

UC Irvine

UC Irvine Previously Published Works

Title

Synthesis of Useful Eight-Bar Linkages as Constrained 6R Loops

Permalink

<https://escholarship.org/uc/item/5wq1q7gq>

ISBN

978-0-7918-4636-0

Authors

Sonawale, Kaustubh H
McCarthy, J Michael

Publication Date

2014-08-17

DOI

10.1115/detc2014-35523

Peer reviewed

SYNTHESIS OF USEFUL EIGHT-BAR LINKAGES AS CONSTRAINED 6R LOOPS

Kaustubh H. Sonawale*

Robotics and Automation Laboratory
University of California
Irvine, California 92697
Email: ksonawal@uci.edu

J. Michael McCarthy

Robotics and Automation Laboratory
University of California
Irvine, California 92697
Email: jmmccart@uci.edu

ABSTRACT

This paper formulates a methodology for designing planar eight-bar linkages for five task positions, by adding two RR constraints to a user specified 6R loop. It is known that there are 32 ways in which these constraints can be added, to yield as many as 340 different linkages. The methodology uses a random search within the tolerance zones around the task specifications to increase the number of candidate linkages. These linkages are analyzed using the Dixon determinant approach, to find all possible linkage configurations over the range of motion. These configuration trajectories are sorted into branches. Linkages that have all the five task configurations on one branch, ensure their smooth movement through the five task positions. The result is an array of branch-free useful eight-bar linkage designs. An example is provided to illustrate the results.

INTRODUCTION

In this paper a design methodology is described for generating useful eight-bar linkages for five finitely separated task positions or poses. The user first discretizes the desired motion requirement into five task positions. In addition, the user also specifies a three degree-of-freedom planar 6R (6 Revolute joints) parallel robot, that can reach all the five task positions. A 6R parallel robot (6R loop) is shown in Fig.1 along with its graph.

The eight-bar linkage is obtained by synthesizing two RR constraints that constrain the 6R loop to a single degree-of-

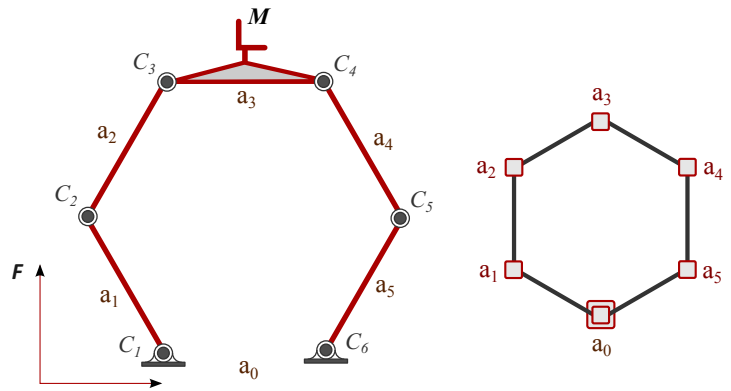


FIGURE 1. SIX-R LOOP (3 DOF PARALLEL ROBOT) WITH GRAPH. LINK a_0 IS THE GROUND LINK, LINK a_1 IS THE INPUT LINK AND LINK a_3 IS THE END-EFFECTOR LINK.

freedom linkage. There are 32 ways in which the two RR constraints can be added to the 6R loop, as reported in Soh and McCarthy(2007) [1]. Our methodology uses all the 32 ways of constraining to yield more candidate linkages. For this paper, link a_0 is chosen as the ground (fixed link), link a_1 is chosen as the input link, and link a_3 is chosen as the end-effector link as shown in Fig.1.

Following the synthesis, each linkage is analyzed to find the forward kinematic solutions. For a particular input link angle, each forward kinematics solution depicts a linkage configuration,

*Address all correspondence to this author.

which represents, a specific way to assemble the various links of the eight-bar linkage. As the input link is rotated in the range, the different possible forward kinematic solutions are tracked and then sorted into branches. These branches give us the overall picture of the different configuration trajectories possible, for the linkage.

A linkage qualifies as a useful linkage if it satisfies two requirements: i) all the five linkage configurations at the five task positions should lie on the same branch (configuration trajectory) and ii) the smallest and the longest link have to be within certain limits, specified by the user. This strategy of finding useful eight-bar linkages ensures smooth movement of the end-effector through the five task positions, when the input link is rotated within the prescribed range.

Providing the user freedom to select the 6R loop permits greater control over the synthesized linkage, especially due to the ability to define both the ground pivots and moving pivots for the linkage. It also reduces the scale of the problem, in terms of the computation requirement, as the number of unknowns are less. A similar remark can be made about specification of only five task positions, as it is well known that an RR dyad can be synthesized exactly for up to five prescribed poses, and extensive research has been reported on this Burmester problem with different approaches. The selection of the 6R loop can be a challenge and does require some intuition on behalf of the designer. It is important to note that, the probability of finding useful eight-bar linkages is heavily dependent on the selection of the 6R loop.

Eight-bar linkages, in general, have the advantage of achieving more complex motion compared to a four-bar or six-bar linkage. Wunderlich [2] derived an expression for the highest degree possible for a coupler curve of a linkage connected by revolute joints. This expression gives degree 6, 18 and 54 for the four-bar, six-bar and eight-bar linkage coupler curves respectively. Freudenstein et al. [3–5] showed the complexity of coupler curves for six-bar linkages is upto degree 16 and for a specific eight-bar topology, it is degree 30. In one case study, the authors observed that a rectilinear motion linkage, synthesized using the proposed method, was able to produce a straight line motion within a deviation of ± 3 microns and $\pm 0.01^\circ$ from the straight line, over a range of 100mm. Thus, the main motivation driving this research, is to design compact linkages that can exhibit complex motion characteristics.

LITERATURE SURVEY

Soni (1973) proposed a technique to synthesize an eight-bar linkage having five links in each of its three loops for a varieties of motion (function, path, motion), along with cases involving constraints on the input and output angle.

Subbian and Flugrad (1994) [7] used continuation methods (secant parameter homotopy) to synthesize an eight-bar to reach six precision points by combining three triads.

Angeles and Chen (2007,2008) [8, 9] developed a method to synthesize an eight-bar that can reach upto 11 poses exactly. Their method couples two four-bar legs, to guide a coupler through the 11 task positions. The formulation involved synthesizes solutions for the four-bar legs, which is a system of 10 equations in 10 unknowns, and then solved them numerically.

Our work follows the design procedure of constraining the user defined, 3 DOF, 6R loop (closed chain) using two RR constraints, introduced by Soh and McCarthy (2007) [1, 10]. Soh and Ying (2013) [11] followed this procedure to design an eight-bar linkage with prismatic joints.

In order to ensure the design of an eight-bar linkage is able, we analyze its movement through the five task positions. Our approach uses the Dixon determinant elimination procedure described in Wampler (2001) [12] and Neilson and Roth (1999) [13], to find all the solutions to the forward kinematics problem. They refer to this procedure as solving an input/output problem for planar linkages. Dhingra et al.(2000) [14] showed that the displacement analysis problem for the eight-bar mechanism can be reduced into a univariate polynomial devoid of any extraneous roots. Apart from the elimination methods, there are numerical methods ranging from Newton's method, which finds a solution near an initial guess, to more sophisticated methods like polynomial continuation, that can find all solutions.

Analyses using the Dixon determinant approach requires the selection of three loops for the synthesized eight-bar linkage, along with generation of the corresponding loop equations. For that we use the procedure developed by Parrish et al.(2013) [15, 16].

Plecnik and McCarthy (2012) [17] demonstrated the effectiveness of random variation of the task positions within tolerance zones in order to increase the number of candidate linkages, for evaluation in the design of a 5SS spatial platform. Sonawale and McCarthy (2013) [18] extended this to spherical six-bar linkages.

Our formulation is intended to find useful eight-bars by verifying the synthesized linkages for branch defects. This ensures smooth movement through the five task positions, when the input link is rotated within the prescribed range. We seek linkage solutions using the 32 different connection types for the two RR constraints combined with a random search in tolerance zones around the task positions. It is found that using this technique, 8 out of 16 eight-bar topologies can be obtained, refer Tsai (2001) [19].

