INFORMATION TO USERS

This reproduction was made from a copy of a document sent to us for microfilming. While the most advanced technology has been used to photograph and reproduce this document, the quality of the reproduction is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help clarify markings or notations which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure complete continuity.

2. When an image on the film is obliterated with a round black mark, it is an indication of either blurred copy because of movement during exposure, duplicate copy, or copyrighted materials that should not have been filmed. For blurred pages, a good image of the page can be found in the adjacent frame. If copyrighted materials were deleted, a target note will appear listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed, a definite method of "sectioning" the material has been followed. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.

4. For illustrations that cannot be satisfactorily reproduced by xerographic means, photographic prints can be purchased at additional cost and inserted into your xerographic copy. These prints are available upon request from the Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases the best available copy has been filmed.
Kinematic and dynamic analyses of cascades of planar four-bar mechanisms

Tsai, Der-Liang, M.S.
The University of Arizona, 1988
PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark √.

1. Glossy photographs or pages ______
2. Colored illustrations, paper or print ______
3. Photographs with dark background ______
4. Illustrations are poor copy ______
5. Pages with black marks, not original copy ______
6. Print shows through as there is text on both sides of page ______
7. Indistinct, broken or small print on several pages ______
8. Print exceeds margin requirements ______
9. Tightly bound copy with print lost in spine ______
10. Computer printout pages with indistinct print ______
11. Page(s) _______ lacking when material received, and not available from school or author.
12. Page(s) _______ seem to be missing in numbering only as text follows.
13. Two pages numbered ______. Text follows.
14. Curling and wrinkled pages ______
15. Dissertation contains pages with print at a slant, filmed as received √
16. Other __________________________________________

________________________________________

________________________________________

UMI
KINEMATIC AND DYNAMIC ANALYSES OF
CASCADES OF PLANAR FOUR-BAR MECHANISMS

by
Der-Liang Tsai

A Thesis Submitted to the Faculty of the
DEPARTMENT OF AEROSPACE AND MECHANICAL ENGINEERING
In Partial Fulfillment of the Requirements
For the Degree of
MASTER OF SCIENCE
WITH A MAJOR IN MECHANICAL ENGINEERING
In the Graduate College
THE UNIVERSITY OF ARIZONA

1988
STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Derliang Isai

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

Parviz E. Nikravesh
Professor of Aerospace and Mechanical Engineering

Date 1/26/88
ACKNOWLEDGMENT

I wish to express my deep appreciation to Professor Parviz E. Nikravesh, my thesis advisor, for his invaluable guidance and constructive criticism during the course of this research.

Sincere thanks are extended to Professor Ara Arabyan of the University of Arizona and Professor Cho-Seng Lee of the University of Malaya in Malaysia, for their careful review and invaluable comments on this thesis.

Special thanks go to Mr. Tung-Kiung Ling, Mr. Gwang-Hun Gim and Mr. Scott Speece for their kind help and patience in the progress of this thesis.

Sincere thanks are also extended to Ms. Linda Harper for her diligence in typing the manuscript, and to many others for their kindness, assistance and encouragement throughout the writing of this thesis.

I am deeply indebted to my parents for financially supporting this work.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LIST OF ILLUSTRATIONS</strong></td>
<td>6</td>
</tr>
<tr>
<td><strong>LIST OF TABLES</strong></td>
<td>7</td>
</tr>
<tr>
<td><strong>ABSTRACT</strong></td>
<td>8</td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>9</td>
</tr>
<tr>
<td>2. KINEMATIC ANALYSIS OF THE TRANSMISSION ELEMENTS</td>
<td>14</td>
</tr>
<tr>
<td>2.1 Kinematic Analysis Using Explicit Equations</td>
<td>14</td>
</tr>
<tr>
<td>2.1.1 Position Angle</td>
<td>15</td>
</tr>
<tr>
<td>2.1.2 Constraint Equations</td>
<td>19</td>
</tr>
<tr>
<td>2.1.3 Angular Velocity and Acceleration</td>
<td>22</td>
</tr>
<tr>
<td>2.2 Kinematic Analysis Using Simultaneous Algebraic Equations</td>
<td>26</td>
</tr>
<tr>
<td>2.2.1 Newton-Raphson Method</td>
<td>27</td>
</tr>
<tr>
<td>2.2.2 Position Angle</td>
<td>27</td>
</tr>
<tr>
<td>2.2.3 Angular Velocity and Acceleration</td>
<td>28</td>
</tr>
<tr>
<td>2.3 Multiple Elements Connected to Side Links</td>
<td>30</td>
</tr>
<tr>
<td>2.3.1 Transmission from Left to Right</td>
<td>30</td>
</tr>
<tr>
<td>2.3.2 Transmission from Right to Left</td>
<td>32</td>
</tr>
<tr>
<td>2.4 Moving Frame</td>
<td>36</td>
</tr>
<tr>
<td>3. DYNAMIC ANALYSIS OF THE TRANSMISSION ELEMENTS</td>
<td>41</td>
</tr>
<tr>
<td>3.1 Lagrange's Equations</td>
<td>42</td>
</tr>
<tr>
<td>3.2 Lagrange Multipliers</td>
<td>45</td>
</tr>
<tr>
<td>3.3 Dynamic Analysis Using Explicit Equations</td>
<td>48</td>
</tr>
<tr>
<td>3.3.1 The Generalized Inertia</td>
<td>48</td>
</tr>
<tr>
<td>3.3.2 The Partial Derivative of the Generalized Inertia</td>
<td>52</td>
</tr>
<tr>
<td>3.3.3 The Generalized Force</td>
<td>54</td>
</tr>
<tr>
<td>3.3.4 Transmission from Side to Side</td>
<td>55</td>
</tr>
<tr>
<td>3.4 Dynamic Analysis Using Simultaneous Equations</td>
<td>59</td>
</tr>
<tr>
<td>3.4.1 Equations of Motion</td>
<td>59</td>
</tr>
<tr>
<td>3.4.2 Constraint Forces</td>
<td>61</td>
</tr>
<tr>
<td>3.5 Moving Frame</td>
<td>64</td>
</tr>
<tr>
<td>3.5.1 Equations of Motion</td>
<td>64</td>
</tr>
<tr>
<td>3.5.2 Terms Appearing in Equations of Motion</td>
<td>68</td>
</tr>
<tr>
<td>3.5.3 Two Elements with a Moving Frame</td>
<td>78</td>
</tr>
</tbody>
</table>
**TABLE OF CONTENTS—continued**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4. APPLICATIONS AND PROGRAMMING ASPECTS</td>
<td>83</td>
</tr>
<tr>
<td>4.1 Introduction to Kinematic and Dynamic Computer Programs</td>
<td>84</td>
</tr>
<tr>
<td>4.1.1 Program Organization of Kinematic Analysis</td>
<td>85</td>
</tr>
<tr>
<td>4.1.2 Program Organization of Dynamic Analysis</td>
<td>90</td>
</tr>
<tr>
<td>4.2 Example Problem</td>
<td>98</td>
</tr>
<tr>
<td>5. CONCLUSIONS AND DISCUSSION</td>
<td>106</td>
</tr>
<tr>
<td>5.1 Computational Efficiency</td>
<td>106</td>
</tr>
<tr>
<td>5.2 Computational Accuracy</td>
<td>107</td>
</tr>
<tr>
<td>5.3 Future Development</td>
<td>111</td>
</tr>
<tr>
<td>APPENDIX A: LISTING OF KINEMATIC ANALYSIS PROGRAM</td>
<td>113</td>
</tr>
<tr>
<td>APPENDIX B: LISTING OF DYNAMIC ANALYSIS PROGRAM</td>
<td>128</td>
</tr>
<tr>
<td>NOTATION</td>
<td>165</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>167</td>
</tr>
</tbody>
</table>
## LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1.1</td>
<td>Four-bar linkage</td>
<td>16</td>
</tr>
<tr>
<td>2.2.1</td>
<td>A four-bar mechanism with relative and absolute coordinates</td>
<td>20</td>
</tr>
<tr>
<td>2.3.1</td>
<td>A transmission from left to right</td>
<td>31</td>
</tr>
<tr>
<td>2.3.2</td>
<td>A transmission from right to left</td>
<td>33</td>
</tr>
<tr>
<td>2.3.3</td>
<td>The image system</td>
<td>35</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Four-bar mechanism with a moving frame</td>
<td>37</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Four-bar mechanism with the inertial properties</td>
<td>49</td>
</tr>
<tr>
<td>3.3.1</td>
<td>A transmission from side to side</td>
<td>57</td>
</tr>
<tr>
<td>3.5.1</td>
<td>A floating four-bar mechanism with the inertial properties</td>
<td>65</td>
</tr>
<tr>
<td>3.5.2</td>
<td>A moving frame connected to an element with fixed frame</td>
<td>79</td>
</tr>
<tr>
<td>4.1.1</td>
<td>The flow chart of kinematic analysis program</td>
<td>86</td>
</tr>
<tr>
<td>4.1.2</td>
<td>The flow chart of dynamic analysis program--main program</td>
<td>92</td>
</tr>
<tr>
<td>4.1.3</td>
<td>The flow chart of dynamic analysis program--subroutines RUNGK4, DIFEQN and MATRIX</td>
<td>93</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Unfolded solar panels of satellite</td>
<td>99</td>
</tr>
<tr>
<td>4.2.2</td>
<td>The process of deploying solar panels in orbit</td>
<td>100</td>
</tr>
<tr>
<td>4.2.3</td>
<td>The first partial motion of solar panels</td>
<td>101</td>
</tr>
<tr>
<td>4.2.4</td>
<td>The second partial motion of solar panels</td>
<td>104</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.1</td>
<td>Input Data of the First Partial Motion of Solar Panels</td>
<td>102</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Input Data of the Second Partial Motion of Solar Panels</td>
<td>105</td>
</tr>
<tr>
<td>5.1.1</td>
<td>The Sizes of the System Equations of Motion by Using Relative Coordinates</td>
<td>108</td>
</tr>
<tr>
<td></td>
<td>(Method 1) and Absolute Coordinates (Method 2)</td>
<td></td>
</tr>
<tr>
<td>5.1.2</td>
<td>The Computation Times of Programs by Using Relative and Absolute Coordinates</td>
<td>109</td>
</tr>
</tbody>
</table>
ABSTRACT

Computer programs have been developed for the kinematic and dynamic analyses of cascades of planar four-bar mechanisms. Since the analytic approach has high efficiency and accuracy in computation, a chain of four-bar linkages is developed horizontally and vertically by using the relative coordinates and the absolute coordinates, from which the explicit equations and the simultaneous equations are respectively derived in the kinematic analysis. In this analysis, the actions of transmission of linkages, from left to right, from right to left and from lower to upper, are performed by the image method and transformation procedures.

Based on the kinematic analysis, the dynamic analysis is also developed by using both sets of coordinate systems. The generalized equation of motion, the general form of Lagrange’s equations, Lagrange multipliers and the theorem of power balance are used to construct various formulations of the governing equations of motion for some particular problems. The problem of a linkage with a moving frame (the ground link) is the most interesting focus in this analysis.
CHAPTER 1

INTRODUCTION

A planar four-bar linkage is a closed loop mechanism with a single degree of freedom when the frame (the ground link) is fixed. Otherwise, a moving frame has four degrees of freedom. By using one four-bar mechanism to drive one or more others, cascades of planar four-bar mechanisms can be used to perform rather complex tasks, such as the unfolding of solar panels in an orbiting satellite and the motion of the wheels of a steam locomotive. To analyze the actions of transmission from one linkage (element) to the others, we require both the kinematic and dynamic analyses which involve complex and numerical computations. The former is the necessary prelude to the latter. Both analyses are developed by constructing mathematical models to describe a particular mechanical system under consideration. Each of these analyses can be performed by using two different approaches, namely, the vectorial (Newtonian) approach and the analytical (variational) approach.

From observation and computational experience, the efficiency and accuracy of a computer program depends not only on the choice of coordinates, but also on the methods of numerical solution. The choice of coordinates, which describes the configuration of the system, directly influences the number and the order of the nonlinearity of the equations of motion. In accordance with the form of these equations, a method of numerical solution is selected, based on its high efficiency and accuracy for solving the set of equations under consideration. Recently, with the advent of digital computers, some general-purpose programs, using vectorial approach in terms of Cartesian coordinates, have been presented in the recent work of Nikravesh [1]. For a planar four-bar linkage, the
formulation of this approach involves nine Cartesian coordinates in eight nonlinear algebraic equations, two equations per revolute joint.

For any problem, one of the main factors that influence the amount of numerical error in the problem solution is the number of equations. In general, the larger the number of equations, the higher the chance for the numerical error to accumulate faster. Therefore, for cascades of four-bar mechanisms, the analytic approach, which uses relative coordinates or absolute coordinates, may improve the efficiency and accuracy a great deal, owing to the fact that it involves a minimum number of coordinates. Here, the analytic approach treats the system as a whole in contrast with the vectorial approach, which deals with individual parts of the mechanical system. The relative coordinates correspond to the internal angles lying between two links of a linkage, while the absolute coordinates are measured with respect to the frame (the ground link) of a linkage. Both coordinate systems can be transferred into the global coordinate system for general usages.

In Chapter 2, the kinematic analysis of the transmission elements are developed by using both the explicit and the simultaneous equations. Based on the analytic kinematics, Shigley and Uicker [2] have solved a four-bar linkage problem by using the loop-closure equation, in which the position angles are first expressed in terms of absolute coordinate in a closed form. By means of a transformation, the explicit equations involving the relative coordinates can be obtained. In order to find the angular velocities and accelerations, four holonomic constraint equations are derived in terms of the relative coordinates and time. The first and the second time derivatives of the constraint equations yield the velocity and acceleration equations. Then, by solving these equations, the angular velocities and accelerations can be expressed in a form involving one generalized coordinate.
In another approach, which uses three absolute coordinates, a set of two nonlinear simultaneous algebraic equations are derived from the configuration of a four-bar mechanism. Given the input angle, the other two position angles can readily be obtained by a numerical iteration procedure on the set of simultaneous equations. In the same manner, given the data of the driving link, the first and the second time derivatives of the constraint equations which come from the loop-closure equation yield the angular velocities and accelerations.

For cascades of four-bar mechanisms, there are three types of actions of transmission. The transmission from left to right is performed by taking the output link of the driving linkage as the input link of the driven linkage. Next, the transmission from right to left is performed by means of the image method. In view of the fact that in the image system, the action of transmission is from left to right, the real system can easily be obtained from the image system by a transformation procedure. Finally, the transmission from lower linkage to upper linkage, which has a moving frame, is by means of the transformation of the local coordinates to global coordinates. By specifying the local coordinates of the moving frame, we can easily obtain the global coordinates. Meanwhile, the angular velocity and acceleration of a moving frame are the same as those of the coupler link of the lower linkage.

Chapter 3 is the main part of this research. In it, the analytic dynamics of a mechanism is developed from its analytic kinematics. The analytic dynamics involves two scalar functions, namely the kinetic energy function and the potential energy function. Based on the theorem of power balance, Paul [3] developed the generalized equation of motion, and then recast it into the general form of Lagrange's equations. For a closed-loop system, the use of relative coordinates will result in one generalized equation of
motion with one generalized coordinate. The form of this generalized equation of motion is more suitable for numerical integration. On the other hand, for a linkage with fixed frame, the general form of Lagrange's equations involves three absolute coordinates. So, in this system, the number of Lagrangian coordinates will be greater than the number of degrees of freedom. As a result, the constraint forces in terms of Lagrange multipliers must appear in the equations of motion. Meanwhile, for a linkage with moving frame, the general form of Lagrange's equations can also be written in terms of a relative coordinate and three Cartesian coordinates (i.e., four independent coordinates). By applying the Lagrange multiplier technique, which was stated by Nikravesh and Greenwood [1,4], two or more linkages can be put together horizontally and vertically to form cascades of four-bar mechanisms. The generalized forces appearing in the governing equations of motion are computed in accordance with the theorem of power balance with respect to the independent generalized coordinates of a system [5]. Finally, we will obtain various forms of system equations of motion for different mechanical systems as well as for different coordinate systems.

The applications and computer programs will be discussed in Chapter 4. The kinematic analysis program consists of a main program and six subroutines. Based on this program, the dynamic analysis program is developed with additional fourteen subroutines. The flow charts and the organizations of input files for both programs will be described in detail; then an excellent example on an orbiting satellite in the process of deploying solar panels will be given.

In the last chapter, the computational efficiency and accuracy of programs based on two kinds of coordinate systems will be discussed in several aspects. It is observed that the approach of using the relative coordinates not only reduces the computation time but also
increases the accuracy. On the other hand, the approach of using the absolute coordinates is more time-consuming in computation and may fail to converge. To make these programs more versatile and more efficient, some recommendations for future development will be discussed.
CHAPTER 2

KINEMATIC ANALYSIS OF THE TRANSMISSION ELEMENTS

Kinematics is the geometry of motion and, in particular, is the study of position, velocity and acceleration of a system regardless of the forces that cause the motion. Hence, kinematic analysis is the process of analyzing the position, velocity and acceleration of a specified mechanism. The kinematic analysis contains an algebraic or analytic mode which enables us to perform digital calculation procedures. In what follows, we will find that the procedure of kinematic analysis is the natural and necessary prelude to dynamic analysis -- the subject of the next chapter.

A typical and one of the widely used planar mechanisms is a four-bar linkage. This planar mechanism is the simplest closed kinematic chain of hinged links with a single degree of freedom. It is interesting to note that the study of a four-bar linkage is a classic problem with solution dating back to over a century ago [2]. Although this solution can be performed by using pencil and paper, a computational method is certainly more useful when the mechanism becomes more complex. With the application of numerical techniques, two different approaches will be developed, namely, the one employing explicit equations and the other employing simultaneous algebraic expressions. Both approaches will supply the necessary solutions for the dynamic analysis. In the subsequent sections, we will discuss both approaches, starting from one element with a fixed frame to several elements with moving frames.

2.1 Kinematic Analysis Using Explicit Equations

In this section, we will illustrate the kinematic analysis by using the explicit equations in terms of relative angles (i.e., the relative coordinates), which can be found
from the absolute coordinates. The comparison between the relative and absolute coordinates, with regard to several aspects, is illustrated in [1].

2.1.1 Position Angle

Figure 2.1.1 shows a typical four-bar linkage with frame $r_1$, coupler $r_3$, and side links $r_2$ and $r_4$; $AB$ is the input link and $CD$ is the output link. Using the notations and loop-closure equation [2], we have the complex polar form:

$$r_2 e^{i\theta_2} + r_3 e^{i\theta_3} = r_1 + r_4 e^{i\theta_4}$$

(2.1)

where the given angle $\theta_2$ and two unknown angles $\theta_3$ and $\theta_4$ are the absolute coordinates.

