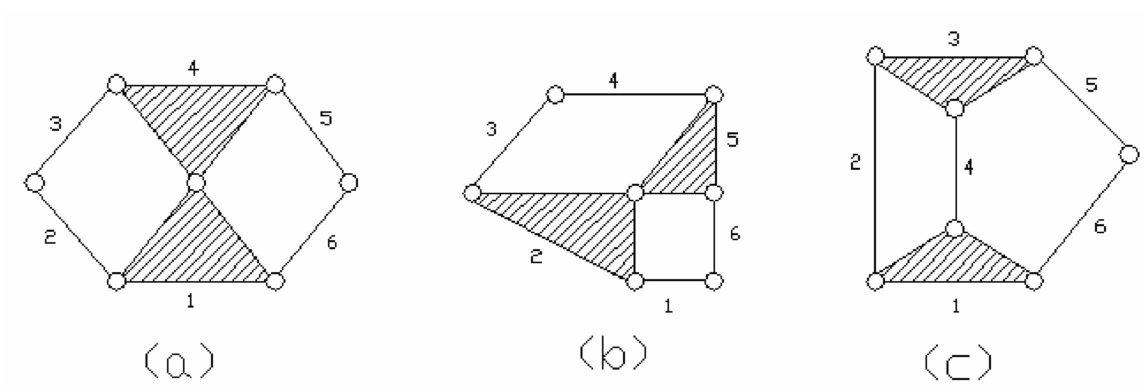


FIG. 4.4 SIX LINKS ONE DEGREE-OF-FREEDOM KINEMATIC CHAINS

4.4 ISOMORPHISM AMONG INVERSIONS OF A GIVEN CHAIN ---

For a given kinematic chain, inversions can be obtained by fixing every link of given kinematic chain. Isomorphism among inversions of a given chain shows their structural similarity i.e. there is one-to-one correspondence of their vertices. As shown in Figure 4.5, Watt chain has 6 links and by fixing each link, 6 different inversions can be obtained but some of them may be structurally similar. Thus test of Isomorphism between inversions of a given chain is done for getting structurally distinct inversions.

As shown in Figure 4.6, two inversions of Watt chain, by fixing link 2 and 5 are isomorphic. One-to-one correspondence of vertices of inversion by fixing link 5 with inversion by fixing link 2 of Watt chain can be obtained by just rearranging structure of inversions by fixing link 5 as shown in Figure 4.7

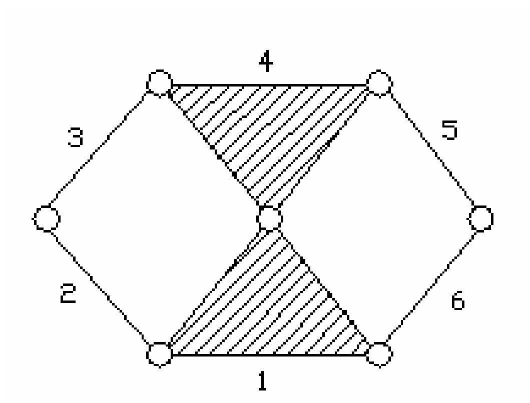


FIG. 4.5 WATT CHAIN

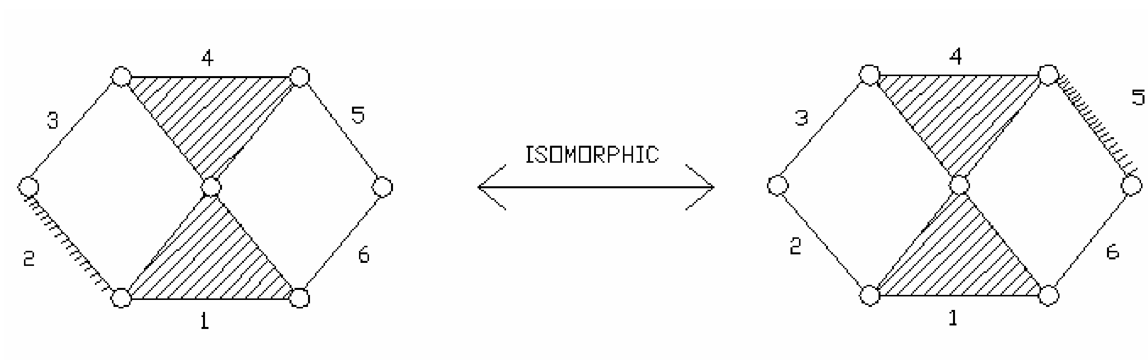


FIG. 4.6 ISOMORPHIC GRAPHS OF WATT CHAIN

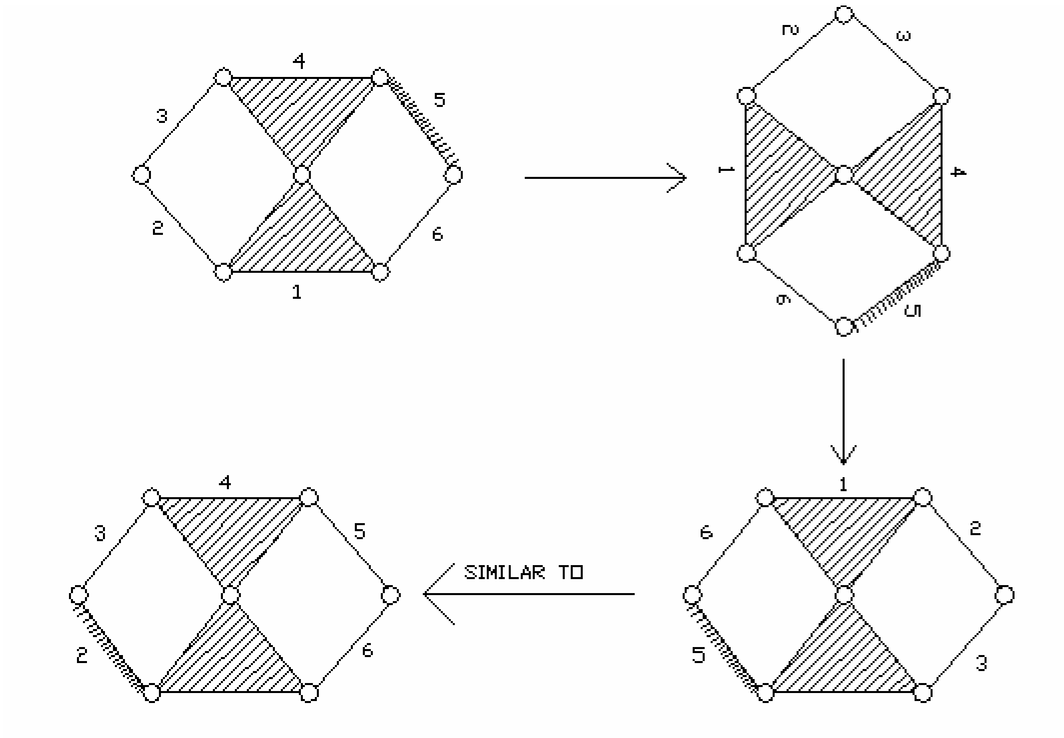


FIG. 4.7 PROOF OF ISOMORPHISM OR STRUCTURAL SIMILARITY OF GRAPHS OF FIG. 4.6

4.5 IMPORTANCE OF ISOMORPHISM IDENTIFICATION ---

Undetected isomers in a complete list of kinematic chains after structural synthesis result in duplicate solutions and an unnecessary effort of mechanism synthesis and falsely identified isomers eliminates possible candidates for new mechanisms. Thus isomorphism identification is very important for identification of isomers from the point of view of time saving and correct synthesis and analysis of mechanism kinematic chain.

4.6 METHODS OF ISOMORPHISM IDENTIFICATION ---

A number of methods have been developed in the past for detecting Isomorphism among planar kinematic chains and some of them were reliable but computationally difficult.

The methods based on characteristic polynomials [4, 5] have the disadvantage of dealing with large numericals and graph theorists noted the existence of counter examples while applying these methods.

In code approaches [7], the Max_Min code of a kinematic chain is unique and decodable. But this method requires highly sophisticated algorithms when applied to large kinematic chains.

The degree code [16] based test overcomes the drawback of Max_Min code method but at the cost of computational efficiency.

The canonical numbering scheme [10] is capable of unique relabeling of the links of a given KC. However it tends to become computationally inefficient when there exists a higher number of symmetry group elements in the KC.

The methods discussed above have a serious handicap that chain has been made, whether or not the min code method can be used to determine the number of inversions of a chain. In characteristic polynomial approach methods [4, 5], considerable amount of additional computation has to be undertaken to get this information.

Most of these demerits are overcome by Hamming code approach [14] which gives a unique string for a chain as well as for identifying distinct inversions besides being computationally efficient and a reliable method as well. But it could derive only 1828 distinct mechanisms from the 230 ten-link, single degree-of-freedom chains as against 1834 already reported in the literature. Most of the methods are generally based on link adjacencies and the connectivity of the links. Then, an invariant is developed called the total distance ranks of all the links arranged in decreasing (or increasing) order, which is based on the concept of distance [19]. This invariant, however, was not found to be sufficient for the test of isomorphism. Further work by the authors on distance concept showed that the discriminating ability of this invariant could be improved by taking into consideration the degrees of links, degrees-of-freedom and types of joints [23]. However, most of these methods require complicated process of solution and time and are difficult to grasp and utilize. In addition, there is a lack of uniqueness or it takes too much time for determining isomorphism of a kinematic chain. So researchers have emphasis on a method which is simple, efficient and reliable as well.

A number of important methods have been proposed till now for detection of isomorphism among kinematic chains and these are -----------

- Linkage Characteristic Polynomials Method
- Hamming Number Technique
- Degree Code Method
- Link Adjacency Table Method
- Distance Concept
- Neural Network Approach
- Fuzzy Logic Approach
- Loop Based Detection Method
- Genetic Algorithm Approach
- Spanning Tree Method
- Adjacency Matrix Method
- Joint-Joint Matrix Method

Six important methods of the above have been studied in details and are discussed in the following chapters. These six methods are listed below:-

- Hamming Number Technique
- Distance Concept
- Loop Based Detection Method
- Genetic Algorithm Approach
- Adjacency Matrix Method
- Joint-Joint Matrix Method

CHAPTER – 5
HAMMING NUMBER TECHNIQUE FOR DETECTION OF ISOMORPHISM AMONG KINEMATIC CHAINS

5.1 INTRODUCTION ---

Rao, A.C. [14] introduced this *Hamming Number Technique.* In this method, a noble approach, which is both reliable and simple, is presented. Use is made of the *Hamming Number*; a concept of digital communication theory. The *Connectivity Matrix* of the various links, a matrix of zeroes and ones, is first formed and the *Hamming Number* matrix is computed. The *Link Hamming String* – which is defined as the string obtained by concatenating the *Link Hamming Number* and the frequency of individual *Hamming Numbers* in that row – is then formed. Finally the *Chain Hamming String*, defined as the string obtained by the concatenation of the *Chain Hamming Number* and the *Link Hamming String* in descending order is formed.

5.2 METHOD ---

Whether or not each link in a chain is connected directly to any other link is spelt out in *Connectivity Matrix* C (also called as an *Adjacency Matrix*), whose elements are either all zero or one. C is of size $n \times n$, for a chain of n links and the value of each element in C is decided as follows:-

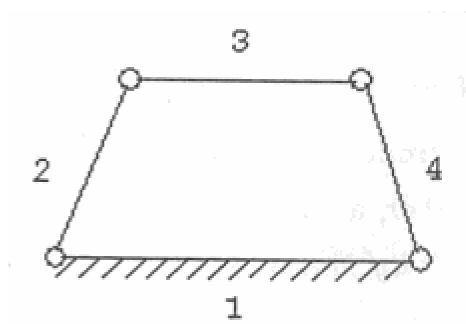


FIG.5.1 FOUR – LINKS ONE DEGREE-OF-FREEDOM KINEMATIC CHAIN

If link i is directly connected to link j then the element in the i^{th} row and j^{th} column of C is 1 and if the links are not connected, the element is 0. Obviously C is always square, symmetric and has zeroes all along its leading diagonal. The four bar chain of Fig. 5.1 has Connectivity matrix C as shown below:

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

It may be observed that every row represents the particular link and every element in that row indicates whether or not that link directly connected to other links. Also, every element of the connectivity matrix informs us merely whether or not two links are directly connected. But relation between two links is much more than this single bit of information. How they relate to every other link of the chain decide total disposition between the two links. For example, in Fig. 5.1, links 1 and 2 are directly connected. Link 3 is connected to link 2 but to link 1, link 4 is connected to link 1 but not to link 2. Hence the total relationship between links 1 and 2 has to take all these facts into consideration. This is precisely what the *Hamming Number* of two links quantifies. It measures the extent by which two links differ from each other.

5.2.1 THE HAMMING NUMBER AND HAMMING NUMBER MATRIX ---

The *Hamming Number* for any two codes each with *n* digits has been defined as the total number of bits in which the two codes differ. Applying this definition to the rows *i* and *j* of C, it becomes:

$$h_{ij} = \sum_{k=1}^{n} S_{k},$$

where $S_k = 0$ if $C_{ik} = C_{jk}$
$= C_{ik} + C_{jk}$ otherwise

The *Hamming Number* between any two rows of size *n* can be any positive integer from *n* (if all the digits are different), down to 0 (if the two rows are exactly identical).

To put this in plain English, *Hamming Number* of any two rows is the sum of all the scores for each of the columns of those rows. A score, in turn, is defined as (i) the sums of the individual elements if they are unequal and (ii) zero if the elements are equal. Thus if elements are (0 and 0) or (1 and 1), the score is 0. But if they are (1 and 0) or (0 and 1), the score is (0+1=1+0=1). In Boolean algebra terminology this score goes by the name XOR for exclusive OR.

Applying this definition to the chain of Fig.5.1, it is-

$h_{12} = h_{14} = 1+1+1+1 = 4$ whereas $h_{13} = 0+0+0+0 = 0$

In the same manner *Hamming Number* for all other pairs of rows are calculated and one obtains the *Hamming Number Matrix* as $H = [h_{ij}]$. Thus for the four-bar chain of Fig. 5.1, the *Hamming Matrix* is:

$$\mathbf{H} = \begin{pmatrix} 0 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \\ 4 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \end{pmatrix}$$

The *Hamming* Matrix is also a square, symmetric matrix and has zeroes all along its leading diagonal. However unlike the *Connectivity Matrix* it contains digits which could be larger than unity.

5.2.2 DEFINITIONS OF TERMINOLOGY ---

- **Link Hamming Number (L.H.N.)** for any link *i* is the sum of all the elements in the i^{th} row of the Hamming Matrix.Thus the link *Hamming Number* for link 1 of Fig. 5.1 is 8 (=0+4+0+4), so also for all the other links.
- **Chain Hamming Number (C.H.N.)** for any chain is the sum of the entire link *Hamming Numbers* of that chain. It also works out to be the sum of all the elements of the *Hamming Matrix* for that chain.The *Chain Hamming Number* for the four-bar chain is 32 (=8+8+8+8).
- **Link Hamming String** for any link *i* is the string obtained by concatenating (a) the link *Hamming Number* of *i* with (b) the frequency of occurrence, of all the integers from *n* down to *0*, in the *Hamming Numbers* of that row *i.* For the example considered, the *link Hamming String* for link 1 is 8, 20002, implying that link *Hamming Number* is 8 and comprises of two 4s, no 1s and two 0s.

The *Link Hamming Strings* for the four links are:

1: 8, 20002

2: 8, 20002

3: 8, 20002

4: 8, 20002

It may be observed that all the four links 1,2,3,4 have the same *Link Hamming Number.*

- **Chain Hamming String** is defined as the concatenation of the (i) Chain Hamming Number and (ii) Link Hamming String placed in decreasing order of magnitude. For the example being considered here, the Chain Hamming Number is 32 and the *Chain Hamming String* is:

  32; 8, 20002; 8, 20002; 8, 20002; 8, 20002

5.3 DETECTION OF ISOMORPHISM AMONG KINEMATIC CHAINS ---

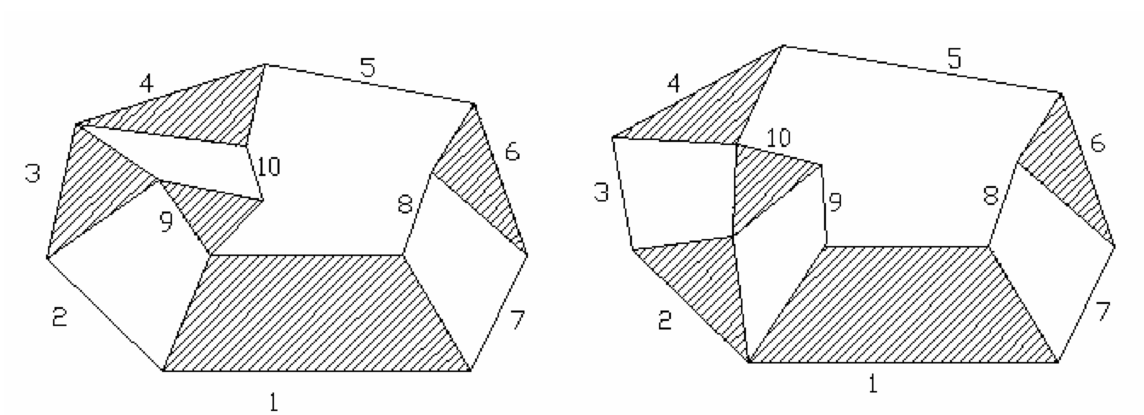
The *Chain Hamming String* is a definitive test for isomorphism among the kinematic chains. This implies that if two chains are known to be isomorphic, their *Chain Hamming String* should be identical and vice-versa. Secondly, if two chains are non-isomorphic their *Chain Hamming String* should differ at some position or other.

5.4 DETECTION OF ISOMORPHISM AMONG INVERSIONS ---

The *Link Neighbourhood String* is a definitive test for isomorphism among inversions of a given chain. The *Link Neighbourhood String* of the various links has the potential to disclose how many distinct inversions can be obtained from a given chain and by fixing which links these inversions are possible. If *Link Neighbourhood Strings* of two links are identical then the inversions are identical structurally.

5.5 <u>ILLUSTRATIVE EXAMPLE -1 (SINGLE DEGREE FREEDOM CHAINS)</u>

The example concerns two kinematic chains with 10 bars and single-degree of freedom as shown in Fig.5.2 (a) and Fig. 5.2 (b). The task is to examine whether these two chains are isomorphic.



(a) **(b)**

FIG. 5.2 TEN-BAR KC, SINGLE-DEGREE OF FREEDOM

CALCULATIONS FOR FIG. 5.2(a)

STEP 1 --- CONNECTIVITY MATRIX

C =

Link → / Link ↓	1	2	3	4	5	6	7	8	9	10
1	0	1	0	0	0	0	1	1	1	0
2	1	0	0	0	0	0	0	0	0	0
3	0	1	0	1	0	0	0	0	1	0
4	0	0	1	0	1	0	0	0	0	1
5	0	0	0	1	0	1	0	0	0	0
6	0	0	0	0	1	0	1	1	0	0
7	1	0	0	0	0	1	0	0	0	0
8	1	0	0	0	0	1	0	0	0	0
9	1	0	1	0	0	0	0	0	0	1
10	0	0	0	1	0	0	0	0	1	0

STEP 2 --- HAMMING MATRIX

H =

Link → / Link ↓	1	2	3	4	5	6	7	8	9	10	L.H.N.
1	0	6	3	7	6	3	6	6	7	4	48
2	6	0	5	3	4	5	2	2	1	4	32
3	3	5	0	6	3	6	5	5	6	1	40
4	7	3	6	0	5	4	5	5	2	5	42
5	6	4	3	5	0	5	2	2	5	2	34
6	3	5	6	4	5	0	5	5	6	5	44
7	6	2	5	5	2	5	0	0	3	4	32
8	6	2	5	5	2	5	0	0	3	4	32
9	7	1	6	2	5	6	3	3	0	5	38
10	4	4	1	5	2	5	4	4	5	0	34

∑ L.H.N. = C.H.N. = 376

STEP 3 --- LINK HAMMING STRING

Link	Link Hamming String
1	48, 10021042
2	32, 11212210
3	40, 11020330
4	42, 10111411
5	34, 10311310
6	44, 10011520
7	32, 20211310
8	32, 20211310
9	38, 11120221
10	34, 11104300

STEP 4 --- CHAIN HAMMING STRING (C.H.S.)

C.H.S. = 376; 48, 10021042; 44, 10011520; 42, 10111411; 40, 11020330; 38, 11120221; 34, 11104300; 34, 10311310; 32, 20211310; 32, 20211310; 32, 11212210

STEP 5 --- LINK NEIGHBOURHOOD STRINGS

Links	Adjacent Links	L.H.S.	Link Neighbourhood Strings
1	2, 7, 8, 9	48, 10021042	32, 11212210; 32, 20211310; 32, 20211310; 38, 11120221
2	1, 3	32, 11212210	48, 10021042; 40, 11020330
3	2, 4, 9	40, 11020330	32, 11212210; 42, 10111411; 38, 11120221
4	3, 5, 10	42, 10111411	40, 11020330; 34, 10311310; 34, 11104300
5	4, 6	34, 10311310	42, 10111411; 44, 10011520
6	5, 7, 8	44, 10011520	34, 10311310; 32, 20211310; 32, 20211310
7	1, 6	32, 20211310	48, 10021042; 44, 10011520
8	1, 6	32, 20211310	48, 10021042; 44, 10011520
9	1, 3, 10	38, 11120221	48, 10021042; 40, 11020330; 34, 11104300
10	4, 9	34, 11104300	42, 10111411; 38, 11120221

Here the Link Neighbourhood Strings of [7, 8] are identical; so taking one link from identical strings together, total nine inversions are possible those are structurally different for the chain in Fig.5.2 (a).

CALCULATIONS FOR FIG. 5.2(b)

STEP 1 --- CONNECTIVITY MATRIX

C =

Link → / Link ↓	1	2	3	4	5	6	7	8	9	10
1	0	1	0	0	0	0	1	1	1	0
2	1	0	1	0	0	0	0	0	0	1
3	0	1	0	1	0	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0	1
5	0	0	0	1	0	1	0	0	0	0
6	0	0	0	0	1	0	1	1	0	0
7	1	0	0	0	0	1	0	0	0	0
8	1	0	0	0	0	1	0	0	0	0
9	1	0	0	0	0	0	0	0	0	1
10	0	1	0	1	0	0	0	0	1	0

STEP 2 --- HAMMING MATRIX

H =

Link → / Link ↓	1	2	3	4	5	6	7	8	9	10	L.H.N.
1	0	7	4	7	6	3	6	6	6	3	48
2	7	0	5	2	5	6	3	3	1	6	38
3	4	5	0	5	2	5	4	4	4	1	34
4	7	2	5	0	5	4	5	5	3	6	42
5	6	5	2	5	0	5	2	2	4	3	34
6	3	6	5	4	5	0	5	5	5	6	44
7	6	3	4	5	2	5	0	0	2	5	32
8	6	3	4	5	2	5	0	0	2	5	32
9	6	1	4	3	4	5	2	2	0	5	32
10	3	6	1	6	3	6	5	5	5	0	40

C.H.N. = ∑ L.H.N. = 376

STEP 3 --- LINK HAMMING STRING

Link	Link Hamming String
1	48, 10021042
2	32, 11120221
3	34, 11104300
4	42, 10111411
5	34, 10311310
6	44, 10011520
7	32, 20211310
8	32, 20211310
9	32, 11212210
10	40, 11020330

STEP 4 --- CHAIN HAMMING STRING (C.H.S.)

C.H.S. = 376; 48, 10021042; 44, 10011520; 42, 10111411; 40, 11020330; 38, 11120221; 34, 11104300; 34, 10311310; 32, 20211310; 32, 20211310; 32, 11212210

Now even a cursory glace reveals that Fig. 5.2(a) and (b) have same C.H.S., hence they are isomorphic.

5.6 CONCLUSIONS ---

Here the *Link Hamming String* of every link together with those of its neighbours is an excellent test for isomorphism among the inversions of given chains. These twin claims have been verified on a computer for all six, eight and ten-bar chains with one degree of freedom as well as ten-bar chains with three-degrees of freedom. It is felt that the greatest advantage of this method is that the *Chain Hamming String* reveals at a glance, without much additional computation, how many inversions are possible out of a given chain. But computations become very long in case of large KC as Link Hamming String and Chain Hamming String are calculated using the Connectivity Matrix, Hamming Matrix, Link Hamming Number and Chain Hamming Number. So it takes a lot of effort & time and is difficult to compute all these things.