SYNTHESIS THEORY

Our synthesis procedure for the eight-bar linkage begins with the specification of a set of five task positions and the 6R loop. The eight-bar is obtained by designing two RR links that constrain the 6R loop to a single degree-of-freedom eight-bar linkage. This synthesis procedure is described in McCarthy and

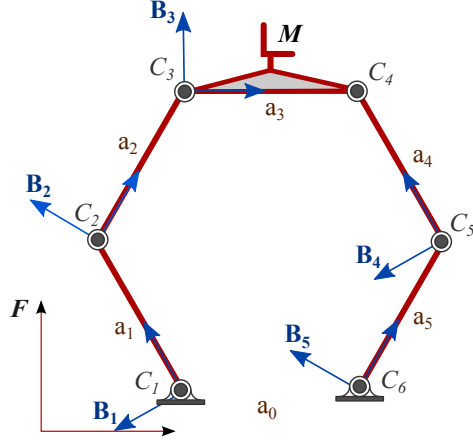


FIGURE 2. COORDINATE TRANSFORMATION FRAMES ATTACHED TO THE 5 LINKS $\{a_1, a_2, a_3, a_4, a_5\}$ OF THE 6R LOOP.

Soh [10], and we summarize it here.

Inverse Kinematics of a 3R chain

Let $[G]$ denote the transformation matrix that maps the base of the 3R chain to the fixed frame F , and let the position of each of the link frames be denoted by $[B_i]$, $i = 1, 2, 3$ and let $[H]$ map the tool frame in the last link of the 3R chain. Then the kinematics equation for the 3R chain is given by

$$[K(\theta_1, \theta_2, \theta_3)] = [G][B_1(\theta_1)][B_2(\theta_2)][B_3(\theta_3)][H], \quad (1)$$

where θ_i , $i = 1, 2, 3$ are the joint rotations for the 3R chain.

With the kinematics equation of the 3R chain $[K(\theta_1, \theta_2, \theta_3)]$ and the five task positions $[T_j]$, $j = 1, \dots, 5$ expressed in the 3×3 homogeneous form, we use the inverse kinematics of the 3R chain to compute the angles $(\theta_{1j}, \theta_{2j}, \theta_{3j})$ such that,

$$[T_j] = [K(\theta_{1j}, \theta_{2j}, \theta_{3j})], \quad j = 1, \dots, 5. \quad (2)$$

The joint values $(\theta_{1j}, \theta_{2j}, \theta_{3j})$, $j = 1, \dots, 5$ allow us to compute the link coordinate frames $[B_{1,j}], [B_{2,j}], [B_{3,j}]$ relative to the ground frame as,

$$\begin{aligned} [B_{1,j}] &= [G][B_1(\theta_{1,j})], \\ [B_{2,j}] &= [G][B_1(\theta_{1,j})][B_2(\theta_{2,j})], \\ [B_{3,j}] &= [G][B_1(\theta_{1,j})][B_2(\theta_{2,j})][B_3(\theta_{3,j})] \\ & \quad j = 1, \dots, 5. \end{aligned} \quad (3)$$

Using this theory, inverse kinematics is performed twice on the two 3R chains of the 6R loop separately, to determine coordinate frames $([B_{1,j}], [B_{2,j}], [B_{3,j}], [B_{4,j}], [B_{5,j}])$ attached to links

$(a_1, a_2, a_3, a_4, a_5)$ at points $(C_1, C_{2,j}, C_{3,j}, C_{5,j}, C_6)$, $j = 1, \dots, 5$ respectively, as shown in Fig.2. These frames will be used for synthesizing the RR constraints.

Synthesis of an RR Constraint

Let $[B_{l,j}]$, $j = 1, \dots, 5$ be the five positions of the l^{th} moving link and let $[B_{k,j}]$, $j = 1, \dots, 5$ be the five positions of the k^{th} moving link measured in the fixed frame F . Let \mathbf{g} be the coordinates of one the pivots of the RR constraint, in the l link, measured in the $[B_l]$ frame. Similarly let \mathbf{w} be the coordinates of the other pivot of the RR constraint, in the k link, measured in the $[B_k]$ frame.

The five positions of these points as the two frames move between the task positions are given by

$$\mathbf{G}_j = [B_{l,j}]\mathbf{g} \quad \text{and} \quad \mathbf{W}_j = [B_{k,j}]\mathbf{w}. \quad (4)$$

Now, we introduce the relative displacements

$$[R_{l,j}] = [B_{l,j}][B_{l,1}]^{-1} \quad \text{and} \quad [S_{k,j}] = [B_{k,j}][B_{k,1}]^{-1}, \quad (5)$$

which allows us to write,

$$\mathbf{G}_j = [R_{l,j}]\mathbf{G}_1 \quad \text{and} \quad \mathbf{W}_j = [S_{k,j}]\mathbf{W}_1, \quad (6)$$

where $[R_{l,1}] = [S_{k,1}] = [I]$ are the identity transformations. The points \mathbf{G}_j and \mathbf{W}_j define the ends of the RR constraint of length R . We can now write the five constraint equations specifying that, this link should have the same length R in all five positions. Mathematically they are expressed as

$$([S_{k,j}]\mathbf{W}_1 - [R_{l,j}]\mathbf{G}_1) \cdot ([S_{k,j}]\mathbf{W}_1 - [R_{l,j}]\mathbf{G}_1) = R^2, \quad j = 1, \dots, 5 \quad (7)$$

These equations can be solved numerically to yield as many four sets of solutions for the RR constraint link (\mathbf{G}, \mathbf{W}) , refer McCarthy [10].

Finding all 32 Different Eight-Bar Linkages

The heart of the synthesis procedure is to find all possible ways in which two RR constraints can be applied to the user defined 6R loop, to make it a single degree-of-freedom eight-bar linkage. We start with the 6R loop links, $\{a_1, a_2, a_3, a_4, a_5\}$. Notice that the ground link a_0 is not part of this list, as we enforce that no new connection to the ground is made, apart from the two ground pivots defined by the user as part of the 6R loop.

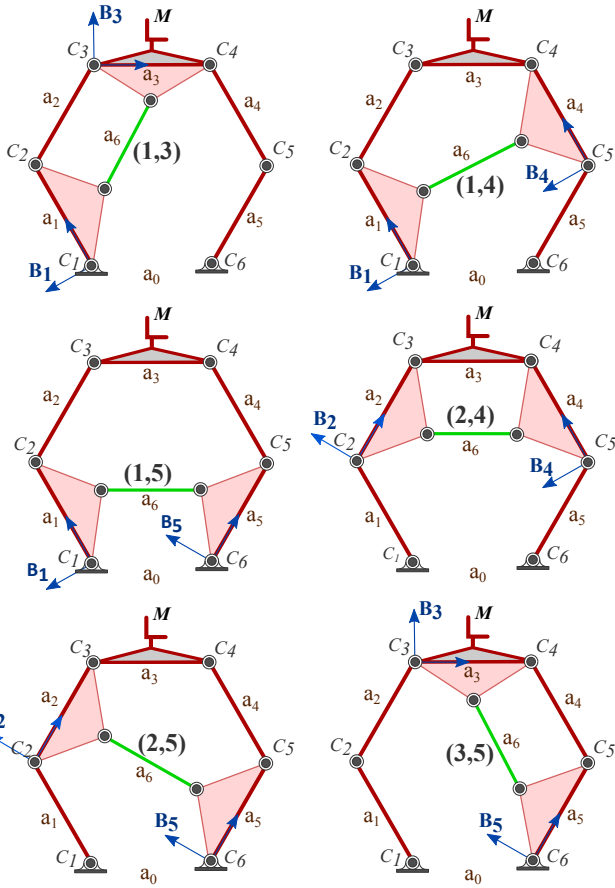


FIGURE 3. 6 DIFFERENT WAYS TO SELECT A PAIR OF LINKS FROM THE 6R LOOP, TO APPLY THE FIRST RR CONSTRAINT.