Using Euler's formula, we separate Eq. (2.1) into the real and imaginary terms:

$$r_2 \cos \theta_2 + r_3 \cos \theta_3 = r_1 + r_4 \cos \theta_4$$

(2.2a)

$$r_2 \sin \theta_2 + r_3 \sin \theta_3 = r_4 \sin \theta_4$$

(2.2b)

Rearranging the terms, Eqs. (2.2) can be put in the form:

$$r_2 \cos \theta_3 = r_1 - r_2 \cos \theta_2 + r_4 \cos \theta_4$$

(2.3a)

$$r_2 \sin \theta_3 = -r_2 \sin \theta_2 + r_4 \sin \theta_4$$

(2.3b)

and therefore,

$$r_3^2 = r_1^2 + r_2^2 + r_4^2 - 2r_2 r_3 \cos \theta_2 + 2r_4 r_3 \cos \theta_4 - 2r_2 r_4 \cos (\theta_4 - \theta_2).$$

(2.4a)

Using the same procedure, we can get another form of expression when angle $\theta_4$ is eliminated:
Figure 2.1.1 Four-bar linkage.
According to the law of cosines, the diagonal distance $d$ can be expressed in terms of the link lengths and angles, as that shown in Figure 2.1.1. This yields two different expressions for $d$:

\begin{align*}
d^2 &= r_1^2 + r_2^2 - 2r_1r_2 \cos \theta_2 - 2r_1r_3 \cos \theta_3 + 2r_2r_3 \cos(\theta_2 - \theta_3). \\
(2.5a) \\
\end{align*}

\begin{align*}
d^2 &= r_3^2 + r_4^2 - 2r_3r_4 \cos \gamma \\
(2.5b) \\
\end{align*}

By substituting Eq. (2.5a) into Eq. (2.5b), we obtain the transmission angle $\gamma$:

\begin{equation}
\gamma = \cos^{-1}\left(\frac{r_3^2 + r_4^2 - r_1^2 - r_2^2 + 2r_1r_2 \cos \theta_2}{2r_3r_4}\right). \\
(2.6) \\
\end{equation}

Dividing Eq. (2.4a) throughout by $2r_3r_4$ and rearranging the terms, we obtain:

\begin{align*}
\frac{r_3^2 + r_4^2 - r_1^2 - r_2^2 + 2r_1r_2 \cos \theta_2}{2r_3r_4} &= \frac{(r_1 - r_2 \cos \theta_2) \cos \theta_4}{r_3} \\
&+ \frac{(r_3 \sin \theta_3) \sin \theta_4 - r_4}{r_3} = 0 \\
(2.7) \\
\end{align*}

where the first term represents the transmission angle $\gamma$. Substituting Eq. (2.6) into Eq. (2.7) and using the half-angle identities of trigonometry, we arrive at

\begin{align*}
(r_1 - r_2 \cos \theta_2) \frac{1 - \tan^2 \frac{\theta_4}{2}}{1 + \tan^2 \frac{\theta_4}{2}} - r_2 \sin \theta_2 \frac{2 \tan \frac{\theta_4}{2}}{1 + \tan^2 \frac{\theta_4}{2}} - r_3 \cos \gamma + r_4 &= 0 \\
(2.8) \\
\end{align*}
Rearranging the terms, Eq. (2.8) reduces to the following quadratic equation:

\[
(-r_1 + r_2 \cos \theta_2 - r_3 \cos \gamma + r_4) \tan^2 \frac{\theta_4}{2} - 2r_2 \sin \theta_2 \tan \frac{\theta_4}{2} \\
+ (r_1 - r_2 \cos \theta_2 - r_3 \cos \gamma + r_4) = 0
\]  
(2.9)

which has two solutions:

\[
\tan \frac{\theta_4}{2} = \frac{r_2 \sin \theta_2 + r_3 \sin \gamma}{r_4 - r_1 + r_2 \cos \theta_2 - r_3 \cos \gamma},
\]  
(2.10)

and so,

\[
\theta_4 = 2 \tan^{-1} \frac{r_2 \sin \theta_2 + r_3 \sin \gamma}{r_4 - r_1 + r_2 \cos \theta_2 - r_3 \cos \gamma}.
\]  
(2.11)

Thus, the output angle \( \theta_4 \) is expressed in terms of the input angle \( \theta_2 \) and the transmission angle \( \gamma \) is given by Eq. (2.6).

Similarly, by repeating the same procedures and using Eq. (2.4b), the unknown angle \( \theta_3 \) is found to be:

\[
\theta_3 = 2 \tan^{-1} \frac{-r_2 \sin \theta_2 \pm r_4 \sin \gamma}{r_1 + r_3 - r_2 \cos \theta_2 - r_4 \cos \gamma}.
\]  
(2.12)

The negative sign in Eq. (2.11) and the positive sign in Eq. (2.12) represent \( \theta_4 \) and \( \theta_3 \) (see Fig. 2.1.1); and the other signs represent \( \theta'_4 \) and \( \theta'_3 \), respectively. Therefore, we refer to the configuration ABCD as the leading form and to ABC'D as the lagging form of the linkage [3].
Next, we will transfer the absolute coordinates $\theta_2$, $\theta_3$ and $\theta_4$ into relative coordinates $q_1$, $q_2$, $q_3$, and $q_4$. Fig. 2.2.1 illustrates the transformation of different sets of coordinate systems. The angles $\theta_2$, $\theta_3$, $\gamma$ and $\theta_4$ are arranged in clockwise sense, but angles $q_1$, $q_2$, $q_3$ and $q_4$ are in counterclockwise sense. Therefore, we obtain the following transformations:

\begin{align*}
q_1 &= \theta_2, \\
q_2 &= \pi - \theta_4, \\
q_3 &= \gamma, \\
q_4 &= q_1 - \theta_3 = \pi, \\
\text{or} \quad q_4 &= \pi - \theta_2 + \theta_3.
\end{align*}

where $\theta_2$ is given, and $\gamma$, $\theta_3$ and $\theta_4$ are calculated from Eqs. (2.6), (2.12) and (2.11), respectively. Since $q_1$, which describes the orientation of the input link with respect to the ground, is the selected coordinate, other relative angles can be calculated from Eqs. (2.13) in terms of angle $q_1$. Therefore, Eqs. (2.13) are called the explicit equations with one generalized coordinate; i.e., angle $q_1$.

2.1.2 Constraint Equations

Having solved the position angles, we will seek the expressions for angular velocities and accelerations in terms of the relative angles. The first step is to derive four holonomic constraint equations, $\Phi = 0$, from which the first and second time derivatives yield the angular velocity and acceleration equations.

The first constraint equation is in the form of a moving constraint equation of the input link (i.e., the crank):
Figure 2.2.1 A four-bar mechanism with relative and absolute coordinates.
\[ \Phi_1 = q_1 - \theta_2 - \omega t - \frac{1}{2} \alpha t^2 = 0 \]  

(2.14)

where

- \( q_1 \): the driving angle,
- \( \theta_2 \): the initial angle,
- \( \omega \): the angular velocity,
- \( \alpha \): the angular acceleration,
- \( t \): the time increment.

The second constraint equation is obtained from the loop-closure equation. It is the same as that shown in Eq. (2.4a), but we need to express this equation in terms of the relative angles. From Eqs. (2.13a) and (2.13b), we have:

\[ \theta_2 = q_1 \]  

(2.15a)

\[ \theta_4 = \pi - q_2 \]  

(2.15b)

Thus, we get two trigonometric identities:

\[ \cos(\theta_4 - \theta_2) = - \cos(q_1 + q_2) \]  

(2.16a)

\[ \cos \theta_4 = - \cos q_2 \]  

(2.16b)

Substituting Eqs. (2.16) into Eq. (2.4a), we obtain the second constraint equation:

\[ \Phi_2 = r_1^2 + r_2^2 + r_4^2 - r_3^2 - 2r_1r_2 \cos \theta_1 - 2r_3r_4 \cos \theta_2 + 2r_2r_4 \cos(q_1 + q_2) = 0 \]  

(2.17)

For the third constraint equation we combine Eqs. (2.5a) and (2.5b) to obtain the identity:
\[ r_1^2 + r_2^2 - 2r_1r_2 \cos \theta_2 = r_3^2 + r_4^2 - 2r_3r_4 \cos \gamma. \quad (2.18a) \]

Since angles \( \theta_2 \) and \( \gamma \) are equal to \( q_1 \) and \( q_3 \) respectively, Eq. (2.18a) becomes:

\[ \Phi_3 \equiv r_1^2 + r_2^2 - r_3^2 - r_4^2 - 2r_3r_4 \cos q_4 - 2r_3r_4 \cos q_3 = 0 \quad (2.18b) \]

Finally, from Eqs. (2.13d) and (2.13e), we have

\[ \theta_3 = q_1 + q_4 - \pi \quad (2.19a) \]
\[ \theta_2 - \theta_3 = \pi - q_4 \quad (2.19b) \]

Thus, we get

\[ \cos \theta_3 = -\cos(q_1 + q_4) \quad (2.20a) \]
\[ \cos(\theta_2 - \theta_3) = -\cos q_4. \quad (2.20b) \]

In the same manner, the fourth constraint equation is obtained by substituting Eqs. (2.20) into Eq. (2.4b):

\[ \Phi_4 = r_1^2 + r_2^2 + r_3^2 - r_4^2 + 2r_4r_3 \cos(q_1 + q_4) - 2r_4r_3 \cos q_1 - 2r_3r_4 \cos q_4 = 0 \quad (2.21) \]

2.1.3 Angular Velocity and Acceleration

The holonomic constraint equations, \( \Phi = 0 \), have been derived in terms of the relative coordinates. Hence, kinematic analysis requires the solution of the following sets of equations [1]:

\[ \Phi_\theta \dot{q} = \nu \quad (2.22a) \]
\[ \Phi_\gamma \ddot{q} = \gamma \quad (2.22b) \]

where
\[ \mathbf{q} = [q_1, q_2, q_3, q_4]^T \]

\[ \nu = -\Phi_t \]

\[ \gamma = - (\Phi_\mathbf{q} \mathbf{q}) \dot{\mathbf{q}} - 2 \Phi_\mathbf{q t} \dot{\mathbf{q}} - \Phi_\mathbf{tt} . \]

Both the velocity and acceleration equations contain the same Jacobian matrix $\Phi_\mathbf{q}$, therefore, in order to obtain the angular velocities and accelerations, we need to determine $\Phi_\mathbf{q}$, $\nu$ and $\gamma$.

From Eqs. (2.14), (2.17), (2.18b) and (2.21), the first time derivatives of the four constraint equations are derived with respect to $q_i$ for $i = 1, ..., 4$:

\[
\frac{\partial \Phi_1}{\partial q_1} = 1 \quad (2.23a)
\]

\[
\frac{\partial \Phi_1}{\partial q_2} = \frac{\partial \Phi_1}{\partial q_3} = \frac{\partial \Phi_1}{\partial q_4} = 0 \quad (2.23b)
\]

\[
\frac{\partial \Phi_2}{\partial q_1} = 2r_1r_2 \sin q_1 - 2r_3r_4 \sin(q_1+q_2) \quad (2.24a)
\]

\[
\frac{\partial \Phi_2}{\partial q_2} = 2r_1r_4 \sin q_2 - 2r_3r_4 \sin(q_1+q_2) \quad (2.24b)
\]

\[
\frac{\partial \Phi_2}{\partial q_3} = \frac{\partial \Phi_2}{\partial q_4} = 0 \quad (2.24c)
\]

\[
\frac{\partial \Phi_3}{\partial q_1} = 2r_1r_2 \sin q_1 \quad (2.25a)
\]

\[
\frac{\partial \Phi_3}{\partial q_2} = \frac{\partial \Phi_3}{\partial q_4} = 0 \quad (2.25b)
\]

\[
\frac{\partial \Phi_3}{\partial q_3} = -2r_3r_4 \sin q_3 \quad (2.25c)
\]

\[
\frac{\partial \Phi_4}{\partial q_1} = -2r_1r_3 \sin(q_1+q_4) + 2r_1r_2 \sin q_1 \quad (2.26a)
\]
\[
\frac{\partial \Phi_4}{\partial q_2} = \frac{\partial \Phi_4}{\partial q_3} = 0 \quad (2.26b)
\]
\[
\frac{\partial \Phi_4}{\partial q_4} = -2r_1r_3 \sin(q_1+q_3) + 2r_3r_4 \sin q_4 \quad (2.26c)
\]
\[
\frac{\partial \Phi_1}{\partial t} = \omega + \alpha t \quad (2.27a)
\]
\[
\frac{\partial \Phi_2}{\partial t} = \frac{\partial \Phi_3}{\partial t} = \frac{\partial \Phi_4}{\partial t} = 0 \quad (2.27b)
\]

from which we obtain the Jacobian matrix \( \Phi_q \). Therefore, Eq. (2.22a) is expanded in the form:

\[
\begin{bmatrix}
\frac{\partial \Phi_1}{\partial q_1} & 0 & 0 & 0 \\
\frac{\partial \Phi_2}{\partial q_1} & \frac{\partial \Phi_2}{\partial q_2} & 0 & 0 \\
\frac{\partial \Phi_3}{\partial q_1} & 0 & \frac{\partial \Phi_3}{\partial q_3} & 0 \\
\frac{\partial \Phi_4}{\partial q_1} & 0 & 0 & \frac{\partial \Phi_4}{\partial q_4}
\end{bmatrix}
\begin{bmatrix}
\dot{q}_1 \\
\dot{q}_2 \\
\dot{q}_3 \\
\dot{q}_4
\end{bmatrix}
= \begin{bmatrix}
\omega + \alpha t \\
0 \\
0 \\
0
\end{bmatrix} \quad (2.28)
\]

In view of the fact that the entities on the upper-triangular matrix are zero, we get the solutions by forward-substitution:

\[
\dot{q}_1 = \omega + \alpha t \quad (2.29a)
\]
\[
\dot{q}_2 = \left( \frac{2r_1r_2 \sin q_1 - 2r_3r_4 \sin(q_1+q_3)}{2r_1r_4 \sin q_2 - 2r_3r_4 \sin(q_1+q_2)} \right) \dot{q}_1 \quad (2.29b)
\]
The angular velocities need to be calculated in order to find the angular accelerations from Eq. (2.22b). The angular acceleration equation contains a zero matrix \((i.e., \Phi_{\Omega t})\), and the matrix \(\Phi_{tt}\) becomes:

$$\Phi_{tt} = \begin{bmatrix} \alpha \\ 0 \\ 0 \\ 0 \end{bmatrix}. $$

The Jacobian matrix \(\Phi_q\) is the same as that in the angular velocity equation, and the matrix \((\Phi_q \dot{q})\) is found to be:

$$ (\Phi_q \dot{q}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 \\ a_{31} & 0 & a_{33} & 0 \\ a_{41} & 0 & 0 & a_{44} \end{bmatrix} \quad (2.30) $$

where

\[
\begin{align*}
a_{21} &= 2r_1 r_2 \cos\theta_1 \dot{\theta}_1 - 2r_2 r_4 \cos(\theta_1 + \theta_2) (\dot{\theta}_1 + \dot{\theta}_2) \\
a_{22} &= 2r_1 r_4 \cos\theta_2 \dot{\theta}_2 - 2r_2 r_4 \cos(\theta_1 + \theta_2) (\dot{\theta}_1 + \dot{\theta}_2) \\
a_{31} &= 2r_2 r_2 \cos\theta_1 \dot{\theta}_1 \\
a_{33} &= -2r_3 r_4 \cos\theta_3 \dot{\theta}_3 \\
a_{41} &= 2r_1 r_2 \cos\theta_1 \dot{\theta}_1 - 2r_3 r_4 \cos(\theta_1 + \theta_4) (\dot{\theta}_1 + \dot{\theta}_4)
\end{align*}
\]
Therefore, combining Eqs. (2.23) - (2.30), Eq. (2.22b) can be put in the form:

\[
\begin{bmatrix}
\ddot{q}_1 \\
\ddot{q}_2 \\
\ddot{q}_3 \\
\ddot{q}_4
\end{bmatrix} = -(\Phi q \dot{q})_q \dot{q} + \begin{bmatrix}
\alpha \\
0 \\
0
\end{bmatrix}
\]  

(2.31)

Similarly, after having calculated the angular velocities from Eqs. (2.29), we apply the forward-substitution approach to obtain the unknown angular accelerations \(\ddot{q}_2\), \(\ddot{q}_3\) and \(\ddot{q}_4\).

Equations (2.13), (2.29) and (2.31) represent the kinematic analysis in terms of the relative coordinates. Another approach which is in terms of absolute coordinates will be illustrated in the next section.

### 2.2 Kinematic Analysis Using Simultaneous Algebraic Equations

Figure 2.2.1 illustrates the positions \(\theta_2\), \(\theta_3\) and \(\theta_4\) with respect to the absolute coordinates. Since the four-bar linkage has only one degree-of-freedom, the three coordinates are not independent.

The scalar equations representing components of the vector loop-closure equation are expressed in Eqs. (2.2). If the input angle \(\theta_2\) is given, a set of two nonlinear simultaneous algebraic equations must be solved for angles \(\theta_3\) and \(\theta_4\):  

\[
\Phi_1 = r_2 \sin \theta_2 + r_3 \sin \theta_3 - r_4 \sin \theta_4 = 0 
\]  

(2.32a)

\[
\Phi_2 = r_2 \cos \theta_2 + r_3 \cos \theta_3 - r_4 \cos \theta_4 - r_1 = 0 
\]  

(2.32b)

For each time step the angles \(\theta_3\) and \(\theta_4\) correspond to the angle \(\theta_2\) when the input
position is incremented. Therefore, the Newton-Raphson method is utilized to solve these nonlinear algebraic equations.

22.1 Newton-Raphson Method

The simultaneous algebraic equations are the constraint equations in terms of the absolute coordinates. These equations are nonlinear algebraic equations with two unknowns $\theta_3$ and $\theta_4$. In order to find the unknown angles $\theta_3$ and $\theta_4$, we need to employ a numerical iteration method, such as the Newton-Raphson method, whose algorithm for n-equations is as stated in [1] as:

\[
\begin{align*}
\phi_X(X^j) \Delta X^j & = -\phi(X^j) \\
X^{j+1} & = X^j + \Delta X^j
\end{align*}
\]

(2.33a)

(2.33b)

where the term $\phi_X(X^j)$ is the Jacobian matrix evaluated at $X = X^j$. The term $\phi(X^j)$ is the vector of residuals, which corresponds to the violation in the equations. Thus, Eq. (2.33a) is a set of n linear equations in unknowns $\Delta X^j$. Then, $X^{j+1}$ is obtained from Eq. (2.33b).

22.2 Position Angle

For the constraints of Eqs. (2.32), the Jacobian matrix is:

\[
\Phi_X = \begin{bmatrix}
  r_3 \cos \theta_3 & -r_4 \cos \theta_4 \\
  -r_3 \sin \theta_3 & r_4 \sin \theta_4
\end{bmatrix}
\]

(2.34)

where

\[
X = [\theta_3, \theta_4]^T
\]

In the Newton-Raphson algorithm, $\Delta X$ is evaluated by solving the solution of the linear system:
Since Eq. (2.35) is simple to solve, there is no need to apply a numerical technique (e.g., L-U Factorization). Hence, the solutions are found by the Cramer's rule:

\[
\begin{bmatrix}
  r_3 \cos \theta_3 & -r_4 \cos \theta_4 \\
  -r_3 \sin \theta_3 & r_4 \sin \theta_4 
\end{bmatrix}
\begin{bmatrix}
  \Delta \theta_3 \\
  \Delta \theta_4 
\end{bmatrix}
= \begin{bmatrix}
  -\Phi_1 \\
  -\Phi_2 
\end{bmatrix}
\]

(2.35)

The convergence of the process can be determined by monitoring $\Phi_1$ and $\Phi_2$ as they tend to zero or by monitoring $\Delta \theta_3$ and $\Delta \theta_4$ as they go to zero. Hence, for finding angles $\theta_3$ and $\theta_4$, we need to start with some estimate values which are close to the solutions. If the convergence is less than a specified error tolerance, the process is terminated. For example,

\[\left| \frac{\Delta \theta_i}{\theta_i} \right| \leq 0.001, \text{ for } i = 3, 4.\]

2.2.3 Angular Velocity and Acceleration

The two unknown angular velocities $\dot{\theta}_3$ and $\dot{\theta}_4$ are obtained by differentiating Eqs. (2.2) with respect to time and by rearranging the terms:

\[
r_3 \cos \theta_3 \dot{\theta}_3 - r_4 \cos \theta_4 \dot{\theta}_4 = -r_2 \cos \theta_2 \dot{\theta}_2 \\
-r_3 \sin \theta_3 \dot{\theta}_3 + r_4 \sin \theta_4 \dot{\theta}_4 = r_2 \sin \theta_2 \dot{\theta}_2
\]

(2.37a)

(2.37b)

The solutions of the above linearized set of equations are obtained by the Cramer's and trigonometric addition rules:
\[ \dot{\theta}_3 = \frac{r_3 \sin(\theta_2 - \theta_3)}{r_3 \sin(\theta_4 - \theta_3)} \dot{\theta}_2 \]  
\[ \dot{\theta}_4 = \frac{r_4 \sin(\theta_2 - \theta_4)}{r_4 \sin(\theta_3 - \theta_2)} \dot{\theta}_2 \]  
\[ (2.38a) \]
\[ (2.38b) \]

Similarly, the two unknown angular accelerations \( \ddot{\theta}_3 \) and \( \ddot{\theta}_4 \) are obtained by differentiating Eqs. (2.37) with respect to time, and then use the Cramer's and trigonometric addition rules.

\[ \ddot{\theta}_3 = \frac{A \sin \theta_3 + B \cos \theta_3}{r_3 \sin(\theta_4 - \theta_3)} \]  
\[ (2.39a) \]
\[ \ddot{\theta}_4 = \frac{A \sin \theta_4 + B \cos \theta_4}{r_4 \sin(\theta_4 - \theta_3)} \]  
\[ (2.39b) \]

where

\[ A = r_2 \dot{\theta}_2^2 \sin \theta_2 + r_3 \dot{\theta}_3^2 \sin \theta_3 - r_4 \dot{\theta}_4^2 \sin \theta_4 - r_2 \dot{\theta}_2 \cos \theta_2 \]
\[ B = r_2 \dot{\theta}_2^2 \cos \theta_2 + r_3 \dot{\theta}_3^2 \cos \theta_3 - r_4 \dot{\theta}_4^2 \cos \theta_4 + r_2 \dot{\theta}_2 \sin \theta_2 \]

The angles \( \theta_3 \) and \( \theta_4 \) and angular velocities \( \dot{\theta}_3 \) and \( \dot{\theta}_4 \) are obtained from Eqs. (2.33) and (2.38), respectively. Note that if the estimate angles \( \theta_3 \) and \( \theta_4 \) in Eqs. (2.33) have not been calculated within a specified error criterion, then the angular velocities and accelerations will directly have some error. This is an important difference in comparison with the results of the explicit equation in Section 2.1.
2.3 Multiple Elements Connected to Side Links

In this section, we will expand the four-bar linkage from one element to several multiple branches, where each branch may consist of several elements. Therefore, more complex mechanisms can be built up by using one four-bar mechanism to drive the others. In kinematic analysis, usually the driving angle (i.e., the input angle) is given since a four-bar mechanism has only one degree-of-freedom. It follows that the actions of transmission from one element to the next element depend on the change in the output angle. Therefore, the output angle of the first element creates the driving angle of the second element. In some applications, the input link and the output link can be specified on any side of the element. These different applications will be illustrated by two kinds of transmissions.