CHAPTER – 6
COMPUTER AIDED DETECTION OF ISOMORPHISM AMONG KINEMATIC CHAINS AND MECHANISMS USING THE CONCEPT OF MODIFIED DISTANCE

6.1 INTRODUCTION ---

A computer aided method has been proposed by Yadav, J.N.; Agrawal, V.P. and Pratap, C.R. [20] for detecting isomorphism among planar kinematic chains with simple joints using a new invariant called *the arranged sequence of modified total distance ranks of all the links (ASMTDRL).* This invariant takes into account the degrees of links, and degrees-of-freedom and types of joints of the kinematic chain. Two computer aided methods have also been developed for identifying distinct mechanisms of a planar kinematic chain with simple joints.

6.2 DEFINITIONS ---

In the process of developing the new invariant for a chain, some new terms have been introduced and defined as follows:

(i) Link – Path: A link-path in a chain is an alternating sequence of distinct links and distinct joints, starting and ending with links, such that each joint connects the links preceding and following it in the sequence. For example, the sequence 1, a, 2, i, 9, h, 8 is a link-path in the chain shown in Fig. 6.1

(ii) Modified Length of a Link-Path: The modified length L of a link-path is defined as: $L = D + W$

where D and W are, respectively, the sum of degrees of all the links and the sum of the weights of all the joints in the link-path.

The degree of a link is 2 for binary link, 3 for ternary link, 4 for quaternary link and so on. The weight of a joint is defined as the sum of its degree-of-freedom and type value. The proposed type values for different types of joints are presented in Table 6.1. The type values are so chosen that no two different types of joints have the same weight. For example, the modified length of the link-path 1, a, 2, i, 9, h, 8 in Fig. 6.1, is 16.

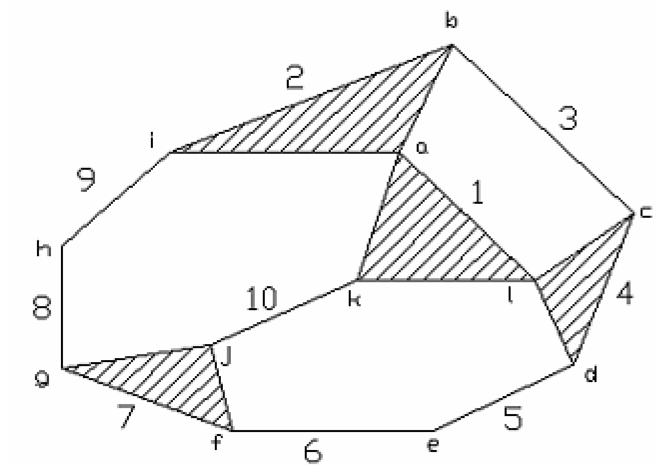


FIG. 6.1 TEN LINK, THREE DEGREE-OF-FREEDOM KINEMATIC CHAIN WITH SIMPLE JOINTS

(iii) Modified Distance between Two Links: It is defined as the modified length of the shortest link-path (based on the concept of modified length) whose terminal links are the given links.

S. No.	Joint	Degree-of-Freedom (a)	Type Value (b)	Weight (a + b)
1.	Revolute	1	1	2
2.	Prismatic	1	2	3
3.	Gear	2	2	4
4.	Cam	2	3	5
5.	Screw	1	5	6
6.	Cylindrical	2	5	7
7.	Spherical	3	5	8
8.	Planar	3	6	9

TABLE 6.1: PROPOSED TYPE VALUES FOR DIFFERENT TYPES OF JOINTS (KINEMATIC PAIRS)

(iv) Modified Link-Link Relation Matrix: The modified link-link relation matrix of a *n*-link chain is defined as an *(n x n)* matrix whose *i, j*th element, R_{ij}, is defined as:

$$R_{ij} = \begin{cases} \text{(weight of the joint between links } i \text{ and } j\text{)} + \text{(degree of link } j\text{)} & \text{if links } i \text{ and } j \text{ are directly connected,} \\ 0 & \text{if } i \text{ is equal to } j, \\ z \text{ (infinity),} & \text{otherwise.} \end{cases}$$

(v) Modified Link-Link Distance Matrix: The modified link-link distance matrix of a *n*-link chain is defined as a *(n x n)* symmetric matrix whose any *i, j*th element,

D_{ij}, is defined as the modified distance between links i and j. Accordingly, D_{ij} is equal to zero for all i.

6.3 MODIFIED LINK-LINK DISTANCE MATRIX ALGORITHM ---

It is cumbersome to generate the modified link-link distance matrix of a given chain directly by its visual inspection. So an algorithm has been developed in the present method to derive the modified link-link distance matrix of a chain from its modified link-link relation matrix with the help of computer. The algorithm consists of the following steps:

Step 1 -- Write down the modified link-link relation matrix of the given chain

Step 2 -- Take any kth column of the matrix.

Step 3 -- Take any finite element, R_{ik}, on this column.

Step 4 -- Take any finite element, R_{kj}, on the kth row.

Step 5 -- Let $S = R_{ik} + R_{kj}$. If S is less than R_{ij}, replace R_{ij} by S.

Step 6 -- Repeat steps 4 and 5 for all other finite elements on the kth row.

Step 7 -- Repeat steps 3-6 for all other finite elements on the kth column.

Step 8 -- Repeat steps 2-7 for all other columns of the matrix.

Step 9 -- Add the degree of each ith link to all the elements, except the main diagonal element, of the ith row of the resulting matrix. The matrix so obtained is the modified link-link distance matrix of the given chain.

In order to make this algorithm suitable for computer implementation, each of the infinite elements, z, of the modified link-link relation matrix has to be replaced by a large but finite number. As per requirement of the algorithm, the value of z should be greater than the largest element of the modified link-link distance matrix of the chain. So z has been replaced by 99 in the present example.

6.4 NEW STRUCTURAL INVARIANTS ---

A new structural invariant for a link of a chain is defined, known as the modified total distance rank. It is the sum of the modified distances of a link to all other links of the chain, and is given by the sum of all the elements of the row (or column) of the modified link-link distance matrix of the chain corresponding to the link. The modified total distance rank is an invariant for the link as it is independent of the labelling of the chain. If the modified total distance ranks of all the links of a chain are arranged in decreasing or increasing order, one gets a finite sequence of natural numbers, which is an invariant for the chain, as it is also independent of the labelling of the chain. This invariant is called the arranged sequence of modified total distance ranks of all the links, and abbreviate as ASMTDRL. For example, consider the chain shown in Fig. 6.1. The modified link-link relation matrix of this chain is given by equation (1), as below:

M_R =

Link	1	2	3	4	5	6	7	8	9	10
1	0	5	99	5	99	99	99	99	99	4
2	5	0	4	99	99	99	99	99	4	99
3	99	5	0	5	99	99	99	99	99	99
4	5	99	4	0	4	99	99	99	99	99
5	99	99	99	5	0	4	99	99	99	99
6	99	99	99	99	4	0	5	99	99	99
7	99	99	99	99	99	4	0	4	99	99
8	99	99	99	99	99	99	5	0	4	99
9	99	5	99	99	99	99	99	4	0	99
10	5	99	99	99	99	99	5	99	99	0

---------------(1)

The modified link-link distance matrix of the chain derived from its modified link-link relation matrix is given by equation (2), as below:

M_D =

Link	1	2	3	4	5	6	7	8	9	10	**DR**
1	0	8	12	8	12	16	12	16	12	7	103
2	8	0	7	12	16	20	16	11	7	12	109
3	12	7	0	7	11	15	20	15	11	16	114
4	8	12	7	0	7	11	16	20	16	12	109
5	12	16	11	7	0	6	11	15	19	15	112
6	16	20	15	11	6	0	7	11	15	11	112
7	12	16	20	16	11	7	0	7	11	7	107
8	16	11	15	20	15	11	7	0	6	11	112
9	12	7	11	16	19	15	11	6	0	15	112
10	7	12	16	12	15	11	7	11	15	0	106

---------(2)

Modified total distance ranks of all the links of the chain are presented in column DR following the matrix. By arranging the total distance ranks of all the links of the chain in decreasing order, one gets the ASMTDRL for the chain as

114, 112, 112, 112, 112, 109, 109, 107, 106, 103 --------------(3)

If the elements of a row of the modified link-link distance matrix of a chain are arranged in decreasing (or increasing) order, one gets a finite sequence of integers, which is also an invariant for the link corresponding to the row. This invariant is called the arranged row.

The computer program in C++ for getting distance matrix by algorithm and ASMTDRL for fig. 6.1 is given in appendix A at the end.

6.5 DETECTION OF ISOMORPHISM AMONG KINEMATIC CHAINS ---

ASMTDRLs for all known non-isomorphic chains with upto 10-1inks and three-degrees-of-freedom have been determined and analysed in the present method. It has been found that all the non-isomorphic chains have distinct ASMTDRLs. So it is proposed that ASMTDRL of a chain may be used as a reliable index for detecting isomorphism among planar chains with simple joints. If any two chains have identical ASMTDRLs, they are isomorphic, otherwise not. The proposed test, being simple and computationally efficient, is probably the most suitable for being incorporated in computerized structural synthesis of kinematic chains.

6.6 IDENTIFICATION OF DISTINCT MECHANISMS (INVERSIONS) OF A KINEMATIC CHAIN ---

The number of distinct mechanisms that can be derived from a given chain is equal to the number of distinct links in the chain. So the problem reduces to that of identifying distinct links of a chain. The following two methods have been developed in the present method for this purpose:

Method 1

The relative disposition of a link is reflected in the arranged row of the link. So in this method, the number of distinct links in a chain is determined by comparing the arranged rows of all the links of the chain. For example, consider the chain shown in Fig. 6.1. By arranging all the rows of the modified link-link distance matrix of the chain, given by equation (2), in decreasing order, it may be concluded that there are only seven distinct links in this chain.

Method 2

In this method, a link of the given chain is removed, and ASMTDRL of the remaining structure is determined. The relative disposition of the removed link is reflected in the ASMTDRL of the remaining structure and so it is taken as the identification code for the removed link. It is important to mention here that the degrees of the links in the remaining structure are taken to be the same as in the original chain.

Link	Identification Code for the Link								
1	224	211	211	211	201	201	197	197	197
2	239	234	211	207	205	197	195	193	193
3	206	206	203	203	198	198	193	191	189
4	239	234	211	207	205	197	195	193	193
5	226	215	213	199	198	195	194	193	192
6	228	213	211	201	200	199	197	191	189
7	241	241	220	213	213	195	195	195	193
8	228	213	211	201	200	199	197	191	189
9	226	215	213	199	198	195	194	193	192
10	211	207	202	202	199	199	199	198	198

TABLE 6.2: IDENTIFICATION CODES FOR ALL THE LINKS OF THE CHAIN SHOWN IN FIG. 6.1

So in this method, the number of distinct links in a chain is determined by comparing the identification codes for all the links of the chain. An algorithm has been developed in the present method to determine the identification code for any *i*th link of a given chain. The algorithm consisting of the following steps:

Step l -- Write down the modified link-link relation matrix of the given chain.

Step 2 -- Replace all the elements of the *i*th row and column of the matrix, except the main diagonal element, by *z*. It implies that the *i*th link has been removed from the chain.

Step 3 --Derive the corresponding modified link-link distance matrix from the resulting modified link-link relation matrix using the modified link-link distance matrix algorithm.

Step 4 --Add all the elements, except that on the *i*th column, of each row of the modified link-link distance matrix.

Step 5 --Arrange the resulting numbers in decreasing order. Omit the first element of the sequence as it corresponds to the removed link *i*. The resulting sequence is the identification code for the *i*th link of the chain.

The identification codes for all the links of the chain, shown in Fig. 6.1, are presented in Table 6.2. It is seen that the links 2 and 4, 5 and 9, 6 and 8 have identical identification codes, whereas links 1, 3, 7 and 10 have distinct codes. So it is concluded that there are only seven distinct links in this chain, and seven distinct mechanisms can be derived from it. The numbers of distinct mechanisms for different cases of planar chains with simple joints, determined in the present work, are presented in Table 6.3. Although the proposed method 1 is simpler and faster than method 2, the latter is more reliable as is evident from Table 6.3 as the number of distinct mechanisms derived by method 2 are more than that derived by method 1.

Category of the Chains	**Total Number of Chains**	**Total Number of Distinct Mechanisms derived by**	
		Method 1	**Method 2**
1-F, 6- link	2	5	5
1-F, 8-link	16	71	71
1-F, 10-link	230	1821	1834
2-F, 7-link	4	14	14
2-F, 9-link	40	253	254
3-F, 6-link	1	1	1
3-F, 8-link	7	26	26
3-F, 10-link	98	683	684

TABLE 6.3: NUMBER OF DISTINCT MECHANISMS OF SOME KNOWN CASES OF PLANAR CHAINS WITH SIMPLE REVOLUTE JOINTS

6.7 EXAMPLE ---

Consider another ten-link three dof chain as shown in Fig. 6.2

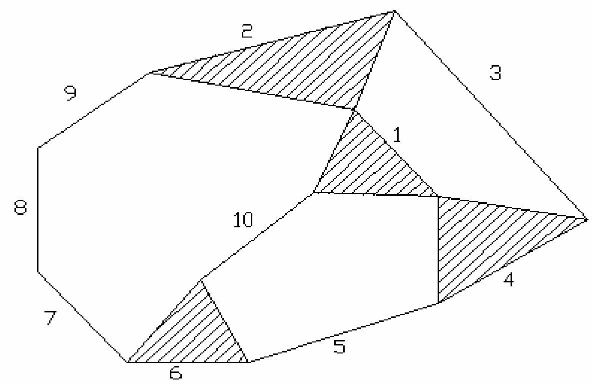


FIG. 6.2 TEN LINK, THREE DEGREE-OF-FREEDOM KINEMATIC CHAIN WITH SIMPLE JOINTS

The distance matrix for the above fig is given by eq.(4) as below:

M_D =

Link	1	2	3	4	5	6	7	8	9	10	**DR**
1	0	8	12	8	12	12	16	16	12	7	103
2	8	0	7	12	16	17	15	11	7	12	105
3	12	7	0	7	11	16	19	15	11	16	114
4	8	12	7	0	7	12	16	20	16	12	110
5	12	16	11	7	0	7	11	15	19	11	109
6	12	17	16	12	7	0	7	11	15	7	104
7	16	15	19	16	11	7	0	6	10	11	111
8	16	11	15	20	15	11	6	0	6	15	115
9	12	7	11	16	19	15	10	6	0	16	112
10	7	12	16	12	11	7	11	15	16	0	107

---------(4)

The ASMTDRL for this chain is

115, 114, 112, 111, 110, 109, 107, 105, 104, 103 --------------(5)

Link	**Identification Code for the Link**								
1	224	214	208	207	204	200	198	197	196
2	246	235	218	213	208	199	198	194	189
3	207	203	202	201	200	193	193	192	190
4	227	226	201	200	199	197	196	196	196
5	215	209	209	206	201	197	194	193	191
6	247	235	224	219	208	199	199	197	190
7	238	219	210	209	202	196	196	192	189
8	157	133	125	124	110	110	110	107	107
9	237	220	214	209	197	196	192	191	191
10	207	207	205	201	200	199	199	198	197

TABLE 6.4: IDENTIFICATION CODES FOR ALL THE LINKS OF THE CHAIN SHOWN IN FIG. 6.2

So, it can be seen that all links have different identification codes. Hence, the chain in fig. 6.2 has 10 distinct inversions.

Further, from the equations (3) and (5), it is seen that both the chains have different ASMTDRL. So, these are non-isomorphic chains.

6.8 CONCLUSIONS ---

In this method, a computer aided method, based on the concept of modified distance, has been proposed for detecting isomorphism between two given planar chains with simple joints. The modified link-link relation matrix can be written easily, even my mere inspection of the KC. Two computer aided methods have also been developed for identifying distinct mechanisms of a planar chain with simple joints. The proposed methods are heuristic and intuitive in nature. Moreover, it is not computationally very efficient. However, they have worked well on all the known cases of planar chains with simple joints.

CHAPTER – 7
LOOP BASED DETECTION OF ISOMORPHISM AMONG CHAINS, INVERSIONS AND TYPE OF FREEDOM IN MULTI DEGREE OF FREEDOM CHAIN

7.1 INTRODUCTION ---

Rao, A.C. [30] introduced this new concept of loop based detection of Isomorphism among chains, inversions and also set out a new concept of detection of Type of Freedom in multi degree of freedom chains. Using the *Loop Concept* method reveals simultaneously chain is isomorphic; link is isomorphic and type of freedom with no extra computational effort. A new invariant for a chain, called the *Chain Loop String* is developed for a planar kinematic chain with simple joints to detect isomorphism among chains. Another invariant called the *Link Adjacency String* is developed, which is a by-product of the same method to detect inversions of a given chain. The proposed method is also applicable to know the type of freedom of a chain in case of multi degree of freedom chains.

7.2 METHOD ---

The following definitions are to be understood clearly before applying this method. Various definitions with their abbreviations are given below:

- **Link Value (L.V.) ---** A numerical value for the link is assigned based on its connectivity to other links. Therefore, quaternary link has "L.V. = 4", and ternary link has "L.V. = 3" etc.
- **Independent Loop (I.L.) ---** The chains shown in the Figs. 7.1 and 7.2 consists of 3 independent loops which are labeled as (1), (2) and (3). Such loops do not contain any other loops within. These loops for any chain can easily be identified by visual inspection and do not depend upon the manner in which the chain is drawn i.e. a chain can be drawn to appear differently but there is always a set of clear and independent loops which can be noted by inspection.
- **Sub Loops (S.L.) ---** They are the combinations of the independent loops. For example, if the independent loops are (1), (2) and (3), the sub loops for that chain are [1-2], [1-3], [2-3] and [1-2-3] avoiding repetition of a combination. Similarly, if the chain has 4 independent loops, all possible sub loops for the chain are [1-2],

[1-3], [1-4], [2-3], [2-4], [3-4], [1-2-3], [1-2-4], [1-3-4], [2-3-4] and [1-2-3-4]. The last sub loop for any chain includes all the loops and is generally known as *Peripheral Loop.*

With the understanding of the above definitions, a table is prepared for each chain, with loops as rows and the links as columns. On the top of each link, its link value has to be written for the purpose which will be explained later. Now refer to Tables 7.1 and 7.2 of Figs. 7.1 and 7.2, respectively, for applying this method which is explained below stepwise.

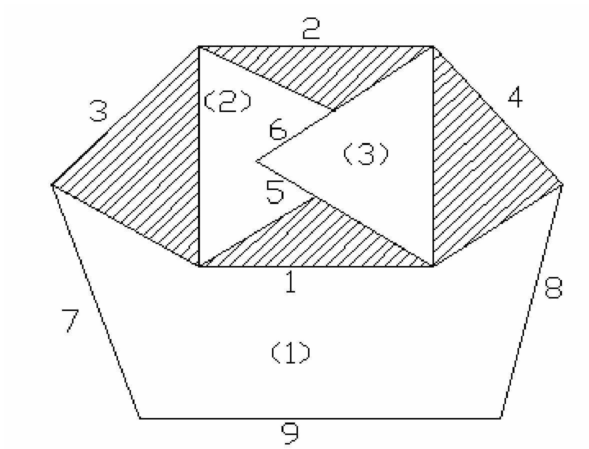


FIG. 7.1 NINE – LINK TWO D.O.F. CHAIN

TABLE 7.1: L.V.C., L.S.L and L.A.S.L of FIG. 7.1

	Link Value (L.V.)	3	3	3	3	2	2	2	2	2	
	Links → Loops ↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Loop Size (L.S.)**
I.L.	**1**	1	0	1	1	0	0	1	1	1	**6**
	2	1	1	1	0	1	1	0	0	0	**5**
	3	1	1	0	1	1	1	0	0	0	**5**
S.L.	**1-2**	1	1	1	1	1	1	1	1	1	**9**
	1-3	1	1	1	1	1	1	1	1	1	**9**
	2-3	1	1	1	1	0	0	0	0	0	**4**
	1-2-3	0	1	1	1	0	0	1	1	1	**6**

Loop Value of a Chain (L.V.C.) = 44

I.L. = Independent Loops
S.L. = Sub-Loops

Links	Loop Value of a Link (L.V.L.)	Loop Frequency String of a Link (L.F.S.L.)	Loop String of a Link (L.S.L.)
1	38	200121	38-200121
2	38	200121	38-200121
3	39	200211	39-200211
4	39	200211	39-200211
5	28	200020	28-200020
6	28	200020	28-200020
7	30	200200	30-200200
8	30	200200	30-200200
9	30	200200	30-200200

L.F.S.C. = 200221
J.V.C = 34
C.L.S. = [44, (200221), (39, 39, 38, 38, 30, 30, 30, 28, 28), 34]

Links	Adjacent Links	Loop Adjacency String of Links (L.A.S.L.)
1	4, 3, 5	38-200121, 39-200211, 39-200211, 28-200020
2	3, 6, 4	38-200121, 39-200211, 39-200211, 28-200020
3	1, 7, 2	39-200211, 38-200121, 38-200121, 30-200200
4	8, 1, 2	39-200211, 38-200121, 38-200121, 30-200200
5	6, 1	28-200020, 38-200121, 28-200020
6	2, 5	28-200020, 38-200121, 28-200020
7	3, 9	30-200200, 39-200211, 30-200200
8	9, 4	30-200200, 39-200211, 30-200200
9	7, 8	30-200200, 30-200200, 30-200200

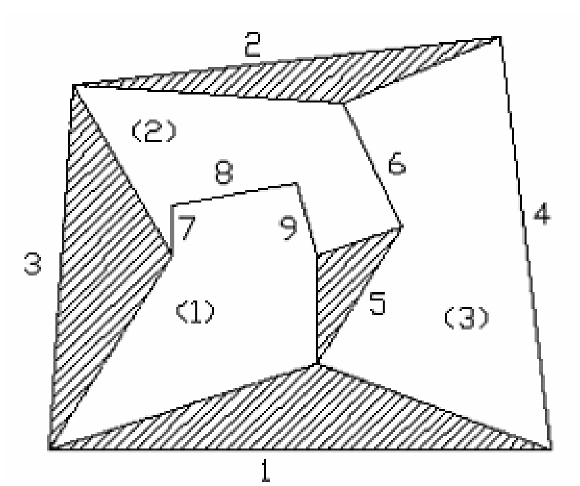


FIG. 7.2 NINE-LINK TWO D.O.F. CHAIN

TABLE 7.2: L.V.C., L.S.L and L.A.S.L of FIG. 7.2

	Link Value (L.V.)	3	3	3	2	3	2	2	2	2	
	Links → Loops ↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Loop Size (L.S.)**
I.L.	**1**	1	0	1	0	1	0	1	1	1	**6**
	2	0	1	1	0	1	1	1	1	1	**7**
	3	1	1	0	1	1	1	0	0	0	**5**
S.L.	**1-2**	1	1	1	0	1	1	0	0	0	**5**
	1-3	1	1	1	1	1	1	1	1	1	**9**
	2-3	1	1	1	1	1	0	1	1	1	**8**
	1-2-3	1	1	1	1	0	0	0	0	0	**4**

Loop Value of a Chain (L.V.C.) = 44

L.F.S.C. = 111121
J.V.C = 34
C.L.S. = [44, (111121), (40, 39, 38, 37, 30, 30, 30, 26, 26), 34]

Links	**Loop Value of a Link (L.V.L.)**	**Loop Frequency String of a Link (L.F.S.L.)**	**Adjacent Links**	**Loop Adjacency String of Links (L.A.S.L.)**
1	37	110121	5, 3, 4	37-110121, 40-111120, 39-111111, 26-110011
2	38	111021	3, 6, 4	38-111021, 39-111111, 26-110011, 26-101020
3	39	111111	1, 7, 2	39-111111, 38-111021, 37-110121, 30-111100
4	26	110011	2, 1	26-110011, 38-111021, 37-110121
5	40	111120	9, 1, 6	40-111120, 37-110121, 30-111100, 26-101020
6	26	101020	2, 5	26-101020, 40-111120, 38-111021
7	30	111100	3, 8	30-111100, 39-111111, 30-111100
8	30	111100	7, 9	30-111100, 30-111100, 30-111100
9	30	111100	8, 5	30-111100, 40-111120, 30-111100

STEP 1: To Fill the Elements in the Rows of Independent Loops.