To apply the first RR constraint, we select 2 links from this list of 5, to form a link pairs, which could be done in $\binom{5}{2} = 10$ ways. The 10 link pairs thus generated are given as, $\{\{a_1, a_2\}, \{a_1, a_3\}, \{a_1, a_4\}, \{a_1, a_5\}, \{a_2, a_3\}, \{a_2, a_4\}, \{a_2, a_5\}, \{a_3, a_4\}, \{a_3, a_5\}, \{a_4, a_5\}\}$. Next we remove the link pairs which consist of adjacent links from the 6R loop. This is to avoid the formation of a structure, between the adjacent links and the newly generated RR constraint link. This removal reduces the list to

$$\{\{a_1, a_3\}, \{a_1, a_4\}, \{a_1, a_5\}, \{a_2, a_4\}, \{a_2, a_5\}, \{a_3, a_5\}\}. \quad (8)$$

These 6 link pairs across which the first RR constraint can be applied are shown in Fig.3. Connection between link pairs $\{\{a_1, a_3\}, \{a_1, a_5\}, \{a_2, a_4\}, \{a_3, a_5\}\}$ divides the 6R loop into two connected sub-loops 4R and 6R, whereas connection between link pairs $\{a_1, a_4\}$ and $\{a_2, a_5\}$ divides the 6R loop in two connected 5R sub-loops. There are two strategies that could be used to add the second RR constraint. They are summarized as follows:

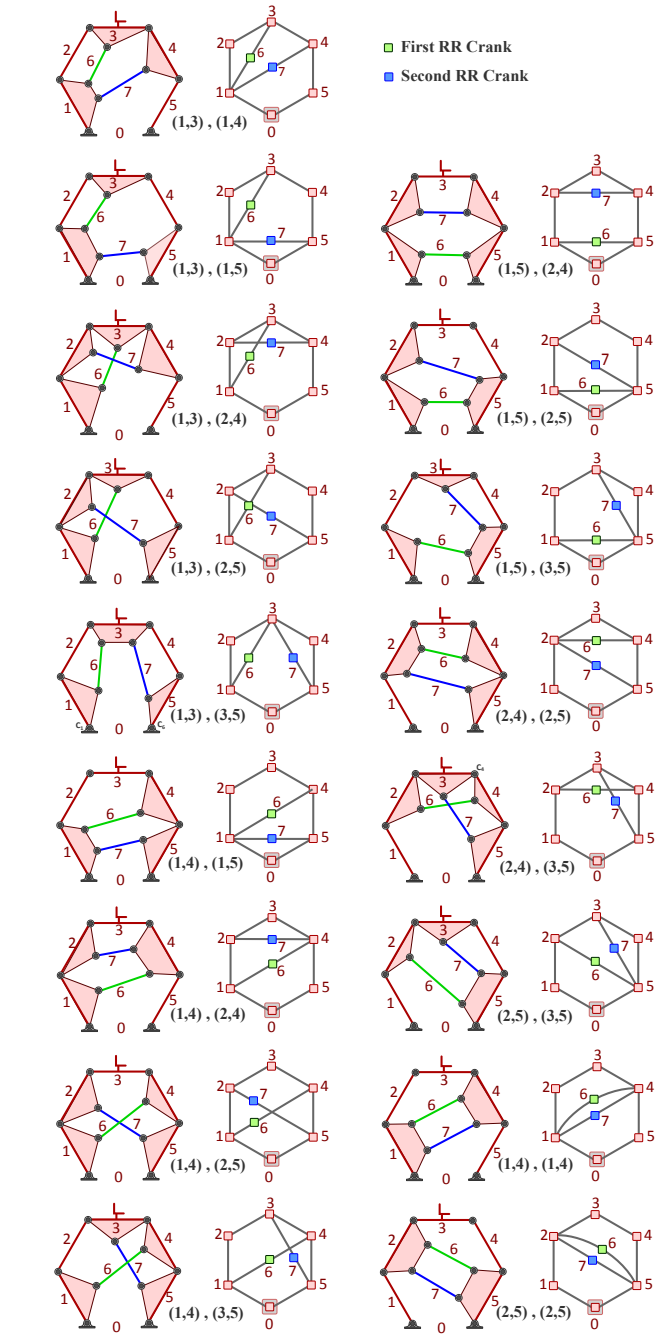


FIGURE 4. 17 EIGHT-BAR LINKAGES OBTAINED BY ADDING TWO INDEPENDENT RR CONSTRAINTS TO THE 6R LOOP.

low:

Two Independent RR Constraints

Here the second RR constraint is added independent of the first. Thus this becomes a problem of selecting two link pairs

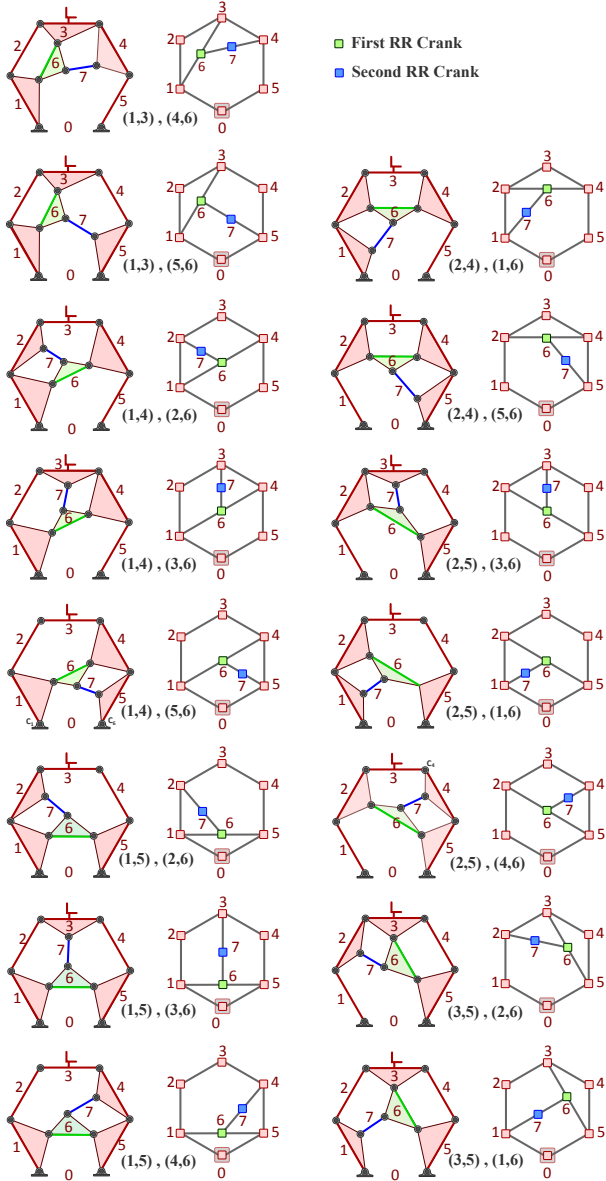


FIGURE 5. 15 EIGHT-BAR LINKAGES OBTAINED BY ADDING A SEQUENCE OF TWO RR CONSTRAINTS TO THE 6R LOOP. THE SECOND CONSTRAINT IS CONNECTED TO THE LINK INTRODUCED BY THE FIRST AND THE AVAILABLE LINKS OF THE 6R LOOP.

from the list of 6 available for the first RR constraint, as shown in Eq.8 and Fig.3. Two connections out of 6 can be selected in ${}^6C_2 = 15$ ways. There is an exception for the link pairs $\{a_1, a_4\}$ and $\{a_2, a_5\}$. Since these connections divide the loop into two connected 5R sub-loops, we can have two RR constraints between the same link pairs. This makes the total count of the synthesized eight-bar linkages, due to application of independent RR

constraints, equal to $15 + 2 = 17$, as shown in refer Fig. 4.

Two Dependent RR Constraints

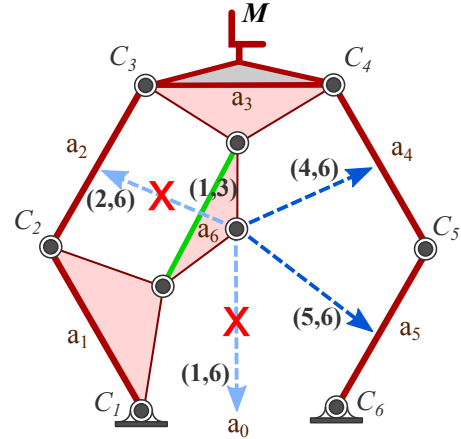


FIGURE 6. FOR THE FIRST RR CONSTRAINT APPLIED BETWEEN LINK PAIR (1,3), LINKS AVAILABLE FROM THE 6R LOOP FOR APPLYING SECOND RR CONSTRAINT ARE 4 AND 5. THE CANDIDATE LINK PAIRS FOR APPLYING THE SECOND RR CONSTRAINTS ARE (4,6) AND (5,6).

Here the second RR constraint is added to the first RR constraint link and one of the available links of the 6R loop (except the ground link a_0). It is to be noted that, the addition of the first RR constraint adds a new link a_6 to the total number of links. In order to apply the second RR constraint, a search is performed to find all the available 6R loop links, that could be used for connection with the newly found a_6 link. This search has to be done for each of the 6 first RR connection pairs, shown in Fig.3.