2.3.1 Transmission from Left to Right

Figure 2.3.1 illustrates the first type of transmission. The second linkage is connected to the right-hand side of the first linkage. The notations \( r_i^* \) and \( r_i \), for \( i = 1, \ldots, 4 \), are the lengths of links. The slopes of fixed frames, angles \( \phi_i^* \) and \( \phi_i \), are global angles with respect to the \( x \)-axis, while angles \( \theta_i^* \) and \( \theta_i \), for \( i = 2, 3, 4 \), are absolute coordinates with respect to the frames. Finally, angle \( \beta \) is the adjacent angle between both linkages.

From the configuration, if the driving angle of the first linkage, angle \( \theta_2^* \), has been given, then the driving angle of the second linkage is:

\[
\theta_2 = \theta_4^* - \beta + \phi_1^* - \phi_1
\]  

(2.40)

where \( \theta_4^* \) is calculated from Eq. (2.11) or Eqs. (2.33). After the driving angle \( \theta_2 \) has been found, we can use the explicit or simultaneous equations to find the other angles, and the procedure is the same for the next connected elements.
Figure 2.3.1 A transmission from left to right.
In a similar manner, the driving angular velocity and acceleration of the second element are:

\[
\dot{\phi}_2 = \dot{\phi}_4^* + \dot{\phi}_1^* - \dot{\phi}_1 \\
\ddot{\phi}_2 = \ddot{\phi}_4^* + \ddot{\phi}_1^* - \ddot{\phi}_1
\]

(2.41a)

(2.41b)

where \(\dot{\phi}_1^*, \dot{\phi}_1, \dot{\phi}_1^*\) and \(\ddot{\phi}_1^*, \ddot{\phi}_1, \ddot{\phi}_1\) are equal to zero if the frame is fixed to the ground, and \(\dot{\phi}_4^*\) and \(\ddot{\phi}_4^*\) are calculated from Eqs. (2.29d), (2.31), (2.38b) or (2.39b).

Therefore, once the angular velocity and acceleration of the driving link have been found, we can calculate the other angular velocities and accelerations by using the velocity and acceleration analysis of one element.

2.3.2 Transmission from Right to Left

Figure 2.3.2 illustrates the second type of transmission. In this case, the second linkage is connected to the left-hand side of the driving link of the first linkage. Angles \(q_i\), for \(i = 1, \ldots, 4\), are the relative angles on the second linkage, while angle \(\theta^*\) is the driving angle of the first linkage with respect to the frame.

Referring to the second linkage, the driving angle \(\theta^*_2\) is on the right-hand side link instead of on the left-hand side link. From the configuration of Fig. 2.3.2, we obtain:

\[
q_1 = \pi - (\theta^* + \phi_1^*) - \beta + \phi_1
\]

(2.42)

where the angles \(\phi_1^*\) and \(\phi_1\) are negative if they are measured in the clockwise sense.

After finding the driving angle \(q_1\) of the second linkage, we will apply the method of image to obtain the other angles, angles \(q_2, q_3\) and \(q_4\), and then transfer them into the global angles \(\phi_2, \phi_3\) and \(\phi_4\).
Figure 2.3.2  A transmission from right to left.
The method of image is shown in Fig. 2.3.3. In the figure, the second linkage is mapped onto the right-hand side with a dotted line, such that angle \( q_i^0 \) is equal to angle \( q_i \), and length \( r_2^0 \) is equal to \( r_4 \), ..., etc. Comparing the dotted line configuration with Fig. 2.2.1, we find that both of them have their driving angles on the left-hand side links. Therefore, it is convenient to use the same equations (i.e., Eqs. 2.12 and 2.13) for calculating the angles \( q_i^0, q_3^0, q_4^0, \) and \( \theta_3^0 \) to obtain the angles \( q_i \) and \( \theta_3 \), which correspond to \( q_i^0 \) and \( -\theta_3^0 \), for \( i = 1, ..., 4 \).

Because the global angles \( \phi_2, \phi_3, \phi_4 \) can easily be understood and applied on the dynamic analysis, we transfer relative angles into global angles by using the following equations.

\[
\begin{align*}
\phi_2 &= q_2 + \phi_1 \\
\phi_3 &= 2\pi + \theta_3 - q_3 + \phi_1 \\
\phi_4 &= \pi - q_1 + \phi_1 
\end{align*}
\]

where angles \( \phi_1 \) and \( \theta_3 \) are negative if they are measured in the clockwise sense.

Referring to Fig. 2.3.2, we find that the driving link of the second linkage is connected to the driving link of the first linkage, therefore, the driving angular velocity and acceleration of the second linkage are transmitted from the first linkage by the following transformation equations:

\[
\begin{align*}
\dot{q}_1 &= -(\dot{\theta}_2^* + \dot{\phi}_1^*) \\
\ddot{q}_1 &= -(\ddot{\theta}_2^* + \ddot{\phi}_1^*)
\end{align*}
\]

Applying the method of image, we use Eqs. (2.29) to obtain other angular velocities \( q_i^0 \), for \( i = 2, 3, 4 \). Because the angular velocities in the image correspond to those in the real configuration, we can find the global velocities by the following equations:
Figure 2.3.3 The image system.
\[
\begin{align*}
\dot{\phi}_2 &= \dot{q}_2 + \dot{\phi}_1 \\
\dot{\phi}_3 &= \dot{q}_2 + \dot{q}_4 + \dot{\phi}_1 \\
\dot{\phi}_4 &= -\dot{q}_4 + \dot{\phi}_1
\end{align*}
\]

where \( q_i \) is equal to \( q_i^0 \), for \( i = 1, \ldots, 4 \).

Similarly, by the method of image, we can find the global angular accelerations:

\[
\begin{align*}
\ddot{\phi}_2 &= \ddot{q}_2 + \ddot{\phi}_1 \\
\ddot{\phi}_3 &= \ddot{q}_2 + \ddot{q}_4 + \ddot{\phi}_1 \\
\ddot{\phi}_4 &= -\ddot{q}_4 + \ddot{\phi}_1
\end{align*}
\]

where \( q_i \) is equal to \( q_i^0 \), for \( i = 1, \ldots, 4 \).

In the application of multiple linkages, Grashof's criterion [2] should only be satisfied for the driving linkage. The subsequent four-bar linkages do not have to satisfy Grashof's law. Therefore, there will be at least one revolving link if the sum of the shortest and longest link lengths is not greater than the sum of the remaining two link lengths.

### 2.4 Moving Frame

A moving frame is the ground link of a four-bar mechanism connected to the coupler of another element. As shown in Fig. 2.4.1, a moving frame is connected to a rigid body (i.e., block BCDE) which is on the coupler of the lower element. Since a four-bar mechanism has only one degree-of-freedom, there are two degrees of freedom in this system. \( \xi_B - \eta_B \) is a local body fixed coordinate system with the origin at point B, and \( x-y \) is the global coordinate with respect to the Cartesian coordinates. Angles \( \theta \)'s are the absolute coordinates with respect to the frame, while angles \( \phi \)'s are the global angles with
Figure 2.4.1 Four-bar mechanism with a moving frame.
respect to the x-axis. By specifying four positions, \((X^A, y^A), (X^F, Y^F), (\xi^C_B, \eta^C_B),\) and \((\xi^D_B, \eta^D_B)\), one can find the global positions \((X^C, Y^C)\) and \((X^D, Y^D)\), and then obtain the slope of the moving frame (i.e., angle \(\phi^u_i\)).

Referring to this configuration, we obtain the slope of the fixed frame of the lower element:

\[
\phi_1 = \arctan \left( \frac{\Delta Y}{\Delta X} \right) \tag{2.47}
\]

where

\[
\Delta Y = Y^F - Y^A \\
\Delta X = X^F - X^A .
\]

If the driving angles \(\phi_2\) has been specified, we can find the other global angles (i.e., \(\phi_3\) and \(\phi_4\)) of the lower element from the previous discussion. Therefore, the global position \((X^B, Y^B)\) and \((X^E, Y^E)\) can be obtained by the following equations:

\[
X^B = X^A + r_2 \cos \phi_2 , \quad Y^B = Y^A + r_2 \sin \phi_2 \tag{2.48a}
\]
\[
X^E = X^F + r_4 \cos \phi_4 , \quad Y^E = Y^F + r_4 \sin \phi_4 \tag{2.48b}
\]

Since there is a rigid body between both elements, we need to transfer the local position \((\xi^D_B, \eta^C_B)\) into global position \((X^C, Y^C)\). The transformation between local and global coordinates of point C is [1]:

\[
r^C = r^B + A s^C \tag{2.49}
\]

where the rotational transformation matrix
\[
A = \begin{bmatrix}
\cos \phi_3 & -\sin \phi_3 \\
\sin \phi_3 & \cos \phi_3
\end{bmatrix}.
\]

\[
\begin{bmatrix}
X^C \\
Y^C
\end{bmatrix} = \begin{bmatrix}
X^B \\
Y^B
\end{bmatrix} + \begin{bmatrix}
\xi^C_B \\
\eta^C_B
\end{bmatrix}, \quad \begin{bmatrix}
\xi^C_C \\
\eta^C_C
\end{bmatrix} = \begin{bmatrix}
\xi^C_B \\
\eta^C_B
\end{bmatrix}
\]

and the vector \( s \) with superscript "\( * \)" is described in terms of the local coordinate system.

Eq. (2.49) in the expanded form can be written as:

\[
X^C = X^B + \xi^C_B \cos \phi_3 - \eta^C_B \sin \phi_3 \quad (2.50a)
\]

\[
Y^C = Y^B + \xi^C_B \sin \phi_3 + \eta^C_B \cos \phi_3 \quad (2.50b)
\]

or

\[
X^C = X^B + L^C \cos(\theta^B + \phi_3) \quad (2.50c)
\]

\[
Y^C = Y^B + L^C \sin(\theta^B + \phi_3) \quad (2.50d)
\]

where

\[
L^C = \left[ \xi^2_B + \eta^2_B \right]^{1/2}
\]

\[
\theta^B = \arctan \left[ \frac{\eta^C_B}{\xi^C_B} \right]
\]

In the same manner, the global position \((X^D, Y^D)\) can be obtained as described above. Therefore, in finding the global positions \((X^C, Y^C)\) and \((X^D, Y^D)\), we can obtain the slope of the moving frame \( \phi^u_1 \) from Eq. (2.47). Hence, the global velocity and acceleration of the moving frame are transmitted from the connected coupler of the lower linkage:
\[ \dot{\phi}_1^u = \dot{\phi}_3 \] \hspace{1cm} (2.51a)
\[ \ddot{\phi}_1^u = \ddot{\phi}_3 \] \hspace{1cm} (2.51b)

where the superscript "u" indicates the "upper" element.

The other angles, angular velocities and accelerations of the upper element can also be found by using the same procedure as the one that we have gone through for the lower linkage with a fixed frame.
CHAPTER 3

DYNAMIC ANALYSIS OF THE TRANSMISSION ELEMENTS

Dynamic analysis involves the study of the motion of the interacting bodies, as well as the description of the relationships of the forces causing the motion. In general, a force acting on a system is a function of its position, velocity and time. In the previous chapter, we discussed the position and velocity analyses of a four-bar mechanism. Hence, the force, or acceleration, which corresponds to the position and velocity, can be found by using two approaches, namely, the vector dynamics and the analytical dynamics. Both approaches lead to the construction of mathematical models for describing a mechanical system. The vector dynamics approach is directly based on Newton's laws of motion; while the analytical dynamics approach is based on the principle of virtual work which includes two scalar functions, i.e., kinetic energy and potential energy.

In this chapter, we will put more emphasis on the analytical dynamics. In view of the fact that this approach treats the system as a whole, we shall find that it is suited to the highly constrained systems by using Lagrange's equations of motion. Using a generalized equation of motion to deal with a single degree-of-freedom mechanism, we shall find that this method suits perfectly the digital methods of calculation. If the number of equations of motion of a system is more than its number of degrees of freedom, then the equations of motion must be considered in their constrained form. In the constrained differential equations, constraint forces are expressed in terms of the Jacobian matrix and a set of Lagrange multipliers.

In the subsequent sections, we will use Lagrange's equations of motion to establish the governing differential equations of motion. This is achieved by using relative and
absolute coordinates. The system analysis will be developed starting from one element with fixed frame to several elements with moving frames. By applying the Lagrange multiplier technique, a set of second order differential equations for the constrained motion is established. Finally, a numerical method, called the L-U factorization method, and a numerical integration algorithm, called the RUNGE-KUTTA method, are used to obtain the solutions.

3.1 Lagrange's Equations

The generalized equation of motion for a system with a single degree of freedom has been stated in [3]. This equation may appropriately be applied to the four-bar mechanism. From Newton's laws of motion, the principle of work and kinetic energy states that the power input of an ideal mechanical system equals the rate of increase of its kinetic energy; this is also called the theorem of power balance, namely,

\[ P = \frac{dT}{dt} . \quad (3.1) \]

Furthermore, the power \( P \) and kinetic energy \( T \) in a system with single degree of freedom can be expressed as:

\[ P = Q \dot{q} \quad (3.2a) \]
\[ T = \frac{1}{2} J \dot{q}^2 \quad (3.2b) \]

where \( Q \) is the generalized force, \( J \) is the generalized inertia and \( \dot{q} \) is the time rate of change in the generalized coordinate.
Using Eqs. (3.1) and (3.2), we obtain

\[ \frac{dt}{dt} \left[ \frac{1}{2} J \dot{q}^2 \right] = Q \dot{q} \quad \text{(3.3a)} \]

or

\[ J \ddot{q} + \frac{1}{2} \frac{dJ}{dt} \dot{q}^2 = Q \dot{q} \quad \text{(3.3b)} \]

Since generalized inertia \( J \) can be expressed as a function of \( q \), we thus have:

\[ \frac{dJ}{dt} = \frac{dJ}{dq} \frac{dq}{dt} \quad \text{(3.4)} \]

By substituting Eq. (3.4) into Eq. (3.3b), we arrive at the generalized equation of motion:

\[ J \ddot{q} + \frac{1}{2} \frac{dJ}{dq} \dot{q}^2 = Q \quad \text{(3.5)} \]

Note that the generalized force \( Q \) may consist of both conservative and nonconservative forces. Eq. (3.5) not only is useful for a single degree of freedom mechanism, it can also be used as the general form of Lagrange's equations for a system with several degrees of freedom.

We express the kinetic energy of a system in terms of \( q \) and \( \dot{q} \), i.e.,

\[ T(q, \dot{q}) = \frac{1}{2} J(q) \dot{q}^2 \quad \text{(3.6)} \]

and then the following two equations can be obtained by differentiating Eq. (3.6) with respect to \( q \) and \( \dot{q} \) respectively.

\[ \frac{\partial T}{\partial q} = \frac{1}{2} \frac{dJ}{dq} \dot{q}^2 \quad \text{(3.7a)} \]

\[ \frac{\partial T}{\partial \dot{q}} = J \dot{q} \quad \text{(3.7b)} \]
Differentiating Eq. (3.7b) with respect to time, we obtain

\[
\frac{d}{dt} \frac{\partial T}{\partial q} = \frac{\partial}{\partial q} \left( \frac{\partial T}{\partial \dot{q}} \right) + \frac{dJ}{dq} \frac{dq}{dt} \quad (3.8a)
\]

or

\[
J \dddot{q} = \frac{d}{dt} \frac{\partial T}{\partial \ddot{q}} - \frac{dJ}{dq} \dot{q}^2 \quad (3.8b)
\]

Finally, substituting Eq. (3.7a) and (3.8b) into Eq. (3.5), we obtain a special case of Lagrange’s equation of motion:

\[
\frac{d}{dt} \frac{\partial T}{\partial \dot{q}} - \frac{\partial}{\partial q} \frac{\partial L}{\partial \dot{q}} = 0 \quad (3.9)
\]

For a system with \( n \) degrees of freedom, Eq. (3.9) becomes

\[
\frac{d}{dt} \frac{\partial T}{\partial \dot{q}_i} - \frac{\partial}{\partial q_i} \frac{\partial L}{\partial \dot{q}_i} = Q_i \quad \text{for } i = 1, 2, \ldots, n \quad (3.10)
\]

and it is usually referred to as Lagrange’s equation.

If the generalized forces \( Q_i \) are conservative [4], then \( Q_i \) is given by

\[
Q_i = -\frac{\partial V}{\partial q_i} \quad (3.11)
\]

where \( V \) denotes the potential energy of the system. However, since \( Q_i \) may consist of nonconservative forces \( Q_i^* \) as well, we rewrite \( Q_i \) as their sum, i.e.,

\[
Q_i = -\frac{\partial V}{\partial q_i} + Q_i^* \quad (3.12)
\]

In view of the fact that \( V \) is only a function of \( q_i \), we obtain:

\[
\frac{\partial V}{\partial q_i} = 0 \quad (3.13)
\]
After substituting Eq. (3.12) and (3.13) into Eq. (3.10), the Lagrange's equation takes the form:

\[ \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q^*_i \]

(3.14)

where

\[ L = T - V \]

is called the Lagrangian function. The \( Q^*_i \) are the nonconservative parts of the generalized forces \( Q_i \), and the subscript "i" refers to the coordinates in the system [5]. Since \( Q^*_i \) are not derivable from a potential function, they may represent frictional forces, time-variant forcing functions and constraint forces [4].

### 3.2 Lagrange Multipliers

In some practical situations, it is not easy to obtain the generalized equation of motion. In that case, we need more Lagrangian coordinates than the degrees of freedom to describe a system. Therefore, the constraint forces must appear in the generalized force terms owing to the existence of constraints [4].

The criteria for calculating the number of constraint equations is given by [1]:

\[ N_f = N_c - N_h \]

(3.15)

where

\[ N_f = \text{number of degrees of freedom} \]

\[ N_c = \text{number of coordinates} \]

\[ N_h = \text{number of holonomic constraints} \]
Thus, we need a Lagrange multiplier technique, as that stated in [1] and [4], to solve a set of equations of motion containing the constraint forces.