Whether or not each link is participating in an independent loop is spelled out in the rows of independent loops, whose elements are either “0” or “1”. For example, the 1st row of Table 7.1 which is [1, 0, 1, 1, 0, 0, 1, 1, 1] implies that links 1, 3, 4, 7, 8 and 9 of Fig. 7.1 are participating in the independent loop (1). Similarly the elements of rows of other independent loops can be filled in.

STEP 2: To Fill the Elements in the Rows of Sub Loops.

Consider the row of sub-loop [1-2] of Table 7.1. For filling the first element of the sub-loop [1-2], add the first elements of the independent loops (1) and (2). If the sum is not equal to zero but less than the link value of link 1, the first element of the sub-loop [1-2] is "1". If the sum is zero or equal to or greater than the link value, the first element of the sub-loop [1-2] is "0".

For the case under consideration, the sum is [1+1] =2, which is less than the link value, i.e. 3. So first element of the sub-loop [1-2] is 1.

Consider as another example to fill the fifth element of the seventh row, i.e. the row of sub-loop [1-2-3], add the corresponding fifth elements of independent loops (1), (2) and (3). The sum is [0+1+1] = 2. And its value is equal to the link value of link value 5. So, the fifth element of the sub-loop [1-2-3] is zero. In the same fashion, fill in all the elements of the rows of sub-loops.

The elements of the row of a sub-loop formed by independent loops which are adjacent to one another indicate whether or not each link is participating on the periphery of that sub loop. For example, the independent loops 2 and 3 of Fig. 7.2 are adjacent to one another. The sub loop [2-3] formed by these two independent loops consists of outermost links 1, 2, 3, 4, 5, 7, 8 and 9. One can easily see that the absence of link 6 in the row of sub loop [2-3], since it is not on the periphery of the sub loop [2-3], even though it is present in both the independent loops (2) and (3).

If the sub loop is formed by independent loops, which are not adjacent to one another, the sub loop consists of all the links of all the corresponding independent loops. For example, the sub loop [1-3] formed by independent loops (1) and (3) of Fig. 7.6 consists of links 1, 5, 4 and 3 of independent loop (1) and 1, 6, 2, 7 and 8 of independent loop (3). Avoiding repetition the row of sub loop [1-3] of Fig. 7.6 consists of links 1, 2, 3, 4, 5, 6, 7 and 8.

STEP 3: Computations.

A number of new terms, pertaining to the table of a given kinematic chain are defined and used as follows:-

(a) Loop Size [L.S.]: It is defined as the size of the loop. It indicates the number of links participating on the periphery of any loop. In order to calculate loop sizes, add all the elements of the rows of Table 7.1 of Fig. 7.1, the loops sizes are respectively 6, 5, 5, 9, 9, 4, 6 for loops 1, 2, 3, 1-2, 1-3, 2-3 and 1-2-3.

(b) Loop Value of a Chain [L.V.C.]: It is defined as the summation of all the loop sizes of a given kinematic chain. For Fig. 7.1 and 7.2, the L.V.C. is 44.

(c) Loop Frequency String of a Chain [L.F.S.C.]: It is a string of numbers, each digit of which represents the frequency of the loop sizes in a descending manner from *n* to 4 in that order where "*n*" is the number of links in a chain and "4" is the minimum loop size of any chain, obviously "*n*" is the maximum loop size.

Example 1: For Fig. 7.1, the L.F.S.C. is 200221. In the above string the last digit is "1" (6^{th} placed), and it indicates presence of one four-bar loop. The fifth placed digit in that string indicates the presence of two 5-bar loops. Similarly fourth, third, second and first placed digits in that string indicate two six-bar loops, no seven-bar loops, no eight-bar loops and two nine-bar loops respectively.

Example 2: For Fig. 7.2, the L.F.S.C. is 111121. The above string contains one four-bar loop, two five-bar loops, one six-bar loop, one seven-bar loop, one eight-bar loop and one nine-bar loop.

(d) Loop Value of a Link [L.V.L.]: It is defined for a link as the summation of the sizes of loops in which it is participating.

For example: From the Table 7.1; for link 1, L.V.L. is 38 which is the sum of loop sizes of the loops [1] [2] [3] [1-2] [1-3] and [2-3], neglecting the loop size [1-2-3] where link 1 is not participating.

(e) Loop Frequency String of a Link [L.F.S.L.]: It is defined as the frequency of occurrence of loop sizes of a link i.e. from "*n*" to "4", in that order where "*n*" is number of links and "4" is the minimum loop size of any chain.

For Example: For the link 1 of Fig. 7.1, the L.F.S.L. is 200121, which implies link 1 is present in two nine-bar loops, one six-bar loop, two five-bar loops and one four-bar loop.

(f) Joint Value of a Chain [J.V.C.]: "Joint value of a chain is defined as the sum of the values of all the joints in a chain." The value of joint in turn is defined as two less than the sum of values of the participating links at a joint.

J.V.C. is an invariable for a particular type of link assortment. If two chains consist of identical link assortment and identical joint assortment, then they will have same joint value. Consider Fig. 7.1 as an example for estimating the joint value. This chain consists of eleven joints. Out of which, there are four ternary-ternary, four ternary-binary and three binary-binary joints.

The joint value of ternary-ternary joint is [3 + 3] – [2] = 4. The joint value of ternary-binary joint is [3 + 2] – [2] = 3. The joint value of binary-binary joint is [2 + 2] – [2] = 2. Therefore the joint value of a chain is equal to [4(4) + 4(3) + 3(2)] = 34.

(g) Chain Loop String [C.L.S.]: It is obtained by concatenating (1) L.V.C. of a given chain, (2) L.F.S.C. of a given chain, (3) L.V.L. of all the links written in descending order, (4) J.V.C.

For Fig. 7.1, the C.L.S. is [44, (200221), (39, 39, 38, 38, 30, 30, 30, 28, 28), (34)].

For Fig. 7.2, the C.L.S. is [44, (111121), (40, 39, 38, 37, 30, 30, 30, 26, 26), (34)].

(h) Loop String of a Link [L.S.L.]: It is obtained by concatenating (1) L.V.L. of a given link, (2) L.F.S.L. of a given link. For example; for link 1, of Fig. 7.1 the L.S.L. is 38-200121.

(i) Loop Adjacency String of a Link [L.A.S.L.]: It is also obtained for a given link by arranging the loop strings of adjacent links of the given link in descending order and concatenate to the loop string of the given link.

For example: Link (1) of Fig. 7.1 has adjacent links 4, 3 and 5. Arrange the loop strings of 4, 3 and 5 in descending order and concatenate to loop string of link 1 to obtain L.A.S.L. So L.A.S.L. of link 1 is 38-200121, 39-200211, 39-200211, 28-200020.

7.3 DETECTION OF ISOMORPHISM AMONG CHAINS ---

The chain loop strings (C.L.S.) is a definitive test for isomorphism among chains. If the two chains are known to be isomorphic, their chain loop strings should be identical and vice-versa. If the two chains are non-isomorphic, the chain loop strings should differ at some position or other.

As an illustration consider the chain loop strings for the Fig. 7.1 and Fig. 7.2

For Fig. 7.1, the C.L.S. is [44, (200221), (39, 39, 38, 38, 30, 30, 30, 28, 28), (34)].

For Fig. 7.2, the C.L.S. is [44, (111121), (40, 39, 38, 37, 30, 30, 30, 26, 26), (34)].

Even a cursory glance reveals that these strings differ from each other considerably and hence they are non-isomorphic.

Consider the Fig. 7.3, which is isomorphic to Fig. 7.2 but redrawn so that its independent loop sizes are now different than Fig. 7.2. For Fig. 7.2, the independent loop sizes are (6), (7) and (5), whereas for Fig. 7.3, the independent loop sizes are (4), (5) and (7). But if the chain loop strings are calculated, it will result in identical strings, since they are isomorphic.

Even if the links are labeled arbitrarily as it is done in the same example, the method presented here will result in identical chain loop strings leading to invariant results.

So this test is not affected by redrawing a chain (so that its independent loop sizes are different) or by relabeling of various links and loops. All the 16 eight-bar single d.o.f. chains, 40 nine-bar two d.o.f. chains, 230 ten-bar single d.o.f. chains and 98 ten-bar three d.o.f. chains have been tested for isomorphism. All of them have yielded distinct chain loop strings.

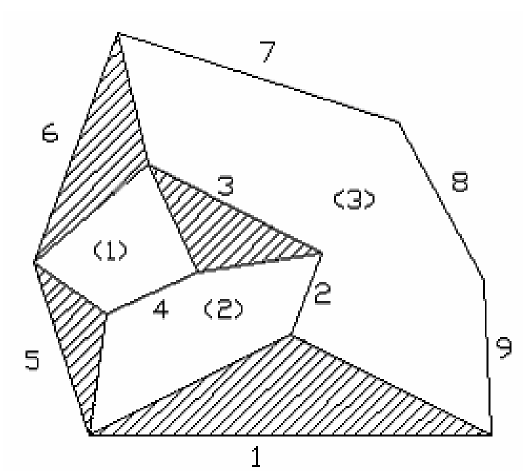


FIG. 7.3 NINE LINK (FIG. 7.2, REDRAWN)

TABLE OF FIG. 7.3

	Link Value (L.V.)	3	2	3	2	3	3	2	2	2	
	Links → Loops ↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Loop Size (L.S.)**
I.L.	**1**	0	0	1	1	1	1	0	0	0	**4**
	2	1	1	1	1	1	0	0	0	0	**5**
	3	1	1	1	0	0	1	1	1	1	**7**
S.L.	**1-2**	1	1	1	0	1	1	0	0	0	**5**
	1-3	1	1	1	1	1	1	1	1	1	**9**
	2-3	1	0	1	1	1	1	1	1	1	**8**
	1-2-3	1	0	0	0	1	1	1	1	1	**6**

(L.V.C.) = 44

L.F.S.C. = 111121
J.V.C = 34
C.L.S. = [44, (111121), (40, 39, 38, 37, 30, 30, 30, 26, 26), 34]

7.4 DETECTION OF ISOMORPHISM AMONG INVERSIONS ---

The link adjacency strings of the various links have the potential to disclose how many inversions can be obtained from a given chain and by fixing which links these inversions are possible. If link adjacency strings of two links are identical then the inversions are identical.

Referring to Table 7.1 of Fig. 7.1, the link adjacency strings of [1, 2] are identical, [3, 4] are identical, [5, 6] are identical, [7, 8] are identical and 9 is distinct. So taking one link each from identical strings together with the distinct link 9, total five inversions are possible for the chain in Fig. 7.1.

Referring to Table 7.2 of Fig. 7.2, all the nine links have distinct link adjacency strings. So there are nine inversions.

The total number of inversions arrived at by this method for 8-link single d.o.f., 9-link two d.o.f., 10-link single d.o.f. and 10-link three d.o.f. are listed below in Table 7.3.

S.No.	Type of Chains	Total No. of Inversions
1	8 link, 1 d.o.f.	71
2	9 link, 2 d.o.f.	254
3	10 link, 3 d.o.f.	684
4	10 link, 1 d.o.f.	1834

TABLE 7.3: LIST OF INVERSIONS

7.5 DETECTION OF TYPE OF FREEDOM OF KINEMATIC CHAINS ---

A multi d.o.f. chain possesses total, partial or fractionated freedom. This is helpful in dividing the frame and input links from the view point of mobility.

Whether a chain can possess a particular type of freedom can easily be predicted by applying this method. It is known that an independent four-bar loop has single d.o.f., a five-bar loop has two d.o.f. and a six-bar loop has three d.o.f. and so on. The smallest loop size including the sub loops of a chain will decide the type of freedom.

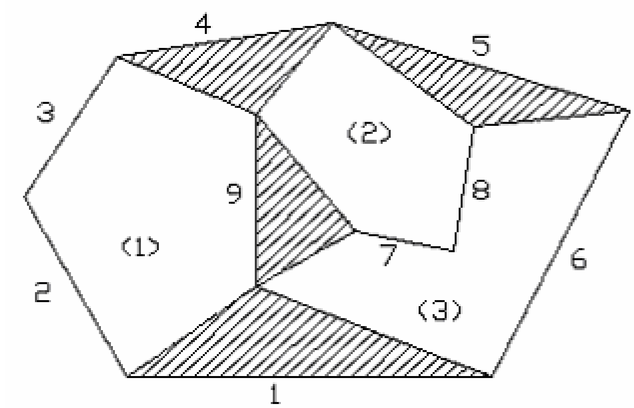


FIG. 7.4 NINE-LINK TWO D.O.F. CHAIN WITH TOTAL FREEDOM

TABLE OF FIG. 7.4

	(L.V.)	3	2	2	3	3	2	2	2	3	
	Links → Loops ↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Loop Size (L.S.)**
I.L.	**1**	1	1	1	1	0	0	0	0	1	**5**
	2	0	0	0	1	1	0	1	1	1	**5**
S.L.	**3**	1	0	0	0	1	1	1	1	1	**6**
	1-2	1	1	1	1	1	0	1	1	1	**8**
	1-3	1	1	1	1	1	1	1	1	1	**9**
	2-3	1	0	0	1	1	1	0	0	1	**5**
	1-2-3	1	1	1	1	1	1	0	0	0	**6**

L.F.S.C. = 110230

Consider chain shown in Fig. 7.4 and calculate loop frequency string for assessing the type of freedom.

1. **Total Freedom:** If the last digit of the loop frequency string of a chain is zero then the chain possesses total freedom indicating the absence of four bar loop in a chain therefore the chain shown in Fig. 7.4 has total freedom.

2. **Partial Freedom:** For a two d.o.f. chain the last digit of the loop frequency string of a chain is not equal to zero then the chain possesses partial freedom, indicating the presence of a four bar loop. The chains, shown in Fig. 7.1 and 7.2 possess partial freedom since the last digit of their loop frequency strings is 1.
3. **Fractionated Freedom:** While working on this aspect, an interesting phenomenon has been observed. All chains belonging to any one family (the chains in a particular family will have same number and type of links) have common loop value. Here the phenomenon has been explained with suitable examples.

   Let n = no. of independent loops.

   s = total no. of sub loops

   m = total no. of loops where ($m = n + s$)

   Consider the following cases for exploring the property of participation of various links in total no. of loops in a chain.

1st Case: Consider six bar chain with its table shown in the Fig. 7.5.

In case of six bar chain, $n = 2$ and $m = 3$. From the computations made, it has been observed that ternary link participates in all the loops, i.e. in "m" loops; binary link is participating in (m-1) loops. Therefore, for four binary links and two ternary links, the total participation in loops is 14 [= 4(m-1) + 2m, where m = 3] and this value is equal to loop value of the chain.

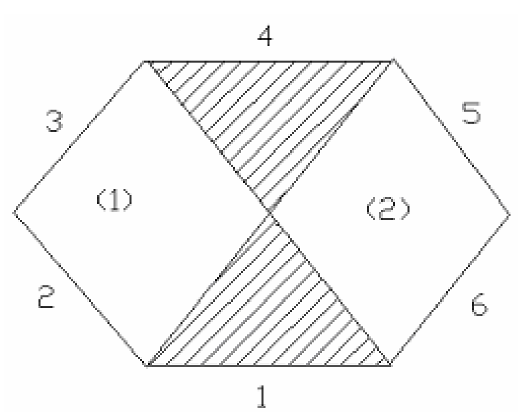


FIG. 7.5 SIX-LINK WATT CHAIN

TABLE OF FIG. 7.5

(L.V.)	3	2	2	3	2	2	
Links → **Loops ↓**	**1**	**2**	**3**	**4**	**5**	**6**	**Loop Size (L.S.)**
I.L. **1**	1	1	1	1	0	0	**4**
I.L. **2**	1	0	0	1	1	1	**4**
S.L. **1-2**	1	1	1	1	1	0	**6**

L.V.C. = 14

2nd Case: When $n = 3$ and $m = 7$.

Consider 9 bar two d.o.f. chain shown in the Fig. 7.1 as an example for this case. Here from the table, it is observed that quarternary link (L.V. = 4) is participating in "*m*" loops and binary link (L.V. = 2) is participating in (m-3) loops and ternary link is participating in (m-1) loops. Therefore if a nine-bar chain consists of one quarternary, two ternary and six binary links, the L.V.C. is estimated as 43 [$1(m) + 2(m-1) + 5(m-3) = 43$, where $m = 7$].

3rd Case: When $n = 4$ and $m = 15$

Consider ten-bar single d.o.f. chain shown in Fig. 7.6 as an example for this case. Here $n = 4$ and $m = 15$ and further it has been observed that quintenary link (L.V. = 5) is participating in all "*m*" loops. Quarternary link (L.V. = 4) is participating in (m-1) loops. Ternary link is participating in (m-3) loops and binary link is participating in (m-7) loops. The chain shown in Fig. 7.6 consists of 1 quintenary, 1 quaternary, 1 ternary and seven binary links. Therefore, the L.V.C. for this family would be equal to $1m + 1(m-1) + 1(m-3) + 7(m-7)$ where $m = 15$ and value is equal to 97.

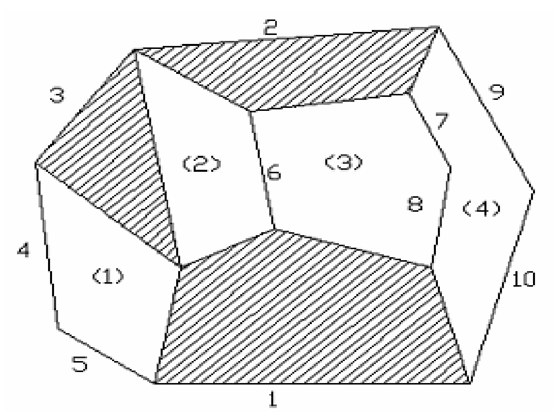


FIG. 7.6 TEN-LINK SINGLE D.O.F. CHAIN

TABLE OF FIG. 7.6

	(L.V.)	5	4	3	2	2	2	2	2	2	2	
	Links → **Loops ↓**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Loop Size (L.S.)**
I.L.	**1**	1	0	1	1	1	0	0	0	0	0	**4**
	2	1	1	1	0	0	1	0	0	0	0	**4**
	3	1	1	0	0	0	1	1	1	0	0	**5**
	4	1	1	0	0	0	0	1	1	1	1	**6**
S.L.	**1-2**	1	1	1	1	1	1	0	0	0	0	**6**
	1-3	1	1	1	1	1	1	1	1	0	0	**8**
	1-4	1	1	1	1	1	0	1	1	1	1	**9**
	2-3	1	1	1	0	0	0	1	1	0	0	**5**
	2-4	1	1	1	0	0	1	1	1	1	1	**8**
	3-4	1	1	0	0	0	1	0	0	1	1	**5**
	1-2-3	1	1	1	1	1	0	1	1	0	0	**7**
	1-2-4	1	1	1	1	1	1	1	1	1	1	**10**
	1-3-4	1	1	1	1	1	1	0	0	1	1	**8**
	2-3-4	1	1	1	0	0	0	0	0	1	1	**5**
	1-2-3-4	1	1	1	1	1	0	0	0	1	1	**7**

L.V.C. = 97

From the above three cases, the following definite relations are established. They are:

(a) If the link value of a link is = $n + 1$, the link participates in all "m" loops.

(b) If the link value of a link is = n, the link participates in "m - 1" loops.

(c) If the link value of a link is = n - 1, the link participates in "m - 3" loops.

(d) If the link value of a link is = n - 2, the link participates in "m - 7" loops.

(e) If the link value of a link is = n - 3, the link participates in "m - 15" loops.

(f) If the link value of a link is = n - 4, the link participates in "m - 31" loops.

With the help of the above-established relation between links and loops, one can easily calculate loop value of a chain. Now based on the above observations, the fractionated freedom is determined as follows. There are two situations in which a chain possesses a fractionated freedom.

First Situation: The L.V.C. is calculated in two ways:

(a) Based on the observations made in the just concluded explanation, calculate loop value of a chain

(b) Estimate the L.V.C. by formulating the table as explained earlier in method.

If L.V.C. found by formulating the table is less than the L.V.C. based on the observations, then the chain possesses fractionated freedom.

Consider Fig. 7.7 as an example. It is seen that, the L.V.C. = 42 when it is estimated by computations made in the table. The L.V.C. by observations is 43 [1(m) + 2(m – 1) + 6(m – 3), where m = 7]. Therefore, this chain possesses fractionated freedom, since the link participation is less than what it should have been.

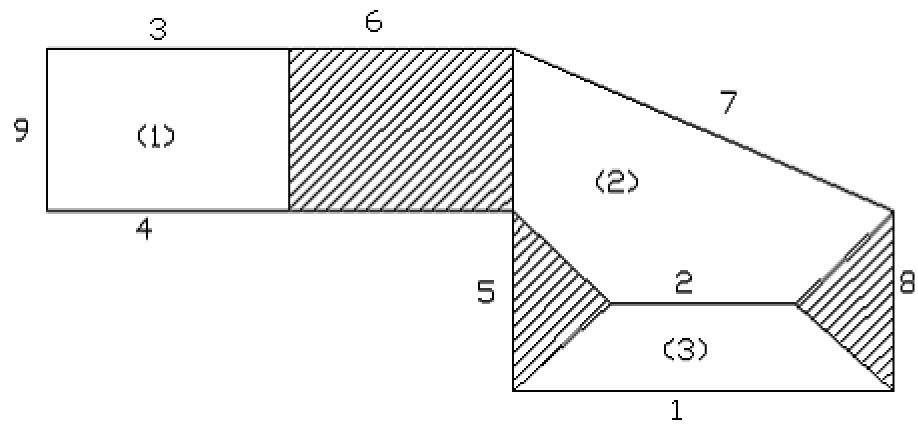


FIG. 7.7 NINE-LINK, 2 D.O.F. CHAIN WITH FRACTIONATED FREEDOM

TABLE OF FIG. 7.7

	(L.V.)	2	2	2	2	3	4	2	3	2	
	Links→ Loops↓	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**Loop Size (L.S.)**
I.L.	**1**	0	0	1	1	0	1	0	0	1	**4**
	2	1	1	0	0	1	0	0	1	0	**4**
	3	0	1	0	0	1	1	1	1	0	**5**
S.L.	**1-2**	1	1	1	1	1	1	0	1	1	**8**
	1-3	0	1	1	1	1	1	1	1	1	**8**
	2-3	1	0	0	0	1	1	1	1	0	**5**
	1-2-3	1	0	1	1	1	1	1	1	1	**8**

L.V.C. = 42

Second Situation: In case of multi d.o.f. chains, if loop value of any link in a chain is more than "n + 1", it automatically possesses fractionated freedom. As shown in Fig.

7.8, the link labeled as “5” has link value 5 and it is equal to “$n + 2$”, so the chain possesses fractionated freedom.

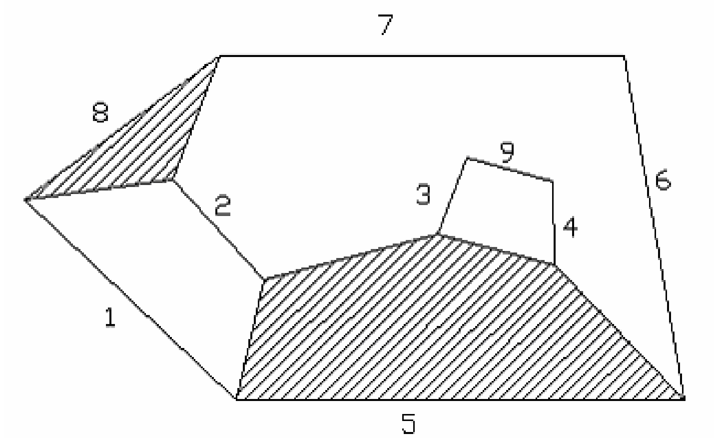


FIG. 7.8 NINE-LINK CHAIN WITH QUINTENARY LINK

7.6 CONCLUSIONS ---

1. Though no proof has been offered, the authors strongly believe that this method is unique and applicable to planar chains of any size and complexity. Unique in the sense that it has taken care of all basic features of a chain viz, links, joints and loops, whereas existing methods have not considered loops.
2. This method is extremely simple in the formulation stage, since the only input necessary is the disposition of various links in a loop.
3. It is also very simple in the execution stage, since the arithmetical computations made are very easy enough to be attempted by hand without the necessity of sophisticated algorithms.
4. The reliability of this method is based on the fact that it has successfully been applied to the counter examples found in the literature [9] and [19].
5. The basic advantage of this method over other methods is exploring three characteristics of a chain at a single stroke. They are isomorphism among chains, isomorphism among inversions and type of freedom in case of multi d.o.f. chains.
6. The required program for executing this method can easily be written by any one with an elementary knowledge of computers. Whereas most of the methods dealing with this topic rely on sophisticated algorithms.
7. Another important aspect of this method is that it is neither affected by relabeling nor by redrawing a chain.