A specific heuristic is employed for eliminating the links of the 6R loop, when the first RR constraint results in a 4R sub-loop. In this case the links of the 6R loop associated with the 4R sub-loop are removed from the list. Consider an example shown in Fig.6. For the first RR constraint between link pair $\{a_1, a_3\}$, the only available links of the 6R loop that could be used for connection to the a_6 link are $\{a_4, a_5\}$. This is because link a_2 has to be eliminated owing to its involvement in the 4R sub-loop formed due to the first RR constraint, and we enforce that no new connection to the ground link a_0 is made.

Similarly, when the first link pair is $\{a_2, a_4\}$ or $\{a_3, a_5\}$, only two more connections are possible. If the first link pair is $\{a_1, a_5\}$, all the three remaining 6R loop links $\{a_2, a_3, a_4\}$ are available thus making a total of $3 * 2 + 3 = 9$ linkages that could be synthesized. For the first constraint link pairs $\{\{a_1, a_4\}$ and $\{a_2, a_5\}\}$, three links of the 6R loop are available for second connection adding to the

total number of eight-bar linkages possible, using this strategy of employing dependent constraints, to $9 + 2 * 3 = 15$ as shown in Fig. 5. Thus the total number of different eight-bar linkages possible using the two strategies, independent and dependent constraints, are 32.

Synthesis of Eight-bar Linkages

The last section described how 32 eight-bar linkages can be synthesized, by adding two RR constraints to the 6R loop. It is to be noted that these 32 eight-bar linkages represent only 8 different topologies out of 16, refer Tsai (2001) [19]. When an RR constraint is synthesized we can get a maximum of 4 solutions for it. This result in the spawning of multiple linkages due to the various combinations of the RR constraint solutions for both first and second constraints. This section describes the maximum number of combinatorial linkages we can get for all the 32 eight-bar linkages combined.

For the first RR constraint, using the connection list $\{\{a_1, a_3\}, \{a_1, a_4\}, \{a_1, a_5\}, \{a_2, a_4\}, \{a_2, a_5\}, \{a_3, a_5\}\}$ and inverse kinematics of the two 3R chains, we can generate the coordinate transformation frame connection list $\{\{B_1, B_3\}, \{B_1, B_4\}, \{B_1, B_5\}, \{B_2, B_4\}, \{B_2, B_5\}, \{B_3, B_5\}\}$ for the five positions, refer Fig.3. Now for the frame connections $\{\{B_1, B_3\}, \{B_1, B_5\}, \{B_2, B_4\}$ and $\{B_3, B_5\}\}$, out of the four RR constraint solutions obtained, one is part of the 6R loop, hence has to be dropped. For frame connections $\{\{B_1, B_4\}, \{B_2, B_5\}\}$ all four RR constraint solutions are available. For Strategy I, when both RR constraints are applied independently, based on the combinations of the frame pairs listed above, there can be a maximum of $(8 * (3 * 4)) + (6 * (3 * 3)) + (1 * (4 * 4)) + (2 * (2 * 3)) = 178$ number of linkages, that could be synthesized.

For adding the second RR constraint between the first RR constraint and the other links of the 6R loop, we need to define in addition to the five frame $\{B_1, \dots, B_5\}$, an additional frame B_6 , which is attached to the newly generated link. The number of solutions we can get for the connections $\{B_6, B_i\}$, $i = 1, \dots, 5$ depends on the link connection pair for the first RR constraint. It is important to note that one of the four solutions obtained for the second RR constraint, will be one of the two sides of the two ternary links which emerge out of the connection link pair for the first RR constraint. In all, using this approach of applying dependent constraints we can get up to $(9 * (3 * 4)) + (6 * (3 * 3)) = 162$. Thus the maximum number of linkage solutions we can get for all the 32 eight-bar linkages combined is $(178 + 162) = 340$.

Using Randomization of the task positions within acceptable variations during each iteration, we can generate more candidate linkage solutions, refer the software flow chart in Fig.7. These linkages are then sorted by the ratio of the longest link to the smallest link. Note that in case of a ternary or quaternary link, all the sides are considered for the link length assessment. The design algorithm relays this list of candidate linkages to the anal-

ysis routine which is explained in the following section.

ANALYSIS OF THE EIGHT-BAR LINKAGES

The design procedure uses an analysis algorithm described in Parrish and McCarthy [15,16] for forward kinematics of multi-loop linkage systems. The algorithm uses an automated loop generation technique based on Graph theory. It uses the Dixon Determinant approach to find all possible solutions for all the unknown joint angles, for a given input angle. This enables us to determine every possible assembly configuration for the linkage. This section describes the information exchange by our program with the analysis algorithm, and post processing the forward kinematic solutions to find useful eight-bar linkages.

Input data to the Analysis Algorithm

The main program, shown in Fig.7, uses the analysis routine seamlessly in it. Since the analysis routine is designed for any generic single dof linkage, the linkage information sent to this routine has to be formatted in a specific way. This section describes the formatting of the input data for the analysis routine.

An adjacency matrix is convenient way to represent the connections (revolute joints) between the various links. For the example linkage shown in Fig.10, the adjacency matrix is given as

$$P_o = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \quad (9)$$

For the input to the analysis routine, each synthesized candidate linkage is expressed as a modified adjacency matrix $[P]$, with the $1s$ in $[P_0]$ replaced by the joint coordinates (x, y) for each of the joints $\{C_1, \dots, C_{10}\}$. For example, $[P_{1,2}]$ and $[P_{2,1}]$ representing the connection between links a_1 and a_2 , will display the coordinates C_2 . The input data \mathbf{V} , for the analysis algorithm is a list, consisting of the modified adjacency matrix $[P]$, ground link a_8 and the input link number a_1 , expressed as:

$$\mathbf{V} = \{[P], a_8, a_1\}. \quad (10)$$

For the example linkage the input is,

$$\mathbf{V} = \left\{ \begin{array}{cccccccc} 0 & C_2 & 0 & 0 & 0 & 0 & C_9 & C_1 \\ C_2 & 0 & C_3 & 0 & 0 & C_7 & 0 & 0 \\ 0 & C_3 & 0 & C_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_4 & 0 & C_5 & C_8 & 0 & 0 \\ 0 & 0 & 0 & C_5 & 0 & 0 & 0 & C_6 \\ 0 & C_7 & 0 & C_8 & 0 & 0 & C_{10} & 0 \\ C_9 & 0 & 0 & 0 & 0 & C_{10} & 0 & 0 \\ C_1 & 0 & 0 & 0 & C_6 & 0 & 0 & 0 \end{array} \right\}, 8, 1 \quad (11)$$

Output data from the Analysis Algorithm

The output \mathbf{W} from the analysis routine is,

$$\mathbf{W} = \{[M], [N], \mathbf{K}, r, \mathbf{s}, \mathbf{t}, [Q], FTLA\}. \quad (12)$$

Matrices $\{[M], [N]\}$ form a matrix pair, which will be used to find the unknown joint angles by solving the generalized eigenvalue problem. Vector \mathbf{K} for the example linkage is

$$\mathbf{K} = \{T_3T_4, T_3T_5, T_4T_5, T_3T_6, T_4T_6, T_5T_6, T_3T_7, T_4T_7, T_5T_7, T_3T_4T_6, T_3T_5T_6, T_4T_5T_6, T_3T_4T_7, T_3T_5T_7, T_4T_5T_7, T_3T_6T_7, T_4T_6T_7, T_5T_6T_7\}. \quad (13)$$

Variable r represents which joint angle will correspond to the eigenvalues. Vector \mathbf{s} represent a list of other five joint angles, which will be obtained from the eigenvectors. Vector \mathbf{t} is the ratio index information. Modified adjacency matrix $[Q]$ enables us to make a one to one correspondence between the joint naming conventions used by the analysis routine and this program, by comparing with matrix $[P]$. For the example linkage,

$$[Q] = \begin{bmatrix} 0 & j1t2 & 0 & 0 & 0 & 0 & j1t7 & j8t1 \\ j1t2 & 0 & j2t3 & 0 & 0 & j2t6 & 0 & 0 \\ 0 & j2t3 & 0 & j3t4 & 0 & 0 & 0 & 0 \\ 0 & 0 & j3t4 & 0 & j4t5 & j6t4 & 0 & 0 \\ 0 & 0 & 0 & j4t5 & 0 & 0 & 0 & j5t8 \\ 0 & j2t6 & 0 & j6t4 & 0 & 0 & j7t6 & 0 \\ j1t7 & 0 & 0 & 0 & 0 & j7t6 & 0 & 0 \\ j8t1 & 0 & 0 & 0 & j5t8 & 0 & 0 & 0 \end{bmatrix}. \quad (14)$$

Comparing with the matrix $[P]$, we find that $j1t2 = C_2, j2t3 = C_3, j3t4 = C_4, j4t5 = C_5, j5t8 = C_6, j2t6 = C_7, j6t4 = C_8, j1t7 = C_9, j7t6 = C_{10}$. $FTLA$ is a compact representation for the three loop equations and will be used not only for the forward kinematics but also for animating the linkage.