From the principle of virtual work, the work done by the constraint forces in a virtual displacement $\delta q$ is zero if the constraints are frictionless; this can be expressed as:

$$ \mathbf{g}^T \delta \mathbf{q} = 0 $$

(3.16)

where vector $\mathbf{g}$ represents a set of constraint forces while vector $\mathbf{q}$ denotes a set of coordinates. We assume that there exists a set of independent constraint equations, i.e.,

$$ \Phi \equiv \Phi(\mathbf{q}, t) = 0 $$

(3.17a)

and a displacement of $\delta \mathbf{q}$ consistent with the constraints such that:

$$ \Phi(\mathbf{q} + \delta \mathbf{q}, t) = 0 . $$

(3.17b)

Using the Taylor series expansion of Eq. (3.17b) about $\mathbf{q}$, and eliminating the higher order terms for infinitesimal $\delta \mathbf{q}$, we find that

$$ \Phi_{\mathbf{q}} \delta \mathbf{q} = 0 . $$

(3.18)

The vector of coordinate $\mathbf{q}$ may be partitioned into a set of dependent coordinates $\mathbf{u}$ and independent coordinates $\mathbf{v}$, i.e., $\mathbf{q} = \begin{bmatrix} \mathbf{u}^T, \mathbf{v}^T \end{bmatrix}^T$. This yields a partitioned vector of virtual displacements $\delta \mathbf{q} = \begin{bmatrix} \delta \mathbf{u}^T, \delta \mathbf{v}^T \end{bmatrix}^T$ and a partitioned Jacobian $\Phi_{\mathbf{q}} = \begin{bmatrix} \Phi_{\mathbf{u}}, \Phi_{\mathbf{v}} \end{bmatrix}$. If vector $\mathbf{g}$ is also partitioned as $\mathbf{g} = \begin{bmatrix} \mathbf{g}(\mathbf{u}), \mathbf{g}(\mathbf{v}) \end{bmatrix}^T$, Eqs. (3.16) and (3.18) can be recast and arranged as:
Since the constraint equations of Eq. (3.17a) are assumed to be independent, the matrix $\Phi_u^T$ is nonsingular. Therefore, the first row of the matrix, $\Phi_u^T$, can be expressed as a linear combination of the other rows of the matrix as:

$$\Phi_u^T \delta u = - \Phi_v^T \delta v$$

(3.19)

Thus, appending Eq. (3.20) to Eq. (3.21b), we get the constraint forces in the form:

$$g = \Phi_v^T \lambda$$

(3.22)

and consequently, the Lagrange's equation, Eq. (3.14), can be rewritten in the following general form:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = g_i \quad \text{for } i = 1, 2, \ldots, N_c$$

(3.23)

where constraint forces $g_i$'s are obtained from Eq. (3.22).

Now we have $N_c$ equations of motion, but there are $N_c + N_h$ unknowns, namely, $N_c$ coordinates and $N_h$ Lagrange's multipliers. In order to solve Eq. (3.23), we need $N_h$ additional equations, and those constraint differential equations are obtained by twice differentiating the constraint equations with respect to time.
3.3 Dynamic Analysis Using Explicit Equations

As discussed in Section 2.1, the explicit equations are expressed in terms of the relative coordinates. We may choose angle $q_j$ as the generalized coordinate because a four-bar mechanism has a single degree of freedom. Thus, the generalized equation of motion, i.e., Eq. (3.5), is suitable for describing the system. It is well known that the Lagrange's equation with only one generalized coordinate allows the analysis to be made without considering the constraint forces.

The inertial properties of a four-bar mechanism are characterized as that shown in Fig. 3.2.1. $m_j$ represents the mass of each link, $u_i$ is the moment of inertia with respect to each mass center of the link, and the coordinate $(X_j, Y_j)$ denotes the position of mass center with respect to the Cartesian coordinate system. Next, we will calculate the kinetic energy of the system from which the generalized inertia is obtained.

3.3.1 The Generalized Inertia

From the configuration of Fig. 3.2.1, the position of the mass center of input link is:

$$X_2 = \frac{1}{2} r_2 \cos q_1$$
$$Y_2 = \frac{1}{2} r_2 \sin q_1.$$  \hspace{1cm} (3.24a, b)

By differentiating Eq. (3.24) with respect to time, its velocity is given by:

$$\dot{X}_2 = -\frac{1}{2} r_2 \sin q_1 \dot{q}_1$$
$$\dot{Y}_2 = \frac{1}{2} r_2 \cos q_1 \dot{q}_1.$$  \hspace{1cm} (3.25a, b)

Therefore, the kinetic energy of the input link is:
Figure 3.2.1 Four-bar mechanism with the inertial properties.
\[ T_2 = \frac{1}{2} [m_2 (\dot{X}_2^2 + \dot{Y}_2^2) + u_2 \dot{q}_1^2] \]
\[ = \frac{1}{2} J_2 \dot{q}_1^2 \]  
\( \text{(3.26)} \)

where
\[ J_2 = \frac{1}{4} m_2 r_2^2 + u_2. \]

For the coupler, the center of mass is:
\[ \begin{align*}
X_3 &= r_2 \cos \alpha_1 + \frac{1}{2} r_3 \cos \theta_3 \\
Y_3 &= r_2 \sin \alpha_1 + \frac{1}{2} r_3 \sin \theta_3
\end{align*} \]  
\( \text{(3.27a)} \)

thus,
\[ \begin{align*}
\dot{X}_3 &= -r_2 \sin \alpha_1 \dot{\alpha}_1 - \frac{1}{2} r_3 \sin \theta_3 \dot{\theta}_3 \\
\dot{Y}_3 &= r_2 \cos \alpha_1 \dot{\alpha}_1 + \frac{1}{2} r_3 \cos \theta_3 \dot{\theta}_3
\end{align*} \]  
\( \text{(3.28a)} \)

Recalling Eqs. (2.38) which are derived from the loop-closure equation, we shall define two additional notations, the use of which will very much simplify our presentation.

Let
\[ R_{23} = \frac{r_2 \sin(\theta_2 - \theta_4)}{r_3 \sin(\theta_4 - \theta_3)} = \frac{\dot{\theta}_3}{\dot{q}_1} \]  
\( \text{(3.29a)} \)

\[ R_{24} = \frac{r_2 \sin(\theta_2 - \theta_3)}{r_4 \sin(\theta_4 - \theta_3)} = \frac{\dot{\theta}_4}{\dot{q}_1} \]  
\( \text{(3.29b)} \)

Thus, substituting \( \dot{\theta}_3 = R_{23} \dot{q}_1 \) and \( \dot{\theta}_4 = R_{24} \dot{q}_1 \) into Eqs. (3.28), we obtain the kinetic energy of the coupler:
\[ \begin{align*}
T_3 &= \frac{1}{2} m_3 (\dot{X}_3^2 + \dot{Y}_3^2) + \frac{1}{2} u_3 \dot{\theta}_3^2 \\
&= \frac{1}{2} J_3 \dot{q}_1^2
\end{align*} \]  
\( \text{(3.30)} \)

where
\[ J_3 = m_3 \left( -r_2 \sin q_1 - \frac{1}{2} r_3 R_{23} \sin \theta_3 \right)^2 + \left( r_2 \cos q_1 + \frac{1}{2} r_3 R_{23} \cos \theta_3 \right)^2 \left( R_{23} \right)^2, \]

and angle \( \theta_3 \) is obtained from Eq. (2.12).

Finally, for the output link, the center of mass is:

\[
\begin{align*}
X_4 &= r_1 + \frac{1}{2} r_4 \cos \theta_4 \\
Y_4 &= \frac{1}{2} r_4 \sin \theta_4
\end{align*}
\]

hence,

\[
\begin{align*}
\dot{X}_4 &= -\frac{1}{2} r_4 \sin \theta_4 \dot{\theta}_4 \\
\dot{Y}_4 &= \frac{1}{2} r_4 \cos \theta_4 \dot{\theta}_4
\end{align*}
\]

Substituting Eq. (3.29b) into Eqs. (3.32), we obtain the kinetic energy of output link:

\[
T_4 = \frac{1}{2} m_4 (\dot{X}_4^2 + \dot{Y}_4^2) + \frac{1}{2} u_4 \dot{\theta}_4^2 = \frac{1}{2} J_4 \dot{q}_4^2
\]

where

\[
J_4 = \left[ \frac{1}{4} m_4 r_4^2 + u_4 \right] \left( R_{24} \right)^2.
\]

Thus, the total kinetic energy of a four-bar mechanism with fixed frame is given by

\[
T = \sum_{i=2}^{4} \sum_{i=2}^{4} T_i = \frac{1}{2} J q_1^2
\]

where the generalized inertia \( J = \sum_{i=2}^{4} J_i. \)
3.3.2 The Partial Derivative of the Generalized Inertia

After getting the generalized inertia $J$, we shall differentiate each of $J_2$, $J_3$ and $J_4$ with respect to the generalized coordinate $q_1$ to obtain the partial derivative of the generalized inertia, i.e.,

$$\frac{dJ}{dq_1} = \sum_{i=2}^{4} \frac{dJ_i}{dq_1}.$$  \hspace{1cm} (3.35)

From $J_2$ of Eq. (3.26), we obtain

$$\frac{dJ_2}{dq_1} = 0.$$ \hspace{1cm} (3.36)

The next step is to find the following five identities from Eqs. (3.29),

$$\frac{d\theta_2}{dq_1} = R_{23}$$ \hspace{1cm} (3.37a)

$$\frac{d\theta_4}{dq_1} = R_{24}$$ \hspace{1cm} (3.37b)

$$\frac{d(q_1 - \theta_2)}{dq_1} = 1 - R_{23}$$ \hspace{1cm} (3.37c)

$$\frac{d(q_1 - \theta_4)}{dq_1} = 1 - R_{24}$$ \hspace{1cm} (3.37d)

$$\frac{d(\theta_4 - \theta_2)}{dq_1} = R_{24} - R_{23}.$$ \hspace{1cm} (3.37e)

Applying the chain rule of differentiation on $J_3$ of Eq. (3.30), and using Eqs. (3.37), we obtain:
\[
\frac{dJ_3}{dq_1} = 2 m_3 \left( k_1 \left[ -r_2 \cos q_1 - \frac{1}{2} r_3 (R_{23})^2 \cos \theta_3 - \frac{1}{2} k_2 r_2 \sin \theta_3 \right] + k_2 \left[ -r_2 \sin q_1 - \frac{1}{2} r_3 (R_{23})^2 \sin \theta_3 + \frac{1}{2} k_2 r_2 \cos \theta_3 \right] \right) \\
+ 2 \frac{r_2}{r_3} k_2 u_4 R_{23}
\]

where

\[
k_1 = -r_2 \sin q_1 - \frac{1}{2} r_3 R_{23} \sin \theta_3
\]

\[
k_2 = \frac{((1 - R_{24}) \sin(\theta_4 - \theta_3) \cos(q_1 - \theta_4))}{\sin^2(\theta_4 - \theta_3)} - \frac{((R_{24} - R_{23}) \sin(q_1 - \theta_3) \cos(\theta_4 - \theta_2))}{\sin^2(\theta_4 - \theta_3)}
\]

\[
k_3 = r_2 \cos q_1 + \frac{1}{2} r_3 R_{23} \cos \theta_3.
\]

In a similar manner, the term \( \frac{dJ_4}{dq_1} \) is obtained from \( J_4 \) of Eq. (3.33) by using the chain rule of differentiation

\[
\frac{dJ_4}{dq_1} = 2 \frac{r_2}{r_4} k_4 R_{24} \left( \frac{1}{4} m_4 r_4^2 + u_4 \right)
\]

where

\[
k_4 = \frac{((1 - R_{23}) \sin(\theta_4 - \theta_3) \cos(q_1 - \theta_4))}{\sin^2(\theta_4 - \theta_3)} - \frac{((R_{24} - R_{23}) \sin(q_1 - \theta_3) \cos(\theta_4 - \theta_2))}{\sin^2(\theta_4 - \theta_3)}
\]

It follows that the partial derivative of the generalized inertia, i.e.,

\[
\frac{dI}{dq_1} = \sum_{i=2}^{4} \frac{dJ_i}{dq_1},
\]

is obtained from Eqs. (3.36), (3.38) and (3.39).
3.3.3 The Generalized Force

If there are active forces applied on the system, say, forces $F_{X_i}$ in the $x$-axis, $F_{Y_i}$ in the $y$-axis and torques $T_{\theta_i}$ in the counterclockwise sense, for $i = 2,3,4$, then we obtain the power of the active forces from Eq. (3.2a):

$$P = \mathbf{Q} \dot{q}_1$$

$$= \sum_{i=2}^{4} (F_{X_i} \dot{X}_i + F_{Y_i} \dot{Y}_i + T_{\theta_i} \dot{\theta}_i) \quad (3.40)$$

Substituting Eqs. (3.25), (3.28) and (3.32) into Eq. (3.40), and then using Eqs. (3.29) while isolating $\dot{q}_1$, we obtain the generalized force as the summation of three separate generalized force terms:

$$\mathbf{Q} = F_X + F_Y + T_{\theta} \quad (3.41)$$

where

$$F_X = F_{X2} \left[ -\frac{1}{2} r_2 \sin q_1 \right] + F_{X3} \left[ -r_2 \sin q_1 - \frac{1}{2} r_3 R_{23} \sin \theta_3 \right]$$

$$+ F_{X4} \left[ -\frac{1}{2} r_4 R_{24} \sin \theta_4 \right]$$

$$F_Y = F_{Y2} \left[ \frac{1}{2} r_2 \cos q_1 \right] + F_{Y3} \left[ r_2 \cos q_1 + \frac{1}{2} r_3 R_{23} \cos \theta_3 \right]$$

$$+ F_{Y4} \left[ \frac{1}{2} r_4 R_{24} \cos \theta_4 \right]$$

$$T_{\theta} = T_{\theta2} + T_{\theta3} R_{23} + T_{\theta4} R_{24}. $$
The generalized force $Q$ may include conservative and nonconservative forces. If there is only gravity affecting the system, we find the generalized force by setting

$$F_X = T_\theta = 0$$  \hspace{1cm} (3.42a)$$
$$F_{Y_i} = m_i g, \text{ for } i = 2, 3, 4.$$  \hspace{1cm} (3.42b)$$

Thus,

$$Q = F_Y$$

$$= g \left( \frac{m_2}{2} + m_3 \right) R_2 \cos q_1 + \frac{1}{2} m_3 r_3 R_{23} \cos \theta_3$$

$$+ \frac{1}{2} m_4 r_4 R_{24} \cos \theta_4 \right)$$  \hspace{1cm} (3.43)$$

where $g$ is the gravitational constant.

Subsequently, from Eq. (3.5), the angular acceleration of input link can be explicitly expressed as:

$$\ddot{q}_1 = \left( Q - \frac{1}{2} \frac{dF}{dq_1} q_1^2 \right) / J$$  \hspace{1cm} (3.44)$$

3.3.4 Transmission from Side to Side

In accordance with the criteria, $Nf = Nc - Nh$, if we use the generalized equation of motion, Eq. (3.5), for one element, then there is no need for a constraint equation. However, if there are two elements in the system, then each element needs one generalized equation of motion, and as the system has only one degree of freedom, we need another constraint equation to describe the relationship between the two elements. Thus, the Lagrange multiplier technique will be applied to solve the problem.
Using the same notations as in Fig. 3.3.1, and referring to the loop-closure equation, we obtain

\[ r_2 e^{i\phi_2} + r_3 e^{i\phi_3} = r_1 e^{i\phi_1} + R_4 e^{i\phi_4} \]  \hspace{1cm} (3.45)

Applying the same procedures just described for Eq. (2.4a), we obtain the constraint equation:

\[ \Phi_3 = r_1^2 + r_2^2 + r_4^2 - r_3^2 - 2r_1r_2 \cos(\phi_2 - \phi_1) \\
+ 2r_1r_4 \cos(\phi_4 - \phi_1) - 2r_2r_4 \cos(\phi_4 - \phi_2). \]  \hspace{1cm} (3.46)

From the configuration of Fig. 3.3.1, we obtain the following relationships:

\[ \phi_2 = \phi_1 + q_1 \]  \hspace{1cm} (3.47a)
\[ \phi_3 = \phi_1 + \phi_1 \]  \hspace{1cm} (3.47b)
\[ \phi_4 = \phi_1 + q_1 + \beta \]  \hspace{1cm} (3.47c)

Substituting Eqs. (3.47) into Eq. (3.46), we obtain the constraint equation in terms of angles \( q_1 \) and \( \phi_1 \). Next, differentiating this equation with respect to time twice, we arrive at:

\[ \Phi_3 q_1 + \Phi_2 q_1 q_1 = \gamma_3 \]  \hspace{1cm} (3.48)

where

\[ \Phi_3 q_1 = r_1r_2 \sin q_1 - r_2r_4 \sin(\phi_1 + \beta - \phi_2) \]
\[ \Phi_2 q_1 = -r_1r_4 \sin(\phi_1 + \phi_2) + r_2r_4 \sin(\phi_1 + \beta - \phi_2) \]
Figure 3.3.1 A transmission from side to side.
\[
\gamma_3 = r_1 r_4 \left[ \cos(\phi_2^* + \beta - \phi_1)\left(\phi_2^* - \phi_1\right) + \sin(\phi_2^* + \beta - \phi_1)\left(\phi_2^* - \phi_1\right) \right] \\
- r_2 r_4 \left[ \cos(\phi_2^* + \beta - \phi_2)\left(\phi_2^* - \phi_2\right) + \sin(\phi_2^* + \beta - \phi_2)\left(\phi_2^* - \phi_2\right) \right] \\
- r_1 r_2 \cos q_1 \dot{q}_1^*
\]

and

\[
\dot{\phi}_1 = \dot{\phi}_1^* = \ddot{\phi}_1^* = 0, \text{ if the frames are fixed.}
\]

Therefore, for a two-element system, we have two generalized equations of motion from Eq. (3.5) and one constraint differential equation from Eq. (3.48). Combining those equations, and applying the Lagrange multiplier technique, we obtain:

\[
\begin{pmatrix}
J & 0 & \Phi_s q_1 & \Phi_s q_1^* \\
0 & J^* & 0 & 0 \\
\Phi_s q_1 & \Phi_s q_1^* & 3x3 \\
\end{pmatrix}
\begin{pmatrix}
\ddot{q}_1 \\
\ddot{q}_1^* \\
\ddot{q}_1 \\
\end{pmatrix}
= 
\begin{pmatrix}
f_1 \\
f_2 \\
\gamma_3 \\
\end{pmatrix}
\]

(3.49)

where

\[
f_i = \left\{ Q - \frac{1}{2} \frac{dJ}{dq} \dot{q}_i^2 \right\}_i
\]

and i refers to different elements.

Finally, we use the L-U factorization method to obtain the solutions, and then apply the RUNGE-KUTTA algorithm to calculate \(\dot{q}_i\) and \(q_i\) at the next time step. The other relative accelerations, velocities and positions can be found by previous kinematic analysis. If there is one additional element in this system, then we will need one additional generalized equation of motion and one constraint equation to describe the system.
3.4 Dynamic Analysis Using Simultaneous Equations

The simultaneous equations used for the analysis are in terms of three absolute coordinates $\theta$'s. Employing three independent coordinates in a four-bar mechanism, three equations of motion are derived from the Lagrangian equations in the form of Eq. (3.23). From the above, we know that the number of Lagrangian coordinates is more than the number of degrees of freedom by two; thus two additional constraint differential equations are included in the set of equations of motion. To solve these equations, we need to find two constraint equations and then use the Lagrangian multiplier technique.

3.4.1 Equations of Motion

In order to obtain the equations of motion, we need to find the Lagrangian function which consists of two scalar functions, namely, the kinetic and potential energy functions. Rewriting Eq. (3.26), (3.30) and (3.33) in terms of absolute coordinates $\theta$'s, and combining them together, we obtain the kinetic energy:

$$T = T_2 + T_3 + T_4 \quad (3.50)$$

where

$$T_2 = \frac{1}{2} \left( \frac{1}{4} m_2 r_2^2 + u_2 \right) \dot{\theta}_2^2$$

$$T_3 = \frac{1}{2} m_3 \left[ r_2^2 \dot{\theta}_2^2 + r_2 r_3 \cos(\theta_2 - \theta_3) \dot{\theta}_2 \dot{\theta}_3 + \frac{1}{4} r_3^2 \dot{\phi}_3^2 \right] + \frac{1}{2} u_3 \dot{\phi}_3^2$$

$$T_4 = \left( \frac{1}{8} m_4 r_4^2 + \frac{1}{2} u_4 \right) \dot{\phi}_4^2$$
The potential energy, due to gravitational force, is obtained by using Eqs. (3.24b), (3.27b) and (3.31b), i.e.,

\[
V = g \sum_{i=2}^{4} m_i Y_i \\
= g \left[ \frac{1}{2} m_2 r_2 \sin \theta_2 + m_3 \left( r_2 \sin \theta_2 + \frac{1}{2} r_3 \sin \theta_3 \right) + \frac{1}{2} m_4 r_4 \sin \theta_4 \right].
\]

(3.51)

Potential energy due to other forces, such as spring and damper, can be added into the right side of Lagrange's equation. Thus, we obtain the Lagrangian function, \( L = T - V \).