CHAPTER – 8
A GENETIC ALGORITHM FOR TOPOLOGICAL CHARACTERISTICS OF KINEMATIC CHAINS

8.1 INTRODUCTION ---

Rao, A.C. [31] made an attempt to utilize the principles (i) to detect isomorphism among kinematic chains, (ii) to compare the chains from the motion generation point of view and (iii) to select the best ground and input links.

Each link of a chain is considered to have a “fitness” equal to its connectivity. For every link in the chain, other links are considered to be the environment. Formation of a chain is then viewed as mating of links. Direct joining of two links is considered to be the first generation mating. Combination of links which are separated by one link with respect to every link in the chain is considered as the second generation mating and so on.

8.2 METHOD ---

A four-link chain is taken as the basic chain and all other chains, six-link, eight-link chains etc. with the same d.o.f. are viewed as families with different population. The links of a four-link chain are either directly joined or separated by only one link. Likewise, in a six-link chain, farthest links are separated by not more than two links and in an eight link single d.o.f. chain; this is limited to three links and so on. Thus, it is only necessary to study the six-link chains to the extent of first generation mating and eight links to second generation mating. Obviously, ten-link chains need investigation only up to third generation mating and so on. The above facts establish that any method based on “mating” concept, to test isomorphism among chains, it is necessary to test isomorphism generation-wise, i.e. first generation, second generation mating, etc., depending upon the number of links; obviously it is adequate to test up to the last generation possible viz. up to the third generation in case of ten link chains. As will be seen later, the effort involved is minimum compared to any other method reported so far. The notable feature is that unlike other methods which propose tests necessary but not sufficient, this method fulfils both necessary and sufficient requirements making it unique.

8.2.1 LINK STRING ---

The basic principle followed is to represent a link by a string of binary numbers (0 and 1), the number of digits in the string being equal to the number of links in the kinematic chains. The number of times 1 occurs in the string is equal to the connectivity of the link, i.e., a binary link will have 1 twice in the string while a ternary link will have 1 at three digit places in the string. All other digits in the string are zeroes. The exact position of 1 (ones) and 0 (zeros) in the string will depend upon its connection to the other links and their labeling. An important fact to be noted is the "fitness" of a link is taken equal to the connectivity of the link since it is indicative of the number of design parameters possessed by the link. It will also be evident that the fitness of a link is equal to sum of all the non-zero elements in the string.

8.2.2 FIRST GENERATION BINARY STRINGS OF LINKS ---

First generation deals with the links that are directly joined. Consider a four-link chain, Fig. 8.1. It is easy to see that it is the basic chain with least number of links having 1 d.o.f..

A matrix *A*, called adjacency matrix, can be written for every chain in which the element

a_{ij} = 0, if there is no direct contact between links *i* and *j;*

a_{ij} = 1, if there is direct contact between links *i* and *j;*

Also a_{ii} = 0, since a link cannot connect to itself.

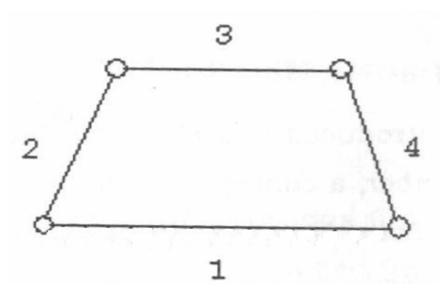


FIG. 8.1 FOUR-BAR CHAIN

For the four-link chain, Fig. 8.1

	Links	1	2	3	4
	1	0	1	0	1
A =	2	1	0	1	0
	3	0	1	0	1
	4	1	0	1	0

or simply

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \qquad \text{----------------------(1)}$$

Each row of the A matrix can be considered as a binary string of the concerned link with well defined digital position for 0s and is depending upon its connectivity with the other links, i.e., the position of 1 in the binary string is dictated by the environment, i.e., disposition of the other links. Also, as stated earlier, the number of times the element 1 occurs in a row (string) is the link's connectivity or "fitness". It is easy to see that the "fitness" of a link is indicative of the number of design parameters it possesses, viz., a binary link has one design parameter while a ternary link has three design parameters.

In a similar manner, the A matrix for the fig. shown in fig. 8.2 can be written as follows:

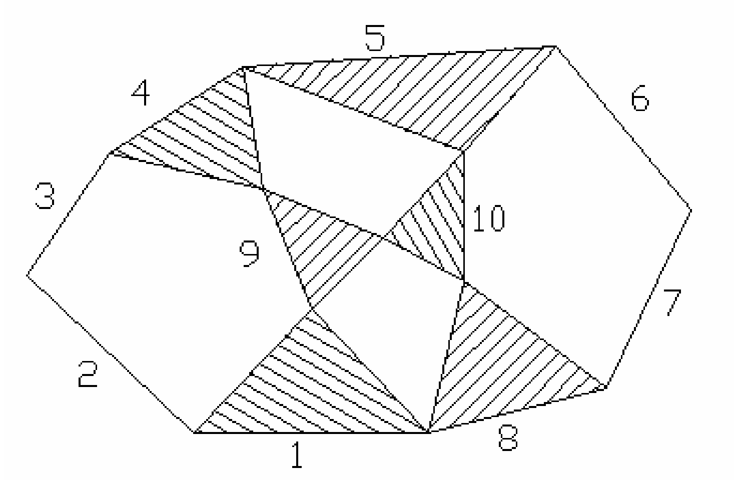


FIG. 8.2 TEN-LINK ONE DEGREE OF FREEDOM CHAIN

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \quad \text{------(2)}$$

Each row of the above matrix can be considered as the binary string of the concerned link with well designed digital positions for the elements 0 and 1 preserving the "fitness" of the link. For example, the binary string of ternary link 4 is

0 0 1 0 1 0 0 0 1 0.

Relabelling of the links does not matter as the relative positions of the links and hence the matrix elements do not change.

8.2.3 MATING ---

From the adjacency matrix, each link, having regard to its environment, i.e. to its disposition in relation to other links can be assigned a binary string with 0s and 1s taking definite digital positions, the sum of non-zero elements being equal to the fitness.

The relationship between any two links or the role of two links taken together can be studied by mating the binary strings of the concerned links. The following rules are followed:

1) Mating only among the bits (elements) occupying the same digital positions in respective binary strings is possible; e.g., Consider two binary strings,

   String *A* - 0 1 0 0 0 0 0 1 1 0

   String *B* - 1 0 1 0 0 0 0 0 0 0

   Mating of the first bit of string *A* is possible only with the first bit of string *B*; similarly second bit of string *A* can mate with the second bit of string *B* only.

2) The mating of the two strings results in a third string (off-spring) *C* of the same order, i.e., same number of digits. It may be noted that mating is not productive among equal bits, i.e., outcome is no off-spring while mating between unlike bits is possible and productive. For example,

Bit of string *A*	+	Bit of string *B*	=	Bit of string *C*	
1	+	1	=	0	
0	+	0	=	0	
1	+	0	=	1	------------------(3)
0	+	1	=	1	

It can be generalized that

(i) Mating between the bits occupying the same digital positions in both binary strings need only be considered.

(ii) Mating among equal bits will result in zero bit value.

(iii) Mating among unequal bits is productive which will result in a bit value equal to the difference in the mating bits.

Following the above, the string *C* for the example strings *A* and *B* will be

1 1 1 0 0 0 0 1 1 0

Therefore, the fitness of the "off-spring", i.e., string *C* is the sum of all the non-zero elements which is equal to five.

8.2.3.1 MATING OF FIRST GENERATION STRINGS ---

For example, consider the links 2 and 5 of the chain, Fig. 8.2. The first generation binary string of link-2 (from the matrix-2) is

1 0 1 0 0 0 0 0 0 0

and the first generation binary string of link-5 is

0 0 0 1 0 1 0 0 0 1

Then the mating of the links 2 and 5 yields an "off-spring" of fitness 5 represented by the binary string 1 0 1 1 0 1 0 0 0 1

In this manner, mating of each link with the other links one-by-one can be considered and the fitness of the resulting "off-spring" can be presented in the form of a matrix F_1. For example, for the ten-link chain, Fig. 8.2, F_1 matrix will be

$$F_1 = \begin{pmatrix} 0 & 5 & 3 & 4 & 6 & 5 & 3 & 6 & 6 & 2 \\ 5 & 0 & 4 & 3 & 5 & 4 & 4 & 3 & 3 & 5 \\ 3 & 4 & 0 & 5 & 3 & 4 & 4 & 5 & 3 & 5 \\ 4 & 3 & 5 & 0 & 6 & 3 & 5 & 6 & 6 & 2 \\ 6 & 5 & 3 & 6 & 0 & 5 & 3 & 4 & 2 & 6 \\ 5 & 4 & 4 & 3 & 5 & 0 & 4 & 3 & 5 & 3 \\ 3 & 4 & 4 & 5 & 3 & 4 & 0 & 5 & 5 & 3 \\ 6 & 3 & 5 & 6 & 4 & 3 & 5 & 0 & 2 & 6 \\ 6 & 3 & 3 & 6 & 2 & 5 & 5 & 2 & 0 & 6 \\ 2 & 5 & 5 & 2 & 6 & 3 & 3 & 6 & 6 & 0 \end{pmatrix} \quad \text{------(4)}$$

It may be seen later that the mating of links with identical binary strings, e.g., links 2 and 4 for the six-link Stephenson chain, do not yield any off-spring as will be evidenced later by the element that $S_{2\text{-}4} = S_{2\text{-}2} = 0$, in the fitness matrix.

Sum of the elements of each row represents the “fitness” of the family of the concerned link.

The fitness of the family of the link-9, for example, can be represented in the form of a string as follows:

38 = 3(6), 2(5), 2(3), 2(2), where 38 is the total fitness of the link-10 family. The diagonal zero elements are not included. Other zeros, if any, must be included. And the elements 6, 5, 3, 2 etc., represent in the descending order, the fitness of the link with respect to the other links. The presence of 3(6) in the above string means that the element 6 appears thrice in the row of the elements. Similarly, 2(5) means that the element 5 appears twice in the row of the elements. Likewise, the string for every link can be written and when all such strings are arranged in the descending order of “fitness”, a string for the chain results. For the chain, Fig. 8.2, the chain string is

4 [40 – 3(6), 2(5), 4, 2(3), 2] – 2 [38 – 3(6), 2(5), 2(3), 2(2)] – 4 [36 – 3(5), 3(4), 3(3)]

------------------------(5)

The presence of 4 and 2 before the square brackets indicates the existence of four and two links respectively with the same strings. A string for every chain can be written in the above manner.

For example, consider another ten-link chain, Fig. 8.3. Working out in the manner explained above, the fitness matrix can be written as

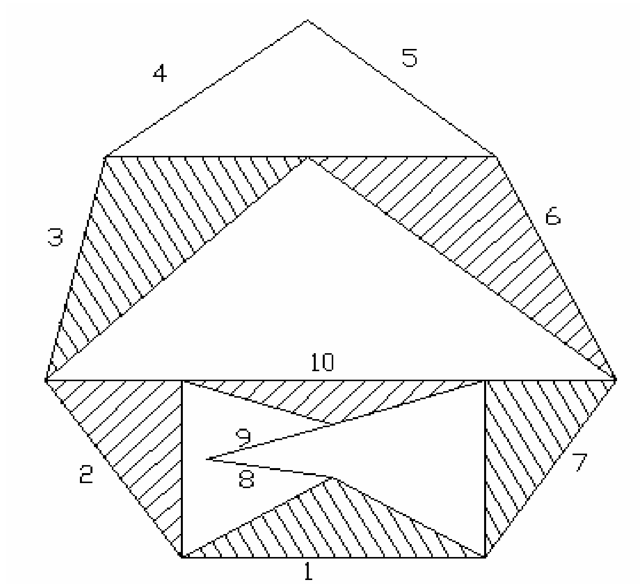


FIG. 8.3 TEN-LINK ONE DEGREE-OF-FREEDOM CHAIN

$$F_1 = \begin{pmatrix} 0 & 6 & 4 & 5 & 5 & 4 & 6 & 5 & 3 & 2 \\ 6 & 0 & 6 & 3 & 5 & 4 & 2 & 3 & 3 & 6 \\ 4 & 6 & 0 & 5 & 1 & 6 & 4 & 5 & 5 & 4 \\ 5 & 3 & 5 & 0 & 4 & 1 & 5 & 4 & 4 & 5 \\ 5 & 5 & 1 & 4 & 0 & 5 & 3 & 4 & 4 & 6 \\ 4 & 4 & 6 & 1 & 5 & 0 & 6 & 5 & 5 & 4 \\ 6 & 2 & 4 & 5 & 3 & 6 & 0 & 3 & 3 & 6 \\ 5 & 3 & 5 & 4 & 4 & 5 & 3 & 0 & 4 & 3 \\ 3 & 3 & 5 & 4 & 4 & 5 & 3 & 4 & 0 & 5 \\ 2 & 6 & 4 & 5 & 5 & 4 & 6 & 3 & 5 & 0 \end{pmatrix}$$

Similarly, the chain string can be written as

2 [40 – 2(6), 3(5), 3(4), 1] – 2 [40 – 2(6), 3(5), 2(4), 3, 2] – 2 [38 – 3(6), 5, 4, 3(3), 2] – 2 [36 – 4(5), 3(4), 3, 1] – 2 [36 – 3(5), 3(4), 3(3)] --------------------(6)

Estimation of row and chain totals of fitness values generation wise can be carried out directly in a simple manner once the generation wise adjacency matrices are formed, by using the formulas (Appendix) without having to formulate *F* matrices.

8.2.3.2 SECOND GENERATION STRINGS AND THEIR MATING ---

Second generation strings are developed in a manner similar to that of the first generation but the elements 1 in the adjacency matrix A_2 correspond to the links separated by only one link and all the other elements will be zero. For example, consider the ten-link chain, Fig. 8.2, with respect to link-2, only links 4, 8 and 9 are separated by one link. Hence only the elements corresponding to links 4, 8 and 9 in the second row of the adjacency matrix A_2 will be 1, all other elements being zero. Similarly, the links 3, 4, 7 and 10 are separated by one link from the link-1. With this understanding, the A_2 matrix for the above chain is

$$A_2 = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \quad \text{------(7)}$$

Each row of the above matrix is considered as the secondary binary string of the concerned link, e.g., for link-5, the binary string is

0 0 1 0 0 0 1 1 1 0

and hence its secondary fitness is four.

The secondary binary strings can be mated following the rules (3) stipulated earlier which in turn yields the secondary fitness matrix, F_2. So, for the chain in Fig. 8.2, F_2 is

$$F_2 = \begin{pmatrix} 0 & 5 & 7 & 6 & 4 & 3 & 5 & 8 & 6 & 4 \\ 5 & 0 & 4 & 7 & 3 & 2 & 6 & 5 & 5 & 5 \\ 7 & 4 & 0 & 5 & 5 & 6 & 2 & 3 & 5 & 5 \\ 6 & 7 & 5 & 0 & 8 & 5 & 3 & 4 & 6 & 4 \\ 4 & 3 & 5 & 8 & 0 & 5 & 7 & 6 & 4 & 6 \\ 3 & 2 & 6 & 5 & 5 & 0 & 4 & 7 & 5 & 5 \\ 5 & 6 & 2 & 3 & 7 & 4 & 0 & 5 & 5 & 5 \\ 8 & 5 & 3 & 4 & 6 & 7 & 5 & 0 & 4 & 6 \\ 6 & 5 & 5 & 6 & 4 & 5 & 5 & 4 & 0 & 8 \\ 4 & 5 & 5 & 4 & 6 & 5 & 5 & 6 & 8 & 0 \end{pmatrix} \quad \text{------(8)}$$

Following the lines of first generation, a chain string for second generation can be written for every chain. For fig. 8.2, it is

4 [48 – 8, 7, 2(6), 2(5), 2(4), 3] – 2 [48 – 8, 2(6), 4(5), 2(4)] – 4[42 – 7, 6, 4(5), 4, 3, 2]

-----------------(9)

Now, working out on the above lines, one can get the secondary fitness matrix and chain string for the chain in Fig. 8.3, as

$$F_2 = \begin{pmatrix} 0 & 5 & 6 & 4 & 4 & 6 & 5 & 5 & 7 & 4 \\ 5 & 0 & 4 & 7 & 3 & 2 & 6 & 5 & 5 & 5 \\ 7 & 4 & 0 & 5 & 5 & 6 & 2 & 3 & 5 & 5 \\ 6 & 7 & 5 & 0 & 8 & 5 & 3 & 4 & 6 & 4 \\ 4 & 3 & 5 & 8 & 0 & 5 & 7 & 6 & 4 & 6 \\ 3 & 2 & 6 & 5 & 5 & 0 & 4 & 7 & 5 & 5 \\ 5 & 6 & 2 & 3 & 7 & 4 & 0 & 5 & 5 & 5 \\ 8 & 5 & 3 & 4 & 6 & 7 & 5 & 0 & 4 & 6 \\ 6 & 5 & 5 & 6 & 4 & 5 & 5 & 4 & 0 & 8 \\ 4 & 5 & 5 & 4 & 6 & 5 & 5 & 6 & 8 & 0 \end{pmatrix}$$

8.2.3.3 THIRD GENERATION STRINGS AND THEIR MATING ---

Link wise, third generation, fourth generation strings, etc. can be generated for every chain. The elements in the adjacency string A_3 correspond to the links separated by two links and all the other elements will be zero. For example, two links in a ten link single d.o.f. chain are separated maximum by four links and it may be necessary to develop strings up to third generation. So, for fig. 8.2 above, third generation fitness matrix F_3 is

$$F_3 = \begin{pmatrix} 0 & 5 & 5 & 6 & 8 & 5 & 5 & 4 & 7 & 3 \\ 5 & 0 & 6 & 5 & 5 & 6 & 2 & 5 & 6 & 6 \\ 5 & 6 & 0 & 5 & 5 & 2 & 6 & 5 & 6 & 6 \\ 6 & 5 & 5 & 0 & 4 & 5 & 5 & 8 & 7 & 3 \\ 8 & 5 & 5 & 4 & 0 & 5 & 5 & 6 & 3 & 7 \\ 5 & 6 & 2 & 5 & 5 & 0 & 6 & 5 & 6 & 6 \\ 5 & 2 & 6 & 5 & 5 & 6 & 0 & 5 & 6 & 6 \\ 4 & 5 & 5 & 8 & 6 & 5 & 5 & 0 & 3 & 7 \\ 7 & 6 & 6 & 7 & 3 & 6 & 6 & 3 & 0 & 6 \\ 3 & 6 & 6 & 3 & 7 & 6 & 6 & 7 & 6 & 0 \end{pmatrix}$$

The chain string is

2 [50 – 2(7), 5(6), 2(3)] – 4 [48 – 8, 7, 6, 4(5), 4, 3] – 4 [46 – 4(6), 4(5), 2]

-----------------(10)

Similarly, fitness matrix for fig. 8.3 can be generated. The chain string for fig. is

4 [44 – 4(6), 2(5), 2(4), 2] – 2 [44 – 3(6), 2(5), 4(4)] – 2 [42 – 4(6), 5, 2(4), 3, 2] – 2 [42 – 7(5), 4, 3] -----------------(11)

8.4 TEST FOR ISOMORPHISM ---

Test for isomorphism among kinematic chains consists of comparing the chain strings generation wise, i.e., first, second and third generation etc. Strings of a chain need be compared respectively with the first, second third generation strings of the other chain. If they are identical, the chains are isomorphic, otherwise distinct. For example, comparison of the strings of both the Fig. 8.2 and 8.3 reveals that the first generation string itself is adequate to show that both the chains are distinct, i.e. non-isomorphic but the comparison of all possible (first, second, third,……) generation strings constitutes the complete test satisfying both the necessary and sufficient requirements, thus making this method unique.

8.5 DISTINCT INVERSIONS ---

A kinematic chain has as many inversions as the number of links in it. However, some of them may be identical. In order to get distinct inversions, it is only needed to compare the family fitness strings of different links of a chain. Usually, first generation strings are adequate but the necessary and sufficient strings will be of first and second generation for eight link chains and so on as described in the earlier sections. If the family fitness strings of two links are identical, the resulting inversions are identical, otherwise distinct.

For the ten-link chain in fig. 8.2, it is easy to see that there are three distinct inversions, i.e., when links (1, 4, 5, 8), (2, 3, 6, 7) and (9, 10) are fixed. On the other hand, the chain in fig. 8.3 has five distinct inversions, i.e., (1, 10), (2, 7), (3, 6), (4, 5) and (8, 9) are fixed.

8.6 BEST CHAINS AND INVERSIONS ---

It may be recalled that the fitness of a link is given by the number of nonzero elements in a binary string. The fitness is in fact related to the number of design parameters such as link invariants (lengths, angles). Thus the first or second generation strings, etc. of a link correspond to the fitness of a link, i.e., greater fitness indicates involvement of more design parameters in generating the motion. Thus it can be noted that greater the family fitness of a link, its contribution to the motion generated by a chain is greater. For example, a chain a greater fitness, i.e. sum of all the elements in the *F* matrix will generate the specified motion such as function generation with greater accuracy or with less structural error. Inversion results when one of the links is grounded (fixed). It is obvious that it is better to fix a link with least fitness, so that more number of design parameters will be left to take part in motion generation. For example, link-2, 3, 6 or 7 in fig. 8.2 has the least fitness as indicated by the sum of the elements in row 2, 3, 6 or 7 in the matrix (4). Hence the inversion obtained by fixing one of these links will be the best inversion of this particular chain. Similar reasoning reveals that the inversion obtained by fixing one of the links 9 or 10 will lead to next best inversion. The third inversion obtained by fixing the links 1, 4, 5 or 8 will not be efficient from the motion generation point of view.

The above chain and other examples not worked out here indicate that in general

(i) Links with least connectivity are preferable for grounding.

(ii) Out of many such links, the sum of the connectivities of all their immediately adjacent links matter, i.e., greater this sum better will be the link for fixing.

(iii) If first generation strings and hence the fitness of two links are equal, compare the second generation strings and their fitness values to find distinct inversions and to rate them. The procedure may be extended if necessary to strings of successive generations.

8.7 BEST INPUT LINK ---

It has been stated in the previous section that it is better to fix a link which has least family fitness value obtained from the fitness matrices. The same logic when extended to the selection of the input link suggests that a link with a maximum or high fitness value will make a good input link. Thus, in fig. 8.2, link 1, 4, 5 or 8 will be the best choice as the input link. The next best choice will be either link 9 or 10. In general higher connectivity links are preferable as input links.

8.8 COMPARISON OF CHAINS ---

Consider two eight-link chains with different link assortments, (i) four binary and four ternary links and (ii) six binary and two quaternary links.

The adjacency matrix and the corresponding fitness matrix can be written for each chain. Sum of all the elements of the F matrix is the fitness value of the chain, generation wise, higher value indicating greater ability of the chain to generate motion, i.e. more accurately. The first generation fitness values of the chains consisting of higher connectivity links will invariably be less irrespective of the chain topology and hence such chains are inferior.

8.9 CONCLUSIONS ---

1) This method presents necessary and sufficient conditions for testing isomorphism uniquely.
2) The genetic approach presented here to detect isomorphism resembles by coincidence the Hamming number technique only up to first generation. But the concept of fitness and successive generation introduced here enables the selection of best chain, best inversion and best input links and makes the test for isomorphism unique.
3) Computations are extremely simple and effort involved is very less.
4) From the motion generation point of view, chains consisting of links with higher connectivity appear to be inferior.
5) Fixing a link with least connectivity will lead to a better inversion and if there are many such links, the one with high connectivity links as adjacent links will be preferable.
6) Links with higher connectivity are preferable as input links.
7) The method is unique as stated earlier; nevertheless, all the 230 distinct ten-link single d.o.f. chains are tested for confirmation.