Forward Kinematics

In this section we will describe, how the solution to the generalized eigenvalue problem is converted to forward kinematic solutions for the joint angles. From the inverse kinematics of the first 3R chain $\{C_1, C_2, C_3\}$ of the 6R loop, we get the five input angles $\theta_{1,j}, j = 1, \dots, 5$ for a_1 corresponding to the five task positions, ref Fig.1. These five angles form four angle ranges between them 1-2, 2-3, 3-4, 4-5. These ranges are discretized to form an array of input angles, $\theta_{1,j}, j = 1, \dots, n$ of length n .

The matrices $[M]$ and $[N]$ have T_1 and T_{c1} as the only unknowns. So a loop of size n is setup, and for each iteration $j = 1, \dots, n$, the values for $T_{1,j}$ and $T_{c1,j}$ are calculated as $T_{1,j} = e^{i\theta_{1,j}}$ and $T_{c1,j} = e^{-i\theta_{1,j}}$ respectively. These values are substituted in the two matrices $[M]$ and $[N]$ and the generalized eigenvalue problem is solved as,

$$[N_j]\mathbf{v} = \lambda[M_j]\mathbf{v}, \quad (15)$$

where the generalized eigenvalues λ correspond to all the possible joint angles, in isotropic form, for the link specified by the variable r . For the linkage discussed in the example, $r = \theta_2$, which means that the eigenvalues correspond to all the possible angles for link 2, for the given input link angle $\theta_{1,j}$. The generalized eigenvectors \mathbf{v} correspond to the other joint angles in isotropic form. It should be noted that $\mathbf{v} = v_i, i = 1, \dots, 18$ is defined up to a constant multiple, say μ . Therefore the other joint angles in isotropic form are found by computing the ratios,

$$T_{3,j} = \frac{v_{10}}{v_5}, \quad T_{4,j} = \frac{v_{10}}{v_4}, \quad T_{5,j} = \frac{v_{11}}{v_4}, \quad (16)$$

$$T_{6,j} = \frac{v_{10}}{v_1}, \quad T_{7,j} = \frac{v_{13}}{v_1}.$$

The ratio information is obtained from the vector \mathbf{t} and can be confirmed by checking the corresponding ratios on the vector $\mathbf{K} = K_i, i = 1, \dots, 18$, as shown below:

$$T_{3,j} = \frac{K_{10}}{K_5}, \quad T_{4,j} = \frac{K_{10}}{K_4}, \quad T_{5,j} = \frac{K_{11}}{K_4}, \quad (17)$$

$$T_{6,j} = \frac{K_{10}}{K_1}, \quad T_{7,j} = \frac{K_{13}}{K_1}.$$

An isotropic angle T is real if its norm is equal to 1. The angle in radians is extracted from the isotropic angle by finding its argument. For an input angle $\theta_{1,j}$ we can get a max of 18 solutions for $\{\theta_{2,j}, \theta_{3,j}, \theta_{4,j}, \theta_{5,j}, \theta_{6,j}, \theta_{7,j}\}$. This means that for a given input angle $\theta_{1,j}$, we can have a maximum of 18 different configurations possible for the linkage. In actuality this number is a lot smaller. For $j = 1, \dots, n$, these solutions are compiled as

a list, given by $\{\theta_{1,j}, \theta_{2,j}, \theta_{3,j}, \theta_{4,j}, \theta_{5,j}, \theta_{6,j}, \theta_{7,j}, \theta_{8,j}\}$. All solution lists are stored in a matrix $[S]$, which is of size $(8 \times 18 \times n)$, such that

$$[S_j] = \begin{bmatrix} \theta_{1,j} & \theta_{1,j} & \cdots & \theta_{1,j} & \theta_{1,j} \\ \theta_{2,1,j} & \theta_{2,2,j} & \cdots & \theta_{2,17,j} & \theta_{2,18,j} \\ \theta_{3,1,j} & \theta_{3,3,j} & \cdots & \theta_{3,17,j} & \theta_{3,18,j} \\ \theta_{4,1,j} & \theta_{4,2,j} & \cdots & \theta_{4,17,j} & \theta_{4,18,j} \\ \theta_{5,1,j} & \theta_{5,2,j} & \cdots & \theta_{5,17,j} & \theta_{5,18,j} \\ \theta_{6,1,j} & \theta_{6,2,j} & \cdots & \theta_{6,17,j} & \theta_{6,18,j} \\ \theta_{7,1,j} & \theta_{7,2,j} & \cdots & \theta_{7,17,j} & \theta_{7,18,j} \\ \theta_8 & \theta_8 & \cdots & \theta_8 & \theta_8 \end{bmatrix}_{(8 \times 18)}, j = 1, \dots, n \quad (18)$$

Note that the angle of the ground link a_0 is constant θ_8 . For every $[S_j]$, each column represents one solution or rather one assembly configuration of the eight-bar linkage.

Sorting Branches

We denote a linkage configuration as a set of angles $\{\theta_2, \dots, \theta_7\}$, made by the links $\{a_2, \dots, a_7\}$. A linkage configuration is a compact representation of how the various links are assembled. For a given input angle θ_1 , there can be many configurations possible as discussed before. The forward kinematic solutions $[S_j]$, $j = 1, \dots, n$ (columns) give us all possible linkage configurations, when the input angle θ_1 is incremented from the starting angle to the ending angle, that is from $\theta_{1,1}$ to $\theta_{1,n}$.

The goal of the sorting algorithm is to track the different configurations and then sort them into branches. Figure 8 displays five linkage configurations sorted into branches in each of the plots for the angles $\{\theta_2, \dots, \theta_7\}$ drawn against the input angle θ_1 . The procedure for sorting the branches is described in Plecnik and McCarthy [20] and is explained briefly below.

An eight-bar linkage has three loops. Each loop will have two loop closure equations respectively for the x and y coordinates of the pivots involved in the loop. The loop equations are obtained from the *FTLA* format, provided by the analysis algorithm and are represented by the vector,

$$\mathbf{f} = \begin{bmatrix} f_{1x}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \\ f_{1y}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \\ f_{2x}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \\ f_{2y}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \\ f_{3x}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \\ f_{3y}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (19)$$

Next we obtain the Jacobian for the loop equations vector, \mathbf{f} , by taking partial derivatives with respect to the configuration angles

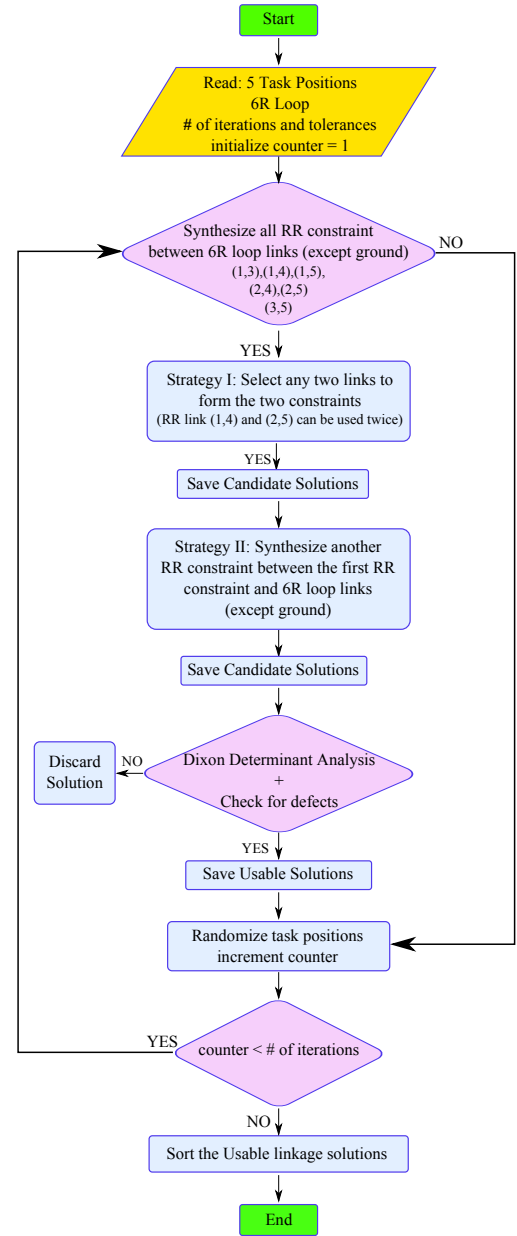


FIGURE 7. DESIGN PROCEDURE FLOW CHART.