Now, we evaluate some of the terms which will be used in writing the equations of motion, i.e.

\[
\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{\theta}_2} \right] = M_{11} \ddot{\theta}_2 + M_{12} \ddot{\theta}_3 + I \dot{\theta}_3,
\quad \frac{\partial L}{\partial \theta_2} = k_1
\]

(3.52a)

\[
\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{\theta}_3} \right] = M_{21} \ddot{\theta}_2 + M_{22} \ddot{\theta}_3 + I \dot{\theta}_3,
\quad \frac{\partial L}{\partial \theta_3} = k_2
\]

(3.52b)

\[
\frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{\theta}_4} \right] = M_{33} \ddot{\theta}_4,
\quad \frac{\partial L}{\partial \theta_4} = k_3
\]

(3.52c)

where

\[
M_{11} = \left( \frac{m_2}{4} + m_3 \right) r_2^2 + u_2,
\]
\[
M_{12} = M_{21} = \frac{1}{2} m_2 r_2 r_3 \cos(\theta_2 - \theta_3),
\]
\[
M_{22} = \frac{1}{4} m_3 r_3^2 + u_3
\]
\[
M_{33} = \frac{1}{4} m_4 r_4^2 + u_4
\]
\[
I = -\frac{1}{2} m_3 r_2 r_3 \sin(\theta_2 - \theta_3)(\dot{\theta}_2 - \dot{\theta}_3).
\]
\[ k_1 = -\frac{1}{2} m_3 r_2 r_3 \sin(\theta_2 - \theta_3) \dot{\theta}_2 \dot{\theta}_3 - \left( \frac{m_2}{2} + m_3 \right) g r_2 \cos\theta_2 \]
\[ k_2 = \frac{1}{2} m_3 r_2 r_3 \sin(\theta_2 - \theta_3) \dot{\theta}_2 \dot{\theta}_3 - \frac{1}{2} m_3 g r_3 \cos\theta_3 \]
\[ k_3 = -\frac{1}{2} m_4 g r_4 \cos\theta_4. \]

Hence, from Eq. (3.23), the equations of motion become:

\[ M_{11} \ddot{\theta}_1 + M_{12} \ddot{\theta}_2 + (I_0 - k_1) = g_1 \]  \hspace{1cm} (3.53a)
\[ M_{21} \ddot{\theta}_1 + M_{22} \ddot{\theta}_2 + (I_2 - k_2) = g_2 \]  \hspace{1cm} (3.53b)
\[ M_{33} \ddot{\theta}_3 - k_3 = g_3 \]  \hspace{1cm} (3.53c)

where the constraint forces \( g_i \)'s will be calculated in the next section.

3.4.2 Constraint Forces

For a four-bar mechanism, there are three equations of motion, but from the criteria \( N_f = N_c - N_h \) we need two additional constraint equations. We now use the same constraint equations as given in Eqs. (2.32). By differentiating Eqs. (2.32) twice with respect to time, we obtain:

\[ \Phi_{\theta} \ddot{\theta} = \gamma \]  \hspace{1cm} (3.54)

where

\[ \theta = [\theta_2, \theta_3, \theta_4]^T \]
\[ \Phi_{\theta} = \begin{bmatrix} \Phi_{1\theta_2} & \Phi_{1\theta_3} & \Phi_{1\theta_4} \\ \Phi_{2\theta_2} & \Phi_{2\theta_3} & \Phi_{2\theta_4} \\ \Phi_{3\theta_2} & \Phi_{3\theta_3} & \Phi_{3\theta_4} \end{bmatrix} \]
\[ = \begin{bmatrix} -r_2 \sin\theta_2 & -r_3 \sin\theta_3 & r_4 \sin\theta_4 \\ r_2 \cos\theta_2 & r_3 \cos\theta_3 & -r_4 \cos\theta_4 \end{bmatrix} \]
Thus, using the results of Eq. (3.54), we obtain the constraint forces from Eq. (3.22), i.e.,

\[ \mathbf{g} = \mathbf{T} \mathbf{A} \]

where

\[ \mathbf{g} = [g_1, g_2, g_3]^T \]
\[ \lambda = [\lambda_1, \lambda_2]^T . \]

Next, combining Eqs. (3.53) - (3.55), we arrive at the result:

\[ \mathbf{M} \begin{bmatrix} \ddot{\theta}_2 \\ \ddot{\theta}_3 \\ \ddot{\theta}_4 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \]

where

\[ f_1 = k_1 - I \dot{\theta}_3 \]
\[ f_2 = k_2 - I \dot{\theta}_2 \]
\[ f_3 = k_3 \]
Similarly, if there are two elements in a system, as that shown in Fig. 2.3.1 or Fig. 3.3.1, then for each element, we shall have a set of equations of motion as that given in Eq. (3.56). These equations, together with Eq. (3.48), describe the two-element system, i.e.,

\[
\begin{bmatrix}
M_{11} & M_{21} & M_{22} \\
M_{21} & M_{33} & 0 \\
0 & 0 & 0
\end{bmatrix}
symmetric
\]

\[
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3
\end{bmatrix} =
\begin{bmatrix}
f_1 \\
f_2 \\
f_3
\end{bmatrix} \tag{3.57}
\]

\[
\begin{bmatrix}
\Phi_{1\theta_2} & \Phi_{1\theta_3} & \Phi_{1\theta_4} \\
\Phi_{2\theta_2} & \Phi_{2\theta_3} & \Phi_{2\theta_4} \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\Phi_{1\theta_2}^* & \Phi_{1\theta_3}^* & \Phi_{1\theta_4}^* \\
\Phi_{2\theta_2}^* & \Phi_{2\theta_3}^* & \Phi_{2\theta_4}^* \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\lambda_1 \\
\lambda_2 \\
\lambda_3
\end{bmatrix}
\begin{bmatrix}
\gamma_1 \\
\gamma_2 \\
\gamma_3
\end{bmatrix}
\]

Finally, we use the L-U factorization method to obtain the solutions of Eq. (3.57), and then employ the RUNGE-KUTTA algorithm to find \( \dot{\theta}_2 \) and \( \theta_2 \) at the next time step. The other accelerations, velocities and positions can then be obtained by kinematic analysis.
If there is one additional element in the system, then we need three additional equations of motion and three constraint equations for the analysis. Therefore, the mass matrix will become very large.

3.5 Moving Frame

As discussed in Section 2.4, a moving frame does not have a fixed ground link. In this section, we will discuss only one floating element, and then use this idea and Lagrange multiplier technique to develop a system with several elements connected to the couplers. To study the process, we start with the derivation of the equations of motion, and then find the constraints between the lower and upper elements.

3.5.1 Equations of Motion

Referring to Fig. 3.5.1, we choose $X_1$, $Y_1$, $\phi_1$ and $\theta_2$ as four independent coordinates since a floating element has four degrees of freedom. Here, $X_1$, $Y_1$ and $\phi_1$ are global coordinates with respect to the Cartesian coordinate system, while $\theta_2$ is the relative angle with respect to the moving frame. In what follows, we shall obtain equations of motion with respect to each independent coordinate.

The kinetic and potential energies of the system are given by

$$ T = \sum_{i=1}^{4} T_i \quad \text{(3.58a)} $$

$$ V = g \sum_{i=1}^{4} (m_i Y_i) \quad \text{(3.58b)} $$

where

$$ T_i = \frac{1}{2} m_i (\dot{X}_i^2 + \dot{Y}_i^2) + \frac{1}{2} u_i \dot{\phi}_1^2. $$
Figure 3.5.1 A floating four-bar mechanism with the inertial properties.
From Eqs. (2.38) and (3.29), the Lagrangian function takes the form:

\[
L = T - V
\]

\[
= \sum_{i=1}^{4} \left[ \frac{1}{2} m_i (\dot{X}_i^2 + \dot{Y}_i^2) - g(m_i Y_i) \right]
+ \frac{1}{2} [u_1 \ddot{\phi}_1^2 + u_2 (\dot{\phi}_1 + \dot{\phi}_2)^2 + u_3 (\dot{\phi}_1 + R_{23} \dot{\phi}_2)^2
+ u_4 (\dot{\phi}_1 + R_{24} \dot{\phi}_2)^2] .
\] (3.59)

Hence, Eq. (3.14), the general form of Lagrange's equation, yields four equations of motion with respect to four independent coordinates, i.e.,

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial X_1} \right) - \frac{\partial L}{\partial \dot{X}_1} = Q_{X_1}
\] (3.60a)

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial Y_1} \right) - \frac{\partial L}{\partial \dot{Y}_1} = Q_{Y_1}
\] (3.60b)

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \phi_1} \right) - \frac{\partial L}{\partial \dot{\phi}_1} = Q_{\phi_1}
\] (3.60c)

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \phi_2} \right) - \frac{\partial L}{\partial \dot{\phi}_2} = Q_{\phi_2}
\] (3.60d)

where

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial X_1} \right) = \sum_{i=1}^{4} m_i \dddot{X}_i
\]
\[
\frac{d}{dt} \left( \frac{\partial L}{\partial Y_1} \right) = \sum_{i=1}^{4} m_i \ddot{Y}_i
\]

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \phi_1} \right) = \sum_{i=2}^{4} m_i \left[ x_i \frac{\partial \dot{X}_i}{\partial \phi_1} + \ddot{X}_i \frac{d}{dt} \left( \frac{\partial \dot{X}_i}{\partial \phi_1} \right) + \dot{Y}_i \frac{\partial Y_i}{\partial \phi_1} + \dot{Y}_i \frac{d}{dt} \left( \frac{\partial Y_i}{\partial \phi_1} \right) \right] + u_1 \ddot{\phi}_1 + u_2 (\ddot{\phi}_1 + \ddot{\phi}_2) + u_3 \left( \theta_2 \frac{d}{dt} R_{22} + \ddot{\phi}_1 + R_{23} \ddot{\theta}_2 \right)
\]

\[
+ u_4 \left( \theta_2 \frac{d}{dt} R_{24} + \ddot{\phi}_1 + R_{24} \ddot{\theta}_2 \right)
\]

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \phi_2} \right) = \sum_{i=2}^{4} m_i \left[ x_i \frac{\partial \dot{X}_i}{\partial \phi_2} + \ddot{X}_i \frac{d}{dt} \left( \frac{\partial \dot{X}_i}{\partial \phi_2} \right) + \dot{Y}_i \frac{\partial Y_i}{\partial \phi_2} + \dot{Y}_i \frac{d}{dt} \left( \frac{\partial Y_i}{\partial \phi_2} \right) \right] + u_2 (\ddot{\phi}_1 + \ddot{\phi}_2) + u_3 \left( \theta_2 \frac{d}{dt} R_{23} + \ddot{\phi}_1 + R_{23} \ddot{\theta}_2 \right) \frac{d}{dt} R_{23}
\]

\[
+ u_4 \left[ \left( \theta_2 \frac{d}{dt} R_{24} + \ddot{\phi}_1 + R_{24} \ddot{\theta}_2 \right) \frac{d}{dt} R_{24} \right]
\]

\[
\frac{\partial L}{\partial x_1} = 0
\]

\[
\frac{\partial L}{\partial Y_1} = -g \left( m_1 + m_2 \frac{\partial Y_2}{\partial Y_1} + m_3 \frac{\partial Y_3}{\partial Y_1} + m_4 \frac{\partial Y_4}{\partial Y_1} \right)
\]

\[
\frac{\partial L}{\partial \phi_1} = \sum_{i=2}^{4} m_i \left[ X_i \frac{\partial \dot{X}_i}{\partial \phi_1} + \dot{Y}_i \frac{\partial Y_i}{\partial \phi_1} - g \frac{\partial Y_i}{\partial \phi_1} \right]
\]
3.5.2 Terms Appearing in Equations of Motion

In reference to Fig. 3.5.1, given the global position of the mass center of floating frame, \((X_1, Y_1, \phi_1)\), we shall use Eq. (2.49) to obtain other global positions of mass centers, i.e.,

\[
\begin{bmatrix}
X_2 \\
Y_2
\end{bmatrix} = \mathbf{r} + \mathbf{A} \begin{bmatrix}
\frac{1}{2} \left(-r_1 + r_2 \cos \theta_2\right) \\
\frac{1}{2} r_2 \sin \theta_2
\end{bmatrix}
\]  

(3.61a)

\[
\begin{bmatrix}
X_3 \\
Y_3
\end{bmatrix} = \mathbf{r} + \mathbf{A} \begin{bmatrix}
-\frac{r_1}{2} + r_2 \cos \theta_2 + \frac{r_3}{2} \cos \theta_3 \\
r_2 \sin \theta_2 + \frac{r_3}{2} \sin \theta_3
\end{bmatrix}
\]  

(3.61b)

\[
\begin{bmatrix}
X_4 \\
Y_4
\end{bmatrix} = \mathbf{r} + \mathbf{A} \begin{bmatrix}
\frac{r_1}{2} + \frac{r_3}{2} \cos \phi_4 \\
\frac{r_4}{2} \sin \phi_4
\end{bmatrix}
\]  

(3.61c)

where
and angles $\theta_2$, $\theta_3$ and $\theta_4$ are obtained from the explicit or simultaneous equations in kinematic analysis.

Differentiating Eqs. (3.61) with respect to time, we obtain the velocities:

\[ \dot{X}_2 = \dot{X}_1 + \frac{1}{2} [r_1 \sin \phi_1 \dot{\phi}_1 - r_2 \sin(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_2)] \]  
(3.62a)

\[ \dot{Y}_2 = \dot{Y}_1 + \frac{1}{2} [-r_1 \cos \phi_1 \dot{\phi}_1 + r_2 \cos(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_2)] \]  
(3.62b)

\[ \dot{X}_3 = \dot{X}_1 + \frac{1}{2} r_1 \sin \phi_1 \dot{\phi}_1 - r_2 \sin(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_2) \]
\[ \quad - \frac{1}{2} r_3 \sin(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_3) \]  
(3.62c)

\[ \dot{Y}_3 = \dot{Y}_1 - \frac{1}{2} r_1 \cos \phi_1 \dot{\phi}_1 + r_2 \cos(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_2) \]
\[ \quad + \frac{1}{2} r_3 \cos(\phi_1 + \theta_2)(\dot{\phi}_1 + \dot{\phi}_3) \]  
(3.62d)

\[ \dot{X}_4 = \dot{X}_1 + \frac{1}{2} [-r_1 \sin \phi_1 \dot{\phi}_1 - r_4 \sin(\phi_1 + \theta_4)(\dot{\phi}_1 + \dot{\phi}_4)] \]  
(3.62e)

\[ \dot{Y}_4 = \dot{Y}_1 + \frac{1}{2} [r_1 \cos \phi_1 \dot{\phi}_1 + r_4 \cos(\phi_1 + \theta_4)(\dot{\phi}_1 + \dot{\phi}_4)] \]  
(3.62f)

Before calculating the accelerations, we first define some notations. Differentiate Eqs. (3.29) with respect to time, i.e.,
\begin{align}
\ddot{\theta}_3 &= R_{23} \ddot{\theta}_2 + \dot{\theta}_2 \frac{d}{dt} R_{23} \\
\ddot{\theta}_4 &= R_{24} \ddot{\theta}_2 + \dot{\theta}_2 \frac{d}{dt} R_{24}
\end{align}

and define:

\begin{align}
\frac{d}{dt} R_{23} &= TR_{23} \\
&= \frac{[r_2 \sin(\theta_4 - \theta_3) \cos(\theta_2 - \theta_4)(\dot{\theta}_2 - \dot{\theta}_4) - r_2 \sin(\theta_2 - \theta_3) \cos(\theta_4 - \theta_3)(\dot{\theta}_4 - \dot{\theta}_3)]}{r_3 \sin^2(\theta_4 - \theta_3)}
\end{align}

\begin{align}
\frac{d}{dt} R_{24} &= TR_{24} \\
&= \frac{[r_2 \sin(\theta_4 - \theta_3) \cos(\theta_2 - \theta_4)(\dot{\theta}_2 - \dot{\theta}_4) - r_2 \sin(\theta_2 - \theta_3) \cos(\theta_4 - \theta_3)(\dot{\theta}_4 - \dot{\theta}_3)]}{r_4 \sin^2(\theta_4 - \theta_3)}
\end{align}

Once again, we differentiate Eqs. (3.62) with respect to time to obtain the accelerations:

\begin{align}
\ddot{X}_2 &= \dot{X}_1 + DDX_2 T_1 \ddot{\phi}_1 + DDX_2 T_2 \ddot{\theta}_2 + DDX_2 T \\
\ddot{Y}_2 &= \dot{Y}_1 + DDY_2 T_1 \ddot{\phi}_1 + DDY_2 T_2 \ddot{\theta}_2 + DDY_2 T \\
\ddot{X}_3 &= \dot{X}_1 + DDX_3 T_1 \ddot{\phi}_1 + DDX_3 T_2 \ddot{\theta}_2 + DDX_3 T \\
\ddot{Y}_3 &= \dot{Y}_1 + DDY_3 T_1 \ddot{\phi}_1 + DDY_3 T_2 \ddot{\theta}_2 + DDY_3 T \\
\ddot{X}_4 &= \dot{X}_1 + DDX_4 T_1 \ddot{\phi}_1 + DDX_4 T_2 \ddot{\theta}_2 + DDX_4 T \\
\ddot{Y}_4 &= \dot{Y}_1 + DDY_4 T_1 \ddot{\phi}_1 + DDY_4 T_2 \ddot{\theta}_2 + DDY_4 T
\end{align}

where

\begin{align}
DDX_1 &= \frac{1}{2} (r_1 \sin \phi_1 - r_2 \sin \phi_2) \\
DDX_2 &= -\frac{1}{2} r_2 \sin \phi_2
\end{align}
Next, we differentiate Eqs. (3.62) with respect to $\dot{\phi}_1$ and $\dot{\phi}_2$ to obtain some identities and then define the following notations:
Taking derivatives of Eqs. (3.66) with respect to time and defining the notations, we have

\[
\frac{d}{dt} \left( \frac{\partial X_1}{\partial \phi_1} \right) = \frac{\partial X_1}{\partial \phi_1} = D_2 DT_1 = DDX_1 T_1 \quad \frac{\partial Y_1}{\partial \phi_1} = D_2 DT_1 = DDY_1 T_1 \quad (3.66a)
\]

\[
\frac{d}{dt} \left( \frac{\partial X_2}{\partial \phi_1} \right) = \frac{\partial X_2}{\partial \phi_1} = D_2 DT_1 = DDX_1 T_1 \quad \frac{\partial Y_2}{\partial \phi_1} = D_2 DT_1 = DDY_1 T_1 \quad (3.66b)
\]

\[
\frac{d}{dt} \left( \frac{\partial X_3}{\partial \phi_2} \right) = \frac{\partial X_3}{\partial \phi_2} = D_2 DT_1 = DDX_2 T_2 \quad \frac{\partial Y_3}{\partial \phi_2} = D_2 DT_1 = DDY_2 T_2 \quad (3.66c)
\]

\[
\frac{d}{dt} \left( \frac{\partial X_4}{\partial \phi_1} \right) = \frac{\partial X_4}{\partial \phi_1} = D_2 DT_1 = DDX_4 T_1 \quad \frac{\partial Y_4}{\partial \phi_1} = D_2 DT_1 = DDY_4 T_1 \quad (3.66d)
\]

\[
\frac{d}{dt} \left( \frac{\partial X_2}{\partial \phi_2} \right) = \frac{\partial X_2}{\partial \phi_2} = D_2 DT_1 = DDX_2 T_2 \quad \frac{\partial Y_2}{\partial \phi_2} = D_2 DT_1 = DDY_2 T_2 \quad (3.66e)
\]

\[
\frac{d}{dt} \left( \frac{\partial X_3}{\partial \phi_2} \right) = \frac{\partial X_3}{\partial \phi_2} = D_2 DT_1 = DDX_3 T_2 \quad \frac{\partial Y_3}{\partial \phi_2} = D_2 DT_1 = DDY_3 T_2 \quad (3.66f)
\]
\[
\frac{d}{dt} \left( \frac{\partial Y_4}{\partial \phi_1} \right) = T D Y_4 D T_1 = \frac{1}{2} (-r_1 \sin \phi_1 \dot{\phi}_1 - r_2 \sin \phi_4 \dot{\phi}_4)
\] (3.67f)

\[
\frac{d}{dt} \left( \frac{\partial X_4}{\partial \phi_2} \right) = T D X_4 D T_2 = -\frac{1}{2} r_2 \cos \phi_2 \dot{\phi}_2
\] (3.67g)

\[
\frac{d}{dt} \left( \frac{\partial Y_4}{\partial \phi_2} \right) = T D Y_4 D T_2 = -\frac{1}{2} r_1 \sin \phi_2 \dot{\phi}_2
\] (3.67h)

\[
\frac{d}{dt} \left( \frac{\partial X_4}{\partial \phi_2} \right) = T D X_4 D T_2 = \frac{1}{2} (-2r_2 \cos \phi_2 \dot{\phi}_2 - r_3 R_{23} \cos \phi_3 \dot{\phi}_3 - r_3 TR_{23} \sin \phi_3)
\] (3.67i)

\[
\frac{d}{dt} \left( \frac{\partial Y_4}{\partial \phi_2} \right) = T D Y_4 D T_2 = \frac{1}{2} (-2r_2 \sin \phi_2 \dot{\phi}_2 - r_3 R_{23} \sin \phi_3 \dot{\phi}_3 + r_3 TR_{23} \cos \phi_2)
\] (3.67j)

\[
\frac{d}{dt} \left( \frac{\partial X_4}{\partial \phi_2} \right) = T D X_4 D T_2 = \frac{1}{2} (-r_4 R_{24} \cos \phi_4 \dot{\phi}_4 - r_4 TR_{24} \sin \phi_4)
\] (3.67k)

\[
\frac{d}{dt} \left( \frac{\partial Y_4}{\partial \phi_2} \right) = T D Y_4 D T_2 = \frac{1}{2} (-r_4 R_{24} \sin \phi_4 \dot{\phi}_4 + r_4 TR_{24} \cos \phi_4)
\] (3.67l)