8.10 ALTERNATIVE FORMULA ---

In general, the chain totals and row totals of every chain are more than adequate except for the special cases, i.e., chains with almost identical topology. If chain totals and row totals of one chain are one-to-one identical to those of the other chain, then the two chains are said to be isomorphic. This saves a lot of effort since the totals are obtained directly from the adjacency matrices by using the formulas given below:

Sum of the fitness values of i^{th} row = $(n - 2)\, C_i + C_n - 2\sum V_i$

Where, C_i = connectivity or adjacency of link-*i*; This is equal to the number of times the element 1 is present in i^{th} row of the adjacency matrix. This will vary for first generation adjacency and second generation adjacency, etc.

n = number of links in the chain

C_n = Sum of the adjacencies of all the links (generation wise)

V_i = Sum of the vertex values of the links directly adjacent to link-*i*. Vertex value of a link is one less than its adjacency value.

For illustration, consider the matrix (2). Here n = 10, C_n = 26

For first row, C_i = 3, V_i = 1 + 2 + 2 = 5

Therefore, sum of the elements of the first row of matrix is = (8 * 3) + 26 – (2*5) = 40

Similarly, for second row, *Ci* = 2, *Vi* = 1 +2 = 3

Sum of the elements = (8*2) + 26 – (2*3) = 36

The sum of the elements for *4*th, *5*th and *8*th row is same as *1*st row and that of *3*rd, *6*th and *7*th row is same as of *2*nd row which shows that these are identical links and hence identical inversions will be obtained by fixing these links.

Now, for *9*th and *10*th row, C_i = 3, V_i = 2 + 2 + 2 = 6

So sum of the elements = (8*3) + 26 – (2*6) = 38

The fitness value of the chain, generation wise, may be obtained by summing up all the row values obtained as above. So fitness value of the chain in fig. 8.1 is = 40 + 40 + 40 + 40 + 36 + 36 + 36 + 36 + 38 + 38 = 380

Also it may directly be determined by using the expression

$$2\left\{\sum_{i=1}^{n}(n-1)\,Ci \;-\; 2\sum_{i=1}^{n}\frac{Ci\,(Ci-1)}{2}\right\}$$

So for fig. 8.2, it is = 2 [9 (12 + 8 + 6) – 2 (12 + 4 + 6)] = 380

Similarly, for fig. 8.3, fitness value is obtained directly as 2 [243 – 50] = 386

CHAPTER – 9
IDENTIFICATION OF ISOMORPHISM AND STRUCTURAL PROPERTIES OF KINEMATIC CHAINS WITH THE HELP OF ADJACENCY MATRICES

9.1 INTRODUCTION ---

Mahere, Vishesh; Nigam, S.P. [37] presented a method based on the Adjacency Matrices of kinematic chains to identify Isomorphism, among kinematic chains and among inversions of given kinematic chain and for identification of structural properties of kinematic chains as Degeneration identification, identification of Type of freedom. The advantage of this method is that it covers all planar kinematic chains of one, two and three degree of freedom at once for identification of Isomorphism and for structural properties.

9.2 MATHEMATICAL FORMULATION OF ADJACENCY MATRICES [35] ---

Suppose A, A' are adjacent matrices of mechanism kinematic chains; $k_1, k_2 \ldots k_n$; $k_1', k_2' \ldots k_n'$ are Eigen values of A, A' and $x_1, x_2 \ldots x_n$; $x_1', x_2' \ldots x_n'$ are Eigen vectors of A, A' and the *n* independent Eigenvectors compose nonsingular matrices X and X', respectively, with them as column vectors.

$X = \{x_1, x_2 \ldots x_n\}$

$X' = \{x_1', x_2' \ldots x_n'\}$

According to matrix theory,

$X^{-1}AX = \text{diagonal}\ (k_1, k_2 \ldots k_n)$ -----------------------(1)

$X^{-1}A'X' = \text{diagonal}\ (k_1', k_2' \ldots k_n')$ ----------------------(2)

If the kinematic chains represented by A, A' are Isomorphic, their eigen values can be modified to be in same order by applying interchange of rows and columns at the same time.

Let $X_n' = X'T_c$ ---------------------(3)

$$\text{diagonal}\ (k_1, k_2 \ldots k_n) = T_c^{-1}\ \text{diagonal}\ (k_1', k_2' \ldots k_n')\ T_c = T_c^{-1}X'^{-1}A'X'T_c$$

Equ. (2) can be written as

$X_n^{-1}A'X_n' = \text{diagonal}\ (k_1, k_2 \ldots k_n)$ ---------------------(4)

Theorem: The kinematic chains are Isomorphic, if and only if their Eigen matrices of corresponding Eigen vectors have the relation of

$X_n' = X'T_c$

Proof. (1) Necessity: If A, A' are isomorphic, they can be transferred by interchange of rows or columns at the same time as Eq. (5)

$T_2 A T_1 = A'$ ---------------------(5)

Where T_2 and T_1 are left multiplication matrix and right multiplication matrix, and can be obtained by same interchange of rows and columns from the identity matrix. Such conclusions can be obtained as

$T_2 = T_1 T$

$T_2 = T_1^{-1}$

$T_2 = T_1$

From Equs. (4) and (5)

$X_n^{-1}(T_1^{-1} A T_1)X_n' = (T_1 X_n')^{-1} A (T_1 X_n') = \text{diagonal}(k_1, k_2 \ldots k_n)$

----------------------(6)

By comparing with Eq. (1), one can obtain

$T_1 X_n' = X$ ---------------------(7)

This is the relationship between the matrices composed by Eigen vectors of the two kinematic chains. So, Eigen values and Eigen vectors of their Adjacent Matrices are equal in correspondence.

(2) Sufficiency: Conversely, suppose the Eigen values and correspondent Eigen vectors of A and A' are satisfied by equation

$T_1 X_n' = X$

According to Eq. (4), one obtains

$$X_n'^{-1} T_1^{-1} T_1 A' T_1^{-1} T_1 X_n' = (T_1 X_n')^{-1} (T_1 A' T_1^{-1}) (T_1 X_n')$$

$$= X^{-1} T_1 A' T_1^{-1} X$$

$$= \text{diagonal}(k_1, k_2 \ldots k_n)$$ ---------------------(8)

So $T_1 A' T_2 = A$ ---------------------(9)

This is to say, the kinematic chains of A and A' are isomorphic.

9.3 ISOMORPHISM IDENTIFICATION AMONG KINEMATIC CHAINS BY ADJACENCY MATRICES ---

For each chain, Adjacency Matrix is formed which is a combination of 0, 1 digits. Adjacency Matrix is a square matrix of *n* x *n*, where *n* is number of links of given chain. In adjacency Matrix, if any link has connectivity with other link then it has digit 1 otherwise 0 and for link itself, digit is 0.

If two chains have same Eigen Values and same Eigen vectors, then according to mathematical theorem of Adjacency Matrices these are isomorphic chains. According to theorem, the Eigen matrices can be equivalent by row exchange.

9.3.1 EXAMPLES ---

(1) Consider example of nine links planar kinematic chain of two degree of freedom as shown in Fig. 9.1(a).

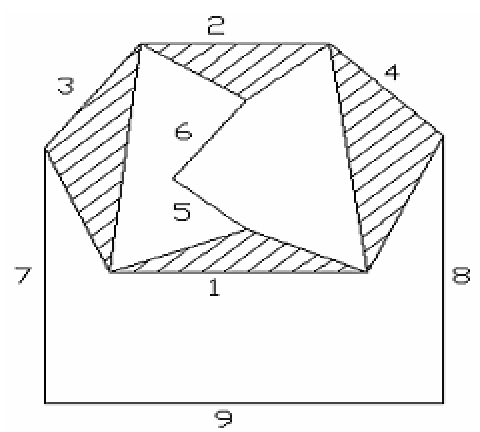


FIG. 9.1(a) NINE LINK TWO DEGREE-OF-FREEDOM KINEMATIC CHAIN

Adjacency Matrix for Fig. 9.1(a)

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

```
>> [eigvec, eigval] = eig (A)
```
---- Matlab Command for getting Eigen value and Eigen vector

Table 9.1:- Eigen values and Eigen vectors of Fig. 9.1(a)

Eigval	-2.4557	-1.6180	-1.2781	-1.0000	0.6180	0.6658	1.0000	1.4781	2.5899

Eigvec

-0.4198	0.3717	-0.2855	0.0000	-0.6015	-0.1732	-0.0000	-0.1715	0.4276
-0.4198	-0.3717	-0.2855	0.0000	0.6015	-0.1732	0.0000	-0.1715	0.4276
0.4547	0.0000	0.1198	-0.5000	0.0000	-0.3167	-0.5000	0.0526	0.4193
0.4547	0.0000	0.1198	0.5000	-0.0000	-0.3167	0.5000	0.0526	0.4193
0.1215	-0.6015	0.1253	-0.0000	-0.3717	0.5182	-0.0000	-0.3587	0.2689
0.1215	0.6015	0.1253	-0.0000	0.3717	0.5182	0.0000	-0.3587	0.2689
-0.2771	-0.0000	0.4179	0.5000	0.0000	0.1355	-0.5000	0.4208	0.2306
-0.2771	-0.0000	0.4179	-0.5000	-0.0000	0.1355	0.5000	0.4208	0.2306
0.2256	0.0000	-0.6539	0.0000	-0.0000	0.4069	0.0000	0.5693	0.1781

(2) Consider another example of nine links planar kinematic chain of two degree of freedom as shown in Fig. 9.1(b).

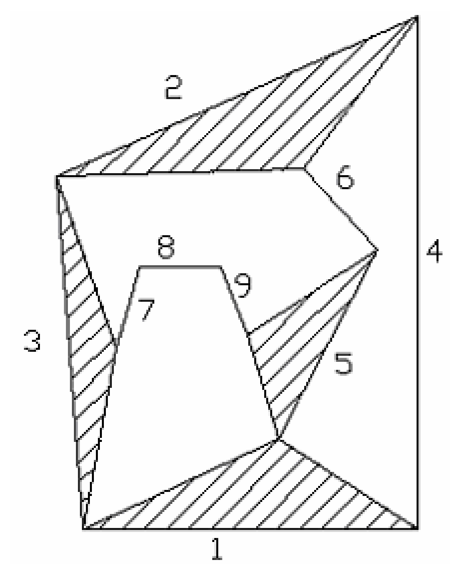


FIG. 9.1(b) NINE LINK TWO DEGREE-OF-FREEDOM KINEMATIC CHAIN

Adjacency Matrix for Fig. 9.1(b)

$$B = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

>> [eigvec, eigval] = eig (B) ---- Matlab Command for getting Eigen value and Eigen vector

<u>Table 9.2</u>:- Eigen values and Eigen vectors of Fig. 9.1(b)

Eigval	-2.3585	-1.9076	-1.2927	-0.5739	0.1994	0.6298	1.2527	1.4834	2.5674

Eigvec

0.4766	-0.1047	0.3822	-0.1358	-0.1152	0.6106	-0.0552	-0.1341	0.4361
0.3842	0.4369	0.0140	0.2920	0.2442	-0.4808	0.0842	-0.3316	0.4102
-0.4917	-0.1537	0.0869	0.4461	-0.3826	0.0490	0.4403	-0.0979	0.4193
-0.3650	-0.1741	-0.3065	-0.2722	0.6467	0.2061	0.0231	-0.3139	0.3296
-0.2675	0.5275	-0.2745	-0.0960	-0.2871	0.1295	-0.5326	0.2129	0.3706
-0.0495	-0.5055	0.2015	-0.3415	-0.2155	-0.5579	-0.3580	-0.0800	0.3041
0.2988	-0.0390	-0.5085	-0.4122	-0.2053	-0.0989	0.5226	0.3204	0.2303
-0.2131	0.2281	0.5704	-0.2096	0.3416	-0.1113	0.2144	0.5733	0.1720
0.2038	-0.3961	-0.2289	0.5325	0.2734	0.0288	-0.2541	0.5300	0.2114

Result shows that Fig. 9.1(a) and Fig. 9.1(b) nine links two degree of freedom chains differ in their Eigen values and in Eigen vectors, thus according to mathematical theorem of Adjacency Matrix, these are non-isomorphic chains.

9.4 ISOMORPHISM IDENTIFICATION AMONG INVERSIONS OF A GIVEN CHAIN BY ADJACENCY MATRICES ---

For any given chain, possible number of inversions can be obtained by fixing its different links but at the same time it may happen that its inversion may be isomorphic or structurally same, thus Isomorphism identification among inversions becomes important from the point of view of getting unique number of possible inversions. This method of Eigen value and Eigen vector has the capability to reveal isomorphism among inversions of a given chain and through that, one can get number of possible unique structurally different inversions of a given chain.

9.4.1 EXAMPLES ---

In fig. 9.1 (a), nine link two d.o.f. chain has row wise similarity of Eigen vectors of corresponding links (without considering its sign), as shown in Table 9.1. This shows that possibly structurally different inversions are – one by fixing any one of links (1, 2), second by fixing any one of links (3, 4), third by fixing any one of links (5, 6), fourth by fixing any one of links (7, 8) and fifth by fixing link 9. Results match with results available in the literature [30]. Thus out of 9 possible inversions there will only be 5 unique inversions of this chain which are structurally distinct.

Similarly, in fig. 9.1 (b), nine link two d.o.f. chain has row wise variation of Eigen vectors of all links (without considering its sign), as shown in Table 9.2. This shows that possibly structurally different inversions are obtained by fixing links 1, 2, 3, 4, 5, 6, 7, 8 and 9. Results match with results available in the literature [30]. Thus out of 9 possible inversions there will be 9 unique inversions of this chain which are structurally distinct.

9.5 DETERMINATION OF STRUCTURAL PROPERTIES OF KINEMATIC CHAIN BY ADJACENCY MATRIX ---

9.5.1 DEGENERATION IDENTIFICATION IN A GIVEN KINEMATIC CHAIN ---

Degeneration is defined as presence of rigid sub chains of zero degree of freedom in a kinematic chain. Detection and isolation of degenerate and isomorphic chains is very important for having the distinct and valid kinematic chains with a specified number of links and degree of freedom.

The chains generated by any method need to be tested for Degeneration, i.e. presence of rigid sub chains. For planar kinematic chains, Grubler's criteria of degree-of-freedom is = 3 (N – 1) – 2J

Here N = Number of links of given kinematic chain

J = Number of joints of given kinematic chain

Thus for having zero degree-of-freedom Grubler's criteria becomes –

$$3 (N - 1) - 2J = 0$$

or $$N = (2J + 3) / 3$$

Thus for –

J = 3, N = 3

J = 6, N = 5

J = 9, N = 7

TYPE -1

For J = 3, N = 3

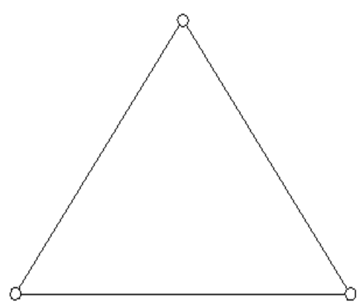


FIG. 9.2 THREE LINKS ZERO DEGREE OF FREEDOM CHAIN

Adjacency Matrix for Fig. 9.2 three links zero d.o.f. chain

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

TYPE – 2

For J = 6, N = 5

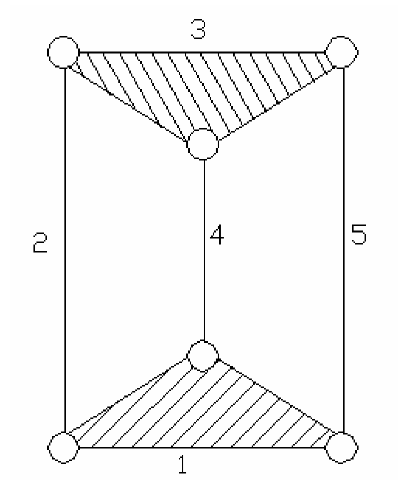


FIG. 9.3 FIVE LINKS ZERO DEGREE OF FREEDOM CHAIN

Adjacency Matrix for Fig. 9.3 five links zero d.o.f. chain

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

TYPE – 3

For J = 9, N = 7

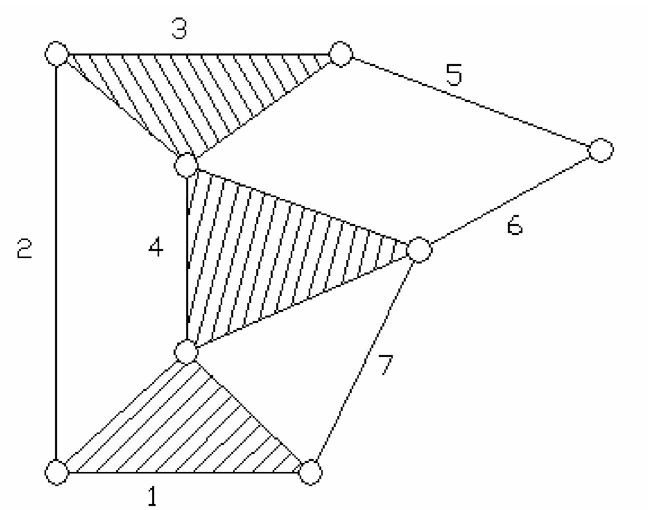


FIG. 9.4 SEVEN LINKS ZERO DEGREE OF FREEDOM CHAIN

In this rigid sub chain, links 1, 4, 7 are already forming a sub chain of Type – 1 thus there is no need of finding this Type – 3 rigid sub chain presence.

9.5.1.1 METHOD OF IDENTIFICATION OF DEGENERATION ---

Identification of Degeneration with the help of Adjacency Matrix Method includes steps:-

1. Identify by observation of type of degenerate loop i.e. loop of zero Degree-of freedom in a given kinematic chain.
2. Confirm presence of Degenerate loop in a given kinematic chain by comparing Degenerate loop's Adjacency matrix present in given kinematic chain's Adjacency matrix with Adjacency matrix of that type of loop defined in Type 1, type 2 and type 3.

EXAMPLE -1

Considering six links one degree-of-freedom planar kinematic chain as shown in fig. 9.5

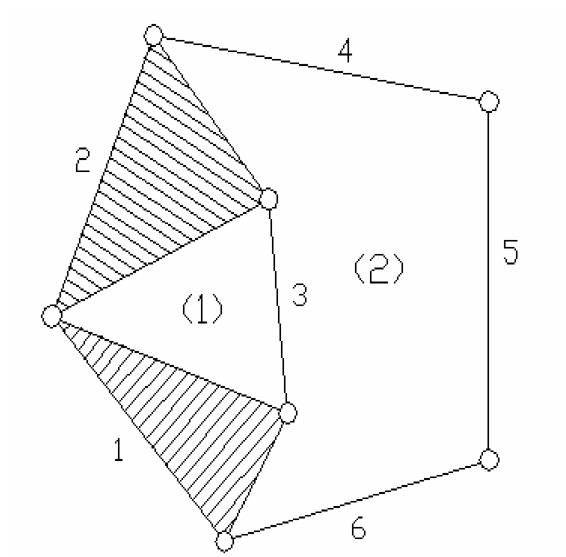


FIG. 9.5 SIX LINKS ONE DEGREE-OF-FREEDOM CHAIN

Adjacency Matrix for Fig. 9.5 six links one d.o.f. chain

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

In fig. 9.5 two loops are present. Out of these two loops, loop 1 formed by links 1, 2 and 3 is of type first loop of degenerate sub chain and contents of this loop in rows

1, 2 and 3 and in columns 1, 2 and 3 in Adjacency matrix [A] of given chain confirm its presence in given kinematic chain.

EXAMPLE – 2

Considering eleven links two degree-of-freedom planar kinematic chain as shown in fig. 9.6

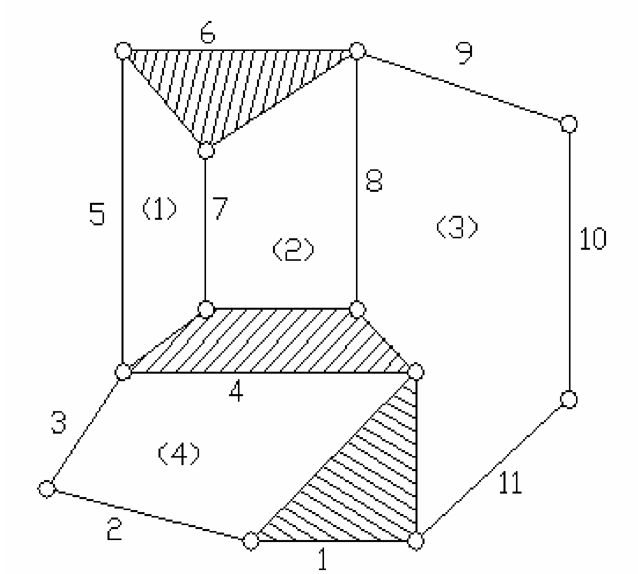


FIG. 9.6 ELEVEN LINKS TWO DEGREE-OF-FREEDOM CHAIN

Adjacency Matrix for Fig. 9.6 eleven links two d.o.f. chain

$$B = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

In fig. 9.6 out of four loops 1, 2, 3 and 4, loops 1 and 2 form a sub chain of type second of degenerate chain. The contents of this sub chain in Adjacency Matrix of given chain [B], formed by loop 1 and loop 2, in rows 4, 5, 6, 7, 8 and in columns 4, 5, 6, 7, 8 confirm presence of second type of degenerate chain.

9.5.2 TYPE OF FREEDOM IN MULTI DEGREE OF FREEDOM CHAIN ---

It is an important consideration in multi degree of freedom chain as what is the Type of freedom of a particular chain, in order to select the actuator (input) links.

A multi degree of freedom kinematic chain possesses one of the following types of freedom –

1. Full degree of freedom
2. Partial degree of freedom
3. Fractionated degree of freedom

9.5.2.1 METHOD OF IDENTIFICATION OF TYPE OF FREEDOM ---

This method includes three steps of identification of type of freedom –

1. Finding degree of freedom of different loops in a given kinematic chain.
2. Identification of loops having degree of freedom less than degree of freedom of a given kinematic chain.
3. Check of Adjacency matrix of loops having degree of freedom less than degree of freedom of a given chain.

9.5.2.1.1 FULL DEGREE OF FREEDOM ---

This gives total freedom of selection of any link of given multi degree of freedom kinematic chain for input. For example, if two d.o.f. kinematic chain has Full degree of freedom then any two links of this chain can be selected as input links.

Given kinematic chain of multi d.o.f. may has Full degree of freedom i.e. freedom of selection of any links as input link of given kinematic chain, if and only if –

"Loops of given kinematic chain have degree of freedom equal or greater than degree of freedom of given kinematic chain."

Step 1 – As shown in Fig. 9.7 nine links two d.o.f chain has loops 1, 2 and 3.

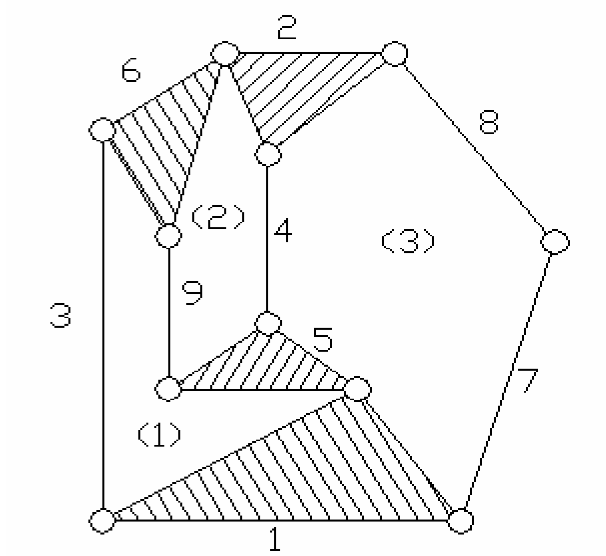


FIG. 9.7 NINE LINKS TWO DEGREE OF FREEDOM CHAIN

Step 2 –

Loop 1 – 5 links loop of links 3, 6, 9, 5, 1 of – 2 degree of freedom

Loop 2 – 5 links loop of links 9, 6, 2, 4, 5 of – 2 degree of freedom

Loop 3 – 6 links loop of links 4, 5, 1, 7, 8, 2 of – 3 degree of freedom

Thus given kinematic chain has loops of degree of freedom equal or greater than degree of freedom of given kinematic chain, thus this kinematic chain has Full degree of freedom.

9.5.2.1.2 PARTIAL DEGREE OF FREEDOM ---

This gives partial freedom of selection of links of given multi degree of freedom kinematic chain for input. For example, if two d.o.f. kinematic chain has Partial degree of freedom then any two links of this chain can not be selected as input links.