$\{\theta_2, \dots, \theta_7\}$ as,

$$[J_f] = \begin{bmatrix} \frac{\partial f_{1x}}{\partial \theta_2} & \cdots & \frac{\partial f_{1x}}{\partial \theta_7} \\ \frac{\partial f_{1y}}{\partial \theta_2} & \cdots & \frac{\partial f_{1y}}{\partial \theta_7} \\ \frac{\partial f_{2x}}{\partial \theta_2} & \cdots & \frac{\partial f_{2x}}{\partial \theta_7} \\ \frac{\partial f_{2y}}{\partial \theta_2} & \cdots & \frac{\partial f_{2y}}{\partial \theta_7} \\ \frac{\partial f_{3x}}{\partial \theta_2} & \cdots & \frac{\partial f_{3x}}{\partial \theta_7} \\ \frac{\partial f_{3y}}{\partial \theta_2} & \cdots & \frac{\partial f_{3y}}{\partial \theta_7} \end{bmatrix}. \quad (20)$$

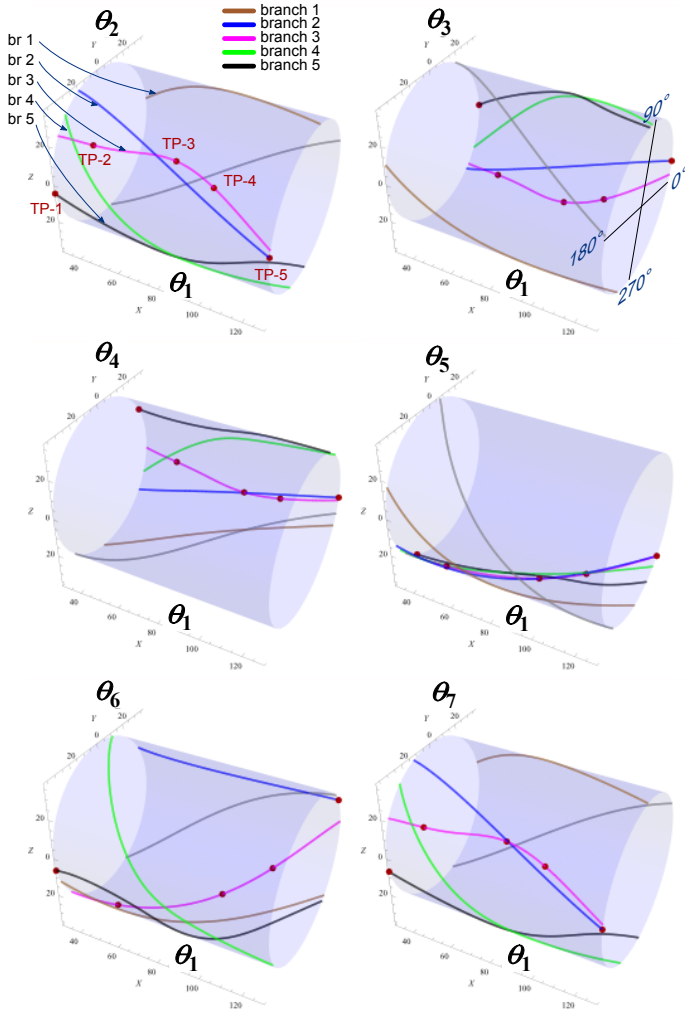


FIGURE 8. LINKAGE CONFIGURATIONS (BRANCHES) FOR AN EIGHT-BAR LINKAGE SUFFERING FROM BRANCH DEFECT. TASK POSITION CONFIGURATIONS 1 AND 5 LIE ON BRANCHES 5 AND 2 RESPECTIVELY, WHILE TASK POSITION CONFIGURATIONS 2, 3 AND 4 LIE ON THE SAME BRANCH 3.

For the sake of being succinct, we will refer to the linkage configurations of $[S_j]$ (columns) as $\mathbf{Z}_{j,k}, k = 1, \dots, 18$. Using $\mathbf{Z}_{1,k}$ from $[S_1]$ as the initial condition, we will now use Newton's Raphson method to find the evolution of each $\mathbf{Z}_{1,k}$, when the input angle θ_1 is incremented from $\theta_{1,1}$ to $\theta_{1,2}$. We refer to it as $\mathbf{Z}_{(2,k)p}$, where p stands for predicted value. This is given by:

$$\mathbf{Z}_{(2,k)p} = \mathbf{Z}_{1,k} + [\mathbf{J}_f(\mathbf{Z}_{1,k})]^{-1} \mathbf{f}(\theta_{1,2}, \mathbf{Z}_{1,k}), \quad k = 1, \dots, 18 \quad (21)$$

Next we match the configurations of $\mathbf{Z}_{(2,k)p}$ with each of the corresponding configurations in $[S_2]$, that is $\mathbf{Z}_{2,k}$ and save the sorted

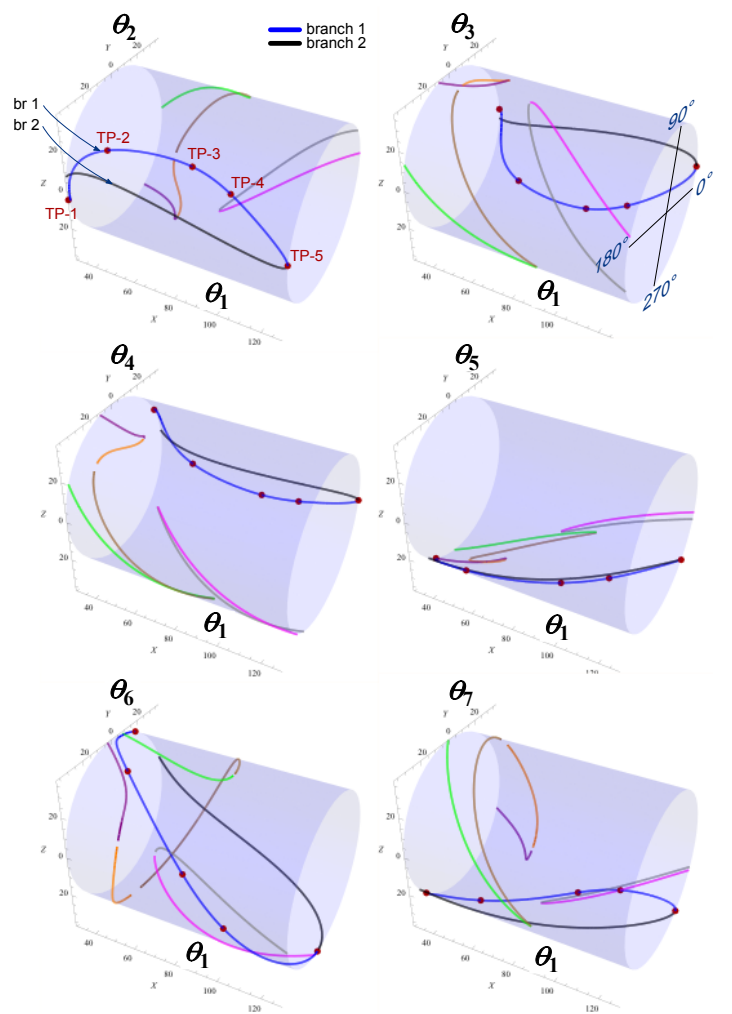


FIGURE 9. LINKAGE CONFIGURATIONS (BRANCHES) FOR THE USEFUL EIGHT-BAR LINKAGE, SELECTED FOR THE CONVERTIBLE SOFA-BED EXAMPLE PROBLEM. ALL FIVE TASK POSITION CONFIGURATIONS LIE ON BRANCH 1.

solutions in the list \mathbf{W} as branches. Now the input angle is incremented to $\theta_{1,3}$ and with the sorted $\mathbf{Z}_{(2,k)}$ as initial conditions, Newton-Raphson method is used again to find $\mathbf{Z}_{(3,k)p}$. This process is continued while $j \leq n$ and after each iteration the branches are updated and saved to the list \mathbf{W} . Figure 8 displays the five sorted branches in each of the 6 plots individually.