Before we proceed any further, we define two notations, and then find two useful identities. Differentiating Eqs. (2.38) with respect to \( \phi_2 \) and using Eqs. (3.29), we get:

\[
\dot{\phi}_2 = \dot{\phi}_2 \frac{\partial}{\partial \phi_2} R_{23}
\] (3.68a)

\[
\dot{\phi}_4 = \dot{\phi}_2 \frac{\partial}{\partial \phi_2} R_{24}
\] (3.68b)

Now define

\[
\frac{\partial}{\partial \phi_2} R_{24} \equiv PR_{23}
\] (3.69a)
\[ \frac{\partial}{\partial \theta_2} R_{24} \equiv PR_{24}. \] (3.69b)

Substituting Eqs. (3.29) into Eqs. (3.68), then expanding its right-hand side and using Eqs. (3.37) and (3.64), we arrive at two identities:

\[ TR_{23} = \dot{\theta}_2 \; PR_{23} = \frac{\partial \phi}{\partial \theta_2} \] (3.70a)

\[ TR_{24} = \dot{\theta}_2 \; PR_{24} = \frac{\partial \phi}{\partial \theta_2} \] (3.70b)

Now, making use of the fact that \( \frac{\partial \theta_1}{\partial \theta_2} = \frac{\partial \phi_1}{\partial \theta_2} = \frac{\partial \phi_2}{\partial \theta_2} = 0 \), we differentiate Eqs. (3.62) with respect to \( \phi_1 \) and \( \theta_2 \) to obtain the following terms after comparison with Eqs. (3.67) has been made and some notations have been defined.

\[ \frac{\partial \dot{X}_1}{\partial \phi_1} = DX_1 T_1 = TDX_2 DT_1, \quad \frac{\partial \dot{Y}_1}{\partial \phi_1} = DY_1 T_1 = TDY_2 DT_1 \] (3.71a)

\[ \frac{\partial \dot{X}_3}{\partial \phi_1} = DX_3 T_1 = TDX_3 DT_1, \quad \frac{\partial \dot{Y}_3}{\partial \phi_1} = DY_3 T_1 = TDY_3 DT_1 \] (3.71b)

\[ \frac{\partial \dot{X}_4}{\partial \phi_1} = DX_4 T_1 = TDX_4 DT_1, \quad \frac{\partial \dot{Y}_4}{\partial \phi_1} = DY_4 T_1 = TDY_4 DT_1 \] (3.71c)

\[ \frac{\partial \dot{X}_2}{\partial \theta_2} = DX_2 T_2 = TDX_2 DT_2, \quad \frac{\partial \dot{Y}_2}{\partial \theta_2} = DY_2 T_2 = TDY_2 DT_2 \] (3.71d)

\[ \frac{\partial \dot{X}_3}{\partial \theta_2} = DX_3 T_2 = TDX_3 DT_2, \quad \frac{\partial \dot{Y}_3}{\partial \theta_2} = DY_3 T_2 = TDY_3 DT_2 \] (3.71e)

\[ \frac{\partial \dot{X}_4}{\partial \theta_2} = DX_4 T_2 = TDX_4 DT_2, \quad \frac{\partial \dot{Y}_4}{\partial \theta_2} = DY_4 T_2 = TDY_4 DT_2. \] (3.71f)
Next, taking derivatives of Eqs. (3.61) with respect to $\phi_1$ and $\theta_2$, we have:

\[
\begin{align*}
\frac{\partial Y_1}{\partial \phi_1} &= Y_1 T_1 = \frac{1}{2} (-r_1 \cos \phi_1 + r_2 \cos \phi_2) \\
\frac{\partial Y_2}{\partial \phi_1} &= Y_2 T_1 = \frac{1}{2} (-r_1 \cos \phi_1 + 2r_2 \cos \phi_2 + r_3 \cos \phi_3) \\
\frac{\partial Y_3}{\partial \phi_1} &= Y_3 T_1 = \frac{1}{2} (r_1 \cos \phi_1 + r_4 \cos \phi_4) \\
\frac{\partial Y_4}{\partial \phi_1} &= Y_4 T_1 = \frac{1}{2} r_3 R_{23} \cos \phi_3 \\
\frac{\partial Y_1}{\partial \theta_2} &= Y_1 T_2 = \frac{1}{2} r_2 \cos \theta_2 \\
\frac{\partial Y_2}{\partial \theta_2} &= Y_2 T_2 = r_2 \cos \theta_2 + \frac{1}{2} r_3 R_{23} \cos \phi_3 \\
\frac{\partial Y_3}{\partial \theta_2} &= Y_3 T_2 = \frac{1}{2} r_4 R_{24} \cos \phi_4.
\end{align*}
\]

Thus, substituting Eqs. (3.62)-(3.72) into Eqs. (3.60), we subsequently obtain the following four equations of motion:

\[
\begin{align*}
AX \ddot{X}_1 + AY \ddot{Y}_1 + AT \ddot{\phi}_1 + AQ \ddot{\theta}_2 &= AF \\
BX \ddot{X}_1 + BY \ddot{Y}_1 + BT \ddot{\phi}_1 + BQ \ddot{\theta}_2 &= BF \\
CX \ddot{X}_1 + CY \ddot{Y}_1 + CT \ddot{\phi}_1 + CQ \ddot{\theta}_2 &= CF \\
DX \ddot{X}_1 + DY \ddot{Y}_1 + DT \ddot{\phi}_1 + DQ \ddot{\theta}_2 &= DF
\end{align*}
\]

where

\[
\begin{align*}
AX &= DY = m_1 + m_2 + m_3 + m_4 \\
AY &= DX = 0 \\
AT &= BX = m_2 \cdot DDX_2 T_1 + m_3 \cdot DDX_3 T_1 + m_4 \cdot DDX_4 T_1 \\
AQ &= CX = m_2 \cdot DDX_2 T_2 + m_3 \cdot DDX_3 T_2 + m_4 \cdot DDX_4 T_2 \\
AF &= -m_2 \cdot DDX_2 T - m_3 \cdot DDX_3 T - m_4 \cdot DDX_4 T + QX_1
\end{align*}
\]
BY = DT = m_2 \cdot \text{DDY}_2T_1 + m_3 \cdot \text{DDY}_3T_1 + m_4 \cdot \text{DDY}_4T_1

BT = m_2 ((\text{DDX}_2T_2)^2 + (\text{DDY}_2T_2)^2) + m_3 ((\text{DDX}_3T_2)^2 + (\text{DDY}_3T_2)^2)
+ m_4 ((\text{DDX}_4T_2)^2 + (\text{DDY}_4T_2)^2) + u_2 + u_3 + u_4

BQ = CT

= m_2(\text{DDX}_2T_1 \cdot \text{DDY}_2T_2 + \text{DDX}_2T_1 \cdot \text{DDY}_2T_2)
+ m_3(\text{DDX}_3T_1 \cdot \text{DDY}_3T_2 + \text{DDY}_3T_1 \cdot \text{DDY}_3T_2)
+ m_4(\text{DDX}_4T_1 \cdot \text{DDY}_4T_2 + \text{DDY}_4T_1 \cdot \text{DDY}_4T_2)
+ u_2 + u_3 \cdot R_{23} + u_4 \cdot R_{24}

BF = - m_2(\text{DDX}_2T_1 \cdot \text{DDX}_2T + \text{DDY}_2T_1 \cdot \text{DDY}_2T)
- m_3(\text{DDX}_3T_1 \cdot \text{DDX}_3T + \text{DDY}_3T_1 \cdot \text{DDY}_3T)
- m_4(\text{DDX}_4T_1 \cdot \text{DDX}_4T + \text{DDY}_4T_1 \cdot \text{DDY}_4T)
- (u_3 \cdot TR_{23} + u_4 \cdot TR_{24}) \dot{\theta}_2
- g(m_2 \cdot Y_2T_1 + m_3 \cdot Y_3T_1 + m_4 \cdot Y_4T_1) + Q \phi_1

CY = DQ = m_2 \cdot \text{DDY}_2T_2 + m_3 \cdot \text{DDY}_3T_2 + m_4 \cdot \text{DDY}_4T_2

CQ = m_2 ((\text{DDX}_2T_2)^2 + (\text{DDY}_2T_2)^2) + m_3 ((\text{DDX}_3T_2)^2 + (\text{DDY}_3T_2)^2)
+ m_4 ((\text{DDX}_4T_2)^2 + (\text{DDY}_4T_2)^2) + u_2 + u_3 \cdot R_{23}^2 + u_4 \cdot R_{24}^2

CF = - m_2(\text{DDX}_2T_2 \cdot \text{DDX}_2T + \text{DDY}_2T_2 \cdot \text{DDY}_2T)
- m_3(\text{DDX}_3T_2 \cdot \text{DDX}_3T + \text{DDY}_3T_2 \cdot \text{DDY}_3T)
- m_4(\text{DDX}_4T_2 \cdot \text{DDX}_4T + \text{DDY}_4T_2 \cdot \text{DDY}_4T)
- u_3 \cdot TR_{23} \cdot \dot{\theta}_3 - u_4 \cdot TR_{24} \cdot \dot{\theta}_4
- g(m_2 \cdot Y_2T_2 + m_3 \cdot Y_3T_2 + m_4 \cdot Y_4T_2) + Q \phi_2

DF = - m_2 \cdot \text{DDY}_2T - m_3 \cdot \text{DDY}_3T - m_4 \cdot \text{DDY}_4T - g \cdot \text{DY} + Q \phi_1

In these equations, the generalized force Q is determined by the same procedure as that described in Section 3.3.3. By using Eq. (3.40), the power of active forces is:
\[ P = \sum_{i=1}^{4} (F_{Xi} \dot{x}_i + F_{Yi} \dot{y}_i + T_{\phi_i} \dot{\phi}_i). \]  

(3.74)

Substituting Eqs. (3.62) into Eq. (3.74), and using Eqs. (3.29), we obtain the power of active forces in the form:

\[ P = QX_1 \dot{x}_1 + QY_1 \dot{y}_1 + Q_{\phi_1} \dot{\phi}_1 + Q_{\theta_2} \dot{\theta}_2 \]  

(3.75)

where

\[ QX_1 = \sum_{i=1}^{4} F_{Xi} \]

\[ QY_1 = \sum_{i=1}^{4} F_{Yi} \]

\[ Q_{\phi_1} = \frac{1}{2} F_{X_2} (r_1 \sin \phi_1 - r_2 \sin \phi_2) \]
\[ + \frac{1}{2} F_{X_3} (r_1 \sin \phi_3 - 2r_2 \sin \phi_2 - r_2 \sin \phi_3) \]
\[ + \frac{1}{2} F_{X_4} (-r_1 \sin \phi_1 + r_2 \sin \phi_2) + \frac{1}{2} F_{Y_2} (-r_2 \cos \phi_1 + r_2 \cos \phi_2) \]
\[ + \frac{1}{2} F_{Y_3} (-r_1 \cos \phi_1 + 2r_2 \cos \phi_2 + r_2 \cos \phi_3) \]
\[ + \frac{1}{2} F_{Y_4} (r_1 \cos \phi_2 + r_4 \cos \phi_4) + \sum_{i=1}^{4} T_{\theta_i} \]

\[ Q_{\theta_2} = F_{X_2} \left[ - \frac{1}{2} r_2 \sin \phi_2 \right] + F_{X_3} \left[ -r_2 \sin \phi_2 - \frac{1}{2} r_3 R_{23} \sin \phi_3 \right] \]
\[ + F_{X_4} \left[ - \frac{1}{2} r_4 R_{24} \sin \phi_4 \right] + F_{Y_2} \left[ \frac{1}{2} r_2 \cos \phi_2 \right] \]
\[ + F_{Y_3} \left[ r_2 \cos \phi_2 + \frac{1}{2} r_3 R_{23} \cos \phi_3 \right] \]
\[ + F_{Y_4} \left[ \frac{1}{2} r_4 R_{24} \cos \phi_4 \right] + T_{\theta_2} + T_{\theta_3} R_{23} + T_{\theta_4} R_{24} \]
Substituting the results of Eqs. (3.75) into Eqs. (3.73), we then have obtained the equations of motion for a floating element in a form suitable for numerical computation. Next, we use the Lagrange multiplier technique to deal with two elements with a moving frame.

3.5.3 Two Elements with a Moving Frame

In order to apply the Lagrange multiplier technique, we need to find the constraint equations between the two elements. Referring to Fig. 2.4.1, we can imaginatively separate the rigid body, block BCDE, into two parts, as is shown in Fig. 3.5.2, such that one part becomes the floating frame of the upper element and the other part belongs to the coupler of the lower element. So, each part corresponds with its own inertia properties, i.e., the moment of inertia and mass. Therefore, the system includes a floating frame with four equations of motion and a fixed frame with one generalized equation of motion. In accordance with the criteria, \( N_f = N_c - N_h \), we need three additional constraint equations because the system has only two degrees of freedom.

It may be assumed that we have put a floating frame on the coupler of the lower element; then we develop the first constraint between \( \theta_2 \) and \( X_1^u \) to constrain the motion of moving frame in the \( X \)-direction. In a similar manner, we develop the second constraint between \( \theta_2 \) and \( Y_1^u \) to constrain the motion in the \( Y \)-direction. The third constraint is between angles \( \theta_2 \) and \( \phi_1^u \) to constrain the floating frame and the coupler with the same rotation.

By using Eq. (3.61b), the mass center of the coupler is:

\[
\begin{bmatrix}
X_3 \\
Y_3
\end{bmatrix} = \begin{bmatrix}
X_1 \\
Y_1
\end{bmatrix} + \begin{bmatrix}
\cos \phi_1 & -\sin \phi_1 \\
\sin \phi_1 & \cos \phi_1
\end{bmatrix} \begin{bmatrix}
-\frac{1}{2} r_1 + r_2 \cos \theta_2 + \frac{1}{2} r_3 \cos \phi_2 \\
r_2 \sin \theta_2 + \frac{1}{2} r_3 \sin \phi_2
\end{bmatrix}.
\]

(3.76)
Figure 3.5.2 A moving frame connected to an element with fixed frame.
Making use of the fact that $X_3 = X^u_1$ and $Y_3 = Y^u_1$, we obtain two constraint equations from Eq. (3.76) as follows:

$$\Phi_1 = X_1 - \frac{1}{2} r_1 \cos \phi_1 + r_2 \cos (\phi_1 + \theta_2) + \frac{1}{2} r_3 \cos (\phi_1 + \theta_3) - X^u_1 \quad (3.77a)$$

$$\Phi_2 = Y_1 - \frac{1}{2} r_1 \sin \phi_1 + r_2 \sin (\phi_1 + \theta_2) + \frac{1}{2} r_3 \sin (\phi_1 + \theta_3) - Y^u_1 \quad (3.77b)$$

Similarly, using Eq. (2.4b) and the fact that $\phi^u_1 = \phi_1 + \theta_3$, we obtain the third equation:

$$\Phi_3 = r_1^2 + r_2^2 + r_3^2 - r_4^2 - 2r_1 r_2 \cos \theta_2 - 2r_1 r_3 \cos (\phi^u_1 + \phi_1) + 2r_2 r_3 \cos (\phi_1 + \phi_2 - \phi^u_1) \quad (3.77c)$$

Differentiating Eqs. (3.77) twice with respect to time and using Eqs. (3.65), we obtain the constraint differential equations in the form:

$$\Phi_1 \dddot{X}^u_1 + \Phi_2 \dddot{Y}^u_1 + \Phi_3 \dddot{\phi}^u_1 + \Phi_4 \dddot{\theta}_2 = \gamma_1 \quad (3.78a)$$

$$\Phi_2 \dddot{\phi}^u_1 + \Phi_3 \dddot{\theta}_2 = \gamma_2 \quad (3.78b)$$

$$\Phi_3 \dddot{Y}^u_1 + \Phi_5 \dddot{\phi}_2 = \gamma_3 \quad (3.78c)$$

where

$$\Phi_1 X^u_1 = -1 \quad , \quad \Phi_1 \theta_2 = \frac{\partial X}{\partial \theta_2} T_2$$

$$\Phi_2 \phi^u_1 = r_2 r_3 \sin (\theta_2 - \theta_3) + r_1 r_3 \sin \theta_2$$

$$\Phi_2 \theta_2 = r_1 r_2 \sin \theta_2 - r_2 r_3 \sin (\theta_2 - \theta_3)$$

$$\Phi_3 Y^u_1 = -1 \quad , \quad \Phi_3 \theta_2 = \frac{\partial Y}{\partial \theta_2} T_2$$

$$\gamma_1 = -\dddot{X}^u_1 T_2 - \dddot{X}^u_1 T_1 \phi_1 - \dddot{X}^u_1 T_1$$
If the frame of the lower element is fixed, then \( \ddot{x}_1 = \ddot{y}_1 = \ddot{\phi}_1 = 0 \).