Given kinematic chain of multi d.o.f. may has Partial degree of freedom i.e. partial freedom of selection of any links as input link of given kinematic chain, if and only if –

"Loops of given kinematic chain have degree of freedom less than degree of freedom of given kinematic chain."

Step 1 – As shown in Fig. 9.8 nine links two d.o.f chain has loops 1, 2 and 3.

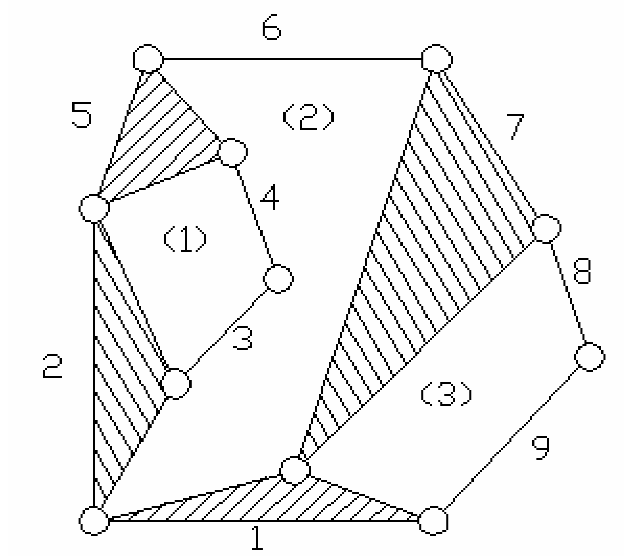


FIG. 9.8 NINE LINKS TWO DEGREE OF FREEDOM CHAIN

Step 2 –

Loop 1 – 4 links loop of links 2, 3, 4, 5 of – 1 degree of freedom

Loop 2 – 7 links loop of links 1, 2, 3, 4, 5, 6, 7 of – 2 degree of freedom

Loop 3 – 4 links loop of links 1, 7, 8, 9 of – 1 degree of freedom

Thus given kinematic chain has loops of degree of freedom less than degree of freedom of given kinematic chain, thus this kinematic chain has Partial degree of freedom.

Step 3 – Adjacency Matrix for Fig. 9.8 nine links two degree of freedom chain –

$$C = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

For loop 1, Adjacency matrix between links 2, 3, 4, 5 contents from Adjacency matrix [C]

$$D = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

This matrix is similar to simple binary link 4 link 1 degree of freedom chain's matrix, as shown below –

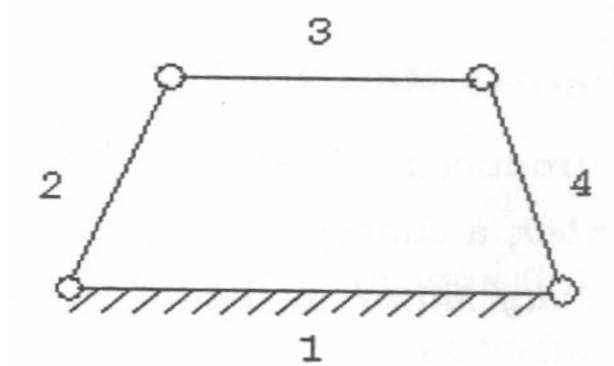


FIG. 9.9 FOUR LINKS ONE DEGREE OF FREEDOM CHAIN

Adjacency Matrix of four bar loop of Fig. 9.9

$$E = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

This gives the confirmation of presence of four bar loop of 1 degree of freedom in a given nine links two degree of freedom kinematic chain, thus it has partial degree of freedom. Thus there is restriction of two links as input links from four bar loop links.

9.5.2.1.3 FRACTIONATED DEGREE OF FREEDOM ---

Fractionated degree of freedom of a particular planar kinematic chain gives freedom of cutting a link of highest connectivity which is sharing all loops of chain so that two separate chains are created. The degree of freedom of each separated chain is such that their combined degree of freedom is equal to that of the original chain.

In a given kinematic chain Fractionated degree of freedom will exist if and only if –

"Highest connectivity link of given kinematic chain sharing all loops and it has connectivity more than 50% of total connectivity i.e. in Adjacency matrix of given kinematic chain, highest connectivity link's row and column consist more '1' than '0'."

<u>Step 1</u> – As shown in Fig. 9.10 nine links two degree of freedom kinematic chain –

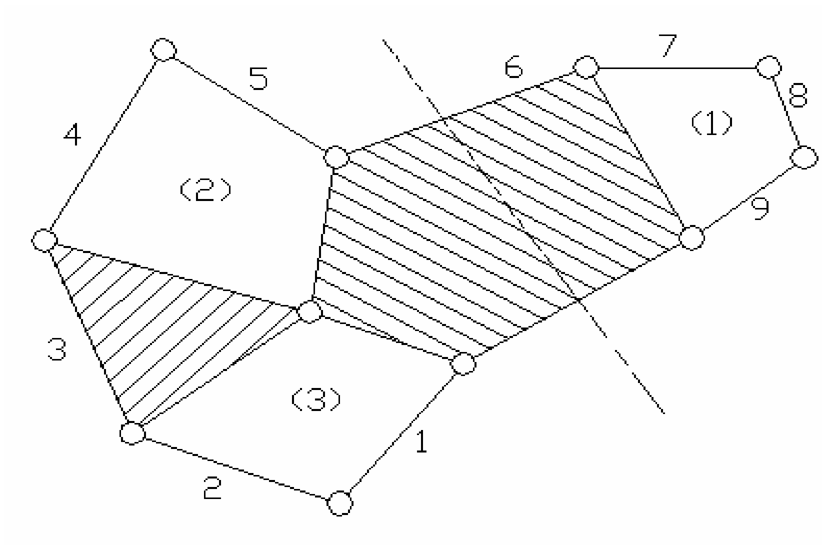


FIG. 9.10 NINE LINKS TWO DEGREE OF FREEDOM CHAIN

Adjacency Matrix for Fig – 9.10 nine links two degree of freedom chain

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Step 2 –

In adjacency matrix shown in step-1, link no. 6 is highest connectivity link and it's row and column consist more '1' digit than '0' digit i.e. more than 50% and as shown in Fig. 37 this link no.6 shares all loops of given nine link two degree of freedom kinematic chain.

This shows presence of Fractionated degree of freedom in this given chain. Now this link no. 6 can be cut into two parts forming two different kinematic chains i.e. first kinematic chain of four links 6, 7, 8, 9 of one degree of freedom and second kinematic chain of six links 1, 2, 3, 4, 5, 6 of one degree of freedom. The degree of freedom of these two separate chains gives a sum of degree of freedom of combined kinematic chain of Fig. 9.10

9.5.3 CONCLUSIONS ---

This method of Adjacency Matrices has simplicity in its process and of time savvy nature with perfect reliability from the point of view of results. It has adopted a very strong tool of mathematics i.e. MATLAB SOFTWARE, for the identification of Isomorphism through Eigen values and Eigen vectors, in both cases of kinematic chains and inversions which has advantage of showing results within a second over showing results after hectic calculations. The main advantage of this method over other previous methods is that it covers all planar chains of one, two and three degree of freedom at once not only for identification of Isomorphism but also for identification of structural properties as Degeneration and Type of freedom. So this method adopts a very simple and compact approach in identification of isomorphism and structural properties of kinematic chains.

CHAPTER – 10
A NEW METHOD TO IDENTIFY ISOMORPHISM IN KINEMATIC CHAINS USING JOINT – JOINT MATRIX

10.1 INTRODUCTION ---

Ali Hasan, Khan R.A., Aas Mohd. [44] proposed a new method to identify the distinct mechanisms (DM) from a given kinematic chain. This method is easy to compute and reliable. The KC are represented in the form of Joint-Joint [JJ] matrices. Two structural invariants derived from the characteristic polynomials of the [JJ] matrix of the KC are the sum of the absolute characteristic polynomial coefficients (shortened as **∑JJ**) and maximum absolute value of the characteristic polynomial coefficient (shortened as **MJJ**). These invariants are used as the composite identification number of a KC and mechanisms. It is capable of detecting isomorphism in all types of planar kinematic chains with same or different kinematic pairs (**KP**). This method is also used to obtain more information among mechanisms KC or in one mechanism kinematic chain. Critical study of kinematic and machine performance has revealed that the type of pair and coefficient of friction affects the performance of joints. In the present work, the degree of freedom (shortened as **d.o.f**) of KP have been taken into account. This study will help the designer to select the best KC and mechanisms to perform the specified task and avoid duplication at the conceptual stage of design. The application this study is in research and development industries.

10.2 OBJECTIVES ---

In this method, the authors' objective is to develop a new, reliable, and efficient method to detect isomorphism in KC. It will help the designer to select the best KC to perform the desired task, at the conceptual stage of design. The proposed method is presented by comparing the structural invariants ∑JJ and MJJ of [JJ] matrices. These invariants may also be used not only to detect isomorphism in the KC and mechanisms having simple joints but also the KC having Co-spectral graph. The method is explained with the help of examples of planar KC having all simple joints.

10.3 DEFINITIONS OF TERMINOLOGY ---

A number of new terms have been developed in the present work for a KC. They are defined in the following paragraph. The joints of a chain are assigned positive integers 1, 2, 3…, n as their names while the small English letters a, b, c, etc. are used for labeling the links of a KC. It may be mentioned here that links and joints of a chain are arbitrarily labeled.

Degree of link d (li): The degree of link actually represents the type of link like binary, ternary, quaternary etc. Let the degree of i^{th} link in a KC be designed as d (l_i).

d (l_i) =2, for binary link, d (l_i) =3, for ternary link,
d (l_i) =4, for quarter nary link, d (l_i) =k, for k-nary link.

10.4 MATRIX REPRESENTATION OF THE KINEMATIC CHAIN ---

The (0, 1) adjacency matrix and the distance matrix are generally used to represent the kinematic graph of a KC. The adjacency matrix shows only the connectivity between adjacent vertices/links. The distance matrix has also the relation between the links that are not directly connected to each other in the form of shortest path distance. However, both adjacency and distance matrices are not able to furnish the information about the types of links those are directly connected with a joint or with the shortest path distances respectively. A generalized matrix representation is made in literature [4] in which the elements of adjacency matrix a_{ij} represent the multiplicity (type) of the joint. The value of a_{ij} is 1 if the joint between i^{th} and j^{th} link is a simple joint, 2 if it is double joint, 3 if it is ternary joint and so on. In the present paper the [JJ] matrix is used which is based upon the connectivity of the joints through a link.

10.4.1 THE JOINT-JOINT [JJ] MATRIX ---

This matrix is based upon the connectivity of the joints through the links and defined, as a square symmetric matrix of size n x n, where n is the number of joints in a KC.

$$[JJ] = \{ L_{ij} \}_{n \times n} \qquad \text{---------- (1)}$$

Where

$$L_{ij} \begin{cases} = \text{Degree of link between } i^{th} \text{ and } j^{th} \text{ joints those are directly connected} \\ = 0, \text{ if joint } i \text{ is not directly connected to joint } j \end{cases}$$

Off course all the diagonal elements

$$L_{ii} = 0$$

Thus the form of [JJ] matrix will be:

$$[JJ] = \begin{pmatrix} 0 & L_{12} & L_{13} \text{---------} L_{1n} \\ L_{21} & 0 & L_{23} \text{---------} L_{2n} \\ - & \text{-------------------} & \\ - & \text{-------------------} & \\ - & \text{-------------------} & \end{pmatrix}$$

10.4.2 CHARACTERISTIC POLYNOMIAL OF [JJ] MATRIX ---

The characteristic polynomial is generally derived from (0, 1) adjacency matrix. The roots of n^{th} order characteristic polynomial are the set of n-eigen values called as Eigen spectrum. Many researchers have reported co-spectral graphs (the non-isomorphic graphs having same Eigen spectrum derived from (0, 1) adjacency matrix). The Proposed [JJ] matrix has additional information about the types of links existing in a KC. Therefore, it is expected that the characteristic polynomial and its coefficient will be unique to clearly identify the KC and even KC with co-spectral graphs. D(λ) gives the characteristic polynomial of [JJ] matrix. The polynomial of degree n is given by equation (2).

$$| (JJ - \lambda I) | = \lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + \text{------} \; a_{n-1}\lambda + a_n \qquad \text{------------(2)}$$

Where

n = number of simple joints in KC and

1, a_1, a_2, a_{n-1}, a_n are the characteristic polynomial coefficients.

The two important properties of the characteristic polynomials are:

1. The sum of the absolute value of the characteristic polynomial coefficients (∑JJ) is an invariant for a [JJ] matrix. i.e.

$| 1 | + | a_1 | + | a_2 | + \text{-----} + | a_{n-1} | + | a_n | = \text{invariant}$

2. The maximum absolute value of the characteristic polynomial coefficient (MJJ) is another invariant for a [JJ] matrix.

10.4.3 STRUCTURAL INVARIANTS '∑JJ' AND 'MJJ' ---

The values of characteristic polynomial coefficients are invariants for a [JJ] matrix. To make these [JJ] matrix characteristic polynomial coefficients as a powerful single number characteristic index, new composite invariants proposed. These indices are **'∑JJ'** and **'MJJ'**. These invariants are unique for a [JJ] matrix and may be used as identification numbers to detect the isomorphism among simple jointed KC and mechanisms. The characteristic polynomial coefficient values are the characteristic invariants for the chains and mechanisms. Many investigators have reported co-spectral graph (non-isomorphic graph having same Eigen spectrum). But these Eigen spectra (Eigen values or characteristic polynomial coefficient) have been determined from (0, 1) adjacency matrices. The proposed [JJ] matrix provides distinct set of characteristic polynomial coefficients of the KC having co-spectral graph. Therefore, it is hoped that the structural invariants **'∑JJ' and 'MJJ'** are capable of characterizing all KC and mechanisms uniquely. Hence, it is possible to detect isomorphism among all the given KC.

10.5 ISOMORPHISM OF KINEMATIC CHAINS ---

Theorem:

Two similar square symmetric matrices have the same characteristic polynomial.

Proof:

Let the two KC are represented by the two similar matrices A and B such that $B = P^{-1}AP$, taking into account that the matrix λI commutes with the matrix P and $| P^{-1} | = | P |^{-1}$. Since the determinant of the product of two square matrices equals the product of their determinants,

$$
\begin{aligned}
| B - \lambda I | &= | P^{-1}AP - \lambda I | \\
&= | P^{-1} (A - \lambda I) P | \\
&= | P^{-1} | \, | (A - \lambda I) | \, | P | \\
&= | A - \lambda I |
\end{aligned}
$$

Hence, $D(\lambda)$ of 'A' matrix = $D(\lambda)$ of 'B' matrix.

$D(\lambda)$ = characteristic polynomial of the matrix.

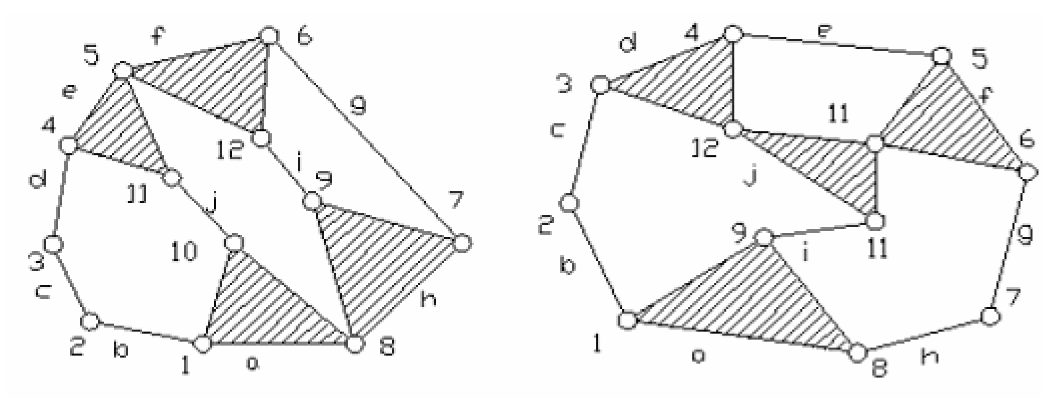
It means that if $D(\lambda)$ of two [JJ] matrices representing two KC is same, their structural invariants '∑JJ' and 'MJJ' will also be same and the two KC are isomorphic otherwise non-isomorphic chains.

10.6 PROPOSED TEST BASIS ---

The kinematic chains are complex chains of combination of binary, ternary, and other higher order links. These links are joined together by simple pin joints. It is the assembly of link/pair combination to form one or more closed circuits. While considering structural equivalence it is essential to consider type of links/joints and layout of the links in the assembly. An identification number is assigned to links. Thus, a binary link has a value of two, ternary three, quaternary four and so on. Link values are used to assign values to [JJ] matrix and these are utilized to identify the layout of the KC. For detecting isomorphism in KC, a [JJ] matrix is determined and composite structural invariants [∑JJ] and [MJJ] of [JJ] matrix are determined using software **MAT LAB.** If the structural invariants '∑JJ' and 'MJJ' of the two KC are same then both the KC are considered as isomorphic otherwise non-isomorphic.

10.7 ILLUSTRATIVE EXAMPLE (MULTIDEGREE FREEDOM CHAINS) ---

The example concerns two kinematic chains with 10 bars, 12 joints, and three-degree of freedom as shown in Fig.10.1. The task is to examine whether these two chains are isomorphic.



10.1 (a) **10.1 (b)**

FIG. 10.1 TEN – BAR KINEMATIC CHAIN, THREE DEGREE OF FREEDOM

[JJ] Matrix of KC shown in Fig. 10.1(a)

[JJ] Matrix of KC shown in Fig. 10.1 (a) using equation (1) is given by equation (3).

$$
[JJ] = \begin{pmatrix}
0 & 2 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 3 & 0 & 0 \\
2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 3 & 0 \\
0 & 0 & 0 & 3 & 0 & 3 & 0 & 0 & 0 & 0 & 3 & 3 \\
0 & 0 & 0 & 0 & 3 & 0 & 3 & 0 & 0 & 0 & 0 & 3 \\
0 & 0 & 0 & 0 & 0 & 3 & 0 & 3 & 3 & 0 & 0 & 0 \\
3 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 3 & 3 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 3 & 3 & 0 & 0 & 0 & 2 \\
3 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 2 & 0 \\
0 & 0 & 0 & 3 & 3 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\
0 & 0 & 0 & 0 & 3 & 3 & 0 & 0 & 2 & 0 & 0 & 0
\end{pmatrix} \quad \text{-------------------(3)}
$$

```
>> t = poly (jj)
   sum (abs(t))
```

---- Matlab Command for getting characteristic polynomial coefficients and structural invariants

The characteristic polynomial coefficients for KC shown in Fig. 10.1 (a)

The characteristic polynomial coefficients derived form [JJ] matrix of KC shown in Fig.10.1 (a) are:

0.0000, -0.0000, -0.0001, -0.0002, 0.0063, 0.0170, -0.1055, -0.3730, 0.4376, 2.1856, 0.0372, -3.5938, -1.6171

Structural Invariants for KC Shown in Fig. 10.1 (a)

The set of structural invariant for KC Shown in Fig.10.1 (a), derived from [JJ] matrix are:

[∑JJ] = 8.3734e+006 [MJJ] =3.5938e+006

Similarly

[JJ] Matrix of KC shown in Fig. 10.1 (b)

[JJ] Matrix of KC shown in Fig. 10.1(b) using equation (1) is given by equation (4).

$$[JJ] = \begin{pmatrix} 0 & 2 & 0 & 0 & 0 & 0 & 0 & 3 & 3 & 0 & 0 & 0 \\ 2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\ 0 & 0 & 3 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & 2 & 0 & 3 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 2 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 3 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 3 & 3 \\ 0 & 0 & 0 & 0 & 3 & 3 & 0 & 0 & 0 & 3 & 0 & 3 \\ 0 & 0 & 3 & 3 & 0 & 0 & 0 & 0 & 0 & 3 & 3 & 0 \end{pmatrix} \quad \text{----------------(4)}$$

```
>> t = poly (jj)
   sum (abs(t))
```

---- Matlab Command for getting characteristic polynomial coefficients and structural invariants

The characteristic polynomial coefficients for KC shown in Fig. 10.1 (b)

The characteristic polynomial coefficients derived form [JJ] matrixes of KC shown in Fig. 10.1(b) are:

0.0000, 0, -0.0001, -0.0002, 0.0058, 0.0161, -0.0965, -0.3525, 0.4307, 2.2696, 0.1572, -2.9393, 0.7465

Structural Invariants for KC Shown In Fig. 10.1 (b)

The set of structural invariant for KC Shown in Fig. 10.1 (b), derived from [JJ] matrixes are:

[∑JJ] = 7.0147e+006 [MJJ] = 2.9393e+006

This method reports that both the KC shown in Fig. 10.1(a) and Fig. 10.1(b) are non-isomorphic as the set of values of [∑JJ] and [MJJ] are different for both the KC. Note that by using other method summation polynomials [18], the same conclusion is obtained.

10.8 RESULTS ---

The proposed composite structural invariants [∑JJ] and [MJJ] of [JJ] matrix of the KC are able to detect isomorphism in the KC and even KC with co-spectral graphs.

All the simple jointed 1-F, 8-links 16 KC and 1-F, 10-links 230 KC along with 2-F, 9-links 40 KC have been tested successfully for their non- isomorphism.

10.9 CONCLUSIONS ---

In this way, a simple, efficient, and reliable method to identify isomorphism is proposed. By this method, the isomorphism of mechanisms kinematic chains can easily be identified. It incorporates all features of the chain and as such, violation of the isomorphism test is rather difficult. In this method, the characteristic polynomials, composite structural invariants [∑JJ] and [MJJ] of [JJ] matrix of the KC are used. The advantage is that they are very easy to compute using MAT LAB software. It is not essential to determine both the composite invariants to compare the given KC, only in case the [∑JJ] is same then it is needed to determine [MJJ] for the KC. The [JJ] matrices can be written with very little effort, even by mere inspection of the KC. The proposed test is quite general in nature and can be used to detect isomorphism of not only planar KC of one degree of freedom, but also KC of multi degree of freedom. The characteristic polynomials and composite structural invariants are very informative and from them valuable information regarding topology of kinematic chains can be predicted.

CHAPTER – 11
CRITICAL COMPARISON OF VARIOUS METHODS FOR DETECTION OF BEST METHOD FOR ISOMORPHISM IDENTIFICATION

A lot of time and effort has been devoted to develop a reliable and computationally efficient technique for isomorphism identification in mechanism kinematic chain and from a long time work is continuing in this field.

For any method of isomorphism identification among chains and among inversions of a given chain, important aspect is its simplicity in calculation, fast in response, reliable in results. From a long time a number of methods have been proposed for isomorphism identification but every method had its shortcomings. So various important methods discussed till now are compared here from the point of view of various attributes like reliability, simplicity, time, applicability etc. to detect the best method of isomorphism identification among chains and inversions of a chain along with the ratings given to them from 1-5. From this rating, it is concluded that the method with the highest total rating may be considered as the best method for isomorphism identification among kinematic chains and inversions of a chain.

For comparing the methods, a few common examples are taken so that comparison is made clear and easy. Consider the six links one degree of freedom Watt chain and Stephenson chain 1 and 2, as shown in Fig. 11.1, 11.2 and 11.3.