Linkage Defect Check

In this section we explain how we determine if a particular eight-bar linkage is useful. First we search for the branches in the list \mathbf{W} that are of length n . Each of these branches represent a linkage configuration, that exist throughout the motion of the

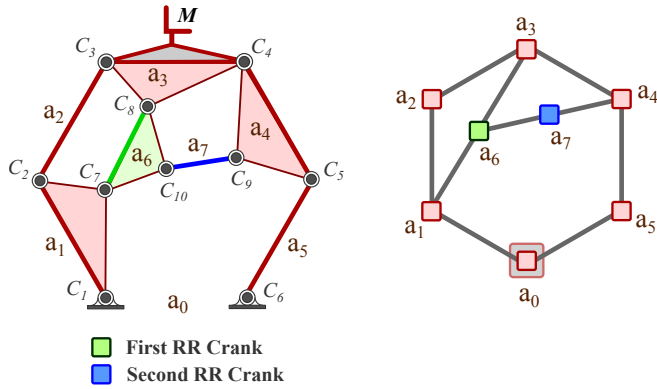


FIGURE 10. TOPOLOGY OF THE SELECTED EIGHT-BAR LINKAGE FOR THE CONVERTIBLE SOFA-BED EXAMPLE PROBLEM.

input link θ_1, j for $j = 1, \dots, n$. This means that, when the input link is rotated, the linkage moves smoothly in that configuration. If a branch is of length less than n , then it means that singularity has occurred at some input angle θ_1, j . This locks the linkage and hence have to be discarded.

If none of the branches in the list \mathbf{W} are of length n , the candidate linkage is discarded. If there are a few branches that are of length n , only these branches are retained in the list \mathbf{W} . Now for each complete branch, it is checked, if the end-effector reaches all the five task positions when the linkage moves in this configuration. This is done by verifying whether the five task position configurations lie on a single branch. If yes, then that particular branch is the correct configuration for the linkage and the linkage is deemed defect-free and the configuration saved.

Figure 8 shows a defective linkage. For this linkage, three task position configurations 2, 3 and 4 lie on branch3, while task position configuration 1 and 5 lie on branches 5 and 2 respectively. So if the linkage moves in the configuration of branch3, only task positions 2, 3 and 4 will be hit. This verifies that this candidate linkage suffers from branch defect and hence it is discarded. Figure 9 shows a defect-free linkage, where all the five task position configurations lie on a single branch (branch1). If the defect-free linkage also satisfies the permissible values for minimum and maximum link length values, specified by the user, it is deemed as useful.

EXAMPLE LINKAGE

In this section, we use the convertible sofa-bed linkage example previously seen in Soh and McCarthy (2007) [1], to verify the design process. The five task positions for this problem are listed in Table 1 and the user defined 6R loop data is given in Table 2. The Tolerance zone for each task position is a list $(\Delta\theta, \Delta x, \Delta y)$, where $\Delta\theta$ is the tolerance on the orientation of the

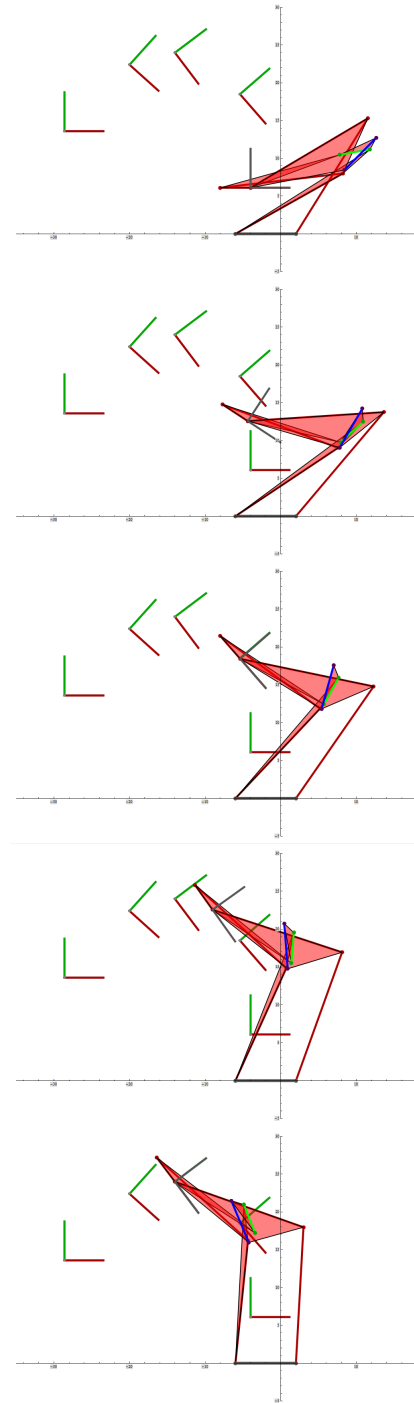


FIGURE 11. THE USEFUL EIGHT-BAR SELECTED FOR THE CONVERTIBLE SOFA-BED EXAMPLE, IS SHOWN MOVING SMOOTHLY THROUGH THE 5 TASK POSITIONS WITH SOME INTERMEDIATE POSITIONS. 5 OUT OF 9 POSITIONS SHOWN.

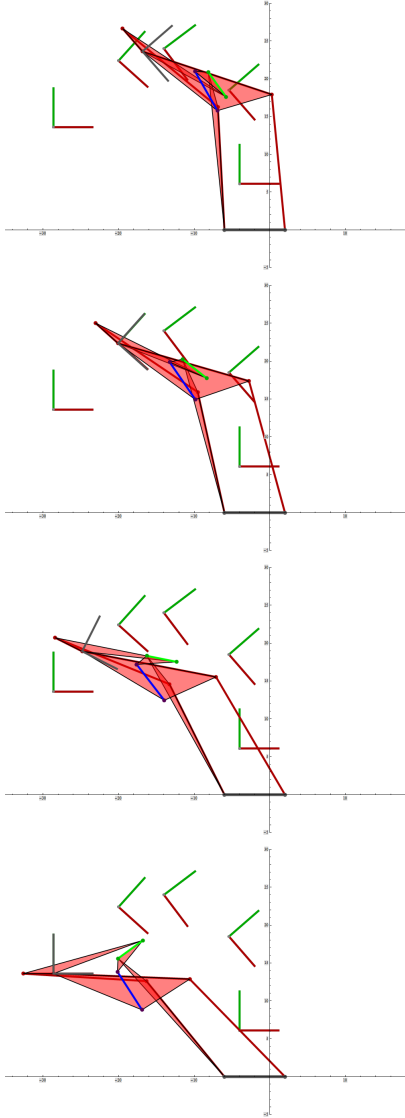


FIGURE 12. THE USEFUL EIGHT-BAR SELECTED FOR THE CONVERTIBLE SOFA-BED EXAMPLE, IS SHOWN MOVING SMOOTHLY THROUGH THE 5 TASK POSITIONS WITH SOME INTERMEDIATE POSITIONS. REMAINING 4 POSITIONS ARE SHOWN.

task position and Δx and Δy are the tolerances on the x and y coordinates of the origin for the task position. These are mentioned in Table 3.

This input data was sent to the synthesis routine to add the two RR constraints using the strategies of adding them independently or in sequence, which could be done in 32 different ways. Note that for a single iteration, the program runs only on the original task positions. For multi-iteration run, the first iteration al-

ways runs on the original task positions and successive iterations use randomized task positions, within the tolerance zones. For this example, a total of 61 candidate linkages were synthesized for the original task positions.

These candidate linkages were analyzed and their configuration trajectories were sorted into branches. Figure 9 shows a candidate linkage with all the five task position configurations lying on a single branch (branch1). This ensures that the linkage is defect free and will smoothly move through the five task positions. Figure 11,12 show the linkage movement through the five task positions along with few intermediate positions.

For a single iteration run, only two linkages were found to be defect-free. In addition to the input data, the user may also add the permissible values for the minimum and maximum allowable link length for the linkage. For the current example, the metric for comparison was derived from the side length of a square fitted around the 5 task positions. This eliminated one solution, thus leaving only one useful linkage. The useful linkage had the RR connections between link pairs (1,3) and (4,6), topology of which is shown in Fig.10. The linkage solution is mentioned in Table 4.

The statistics for this example problem, with the program running on different machines with several iterations, is shown in Table 5. The program was run in serial and parallel, and substantial speed improvement was reported using Mathematica's automatic parallel computation. The first two rows of the Table 5 show that for a 4-core Intel machine, the time went down from 9.29 min to 2.75 min for one iteration. We tested the program on a 12 core Intel machine for 1, 10 and 100 iterations and the speed increase was even greater. The next step towards speed improvement is by GPU computing. Mathematica provides an interface called *CUDALink* to program the GPU efficiently.