Finally, we combine Eqs. (3.5), (3.73) and (3.78) to obtain the result as follows:

\[
\begin{bmatrix}
A \chi \\
B \chi & B T & \text{symmetric} \\
C \chi & C T & C Q \\
0 & D T & D Q & D Y \\
0 & 0 & 0 & 0 & J \\
\Phi_1 \chi & 0 & 0 & 0 & \Phi_1 \theta_2 & 0 \\
0 & \Phi_2 \phi_1 & 0 & 0 & \Phi_2 \theta_2 & 0 & 0 \\
0 & 0 & 0 & 0 & \Phi_3 Y_1 & \Phi_3 \theta_2 & 0 & 0 & 0
\end{bmatrix}
= \begin{bmatrix}
\dddot{X}_1^u \\
\dddot{\phi}_1^u \\
\dddot{\theta}_2^u \\
\dddot{Y}_1^u \\
\dddot{\theta}_2 \\
-\lambda_1 \\
-\lambda_2 \\
-\lambda_3
\end{bmatrix}
\begin{bmatrix}
AF \\
BF \\
CF \\
DF \\
QF \\
\gamma_1 \\
\gamma_2 \\
\gamma_3
\end{bmatrix}
\]

(3.79)

where constraint equations \( \Phi \)'s are obtained from the lower element, and

\[
Q F = Q_{\theta_2} - \frac{1}{2} \frac{dI}{d\theta_2} \dddot{\theta}_2^2 .
\]
In view of the fact that there are two degrees of freedom in this system, i.e., angles \( \theta_2 \) and \( \theta_u^2 \), we need to specify the initial conditions of upper and lower input links. Hence, the angular accelerations of both input links are determined by Eq. (3.79). Next, we apply the L-U factorization method on Eq. (3.79) to obtain the solution, and then use the RUNGE-KUTTA algorithm to integrate \( \ddot{\theta}_2 \) and \( \ddot{\theta}_u^2 \). Finally, we use these results to calculate other accelerations, velocities and positions by using kinematic analysis.

If there is one additional moving frame connected to the coupler of any element, the system will have its degrees of freedom increased by one. At the same time, four additional equations of motion and three constraint equations are needed for the solutions. Thus, the mass matrix of Eq. (3.79) will become very large. For the solutions of these equations, several numerical methods with algorithms are available in [1] and other textbooks on numerical analysis.
CHAPTER 4

APPLICATIONS AND PROGRAMMING ASPECTS

The four-bar mechanism is one of the simplest closed-loop mechanisms with a single degree of freedom after the frame is fixed. More complex mechanisms can be developed by connecting different numbers of elements together and specifying the geometric and inertial properties, e.g., link lengths, mass, etc. Because of those properties, a wide variety of motions can be generated by four-bar mechanisms. Several common applications of such chains of four-bar linkages are often found in the heart of machines and subsystems, such as function generators, steam locomotives, semi-automatic power looms, dough-kneaders, web cutters, punch presses, film transports, quick returns, analog computers, wheel-suspension systems, solar panels of satellites, and so on [1,3].

Because there are so many application possibilities for this mechanism, versatile computer programs are utilized to do the kinematic and dynamic analyses. These special-purpose programs are rigidly structured computer codes written only for dealing with four-bar mechanisms. Formulation of the governing equations for each of these applications has been derived in the previous chapters, and computer programs are then written for solving them. Since these programs have a special-purpose property, they are often more efficient in numerical computation than the general-purpose programs. Engineering data describing the mechanical systems, such as the dimensions and physical characteristics of each part, are input into the program; then it will automatically solve the governing equations of motion. Once the sequential simulation data which indicate the positions of all bodies per time increment are transferred into animated graphics, the animated images provide an extremely effective medium for communication between designer and computer.
In the following section, we will introduce the kinematic and dynamic computer programs and show how to use them; then, as an interesting example of the application of these programs, we will discuss the unfolding of the solar panels of a satellite.

4.1 Introduction to Kinematic and Dynamic Computer Programs

The kinematic and dynamic computer programs are developed by using explicit and simultaneous equations, respectively. Here, only the programs which use explicit equations are described and listed in Appendices A and B. Other programs which use simultaneous equations are employed only for a comparison of different approaches. All of the programs have a common organization and are described in the following.

The programs are written in FORTRAN 77 with double precision. The current version of the programs can handle a total of five branches, and each branch can consist of five linkages. To handle more data, the dimension statements must be modified slightly. The dynamic analysis program can automatically handle a system which includes a chain of two linkages developed horizontally or vertically. But for more linkages in the system, the entities of the mass matrix in the programs need to be rearranged by the user. Additionally, the user, by using the FORTRAN expressions, can specify applied forces and moments which may be arbitrary functions of positions and velocities. These applied forces are those terms appearing in the equations of motion and they have been discussed in the previous sections. The expressions of the active forces for rotational spring and damper will be shown in the next sections.

The flow charts for the computer programs, the description of each subroutine and the input files will enable users to easily understand and use the computer programs. All of these are described in the following sections.
4.1.1 Program Organization of Kinematic Analysis

The kinematic analysis program is organized in such a way that input of data is made easy for the users. It contains a main program and six subroutines. A flow chart of the algorithm is shown in Fig. 4.1.1. First, an algorithm is developed to analyze only one linkage mechanism. Then, other functions such as different time steps, provisions for multi-branches and moving frames are gradually added. To satisfy the need for a kinematic analysis as a function of time and multiple branches of linkages, it is necessary to use 2-D arrays in the program. Do-loops are incorporated into the code to vary time, provide for additional branches and cater for the number of linkages in a branch. The main program and subroutines are described in detail in what follows:

**Program MAIN:** This program serves as the link between the user, the data input and other calling subroutines. It starts with a few interactive questions, requesting the user to specify the names of input and output files of the user's choice. It then opens the files accordingly and calls the subroutine DINPUT. Then the program starts by reading the data from the file specified by the user in a free format and a specific sequence.

**Subroutine CPUTM:** This routine reads the Central Processing Unit (CPU) clock time once before and once after the execution of the program. The difference in time gives a measure of the CPU time used.

**Subroutine DINPUT:** This subroutine reads the required input data from the user's specified file and stores it in the memory. The number of branches, the number of linkages in each branch, the link lengths, the orientations of frames and initial positions, velocities and accelerations of input links are read. The start time, end time and the time increment are also read. The descriptions of an input file are as follows:
Main Program

Start

Call CPUTM

Input Data

TIME = T₀
NB = 1
NL = 1

Yes

Check the Grashof's Law

Error

No

Kinematic Analysis

More Data/Time

No

Output Data

Call CPUTM

Stop

Figure 4.1.1 The flow chart of kinematic analysis program.
Data Input

1. Number of branches in a system.

2. Connectivity details ... (Branch 1),
   e.g. IBN, ICV, ILN, ITYPE, ILINK, ISIDE
   where all variables are integers as defined in the following:
   IBN = The i-th branch which will be added into a system.
   ICV = The j-th branch which will be connected to the i-th branch.
   ILN = The k-th linkage of the j-th branch where the i-th branch is connected.
   ITYPE = A flag to show the configuration of the linkages of the i-th branch,
   EQ. 1; the linkages have leading form.
   EQ. 2; the linkages have lagging form.
   EQ. 3; the linkages have lengths, \( r_1 = r_3 \) and \( r_2 = r_4 \).
   ILINK = A link number of the k-th linkage where the i-th branch is connected,
   EQ. 2; the input link.
   EQ. 3; the coupler.
   EQ. 4; the output link.
   ISIDE = A flag to show the direction of the transmission of the i-th branch,
   EQ. 1; a transmission from right to left.
   EQ. 2; a transmission from left to right.

3. Number of linkages in the i-th branch.
The global coordinates of the fixed frame \( \ldots \) (linkage 1),
i.e. \((X^A, Y^A)\) and \((X^F, Y^F)\) as shown in Fig. 2.4.1.

Lengths of other links and adjacent angle

i.e., \(r_2, r_3, r_4, \beta\)

where the adjacent angle \(\beta\) is always measured with respect to the output link of connected linkage.

Initial global angle, angular velocity and acceleration of the input link.

Start time, finish time and time increment.

Repeating line 4 and 5, if more linkages exist.

The same as in line 2 \(\ldots\) (branch 2).

The same as in line 3.

If the frame is fixed, then input the same as in line 4 \(\ldots\) (linkage 1).

If the frame is moving, then input the local coordinates,
i.e. \((\zeta^C, \eta^C)\) and \((\xi^D, \eta^D)\)

where both local coordinates are with origins at point B and E, respectively, as shown in Fig. 2.4.1.

IDRIVER, initial global angle, angular velocity and acceleration of the input link, where IDRIVER is a flag as follows:

EQ. 0; the angular velocity and acceleration are measured with respect to the global axis.

EQ. 1; the angular velocity and acceleration are measured with respect to the moving frame.
EQ. 2; the driving link is connected to the output link of other linkage which is specified by the user. In this case, the initial angle, angular velocity and acceleration depend on those of other output link, such that we can give arbitrary values to them. The program will correct them internally.

K+4 The linkage and branch numbers, if the flag IDRIVER=2.
K+5 The same as in line 5.

Repeating line K+2, ..., K+5, if more linkages exist.
Repeating line K, ..., t, if more branches exist.

m Printing choice for positions, velocities and accelerations,
e.g., 1, 1, 1, if print all of them
0, 0, 0, if don’t print all of them.

Once the data have been read in, the program MAIN takes over and calls another routine KINEMAT for the kinematic analysis of the given problem.

Subroutine KINEMAT: Here, the main analysis of the program is done. The time step is the outer most loop and it is varied from start time to end time as specified by the user. The number of branches is the next inner loop which is varied accordingly. The number of linkages in each branch is the inner most loop in the program. So, for only one branch and one linkage, the calculations are done only once.

In the first time step only, in order to check whether the driving linkage satisfies Grashof’s law, this subroutine calls another routine, DCHECK, to check the link lengths of
the driving linkage. The subsequent linkages do not have to satisfy Grashof's law. In some applications, if the driving link does not need a full cycle rotation, then the user can delete the DCHECK routine.

For each time step, all the elements of the respective branches are analyzed for positions, velocities and accelerations. After computation of one linkage of a branch, a calling routine calls the subroutine DOUTPUT and writes the requisite values in the output file.

**Subroutine DCHECK:** As stated earlier, this subroutine checks the data for input link. If the data do not satisfy Grashof’s law, this routine calls another routine ERROR and prints a message on the screen, indicating the linkage number and corresponding branch. If the data are correct, it returns to the routine KINEMAT which continues to perform the kinematic analysis.

**Subroutine DOUTPUT:** The main function of this routine is reflected in its name. It writes the output in the user specified file which is then stored in the user area and can also be printed out.

It should be noted that for both input and output, the units used are consistent and the angles are in radians only. All the angles are measured from the global axis. But the angular velocities and accelerations are measured from either the global axis or the frames, which depend on those of the input links. The user has a choice in printing the output. Flags are set accordingly in the input file. Either of the outputs, viz., positions, velocities, accelerations or any of the combinations of them, can be requested in the printout.

4.1.2 Program Organization of Dynamic Analysis

This program is developed from the kinematic analysis program. The equation of motion for a fixed frame comes from the generalized equation of motion. Hence, for a
moving frame, we use the general form of Lagrange's equations. Besides, there is one constraint equation between two adjoining elements and three constraint equations between upper and lower elements.

The program contains a main program and twenty subroutines. A flow chart is given in Figs. 4.1.2 and 4.1.3 to picture the flow of the algorithm. Basically, the organization of the main program and six subroutines, i.e., subroutines CPUTM, DINPUT, KINEMAT, DCHECK, ERROR and DOUTPUT, are similar to those in the kinematic analysis program. Here, we will describe only the input file and those additional subroutines which are significantly different from the previous program.

The Input File: This file is similar to the input file of the kinematic analysis program. The only difference is that we need to specify the inertial properties and the active forces of the system. The description of the input file is shown below:

<table>
<thead>
<tr>
<th>Line #</th>
<th>Data Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Number of branches in the system.</td>
</tr>
<tr>
<td>2</td>
<td>Connectivity details ... (branch 1)</td>
</tr>
<tr>
<td>3</td>
<td>Number of linkages in this branch.</td>
</tr>
<tr>
<td>4</td>
<td>The global coordinates of the fixed frame ... (linkage 1).</td>
</tr>
<tr>
<td>5</td>
<td>Lengths of other links and adjacent angle.</td>
</tr>
<tr>
<td>6</td>
<td>Mass of each link.</td>
</tr>
<tr>
<td>7</td>
<td>Moment of inertia of each link.</td>
</tr>
<tr>
<td>8</td>
<td>Applied torque of each link.</td>
</tr>
<tr>
<td>9</td>
<td>Initial global angle and angular velocity of the input link.</td>
</tr>
</tbody>
</table>
Figure 4.12 The flow chart of dynamic analysis program—main program.
Figure 4.1.3  The flow chart of dynamic analysis program—subroutines RUNGK4, DIFEQN and MATRIX.
10  Start time, finish time and time increment.

  *  Repeating line 4, ..., 8, if more linkages exist.

K  The same as in line 2 ... (branch 2)
K+1 The same as in line 3.
K+2 If the frame is fixed, then input the same as in line 4 ... (linkage 1).
If the frame is moving, then input the local coordinates.
K+3 IDRIVER, initial global angle and angular velocity of the input link.
K+4 The linkage and branch numbers, if the flag IDRIVER=2
  *  Repeating line 5, ..., 8.
  *  Repeating line K+2, ..., K+8, if more linkages exist.
  *  Repeating line K, ..., t, if more branches exist.

m  Number of rotational spring-damper actuators in the system.
m+1 Location and characteristics of the spring-damper actuator,
i.e., N, N, N, N, RS, DC, ZTE
where  N  = The i-th branch,
    N  = The j-th linkage of the i-th branch,
    N  = The k-th link of the j-th linkage,
    N  = The ℓ-th link of the j-th linkage,
    RS = The spring stiffness,
    DC = The damping coefficient
    ZTE = Zero torque angle of the rotational spring.
Repeating line m+1, if there are more than one rotational spring-damper actuator.

Printing choice for positions, velocities and accelerations.

Subroutine \textsc{INITL}: This subroutine is used to specify the initial conditions, such as the initial positions and velocities of all driving links which correspond to the degrees of freedom of the system. After this routine, the Runge-Kutta algorithm is used for numerical integration.

Subroutine \textsc{RUNGK4}: This subroutine uses the fourth-order Runge-Kutta algorithm to solve a set of first order differential equations. It calls subroutine \textsc{DIFEQN} for evaluating $\dot{Y} = f(Y,t)$. In our problem, the second order differential equations can be converted to first order equations, so that by integrating the velocities and accelerations per time increment, we can obtain the positions and velocities, respectively. This subroutine first calls subroutines \textsc{DIFEQN}, \textsc{ENERGY} and \textsc{DOUTPUT} sequentially. Before printing the output, we set a flag to call subroutine \textsc{KINEMAT} to calculate other accelerations. After calling subroutine \textsc{DOUTPUT}, we reset the flag to stop calculating other accelerations, so that the cost of the numerical integration can be decreased.

Subroutine \textsc{DIFEQN}: This subroutine calls subroutine \textsc{KINEMAT} first to supply the necessary data for other subroutines, such as subroutines \textsc{RSD}, \textsc{TRIG}, \textsc{CENTER}, \textsc{LOWERLINK}, \textsc{UPPERLINK}, \textsc{CONSTRAIN} and \textsc{MATRIX}. Finally, we obtain the accelerations of all driving links. In accordance with a flag, this routine either calls subroutine \textsc{KINEMAT} again to calculate other accelerations, or directly returns to subroutine \textsc{RUNGK4} to continue integration.
Subroutine RSD: This subroutine computes the moment of a rotational spring-damper actuator. As stated in [1], a rotational spring acts between two links which are hinged on a revolute joint. It applies pure moment on the links with equal magnitude but in opposite directions. The moment can be expressed as:

\[ n(r-s) = k(\phi - \phi^o) \]  

(4.1)

where \( k \) is the spring stiffness, \( \phi \) is the deformed angle and \( \phi^o \) is the undeformed angle (the zero torque angle) of the spring. The deformed angle \( \phi \) depends on the orientations of two connected links, and the zero torque angle \( \phi^o \) is specified by the user based on engineering judgment.

Similarly, for a rotational damper, the moment is expressed as:

\[ n(r-d) = d(\dot{\phi}_j - \dot{\phi}_i) \]  

(4.2)

where \( d \) is the damping coefficient, and \( \dot{\phi}_i \) and \( \dot{\phi}_j \) are the global angular velocities of two connected links. The moments due to rotational spring and damper can be treated as pairs of external forces acting on the links, in this way they become one part of the generalized forces in the equations of motion.

Subroutine TRIG: This subroutine calculates some variables which are in terms of the trigonometric functions. In order to reduce the cost of calculation, in each time step new values of those variables are computed once and then stored and transferred to other subroutines.

Subroutine CENTER: This subroutine computes the velocity and acceleration of the mass center of each link, and calculates the necessary terms which appear in the equations of motion.
Subroutine LOWERLINK: this subroutine computes the terms which appear in the generalized equation of motion for a linkage with a fixed frame. The terms include the generalized moment of inertia, the derivative of generalized moment of inertia and generalized force.

Subroutine UPPERLINK: This subroutine computes the terms which appear in the general form of Lagrange’s equations of motion for a linkage with a moving frame.

Subroutine CONSTRAINT: This subroutine computes the Jacobian matrix and the right side of constraint differential equations. The constraints are built between two elements. Both elements have fixed frames or moving frames or one with a fixed frame and the other with a moving frame.

Subroutine MATRIX: This subroutine calls subroutine SOLVE to obtain the solutions of the governing equations of motion.

Subroutine SOLVE: This subroutine arranges the entries of the mass matrix and the right side of the system equations of motion, and then calls subroutine LINEAR to obtain the solutions of a set of differential equations.

Subroutines LINEAR and LU: Both subroutines are stated in [1] with a detailed illustration and computer algorithm. Subroutine LINEAR solves a set of linear equations in the form $A X = C$ by calling subroutine LU to factorize $A$ into $L$ and $U$ matrices.

Subroutine ENERGY: This subroutine computes the kinematic energy, potential energy and mechanical energy of the system. In a conservative system, by monitoring the variation of mechanical energy, one can check the numerical error after several time increments.
4.2 Example Problem

Cascades of simple planar four-bar mechanisms can be put together to perform rather complex tasks. As stated in [1], an orbiting satellite in the process of deploying solar panels supplies an excellent example. This application lets us obtain the detailed knowledge of dynamic analysis for further technical performance estimation. As shown in Fig. 4.2.1, the deployable solar panels consist of a cascade of six four-bar linkages. These panels are folded in a compact form to occupy minimum space. As the satellite moves into orbit, where the gravity is equal to zero, the rotational springs and dampers deploy the panels in four predefined sequences, as shown in Fig. 4.2.2. After the final deploying process, the six four-bar linkages become a truss structure to support the solar panels. Because this is a symmetric structure, we only present the analytic scheme to simulate the first two partial motions, and then use the same approach for the other two partial motions.

The configuration of the first partial motion is shown in Fig. 4.2.3, where we assume that the base is temporarily fixed, such that there is only one branch consisting of two linkages with fixed frames and one spring-damper actuator in the system. The zero torque angle of the rotational spring equals zero by measuring the orientations of two links to which the spring is attached. Giving the dimensions and inertial properties of links, specifying reasonable physical characteristics of the spring and damper, and referring to the description of the input file, as described in Section 4.1.2, we can obtain the input data for the dynamic analysis program, as shown in Table 4.2.1.