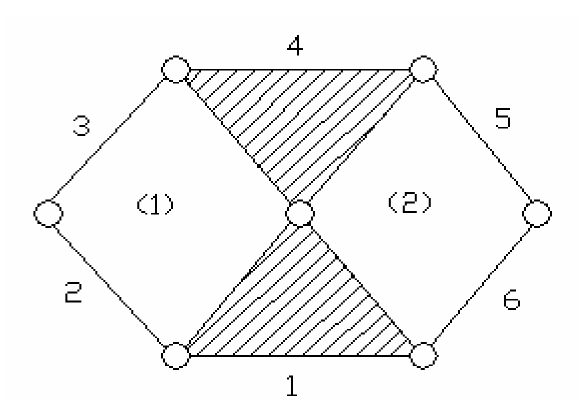


FIG. 11.1 WATT CHAIN

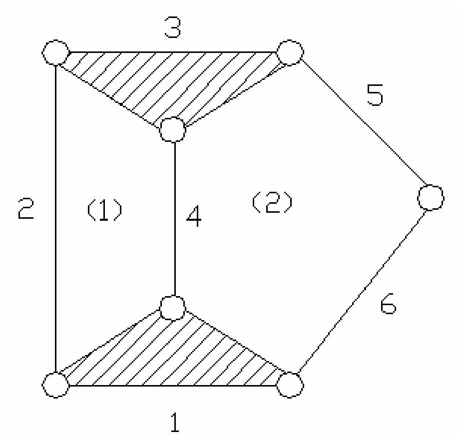


FIG. 11.2 STEPHENSON CHAIN 1

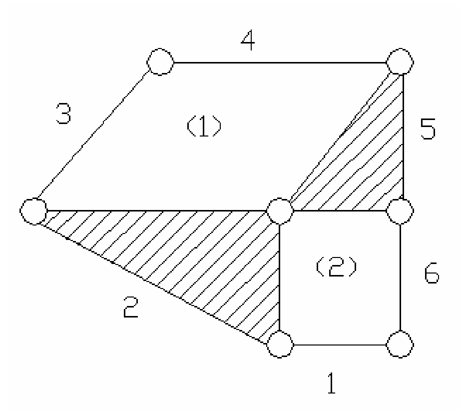


FIG. 11.3 STEPHENSON CHAIN 2

11.1 HAMMING NUMBER TECHNIQUE ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

Hamming Matrix

H =

Link → / Link ↓	1	2	3	4	5	6	L.H.N.
1	0	5	1	6	1	5	18
2	5	0	4	1	4	2	16
3	1	4	0	5	2	4	16
4	6	1	5	0	5	1	18
5	1	4	2	5	0	4	16
6	5	2	4	1	4	0	16

∑ L.H.N.
= C.H.N.
= 100

Link Hamming String

Link	Link Hamming String
1	18, 1200021
2	16, 0120111
3	16, 0120111
4	18, 1200021
5	16, 0120111
6	16, 0120111

Chain Hamming String (C.H.S.)

C.H.S. = 100; 18, 1200021; 18, 1200021; 16, 0120111; 16, 0120111; 16, 0120111; 16, 0120111

Links	Adjacent Links	L.H.S.	Link Neighbourhood Strings
1	2, 4, 6	18, 1200021	16, 0120111; 18, 1200021; 16, 0120111
2	3, 1	16, 0120111	16, 0120111; 18, 1200021
3	2, 4	16, 0120111	16, 0120111; 18, 1200021
4	3, 1, 5	18, 1200021	16, 0120111; 18, 1200021; 16, 0120111
5	4, 6	16, 0120111	16, 0120111; 18, 1200021
6	1, 5	16, 0120111	16, 0120111; 18, 1200021

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

Hamming Matrix

H =

Link → / Link ↓	1	2	3	4	5	6	L.H.N.
1	0	5	2	5	3	5	20
2	5	0	5	0	2	2	14
3	2	5	0	5	5	3	20
4	5	0	5	0	2	2	14
5	3	2	5	2	0	4	16
6	5	2	3	2	4	0	16

∑ L.H.N. = C.H.N. = 100

Link Hamming String

Link	Link Hamming String
1	20, 0301101
2	14, 0200202
3	20, 0301101
4	14, 0200202
5	16, 0111201
6	16, 0111201

Chain Hamming String (C.H.S.)

C.H.S. = 100; 20, 0301101; 20, 0301101; 16, 0111201; 16, 0111201; 14, 0200202; 14, 0200202

Links	Adjacent Links	L.H.S.	Link Neighbourhood Strings
1	2, 4, 6	20, 0301101	14, 0200202; 14, 0200202; 16, 0111201
2	3, 1	14, 0200202	20, 0301101; 20, 0301101
3	2, 4, 5	20, 0301101	14, 0200202; 14, 0200202; 16, 0111201
4	3, 1	14, 0200202	20, 0301101; 20, 0301101
5	3, 6	16, 0111201	20, 0301101; 16, 0111201
6	1, 5	16, 0111201	20, 0301101; 16, 0111201

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

Hamming Matrix

H =

Link → / Link ↓	1	2	3	4	5	6	L.H.N.
1	0	5	2	4	1	4	16
2	5	0	5	1	6	1	18
3	2	5	0	4	1	4	16
4	4	1	4	0	5	2	16
5	1	6	1	5	0	5	18
6	4	1	4	2	5	0	16

∑ L.H.N.
= C.H.N.
= 100

Link Hamming String

Link	Link Hamming String
1	16, 0120111
2	18, 1200021
3	16, 0120111
4	16, 0120111
5	18, 1200021
6	16, 0120111

Chain Hamming String (C.H.S.)

C.H.S. = 100; 18, 1200021; 18, 1200021; 16, 0120111; 16, 0120111; 16, 0120111; 16, 0120111

Links	Adjacent Links	L.H.S.	Link Neighbourhood Strings
1	2, 6	16, 0120111	18, 1200021; 16, 0120111
2	3, 1, 5	18, 1200021	16, 0120111; 16, 0120111; 18, 1200021
3	2, 4	16, 0120111	18, 1200021; 16, 0120111
4	3, 5	16, 0120111	18, 1200021; 16, 0120111
5	2, 4, 6	18, 1200021	18, 1200021; 16, 0120111; 16, 0120111
6	1, 5	16, 0120111	16, 0120111; 18, 1200021

11.1.1 RESULTS ---

It is revealed that Watt Chain and Stephenson Chain 2 have same C.H.S. hence they are isomorphic. But C.H.S. of Watt Chain and Stephenson Chain 1 differ from each other considerably and hence they are non-isomorphic.

	IDENTICAL LINK NEIGHBOUHOOD STRINGS	TOTAL DISTINCT INVERSIONS
WATT CHAIN	[1, 4], [2, 3, 5, 6]	2
STEPHENSON CHAIN 1	[1, 3], [2, 4], [5, 6]	3
STEPHENSON CHAIN 2	[2, 5], [1, 3, 4, 6]	2

TABLE 11.1: TOTAL DISTINCT INVERSIONS POSSIBLE BY HAMMING NUMBER TECHNIQUE

11.2 CONCEPT OF MODIFIED DISTANCE ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

Distance Matrix

Link	1	2	3	4	5	6	**DR**
1	0	7	11	8	11	7	44
2	7	0	6	11	15	11	50
3	11	6	0	7	11	15	50
4	8	11	7	0	7	11	44
5	11	15	11	7	0	6	50
6	7	11	15	11	6	0	50

M_D = (matrix above)

The ASMTDRL for this chain is: 50, 50, 50, 50, 44, 44

Link	**Identification Code for the Link**				
1	152	152	140	140	138
2	146	140	140	136	135
3	146	140	140	136	135
4	152	152	140	140	138
5	146	140	140	136	135
6	146	140	140	136	135

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

Distance Matrix

Link	1	2	3	4	5	6	**DR**
1	0	7	12	7	11	7	44
2	7	0	7	11	11	11	47
3	12	7	0	7	7	11	44
4	7	11	7	0	11	11	44
5	11	11	7	11	0	6	46
6	7	11	11	11	6	0	46

M_D = (matrix above)

The ASMTDRL for this chain is: 47, 47, 46, 46, 44, 44

Link	Identification Code for the Link				
1	148	145	145	136	134
2	139	139	137	136	136
3	148	145	145	136	134
4	139	139	137	136	136
5	146	144	137	137	135
6	146	144	137	137	135

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

Distance Matrix

Link	1	2	3	4	5	6	**DR**
1	7	11	15	11	6	0	44
2	0	7	11	8	11	7	44
3	7	0	6	11	15	11	50
4	11	6	0	7	11	15	50
5	8	11	7	0	7	11	44
6	11	15	11	7	0	6	50

$M_D =$ (matrix above)

The ASMTDRL for this chain is: 50, 50, 50, 50, 44, 44

Link	Identification Code for the Link				
1	146	140	140	136	135
2	152	152	140	140	138
3	146	140	140	136	135
4	146	140	140	136	135
5	152	152	140	140	138
6	146	140	140	136	135

11.2.1 RESULTS ---

It can be seen that that Watt Chain and Stephenson Chain 2 have same ASMTDRL hence they are isomorphic. But ASMTDRL of Watt Chain and Stephenson Chain 1 differ from each other considerably and hence they are non-isomorphic.

	LINKS WITH IDENTICAL IDENTIFICATION CODES	TOTAL DISTINCT INVERSIONS
WATT CHAIN	[1, 4], [2, 3, 5, 6]	2
STEPHENSON CHAIN 1	[1, 3], [2, 4], [5, 6]	3
STEPHENSON CHAIN 2	[2, 5], [1, 3, 4, 6]	2

TABLE 11.2: TOTAL DISTINCT INVERSIONS POSSIBLE BY MODIFIED DISTANCE METHOD

11.3 LOOP BASED METHOD ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

LINKS	L.S.L.	L.A.S.L.
1	14-102	14-102, 14-102, 10-101, 10-101
2	10-101	10-101, 14-102, 10-101
3	10-101	10-101, 14-102, 10-101
4	14-102	14-102, 14-102, 10-101, 10-101
5	10-101	10-101, 14-102, 10-101
6	10-101	10-101, 14-102, 10-101

JVC = (1*4) + (4*3) + (2*2) = 20
CLS = [14, (102), (14, 14, 10, 10, 10, 10), 20]

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

LINKS	L.S.L.	L.A.S.L.
1	14-021	14-021, 10-020, 9-011, 9-011
2	9-011	9-011, 14-021, 14-021
3	14-021	14-021, 10-020, 9-011, 9-011
4	9-011	9-011, 14-021, 14-021
5	10-020	10-020, 14-021, 10-020
6	10-020	10-020, 14-021, 10-020

JVC = (6*3) + (1*2) = 20
CLS = [14, (021), (14, 14, 10, 10, 9, 9), 20]

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

LINKS	L.S.L.	L.A.S.L.
1	10-101	10-101, 14-102, 10-101
2	14-102	14-102, 14-102, 10-101, 10-101
3	10-101	10-101, 14-102, 10-101
4	10-101	10-101, 14-102, 10-101
5	14-102	14-102, 14-102, 10-101, 10-101
6	10-101	10-101, 14-102, 10-101

JVC = (1*4) + (4*3) + (2*2) = 20
CLS = [14, (102), (14, 14, 10, 10, 10, 10), 20]

11.3.1 RESULTS ---

It is seen that Watt Chain and Stephenson Chain 2 have same C.L.S. hence they are isomorphic. But C.L.S. of Watt Chain and Stephenson Chain 1 differ from each other considerably and hence they are non-isomorphic.

	IDENTICAL L.A.S.L	TOTAL DISTINCT INVERSIONS
WATT CHAIN	[1, 4], [2, 3, 5, 6]	2
STEPHENSON CHAIN 1	[1, 3], [2, 4], [5, 6]	3
STEPHENSON CHAIN 2	[2, 5], [1, 3, 4, 6]	2

TABLE 11.3: TOTAL DISTINCT INVERSIONS POSSIBLE BY LOOP BASED METHOD

11.4 A GENETIC ALGORITHM ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

$$F_1 = \begin{pmatrix} 0 & 5 & 1 & 6 & 1 & 5 \\ 5 & 0 & 4 & 1 & 4 & 2 \\ 1 & 4 & 0 & 5 & 2 & 4 \\ 6 & 1 & 5 & 0 & 5 & 1 \\ 1 & 4 & 2 & 5 & 0 & 4 \\ 5 & 2 & 4 & 1 & 4 & 0 \end{pmatrix}$$

First Generation Chain String = 2 [18 – 6, 2 (5), 2 (1)] – 4 [16 – 5, 2 (4), 2, 1]

$$F_2 = \begin{pmatrix} 0 & 4 & 2 & 4 & 2 & 4 \\ 4 & 0 & 4 & 2 & 4 & 2 \\ 2 & 4 & 0 & 4 & 2 & 4 \\ 4 & 2 & 4 & 0 & 4 & 2 \\ 2 & 4 & 2 & 4 & 0 & 4 \\ 4 & 2 & 4 & 2 & 4 & 0 \end{pmatrix}$$

Second Generation Chain String = 6 [16 – 3 (4), 2 (2)]

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

$$F_1 = \begin{pmatrix} 0 & 5 & 2 & 5 & 3 & 5 \\ 5 & 0 & 5 & 0 & 2 & 2 \\ 2 & 5 & 0 & 5 & 5 & 3 \\ 5 & 0 & 5 & 0 & 2 & 2 \\ 3 & 2 & 5 & 2 & 0 & 4 \\ 5 & 2 & 3 & 2 & 4 & 0 \end{pmatrix}$$

First Generation Chain String = 2 [20 – 3(5), 3, 2] - 2 [16 – 5, 4, 3, 2 (2)] – 2 [14 – 2 (5), 2 (2), 0]

$$F_2 = \begin{pmatrix} 0 & 3 & 4 & 3 & 5 & 3 \\ 3 & 0 & 3 & 2 & 4 & 4 \\ 4 & 3 & 0 & 3 & 3 & 5 \\ 3 & 2 & 3 & 0 & 4 & 4 \\ 5 & 4 & 3 & 4 & 0 & 2 \\ 3 & 4 & 5 & 4 & 2 & 0 \end{pmatrix}$$

Second Generation Chain String = 2 [18 – 5, 2 (4), 3, 2] – 2 [18 – 5, 4, 2 (3)] – 2 [16 – 2 (4), 2 (3), 2]

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

$$F_1 = \begin{pmatrix} 0 & 5 & 2 & 4 & 1 & 4 \\ 5 & 0 & 5 & 1 & 6 & 1 \\ 2 & 5 & 0 & 4 & 1 & 4 \\ 4 & 1 & 4 & 0 & 5 & 2 \\ 1 & 6 & 1 & 5 & 0 & 5 \\ 4 & 1 & 4 & 2 & 5 & 0 \end{pmatrix}$$

First Generation Chain String = 2 [18 – 6, 2 (5), 2 (1)] – 4 [16 – 5, 2 (4), 2, 1]

$$F_2 = \begin{pmatrix} 0 & 4 & 2 & 4 & 2 & 4 \\ 4 & 0 & 4 & 2 & 4 & 2 \\ 2 & 4 & 0 & 4 & 2 & 4 \\ 4 & 2 & 4 & 0 & 4 & 2 \\ 2 & 4 & 2 & 4 & 0 & 4 \\ 4 & 2 & 4 & 2 & 4 & 0 \end{pmatrix}$$

Second Generation Chain String = 6 [16 – 3 (4), 2 (2)]

11.4.1 RESULTS ---

It is revealed that Watt Chain and Stephenson Chain 2 have same chain strings hence they are isomorphic. But chain strings of Watt Chain and Stephenson Chain 1 differ from each other considerably and hence they are non-isomorphic.

	LINKS WITH IDENTICAL FITNESS STRINGS	TOTAL DISTINCT INVERSIONS
WATT CHAIN	[1, 4], [2, 3, 5, 6]	2
STEPHENSON CHAIN 1	[1, 3], [2, 4], [5, 6]	3
STEPHENSON CHAIN 2	[2, 5], [1, 3, 4, 6]	2

TABLE 11.4: TOTAL DISTINCT INVERSIONS POSSIBLE BY GENETIC ALGORITHM

11.5 ADJACENCY MATRIX METHOD ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Eigen values and Eigen vectors of Watt Chain

Eigval	-2.4142	-1.000	-0.4142	-0.4142	1.000	2.4142

Eigvec	0.5000	0.0000	0.5000	0.5000	0.0000	0.5000
	-0.3536	0.5000	-0.3536	0.3536	0.5000	0.3536
	0.3536	-0.5000	-0.3536	-0.3536	0.5000	0.3536
	-0.5000	0.0000	0.5000	-0.5000	0.0000	0.5000
	0.3536	0.5000	-0.3536	-0.3536	-0.5000	0.3536
	-0.3536	-0.5000	-0.3536	0.3536	-0.5000	0.3536

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Eigen values and Eigen vectors of Stephenson Chain 1

Eigval	-2.1642	-1.6180	-0.0000	0.6180	0.7729	2.3914

Eigvec	-0.5059	0.3717	0.0000	-0.6015	-0.1359	0.4750
	0.4675	0.0000	-0.7071	0.0000	-0.3516	0.3973
	-0.5059	-0.3717	0.0000	0.6015	-0.1359	0.4750
	0.4675	0.0000	0.7071	0.0000	-0.3516	0.3973
	0.1599	0.6015	0.0000	0.3717	0.5982	0.3414
	0.1599	-0.6015	0.0000	-0.3717	0.5982	0.3414

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Eigen values and Eigen vectors of Stephenson Chain 2

Eigval	-2.4142	-1.000	-0.4142	-0.4142	1.000	2.4142

Eigvec	0.5000	0.0000	0.5000	0.5000	0.0000	0.5000
	-0.3536	0.5000	-0.3536	0.3536	0.5000	0.3536
	0.3536	-0.5000	-0.3536	-0.3536	0.5000	0.3536
	-0.5000	0.0000	0.5000	-0.5000	0.0000	0.5000
	0.3536	0.5000	-0.3536	-0.3536	-0.5000	0.3536
	-0.3536	-0.5000	-0.3536	0.3536	-0.5000	0.3536

11.5.1 RESULTS ---

It is seen that Watt chain and Stephenson Chain 2 have same Eigen values and same Eigen vectors hence they are isomorphic. But Watt chain and Stephenson Chain 1 differ in their Eigen values and Eigen vectors and hence they are non-isomorphic.

	LINKS SHOWING ROW WISE SIMILARITY OF EIGEN VECTORS	TOTAL DISTINCT INVERSIONS
WATT CHAIN	[1, 4], [2, 3, 5, 6]	2
STEPHENSON CHAIN 1	[1, 3], [2, 4], [5, 6]	3
STEPHENSON CHAIN 2	[2, 5], [1, 3, 4, 6]	2

TABLE 11.5: TOTAL DISTINCT INVERSIONS POSSIBLE BY ADJACENCY MATRICES METHOD

11.6 JOINT – JOINT MATRIX METHOD ---

EVALUATION OF PARAMETERS FOR WATT CHAIN

$$JJ = \begin{pmatrix} 0 & 2 & 0 & 0 & 0 & 3 & 3 \\ 2 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 3 & 0 & 0 & 3 \\ 0 & 0 & 3 & 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 2 & 0 & 2 & 0 \\ 3 & 0 & 0 & 0 & 2 & 0 & 3 \\ 3 & 0 & 3 & 3 & 0 & 3 & 0 \end{pmatrix}$$

Characteristic Polynomial Coefficients:
0.0010 0.0000 -0.0700 -0.1080 0.9010 0.9720 -3.1680 0.0000
Structural Invariants:
$[\sum JJ] = 5.2200e + 003$, $[MJJ] = 3.1680e + 003$

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 1

$$JJ = \begin{pmatrix} 0 & 3 & 0 & 0 & 2 & 3 & 0 \\ 3 & 0 & 2 & 0 & 0 & 3 & 0 \\ 0 & 2 & 0 & 3 & 0 & 0 & 3 \\ 0 & 0 & 3 & 0 & 2 & 0 & 3 \\ 2 & 0 & 0 & 2 & 0 & 0 & 0 \\ 3 & 3 & 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 3 & 3 & 0 & 2 & 0 \end{pmatrix}$$

Characteristic Polynomial Coefficients:
0.0010   0.0000   -0.0700   -0.1080   1.1690   2.6280   -2.1800   -3.6000
Structural Invariants:
[∑JJ] = 9.7560e + 003, [MJJ] = 3.6000e + 003

EVALUATION OF PARAMETERS FOR STEPHENSON CHAIN 2

$$JJ = \begin{pmatrix} 0 & 3 & 0 & 0 & 0 & 2 & 3 \\ 3 & 0 & 2 & 0 & 0 & 0 & 3 \\ 0 & 2 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 3 & 0 & 3 \\ 0 & 0 & 0 & 3 & 0 & 2 & 3 \\ 2 & 0 & 0 & 0 & 2 & 0 & 0 \\ 3 & 3 & 0 & 3 & 3 & 0 & 0 \end{pmatrix}$$

Characteristic Polynomial Coefficients:
0.0010   0.0000   -0.0700   -0.1080   0.9010   0.9720   -3.1680   0.0000
Structural Invariants:
[∑JJ] = 5.2200e + 003, [MJJ] = 3.1680e + 003

11.6.1 RESULTS ---

It can be seen that Watt chain and Stephenson Chain 2 have same set of values of [∑JJ] and [MJJ] hence they are isomorphic. But Watt chain and Stephenson Chain 1 differ in their set of values of [∑JJ] and [MJJ] and hence they are non-isomorphic.

11.7 FINAL RESULTS ---

Now after comparing the above six methods for common chains, a generalized table is prepared of comparison of all the methods from various aspects such as reliability, computational ease, computational time, detection of inversions, applicability to structural properties and other features. This table is shown below as Table 11.6. An attempt has been made in this work to assign ranks to different methods on the basis of various attributes as explained below:

1. **Reliability:**

By reliability, it means that how correct the method applies to various types of chains, i.e. six-bar, eight-bar, nine-bar and ten bar chains with single and multi d.o.f. Least rating of 3 is given to a genetic algorithm as it works only for ten-link single d.o.f. chains and not for multi d.o.f. chains. Least rating of 1 or 2 may be given if it works only for six or eight-bar chains and not for ten-bar chains because of the complexities involved. Here it is not so and hence rating is given 3. Similarly, rating of 4 is given to concept of modified distance as it woks well in all the cases of planar chains but it is obtained by using the feelings rather than by considering the facts, i.e. it is heuristic and intuitive so it is not much reliable. In the same manner, rating of 5 is given to three methods, i.e. Hamming number technique, loop based method and adjacency matrix method, as all these methods have been verified for all the chains of one, two and three d.o.f, as compared to joint-joint matrix method which has worked only in one and two d.o.f. chains.

2. **Computational Ease:**

In this attribute, three methods, i.e. loop based method, adjacency matrix method and joint- joint matrix method, have been given the highest rating of 5 as only one input is necessary to get the final results in all three methods and computations are very easy, either manual or using MATLAB software. The concept of modified distance and genetic algorithm has been given the rating of 4 as computations are somewhat difficult as compared to the other methods. Similarly, hamming number technique involves very difficult computations hence rating of 3 has been given to this method.

3. **Computational Time:**

The calculations are done by computer in the concept of modified distance, adjacency matrix method and joint-joint matrix method. Therefore, all these three methods takes few seconds to get the final results and hence they have been given the rating of 5. The loop based method takes more time than these as the calculations are easy but usually done by hand, so rating of 4 has been given to it. Least rating of 3 has been given to hamming number technique as the manual calculations are very long and it takes a lot of time.

4. **Detection of Inversions:**

Here rating can be either 0 or 5 only because either the method can detect the number of distinct inversions or it cannot detect. In this manner, all the methods considered here except joint-joint matrix method have been given the rating of 5. Only joint-joint matrix method has been given 0 rating as it cannot detect the number of inversions of a given chain.

5. **Applicability to Structural Properties:**

Hamming number technique, concept of modified distance and joint-joint matrix method cannot identify any structural property of a KC. So, all these three methods have been given 0 rating. Whereas loop based method explores one property involving three sub-properties, hence it has been given rating of 3. Rating of 4 has been given to a genetic algorithm as it identifies three properties but these are not of much importance as compared to the properties identified by adjacency matrix method, which has been given highest rating of 5 as it identifies two very important structural properties, i.e. degeneration and type of freedom with regard to motion of the KC.

6. **Other Features:**

Highest rating of 5 has been given to loop based method and a genetic algorithm as these two methods are unique in themselves as compared to the other four methods. The concept of modified distance, adjacency matrix method and joint-joint matrix method are computer based methods. Therefore, these methods have been given the rating of 4. Similarly, rating of 3 has been given to hamming number technique as it doesn't involve the use of computer.

In table 11.6, rating from 1-5 is given for each point in brackets and then total rating is calculated in the end to detect the best method for isomorphism identification among kinematic chains. Overall ranking of various methods has also been done based on the total rating. A bar graph (Fig. 11.4) is also drawn for total ratings of all methods so that the comparison to detect the best method can easily be seen. As shown in Table 11.6 and Fig. 11.4, Method of Adjacency Matrices has the highest rating of 29 so it can be said that this method is the best method till now for isomorphism identification among chains. Then second best method is Loop based Method with the rating of 27. Therefore, any one of these two methods can be used to detect isomorphism in kinematic chains.