TABLE 1. FIVE TASK POSITIONS

<i>Task</i>	<i>Orientation (θ)</i>	<i>Location(x,y)</i>
<i>(degrees)</i>		
1	0°	(-4.0, 6.1)
2	-49°	(-5.4, 18.5)
3	-53°	(-14.0, 24.0)
4	-42°	(-20.0, 22.4)
5	0°	(-28.6, 13.6)

TABLE 2. SIX-R LOOP COORDINATES

<i>Pivot</i>	<i>Location Data (x,y)</i>
C_1	(-6,0)
C_2	(8.2,8)
C_3	(-8,6.1)
C_4	(-4,6.1)
C_5	(11.5,15.3)
C_6	(2.0)

TABLE 3. FIVE TOLERANCES ON TASK POSITIONS

<i>Task</i>	<i>Tolerance Data ($\Delta\theta, \Delta x, \Delta y$)</i>
1	(0°, 1.0, 1.0)
2	(5°, 2.0, 2.0)
3	(5°, 2.0, 2.0)
4	(5°, 2.0, 2.0)
5	(0°, 1.0, 1.0)

TABLE 4. USEFUL EIGHT-BAR LINKAGE SOLUTION

<i>Pivot</i>	<i>Location Data (x,y)</i>
C_1	(-6,0)
C_2	(8.2,8)
C_3	(-8,6.1)
C_4	(-4,6.1)
C_5	(11.5,15.3)
C_6	(2.0)
C_7	(11.781, 11.185)
C_8	(7.799, 10.478)
C_9	(8.415, 8.56)
C_{10}	(12.616, 12.723)

CONCLUSION

This paper describes a methodology for the design of useful (defect-free), single degree-of-freedom eight-bar linkages, for five finitely separated task position or poses. The input to the design procedure is a set of five task positions, tolerances acceptable for the task variations, and the 6R loop that is capable of guiding the end-effector through the five task positions.

TABLE 5. STATISTICS FOR THE CONVERTIBLE SOFA-BED LINKAGE DESIGN PROBLEM

<i>Iterations</i>	<i>No. of Linkages Synthesized</i>	<i>No. of Useful Linkages</i>	<i>No. of Useful Linkages satisfying Criteria</i>	<i>Time (min)</i>	<i>Notes</i> S: (Serial) P: (Parallel) (Cores/Threads)
1	61	2	1	9.29	S, Intel (4/8)
1	61	2	1	2.75	P, Intel (4/8)
1	61	2	1	2.32	P, AMD (6)
10	559	3	2	19.231	P, AMD (6)
1	61	2	1	1.035	P, Intel (12/24)
10	469	4	2	6.949	P, Intel (12/24)
100	4191	9	4	62.023	P, Intel (12/24)

The synthesis procedure uses two strategies to apply the two RR constraints. Strategy I (independent constraints) picks any two link pairs of the 6R loop and applies an RR constraint between each pair. There are 17 ways to do this. Strategy II (dependent constraints) applies the two RR constraints in sequence. The first RR constraint is applied between any allowable link pair of the 6R loop. Then the second RR constraint is applied between, the newly generated link due to the first RR constraint, and any of the available links of the 6R loop. There are 15 ways to do this. Overall, there are 32 different ways in which the two RR constraints can be applied. It is important to note that even though the ground link, a_0 , is part of the 6R loop, it is excluded from the synthesis process. This constraining procedure allows the designer greater control on the form and size of the synthesized linkage.

Next, Dixon determinant approach is used to analyze the synthesized linkages for generating trajectories for the possible linkage configurations, when the input link is rotated within the prescribed range. These trajectories are then sorted into branches to check for defects. A linkage is deemed defect-free if the end-effector moves smoothly through the five task positions, when the input link is rotated within the range. If the defect-free linkages also satisfy the minimum and maximum allowable link length requirement, specified by the user, they are termed as useful. As an example, the design of an convertible sofa-bed linkage is presented, to discuss the design methodology.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 1066082.

REFERENCES

- [1] Soh, G. S., and McCarthy, J. M., "Synthesis of Eight-Bar Linkages as Mechanically Constrained Parallel Robots.", *12th IFToMM world congress A*, Vol. 653, 2007.
- [2] Wunderlich, W., "Hahere Koppelkurven." *Osterreichisches Ingenieur Archiv*, XVII(3), p.162-165, 1963.
- [3] Primrose, E. J. F., Freudenstein, F. and Roth, B., "Six-Bar Motion I. The Watt Mechanism.", *Archive for Rational Mechanics and Analysis*, 24.1, p.22-41, 1967.
- [4] Primrose, E. J. F., Freudenstein, F. and Roth, B., "Six-Bar Motion II. The Stephenson-1 and Stephenson-2 Mechanisms.", *Archive for Rational Mechanics and Analysis*, 24.1, p.42-72, 1967.
- [5] Primrose, E. J. F., Freudenstein, F. and Roth, B., "Six-Bar Motion III. Extension of the Six-Bar Techniques to Eight-Bar and 2n-Bar Mechanisms.", *Archive for Rational Mechanics and Analysis*, 24.1, p.73-77, 1967.
- [6] Hamid, S., and Soni A. H., "Synthesis of an Eight-Link Mechanism for Varieties of Motion Programs." *Journal of Engineering for Industry*, 95, no. 3, pp. 744-750, 1973.
- [7] Subbian, T., and Flugrad, D. R., "6 and 7 Position Triad Synthesis using Continuation Methods," *Journal of Mechanical Design*, 116(2):660-665, 1994.
- [8] Chen, C., Angeles, J., and Univerisity, M., "Kinematic synthesis of an eight-bar linkage to visit eleven poses exactly.", *In Proc. CDEN/C2E2 2007 Conference*, 2007.
- [9] Chen, C., and Angeles, J., "A novel family of linkages for advanced motion synthesis.", *Mechanism and Machine Theory*, 43(7), 882-890, 2008.
- [10] McCarthy, J. M. and Soh, G. S., *Geometric Design of Linkages. 2nd Ed.*, Springer-Verlag, 2010.
- [11] Soh, G. S., and Ying, F., "Dimensional Synthesis of Planar Eight-Bar Linkages based on a Parallel Robot with a Prismatic Base Joint.", *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Paper No. DETC2013-12799, August 4-7, 2013, Portland, Oregon, USA.
- [12] Wampler C. W., "Solving the Kinematics of Planar Mechanisms by Dixon Determinant and a Complex Plane Formulation", *ASME Journal of Mechanical Design*, 123(3), pp. 382-387, 2001.
- [13] Nielsen, J., and Roth, B., "Solving the input/output problem for planar mechanisms.", *Journal of Mechanical Design*, 121, 206, 1999.
- [14] Dhingra A. K., Almadi A. N. and D. Kohli D., "Closed-form displacement analysis of 8, 9, and 10-link mechanisms, Part I: 8-link 1-DOF mechanisms," *Mechanism and Machine Theory*, 35:821-850, 2000.
- [15] Parrish, B. E. and McCarthy, J. M., "Use of the Jacobian to Verify Smooth Movement in Watt I and Stephenson I Six-Bar Linkages.", *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DET2013-13454, August 4-7, 2013, Portland, Oregon, USA.
- [16] Parrish, B. E., McCarthy, J. M., and Eppstein, D., "Automated Generation of Linkage Loop Equations for Planar 1-DOF Linkages, Demonstrated up to 8-bar.", *Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Paper No. DETC2014-35263, August 17-20, 2014, Buffalo, New York, USA.
- [17] Plecnik, M. M. and McCarthy, J. M., "Design of a 5-SS Spatial Steering Linkage," DETC 2012-71405, *Proceedings of the ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conferences*, August 12-15, 2012, Chicago, IL, USA.
- [18] Sonawale, K. H., Arredondo, A., and McCarthy, J. M., "Computer Aided Design of Useful Spherical Watt I Six-bar Linkages," *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DET2013-13454, August 4-7, 2013, Portland, Oregon USA.
- [19] Tsai L. W., *Enumeration of Kinematic Structures According to Function*, CRC Press, 2001.
- [20] Plecnik, M. M. and McCarthy, J. M., "Numerical Synthesis of Six-bar Linkages for Mechanical Computation.", *Journal of Mechanisms and Robotics*, 2013.