TABLE 11.6: COMPARISON OF VARIOUS METHODS

	HAMMING NUMBER TECHNIQUE	CONCEPT OF MODIFIED DISTANCE	LOOP BASED METHOD	A GENETIC ALGORITHM	ADJACENCY MATRIX METHOD	JOINT-JOINT MATRIX METHOD
1. RELIABILITY	It has been verified for all six, eight and ten-bar chains with one d.o.f. as well as ten-bar chains with three d.o.f. **(5)**	It has worked well in all the known cases of planar chains with simple joints. But it is heuristic and intuitive in nature. **(4)**	All the 16 eight-bar single d.o.f. chains, 40 nine-bar two d.o.f. chains, 230 ten-bar single d.o.f. chains and 98 ten-bar three d.o.f. chains have been tested for isomorphism. **(5)**	All the 230 distinct ten-link single d.o.f. chains are tested for confirmation but it doesn't work for multi d.o.f. chains. **(3)**	It covers all planar chains of one, two and three d.o.f. So it has perfect reliability from the point of view of results. **(5)**	All the simple jointed 1-F, 8-links 16 KC and 1-F, 10-links 230 KC along with 2-F, 9-links 40 KC have been tested successfully for their non-isomorphism. **(4)**
2. COMPUTATIONAL EASE	Computations are very long, especially in case of large KC. So it takes lot of effort to compute many things for single KC. **(3)**	The relation matrix can be written easily, even by mere inspection of the KC. So it is not very difficult to get results. **(4)**	This method is extremely simple in the formulation and execution stage, since the only input necessary is the disposition of various links in a loop. **(5)**	Computation is extremely simple as effort involved is very less. But these are lengthy. **(4)**	It has simplicity in its process and is very easy to get the final results. **(5)**	It is very simple as computations are very easy using MATLAB software. Also, the [JJ] matrix can be written with very little effort, even by mere inspection of the KC. **(5)**

	HAMMING NUMBER TECHNIQUE	CONCEPT OF MODIFIED DISTANCE	LOOP BASED METHOD	A GENETIC ALGORITHM	ADJACENCY MATRIX METHOD	JOINT-JOINT MATRIX METHOD
3. COMPUTATIONAL TIME	It takes lot of time as calculations are very long. **(3)**	It is a computer aided method hence fast in calculations. **(5)**	The arithmetical computations made are very easy enough to be attempted by hand without the necessity of sophisticated algorithms. **(4)**	It takes some time as computations become very long in case of large KC. **(3)**	It takes few seconds to get results due to use of MATLAB software. **(5)**	It takes very less time as its not essential to determine both the composite invariants, only in case [∑JJ] is same then it is needed to determine [MJJ] for the KC. **(5)**
4. DETECTION OF INVERSIONS	It has the potential to disclose how many structurally different inversions can be obtained from a given chain. **(5)**	It can identify distinct mechanisms (inversions) of a planar chain. **(5)**	It can detect isomorphism among inversions of a of a given chain. **(5)**	It can detect distinct inversions of a given KC. **(5)**	It can detect isomorphism among inversions of a given chain. **(5)**	It can't be known by this method that how many distinct inversions can be obtained from a given chain. **(0)**
5. APPLICABILITY TO STRUCTURAL PROPERTIES	It can't identify any other property of a KC. **(0)**	Structural properties of a chain cannot be identified by this method. **(0)**	It explores only one property, i.e. type of freedom in case of multi d.o.f. chains. **(3)**	This method enables the selection of best chain, best inversion and best input links for a chain. **(4)**	It can identify structural properties as Degeneration and type of freedom of a KC. **(5)**	Structural properties of a chain cannot be identified by this method. **(0)**

	HAMMING NUMBER TECHNIQUE	CONCEPT OF MODIFIED DISTANCE	LOOP BASED METHOD	A GENETIC ALGORITHM	ADJACENCY MATRIX METHOD	JOINT-JOINT MATRIX METHOD
6. OTHER FEATURES	All calculations are manual. No software dependence is there. **(3)**	Two computer aided methods are developed here. **(4)**	This method is unique as it has taken care of all basic features of chain viz, links, joints and loops, whereas other methods have not considered loops. Also, it is neither affected by relabelling or redrawing a chain. **(5)**	Unlike other methods, this method fulfills both necessary and sufficient requirements, making it unique. **(5)**	Computer based MATLAB software is used to get final results. **(4)**	Computer based MATLAB software is used to get final results. **(4)**
TOTAL RATING	**19**	**22**	**27**	**24**	**29**	**18**
OVERALL RANKING	**5**	**4**	**2**	**3**	**1**	**6**

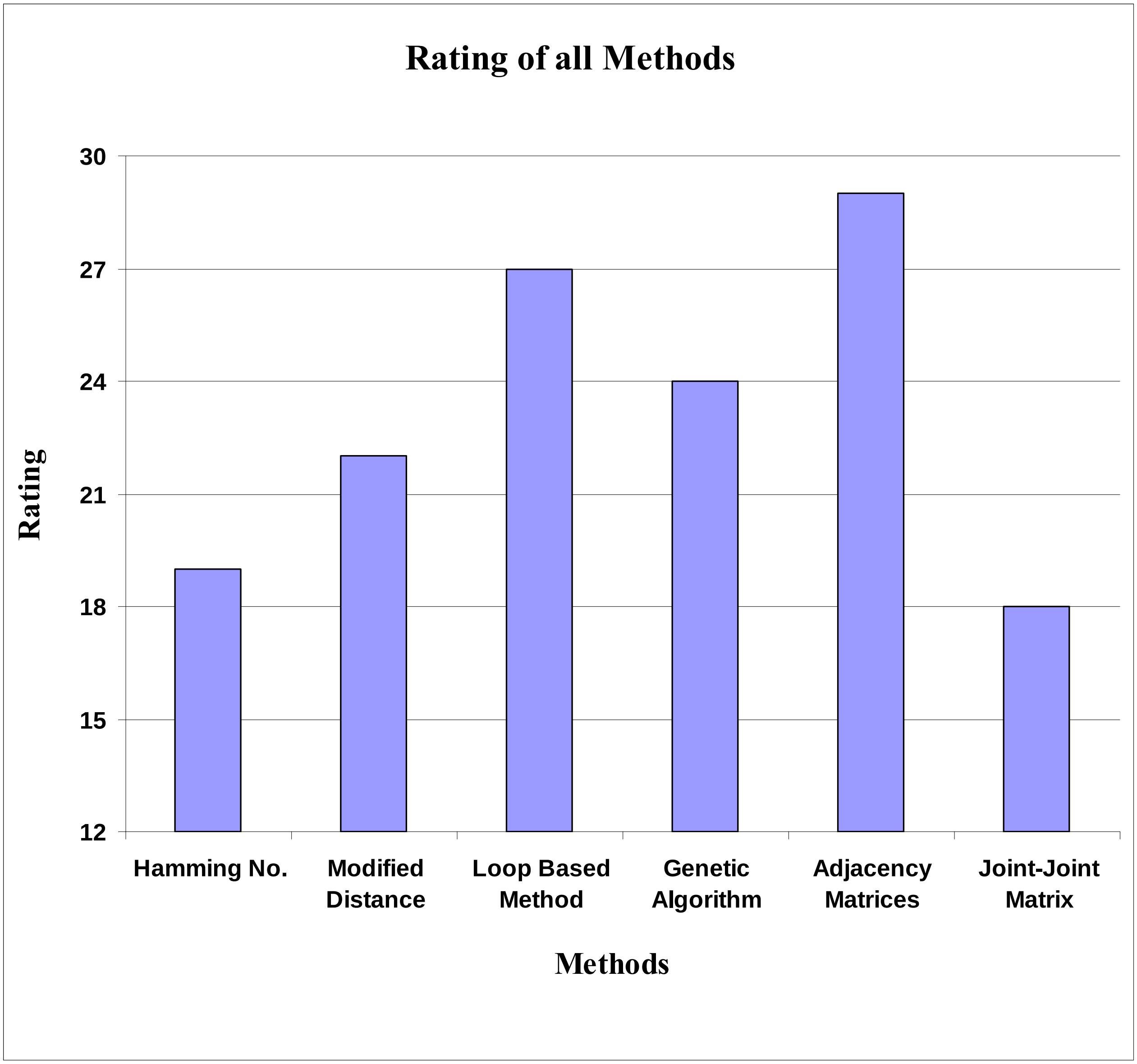


FIG. 11.4 GRAPH OF COMPARISON OF VARIOUS METHODS

CHAPTER – 12
CONCLUSIONS AND SCOPE FOR FUTURE WORK

12.1 CONCLUSIONS ---

The present work is an attempt to compare some of the existing methods for identification of isomorphism among kinematic chains and among inversions of a given chain from various aspects such as identification of structural properties, reliability, time, computational ease, detection of inversions and other features.

Thus the objective of a critical comparison of various methods for isomorphism identification among kinematic chains has been achieved in this work as six important methods, developed by different kinematicians have been clearly compared and on the basis of presented work, following conclusions are drawn:-

- This new approach of comparison has shown that Method of Adjacency Matrices is the best method for the identification of Isomorphism among kinematic chains and inversions. It has adopted a very strong tool of mathematics i.e. MATLAB software, which has the advantage of showing results within few seconds through Eigen values and Eigen Vectors over showing results after hectic calculations by other methods.
- Method of Loop Based Detection can also be considered as a good method after Adjacency Matrices Method but it does not have capability to identify one of the structural properties of kinematic chains i.e. degeneration identification in kinematic chains. On the other hand, the method of Adjacency Matrices adopts a very simple and compact approach in identification of isomorphism and structural properties of kinematic chains.

Therefore, the ranking of six methods discussed here is as follows:

1. Adjacency Matrix Method
2. Loop Based Method
3. A Genetic Algorithm
4. Concept of Modified Distance

5. Hamming Number Technique
6. Joint-Joint Matrix Method

12.2 SCOPE FOR FUTURE WORK ---

The present work may be extended in one of the following directions:-

- The present work mainly concentrated over planar kinematic chains, it can be extended for the spatial chains which find its application in the field of robotics.
- Comparison can be made for the identification of other structural properties of kinematic chains like rigidity, compactness or parallelism, work-space, mobility, symmetry etc.
- Identification of an important structural property of kinematic chain i.e. compactness or parallelism, can be done which finds its application in the field of robotics for particular use in platform type of robot.
- The work can be extended to complex mechanisms with higher order links and multiple inputs and outputs.

CHAPTER -13
REFERENCES AND BIBLIOGRAPHY

1. **Manolescu, N.I.;** 1973. A Method Based on Baranov Trusses, and using Graph Theory to find the set of Planar Jointed Kinematic Chains and Mechanisms. *Mechanism and Machine Theory, Vol. 8, Issue 1, pp 3-22.*
2. **Hopcroft, J.E. and Tarjan, R.E.;** June 1973. A V log V algorithm for Isomorphism of Triconnected Planar Graphs. *Journal of Computer and System Science, Vol. 7, Issue 3, pp 323-331.*
3. **Uicker, J.J.; Raicu A.;** October 1975. A Method for the Identification and Recognition of Equivalence of Kinematic Chains. *Mechanism and Machine Theory, Vol. 10, Issue 5, pp 375-383.*
4. **Mruthyunjaya, T.S.; Raghavan M.R.;** July 1979. Structural Analysis of Kinematic Chains and Mechanisms based on Matrix Representation. *Transaction of the ASME Journal of Mechanical Design, Vol. 101, pp 488-518.*
5. **Yan, H.S. and Hall, A.S.;** July 1981. Linkage Characteristic Polynomials: Definitions, Coefficients by Inspection. *ASME Journal of Mechanical Design, Vol. 103, pp 578-584.*
6. **Yan, H.S. and Hall, A.S.;** January 1982. Linkage Characteristic Polynomials: Assembly Theorem, Uniqueness. *ASME Journal of Mechanical Design, Vol. 104, pp 11-20.*
7. **Yan, Hong-Sen and Hwang, Wen-Miin;** 1984. Linkage Path Code. *Mechanism and Machine Theory, Vol. 19, Issue 4, pp 425-529.*
8. **Mruthyunjaya, T.S.;** 1984. A Computerized Methodology for Structural Synthesis of Kinematic Chains. *Mechanism and Machine Theory, Vol. 19, Issue 6, pp 487-495.*
9. **Mruthyunjaya, T.S. and Balasubramanian, H.R.;** 1987. In Quest of a Reliable and Efficient Computational Test for detection of Isomorphism in Kinematic Chains. *Mechanism and Machine Theory, Vol. 22, Issue 2, pp 131-139.*
10. **Ambedkar, A.G. and Agrawal, V.P.;** 1987. Canonical Numbering of Kinematic Chains and Isomorphism Problem: Min. Code. *Mechanism and Machine Theory, Vol. 22, Issue 5, pp 453-461.*

11. **Ambedkar, A.G. and Agrawal, V.P.;** 1987. Identification of Kinematic Chains, Mechanisms, Path generators and Function Generators using Min. Code. *Mechanism and Machine Theory, Vol. 22, Issue 5, pp 463-471.*
12. **Agrawal, V.P. and Rao, J.S.;** 1987. Structural classification of kinematic chains and mechanisms. *Mechanism and Machine Theory, Vol. 22, Issue 5, pp 489-496.*
13. **Agrawal, V.P. and Rao, J.S.;** 1989. Identification and Isomorphism of Kinematic Chains and Mechanisms. *Mechanism and Machine Theory, Vol. 24, Issue 4, pp 309-321.*
14. **Rao, A.C.;** 1991. Application of the Hamming Number Technique to detect Isomorphism among Kinematic Chains and Inversions. *Mechanism and Machine Theory, Vol. 26, Issue 1, pp 55-75.*
15. **Rao, A.C. and Rao, C.N.;** January 1993. Loop based Pseudo Hamming Values-I Testing Isomorphism and Rating Kinematic Chains. *Mechanism and Machine Theory, Vol. 28, Issue 1, pp 113-127.*
16. **Tang, C.S. and Liu, Tyang;** December 1993. The Degree Code-A New Mechanism Identifier. *ASME Journal of Mechanical Design, Vol. 115, pp 627-630.*
17. **Jin-Kui, Chu and Wei-Qing, Cao;** January 1994. Identification of Isomorphism among Kinematic Chains and Inversions using Link's Adjacent-Chain Table. *Mechanism and Machine Theory, Vol. 29, Issue 1, pp 53-58.*
18. **Shende, S. and Rao, A.C.;** October 1994. Isomorphism in Kinematic Chains, *Mechanism and Machine Theory, Vol. 29, Issue 7, pp 1065-1070.*
19. **Tischler, C.R.; Samuel. A.E. and Hunt K.H.;** November 1995. Kinematic Chains for Robotic Hands – I, Orderly Number Synthesis. *Mechanism and Machine Theory, Vol. 30, Issue87, pp 1193-1215.*
20. **Yadav, J.N.; Agrawal, V.P. and Pratap, C.R.;** December 1995. Detection of Isomorphism among Kinematic Chains using the Distance Concept. *ASME Journal of Mechanical Design, Vol. 117, pp 607-611.*
21. **Murphy, D. Morgan; Midha, Ashok and Howell, L. Larry;** February 1996. The Topological Synthesis of Compliant Mechanisms. *Mechanism and Machine Theory, Vol. 31, Issue 2, pp 185-199.*

22. **Tuttle, E.R.;** August 1996. Generation of Planar Kinematic Chains. *Mechanism and Machine Theory, Vol. 31, Issue 6, pp 729-748.*

23. **Yadav, J.N. and Pratap, C.R.;** October 1996. Mechanism of Kinematic chain and the Degree of Structural Similarity based on the concept of Link – Path Code. *Mechanism and Machine Theory, Vol. 31, Issue 7, pp 865-871.*

24. **Yadav, J.N.; Agrawal, V.P. and Pratap, C.R.;** October 1996. Computer Aided Detection of Isomorphism among Binary Chains using the Link-Link Multiplicity Distance Concept. *Mechanism and Machine Theory, Vol. 31, Issue 7, pp 873-877.*

25. **Rao, A.C.;** May 1997. Hamming Number Technique-I. Further Application. *Mechanism and Machine Theory, Vol. 32, Issue 4, pp 477-488.*

26. **Rao, A.C.;** May 1997. Hamming Number Technique-II. Generation of Planar Kinematic Chains. *Mechanism and Machine Theory, Vol. 32, Issue 4, pp 489-499.*

27. **Rao, A.C. and Jagdeesh, Anne;** July 1998. Topology Based Characteristics of Kinematic Chain: Work Space, Rigidity, Input-Output and Isomorphism. *Mechanism and Machine Theory, Vol. 33, Issue 5, pp 625-638.*

28. **Zhang, W.J.; Lib, Q. and Kong, F.G.;** February 1999. An Artificial Neural Network Approach to Mechanism Kinematic Chain Isomorphism Identification. *Mechanism and Machine Theory, Vol. 34, Issue 2, pp 271-283.*

29. **Zhang, W.J. and Lib, Q.;** March 1999. A New Approach to Mechanism Topology Identification. *ASME Journal of Mechanical Design, Vol. 121, pp 57-64.*

30. **Rao, A.C.;** March 2000. Loop Based Detection of Isomorphism among Chains, Inversions and Type of Freedom in Multi Degree-of-Freedom Chain. *ASME Journal of Mechanical Design, Vol. 122, pp 31-42.*

31. **Rao, A.C.;** June 2000. A Genetic Algorithm for Topological Characteristics of Kinematic Chains. *ASME Journal of Mechanical Design, Vol. 122, pp 228-231.*

32. **Rao, A.C.;** August 2000. Application of Fuzzy Logic for the study of Isomorphism, Inversions, Symmetry, Parallelism and Mobility in Kinematic Chains. *Mechanism and Machine Theory, Vol. 35, Issue 8, pp 1103-1116.*

33. **Schmidt, C. Linda and Harshawardhan, Shetty;** December 2000. A Graph Grammar Approach for Structure Synthesis of Mechanisms. *ASME Journal of Mechanical Design, Vol. 122, pp 371-376.*

34. **Rao, A.C. and Deshmukh, B. Pratap;** April 2001. Computer Aided Structural Synthesis of Planar Kinematic Chains obviating the Test for Isomorphism. *Mechanism and Machine Theory, Vol. 36, Issue 4, pp 489-506.*

35. **Zongyu, Chang ; Ce, Zhang ; Yuhu, Yang and Yuxin, Wang ;** April 2002. A New Method to Mechanism Kinematic Chain Isomorphism Identification, *Mechanism and Machine Theory, Vol. 37, Issue 4, pp 411-417.*

36. **Mruthyunjaya, T.S.,** April 2003. Kinematic Structure of Mechanisms Revisited. *Mechanism and Machine Theory, Vol. 38, Issue 4, pp 279-320.*

37. **Mahere, Vishesh; Nigam, S.P.;** June 2004. Identification of Isomorphism and Structural Properties of Kinematic Chains with the help of Adjacency Matrices. *Communicated.*

38. **Sarkar, S.C.; Khare, A.K.;** August 2004. Detecting the Effect of Uncertainty and Isomorphism in 10-bar Kinematic Chains using all possible paths for motion transmission. *Mechanism and Machine Theory, Vol. 39, Issue 8, pp 893-900.*

39. **Cubillo, J.P. and Wan Jinbao;** February 2005. Comments on Mechanism Kinematic Chain Isomorphism Identification using Adjacent Matrices. *Mechanism and Machine Theory, Vol. 40, Issue 2, pp 131-139.*

40. **Butcher, A. Eric; Hartman Chris;** September 2005. Efficient Enumeration and Hierarchical Classification of Planar Simple-Jointed Kinematic Chains: Application to 12-bar and 14-bar single degree-of-freedom chains. *Mechanism and Machine Theory, Vol. 40, Issue 9, pp 1030-1050.*

41. **P.R., He; W.J., Zhang and Li, Q.;** September 2005. Some Further Development on the Eigen System Approach for Graph Isomorphism Detection. *Journal of the Franklin Institute, Vol. 342, Issue 6, pp 657-673.*

42. **Srinath, A. and Rao, A.C.;** June 2006. Correlation to detect Isomorphism, Parallelism and Type of Freedom. *Mechanism and Machine Theory, Vol. 41, Issue 6, pp 646-655.*

43. **Sunkari, P. Rajesh; Schmidt, C. Linda;** September 2006. Structural Synthesis of Planar Kinematic Chains by adapting a Mckay - type Algorithm. *Mechanism and Machine Theory, Vol. 41, Issue 9, pp 1021-1030.*

44. **Ali, Hasan; Khan, R.A.; Aas, Mohd.;** January 2007. A New Method to Detect Isomorphism in Kinematic Chains. *Kathmandu University Journal of Science, Engineering and Technology, Vol. 1, No. III.*

45. **Hung, Chih-Ching; Yan, Hong–Sen; Pennock, R. Gordon;** August 2007. A Procedure to count the Number of Planar Mechanisms subject to design constraints from Kinematic Chains. *Mechanism and Machine Theory.*

46. **Ding, Huafeng; Huang, Zhen;** October 2007. A New Theory for the Topological Structure Analysis of Kinematic Chains and its Applications. *Mechanism and Machine Theory, Vol. 42, Issue 10, pp 1264-1279.*

BOOKS :-

47. **Arthur G. Erdman, George N. Sandor,** 1988. *Mechanism Design: Analysis and Synthesis, Volume 1* **(Prentice Hall of India Private Limited, New Delhi).**

48. **Robert L. Norton,** 1999. *Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms of Machines, Second Edition* **(WCB McGraw-Hill).**

49. **R.S. Khurmi, J.K. Gupta,** 2000. *Theory of Machines* **(Eurasia Publishing House Pvt. Ltd., New Delhi).**

APPENDIX A

The computer program in C++ for getting distance matrix by algorithm and ASMTDRL for the concept of modified distance is written below:

```
#include<iostream.h>
#include<conio.h>
void main( )
{
	clrscr( );
	int i,j,k,S;
	int w=2,temp;			// weight of simple revolute joint = 2
	int R[10][10],d[10]={0,0,0,0,0,0,0,0,0,0};
	cout<<"Relation matrix R:\n\n";
	for(i=0;i<10;i++)
	{
		for(j=0;j<10;j++)
		{
			if(i==j)
				R[i][j]=0;
			else
				R[i][j]=99;
			R[0][1]=w+3;
			R[0][3]=w+3;
			R[0][9]=w+2;
			R[1][0]=w+3;
			R[1][2]=w+2;
			R[1][8]=w+2;
			R[2][1]=w+3;
			R[2][3]=w+3;
			R[3][0]=w+3;
			R[3][2]=w+2;
```

```
                R[3][4]=w+2;
                R[4][3]=w+3;
                R[4][5]=w+2;
                R[5][6]=w+3;
                R[5][4]=w+2;
                R[6][5]=w+2;
                R[6][9]=w+2;
                R[6][7]=w+2;
                R[7][6]=w+3;
                R[7][8]=w+2;
                R[8][7]=w+2;
                R[8][1]=w+3;
                R[9][6]=w+3;
                R[9][0]=w+3;
                cout<<R[i][j];
                cout<<"\t";
        }
        cout<<"\n";
}
for(k=0;k<10;k++)
{
        for(i=0;i<10;i++)
        {
          for(j=0;j<10;j++)
          {
                S=R[i][k] + R[k][j];
                if(S<R[i][j])
                R[i][j] = S;
          }
        }
}
for(j=0;j<10;j++)
```

```
{
	R[0][j]+=2;
	R[1][j]+=3;
	R[2][j]+=3;
	R[3][j]+=2;
	R[4][j]+=2;
	R[5][j]+=3;
	R[6][j]+=3;
	R[7][j]+=3;
	R[8][j]+=2;
	R[9][j]+=3;
}
for(i=0;i<10;i++)
{
	for(j=0;j<10;j++)
	{
		if(i==j)
		R[i][j]=0;
	}
}
cout<<"Distance Matrix D:\n\n";
for(i=0;i<10;i++)
{
	for(j=0;j<10;j++)
	{
		cout<<R[i][j];
		cout<<"\t";
		d[i]+=R[i][j];
	}
	cout<<d[i];
	cout<<"\n";
}
```

```
        for(i=0;i<10;i++)
        {
                for(j=i+1;j<10;j++)
                {
                        if(d[i]<d[j])
                        {
                                temp=d[j];
                                d[j]=d[i];
                                d[i]=temp;
                        }
                }
        }
        cout<<"\nASMTDRL:\n";
        for(i=0;i<10;i++)
        {
                cout<<d[i]<<"\t";
        }
        getch( );
}
```