During the first partial motion, the input link and the coupler oscillate about the revolute joint where the spring-damper actuator is installed. After a while, the system will arrive at an equilibrium condition as the first linkage becomes a triangular structure.
Figure 4.2.1 Unfolded solar panels of satellite.
Figure 4.2.2 The process of deploying solar panels in orbit.
\[ r_2 = r_3 = r_4 = r_5 = 2.0 \text{ m} \]
\[ r_1 = r_4 = r_5 = r_6 = 2.828 \text{ m} \]
\[ m_i = m_j = 1 \text{ kg} \]
\[ u_i = u_j = 0.2 \text{ kg}\cdot\text{m}^2, \text{ for } i = 1, \ldots, 4 \]
\[ k = 100 \text{ N}\cdot\text{m}/\text{rad.}, \text{ and } d = 25 \text{ N}\cdot\text{m}\cdot\text{s}/\text{rad.} \]
\[ \phi^0 = 0.0 \text{ rad.} \]

Figure 4.2.3 The first partial motion of solar panels.
<table>
<thead>
<tr>
<th>Line #</th>
<th>Input Data (Free Format)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1, 1, 1, 1, 4, 2</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>-0.828, 2.0, 2.0, 2.0</td>
</tr>
<tr>
<td>5</td>
<td>2.0, 2.0, 2.828, 0.0</td>
</tr>
<tr>
<td>6</td>
<td>1.0, 1.0, 1.0, 1.0</td>
</tr>
<tr>
<td>7</td>
<td>0.2, 0.2, 0.2, 0.2</td>
</tr>
<tr>
<td>8</td>
<td>0.0, 0.0, 0.0, 0.0</td>
</tr>
<tr>
<td>9</td>
<td>0.05, 0.0</td>
</tr>
<tr>
<td>10</td>
<td>0.0, 2.5, 0.05</td>
</tr>
<tr>
<td>11</td>
<td>2.0, 2.0, 4.0, 2.0</td>
</tr>
<tr>
<td>12</td>
<td>2.828, 2.828, 2.0, 0.0</td>
</tr>
<tr>
<td>13</td>
<td>1.0, 1.0, 1.0, 1.0</td>
</tr>
<tr>
<td>14</td>
<td>0.2, 0.2, 0.2, 0.2</td>
</tr>
<tr>
<td>15</td>
<td>0.0, 0.0, 0.0, 0.0</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>1, 1, 2, 3, 100.0, 25.0, 0.0</td>
</tr>
<tr>
<td>18</td>
<td>1, 1, 1</td>
</tr>
</tbody>
</table>
We take the triangular structure as a rigid body, and then start the second partial motion, as shown in Fig. 4.2.4. In this step, there are two branches in the system, and each branch consists of only one linkage. The first linkage has a fixed frame, but the second linkage has a moving frame, where a spring-damper actuator is installed between the ground and the input link of the first linkage. The moving frame is connected to the triangular rigid body, which is one the coupler of the first linkage. Note that the driving link of the floating linkage is link 4, and its initial global angle and angular velocity depend upon those of link 4 of the first linkage.

Specifying the zero torque angle of rotational spring (i.e., 3.1416 rad.) and referring to the description of the input file, we can obtain the input data, as shown in Table 4.2.2. After the system arrives at an equilibrium condition, the partial motions of steps 3 and 4 will be started sequentially. In the same manner, the input data of the third and fourth partial motions can easily be obtained because the structure of the solar panels is symmetric.

In the computer program, by setting gravity equal to zero and inputting four data files, we can obtain the simulation output by combining four data outputs. Sending the simulation output into the animated graphics and creating the geometry files describing the shapes of the bodies in the system, we can obtain the full cycle animated images as that shown in Fig. 4.2.2. The variation in the characteristics of the springs and dampers can create different behaviors of the unfolding process.
Figure 4.2.4 The second partial motion of solar panels.
Table 4.22 Input Data of the Second Partial Motion of Solar Panels

<table>
<thead>
<tr>
<th>Line #</th>
<th>Input Data (Free Format)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>1, 1, 1, 1, 4, 2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0.0, 1.172, 0.0, 2.0</td>
</tr>
<tr>
<td>5</td>
<td>1.172, 2.828, 2.0, 0.0</td>
</tr>
<tr>
<td>6</td>
<td>1.0, 1.0, 1.0, 1.0</td>
</tr>
<tr>
<td>7</td>
<td>0.2, 0.2, 0.2, 0.2</td>
</tr>
<tr>
<td>8</td>
<td>0.0, 0.0, 0.0, 0.0</td>
</tr>
<tr>
<td>9</td>
<td>2.356, 0.0</td>
</tr>
<tr>
<td>10</td>
<td>0.0, 12.0, 0.05</td>
</tr>
<tr>
<td>11</td>
<td>2, 1, 1, 1, 3, 1</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>2.828, 2.828, 0.0, 0.0</td>
</tr>
<tr>
<td>14</td>
<td>2, 0.0, 0.0</td>
</tr>
<tr>
<td>15</td>
<td>1, 1</td>
</tr>
<tr>
<td>16</td>
<td>2.828, 2.0, 2.0, 0.0</td>
</tr>
<tr>
<td>17</td>
<td>3.0, 1.0, 1.0, 1.0</td>
</tr>
<tr>
<td>18</td>
<td>0.6, 0.2, 0.2, 0.2</td>
</tr>
<tr>
<td>19</td>
<td>0.0, 0.0, 0.0, 0.0</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>1, 1, 2, 1, 100.0, 200.0, 3.1416</td>
</tr>
<tr>
<td>22</td>
<td>1, 1, 1</td>
</tr>
</tbody>
</table>
CHAPTER 5

CONCLUSIONS AND DISCUSSION

The objective of this research is to seek an efficient and precise computer program for dealing with cascades of planar four-bar mechanisms. These special-purpose computer programs have been achieved by using two different approaches. Here, the approach that uses the explicit equations in terms of four relative coordinates is called Method 1; while the other one that uses the simultaneous equations in terms of three absolute coordinates is called Method 2. These coordinate systems, i.e., the relative and the absolute coordinates, can both be transferred into the global coordinates for the general usage such as that for the animation in a high-speed computer graphics.

From the simulation results, the efficiency and accuracy of these special-purpose programs are comparable or superior to those of the general-purpose programs, such as programs KAP and DAP, which have been developed from the method of vector analysis by using the Cartesian coordinate system, as stated in [1]. It will be seen that constrained systems are best described in terms of the so-called Lagrangian coordinates, which include both the relative and absolute coordinates as that used in Methods 1 and 2, respectively. In what follows, we will compare the efficiency and the accuracy of these programs that are based on these methods. The future development of these computer programs will also be recommended.

5.1 Computational Efficiency

The computational efficiency of a program depends on the kind of method used. In each method, there are different numbers of governing equations of motion and constraint equations. As a result, the dimension of mass matrix is different.
The comparison between Methods 1 and 2 with regard to several typical models of four-bar mechanisms and the sizes of the system equations of motion is summarized in Table 5.1.1. From it, we conclude that the size of the mass matrix of Method 1 is smaller than that of Method 2. Besides, Method 1 uses the closed-form expressions for finding the position angles while Method 2 uses iteration method for finding them, in view of that, the computation time for convergence is affected by the proximity of the initial estimate of the unknowns. However, Method 1 actually has higher order of nonlinearity and complexity in the governing equations of motion than those in Method 2. Therefore, we use four prototype programs to compare the efficiency between both methods. Each method is developed into kinematic and dynamic analysis programs.

Choosing a system consisting of two linkages with fixed frames, as the first model shown in Table 5.1.1, we obtain input files for simulating twenty time steps with step size 0.03 second. As a result, from the outputs of programs, which only indicate the discrete time without other data, the CPU times with regard to different programs are obtained as shown in Table 5.1.2. From this, we reach a conclusion that Method 1 saves more computation time in kinematic analysis and dynamic analysis than Method 2 does. Furthermore, Method 1 requires less storage space on a computer than Method 2, since the resulting mass matrix has a smaller dimension.

5.2 Computational Accuracy

As Method 1 uses the explicit equations directly for finding the position angles, it generates less numerical error. On the other hand, Method 2 uses a set of simultaneous equations, the solution of which requires the Newton-Raphson approach. This iteration method naturally creates some numerical error after several iteration procedures. Furthermore, in dynamic analysis, the acceleration is a function of positions, velocities and time.
Table 5.1.1: The Sizes of the System Equations of Motion by Using Relative Coordinates (Method 1) and Absolute Coordinates (Method 2).

<table>
<thead>
<tr>
<th>Four-Bar Mechanism</th>
<th>Number of Degrees of Freedom</th>
<th>Number of Equations of Motion Method 1</th>
<th>Number of Equations of Motion Method 2</th>
<th>Number of Constraint Equations Method 1</th>
<th>Number of Constraint Equations Method 2</th>
<th>Dimension of Mass Matrix Method 1</th>
<th>Dimension of Mass Matrix Method 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td>2</td>
<td></td>
<td>6</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>5</td>
<td></td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>6</td>
<td></td>
<td>12</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>10</td>
<td></td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td>18</td>
<td></td>
<td>34</td>
<td>34</td>
</tr>
</tbody>
</table>
Table 5.1.2 The Computation Times of Programs by Using Relative and Absolute Coordinates.

<table>
<thead>
<tr>
<th>Method (coordinates)</th>
<th>Kinematic</th>
<th>Analysis</th>
<th>Dynamic</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method 1 (relative)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Method 2 (absolute)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of Time Steps</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Step Size (second)</td>
<td>0.03</td>
<td>0.03</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>The CPU Time (second)</td>
<td>0.41</td>
<td>0.49</td>
<td>2.71</td>
<td>4.07</td>
</tr>
</tbody>
</table>
Thus, if the positions and velocities, which are obtained from numerical integration, have some numerical error, then they will create more error for the accelerations. In accordance with this observation and the comparison test, as discussed in Table 5.1.2, we find that by monitoring the variation of system mechanical energy in the dynamic analysis, Method 1 is more precise than Method 2, even though in kinematic analysis, the gaps between the output data of both methods are under ± 0.00005 after twenty time steps of simulation. Besides, Method 2 may encounter convergence difficulties if the initial estimates of the unknown parameters are far from their actual values or the error criterion of convergence is not satisfied.

In some situations, the explicit equations may be unacceptable by a computer. For instance, Eq. (2.11) is invalid, as \( r_1 = r_3, \ r_2 = r_4 \) and \( q_1 = \pi \) radians. In this case, \( \theta_4 = 2 \tan^{-1} \frac{0}{0} \) is an indeterminate form; hence, we need to design a routine to specify \( \theta_4 = q_1 \). In a similar manner, as \( r_1 = r_4 = 2r_2 = 2r_3 \) and \( q_1 = \frac{\pi}{3} \) radians, angle \( \theta_4 \) is also indeterminate; hence we have to create a new routine to define \( \theta_4 = \pi - (2\pi - q_1 - q_3 - q_4) \).

There is another problem that arises from the nature of four-bar mechanisms. As stated in [1], a mechanism becomes kinematically indeterminate when the configuration of the mechanism is in a change-point state, which gives rise to a leading form or a lagging form of the linkage. During numerical analysis, if the time steps are chosen such that the constraint equations are assembled in the change-point configuration, the Jacobian matrix will become singular, so that the solution of Eqs. (2.22) cannot be obtained. Therefore, in the change-point state, we must specify auxiliary devices or dynamic considerations.

Finally, some numerical error may be created during the numerical integration. The fourth-order Runge-Kutta algorithm is most widely used; hence the algorithm can be improved by changing the order and step size [1]. Usually increasing the order and
changing the step size can improve the accuracy, but this requires additional computation time. Once a certain order is selected, the order must remain fixed during the entire integration process.

After we have obtained the solutions of the system equations of motion, it is not necessary to integrate all angular accelerations. The numerical integration can only apply to the angular velocities and accelerations of the driving links which correspond to the degrees of freedom of the system. Once we have got the position angles, angular velocities and accelerations of the driving links, then we can use them to obtain the data of other links by means of the kinematic analysis. Therefore, there is no constraint violation created after several integration procedures.

5.3 Future Development

As the mass matrix is the main part of the system equations of motion, an improvement of the computer programs will be to modify the program so that it will internally and automatically formulate the mass matrix and the right side of the system equations of motion. Referring to Table 5.1.1, we see that the formulation of the mass matrix is dependent on the model of the system. Once the model is determined, the entries of the mass matrix and the right side of the system equations of motion need to be arranged into suitable locations of the working arrays, in such a way that we can apply the L-U factorization to get the solutions. As shown in subroutine SOLVE, the locations of entries in the working arrays are currently arranged by a user, consequently, the program is lack of flexibility. We anticipate that the use of programming techniques will be able to design a routine which will make the program more versatile.

As a system becomes more complex, the dimension of its mass matrix will expand quickly, as shown in Table 5.1.1. Since there are many zero entries distributing over the
mass matrix, this not only wastes a lot of computer memory, but also makes the computation inefficient. Therefore, a compact matrix technique is required to deal with the non-zero entries in a symmetric matrix. Meanwhile, we need to define pointers which indicate the addresses in working arrays for locating variables. By doing that, some COMMON statements in computer programs can be replaced by pointers to transfer variables among subroutines; this will save more computer memory.

The program can handle a system loaded with rotational springs and dampers. Other functions concerning different kinds of mechanical devices can also be added into the program. In order to satisfy a wide range of problems, the program can be improved by additional subroutines which deal with user-specified mechanical devices. These subroutines are written in FORTRAN expressions in accordance with the theorem of power balance. For instance, a linear spring and damper can be installed at a link to separate it into two parts such that its length is variable, depending on active forces. Functions with this property will certainly enable the program to handle more variety of mechanical systems.

Finally, an additional subroutine to compute the data of user-selected points of interest is necessary for wider applications, such that one can obtain the data of specified points on any of the links. For further development, the analysis of spatial four-bar mechanisms is recommended. It can be anticipated than Method 1 will save more computation time and computer memory than Method 2 for three-dimensional problems because Method 1 is in terms of generalized coordinates with the minimum number of variables.
APPENDIX A

LISTING OF KINEMATIC ANALYSIS PROGRAM
**KINEMATIC ANALYSIS OF FOUR-BAR LINKAGES**

This program analyses cascades of planar four-bar mechanisms horizontally and vertically by given link lengths and initial angles, angular velocities and angular accelerations of the input links (cranks).

**Dictionary of Variables:**

<table>
<thead>
<tr>
<th>NAME</th>
<th>TYPE</th>
<th>USAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>REAL</td>
<td>Length of the ground link (Link 1)</td>
</tr>
<tr>
<td>R2,RR2</td>
<td>REAL</td>
<td>Length of the input link (Link 2)</td>
</tr>
<tr>
<td>R3</td>
<td>REAL</td>
<td>Length of the coupler link (Link 3)</td>
</tr>
<tr>
<td>R4,RR4</td>
<td>REAL</td>
<td>Length of the output link (Link 4)</td>
</tr>
<tr>
<td>XX,YY</td>
<td>REAL</td>
<td>Global coord. of the end of link</td>
</tr>
<tr>
<td>PHI1</td>
<td>REAL</td>
<td>Global angle of the ground link</td>
</tr>
<tr>
<td>PHI2</td>
<td>REAL</td>
<td>Global angle of the input link</td>
</tr>
<tr>
<td>PHI3</td>
<td>REAL</td>
<td>Global angle of the coupler link</td>
</tr>
<tr>
<td>PHI4</td>
<td>REAL</td>
<td>Global angle of the output link</td>
</tr>
<tr>
<td>BETA</td>
<td>REAL</td>
<td>Angle between two adjacent linkages</td>
</tr>
<tr>
<td>Q1,QQ1</td>
<td>REAL</td>
<td>Relative angle of crank</td>
</tr>
<tr>
<td>QD1,QQD1</td>
<td>REAL</td>
<td>Relative angular velocity of crank</td>
</tr>
<tr>
<td>QDD1,QQDD1</td>
<td>REAL</td>
<td>Relative angular accel. of crank</td>
</tr>
<tr>
<td>T0</td>
<td>REAL</td>
<td>Starting time</td>
</tr>
<tr>
<td>DT</td>
<td>REAL</td>
<td>Time step</td>
</tr>
<tr>
<td>TE</td>
<td>REAL</td>
<td>Finish time</td>
</tr>
<tr>
<td>NB</td>
<td>INTEGER</td>
<td>Number of branches in a system</td>
</tr>
<tr>
<td>NL</td>
<td>INTEGER</td>
<td>Number of links in each branch</td>
</tr>
<tr>
<td>IBN</td>
<td>INTEGER</td>
<td>Additional branch number</td>
</tr>
<tr>
<td>ICV</td>
<td>INTEGER</td>
<td>Branch number to be connected</td>
</tr>
<tr>
<td>ILN</td>
<td>INTEGER</td>
<td>Linkage number to be connected</td>
</tr>
<tr>
<td>ITYPE</td>
<td>INTEGER</td>
<td>Type of linkage</td>
</tr>
<tr>
<td>ILINK</td>
<td>INTEGER</td>
<td>Connected link number</td>
</tr>
<tr>
<td>ISIDE</td>
<td>INTEGER</td>
<td>Connected side</td>
</tr>
<tr>
<td>IDRIVER</td>
<td>INTEGER</td>
<td>Flag of angular velocity and accel. (0: global, 1: relative, 2: dependent)</td>
</tr>
</tbody>
</table>

**Main Program**

**External Subroutines:**

- **DINPUT** — For acquiring input data for all linkages
- **DCHECK** — Checking of link lengths (Grashof's law)
- **ERROR** — Error message
- **KINEMAT** — Kinematic analysis
- **DOUTPUT** — For data output
- **CPUT** — Measurement of the CPU time
C

***********************************************************************
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(7,7),Q2(7,7),Q3(7,7),Q4(7,7),ANGLE(7,7)
COMMON/BLOC2/QD1(7,7),QD2(7,7),QD3(7,7),QD4(7,7),VEL(7,7)
COMMON/BLOC3/QDD1(7,7),QDD2(7,7),QDD3(7,7),QDD4(7,7)
+ ,ACCEL(7,7)
COMMON/BLOC4/R1(7,7),R2(7,7),R3(7,7),R4(7,7)
COMMON/TRANS/QQ1(7,7),QQD1(7,7),QQDD1(7,7),RR2(7,7)
+ ,RR4(7,7)
COMMON/BLOC5/T0,TE,DT,T
COMMON/BLOC6/PHI1(7,7),PHI2(7,7),PHI3(7,7),PHI4(7,7)
COMMON/BLOC7/XX1(7,7),XX1(7,7),XX4(7,7),XY4(7,7),
+ ,YY2(7,7),YY2(7,7),YY3(7,7),YY3(7,7),
COMMON/BLOC8/ICV(20),ILN(20),ITYPE(20),ILINK(20),
+ ,ISIDE(20),IDRIVER(20),ICB(20),IL(20)
COMMON/BLOC9/BETA(7,7),NB,NL(20),NPOS,NVEL,NACC
COMMON/BLOC10/PHI1(7,7),PHI2(7,7),PHI3(7,7),PHI4(7,7),
+ ,PHI1DD(7,7),PHI2DD(7,7),PHI3DD(7,7),PHI4DD(7,7)
COMMON/BLOC11/FL1(5,5),FL4(5,5),S1(5,5),S4(5,5)
CHARACTER*80 FIN,FOUT
WRITE(5,11)
11 FORMAT(/' * KINEMATIC ANALYSIS OF FOUR-BAR LINKAGES *',
+ /3X,39(' -')//)
WRITE(5,*) ' * ENTER INPUT DATA FILE NAME *'
READ (5,21) FIN
WRITE(5,*) ' * ENTER OUTPUT FILE NAME *'
READ (5,21) FOUT
21 FORMAT(A)
C....OPENING OF INPUT AND OUTPUT FILES ......
OPEN(2,FILE=FIN,STATUS='OLD',PAD='YES',IOINTENT='INPUT')
OPEN(3,FILE=FOUT,STATUS='FRESH',IOINTENT='OUTPUT')
WRITE(3,31)
31 FORMAT(/5X,' * KINEMATIC ANALYSIS OF FOUR-BAR LINKAGES * ',
+ /8X,39(' -')//)
C
CALL CPUTM(SEC)
TIMEB = SEC
C....READ DATA FROM INPUT FILE ......
CALL DINPUT
DO 10 T = T0,TE,DT
CALL KINEMAT
CALL DOUTPUT
ANGLE(1,1) = ANGLE(1,1)+VEL(1,1)*DT+0.5*ACCEL(1,1)*DT*DT
VEL(1,1) = VEL(1,1)+ACCEL(1,1)*DT
DO 51 I = 1,NB
IF(ILINK(I).EQ.3) THEN
   ANGLE(I,1)=ANGLE(I,1)+VEL(I,1)*DT+0.5*ACCEL(I,1)*DT*DT
   VEL(I,1) = VEL(I,1)+ACCEL(I,1)*DT
ENDIF
51 CONTINUE
10 CONTINUE
CALL CPUTM(SEC)
TIMEE = SEC
TEND = TIMEE-TIMEB
WRITE(5,*)' TOTAL CPU TIME = ',TEND,' SECS'
WRITE(3,41) TEND
CLOSE(2)
CLOSE(3)
STOP
END

SUBROUTINE CPUTM (SEC)
C
C....RETURN CPU TIME IN SECONDS
C
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DATA T1000 /1000.00/
C
%INCLUDE "QSYM.F77"
INTEGER*2 IBUF(0:ISYS_GRLTH)
EQUIVALENCE (ICPUMS,IBUF(ISYS_GRCH)
IAC0 = -1
IAC1 = 0
IAC2 = WORDADDR (IBUF)
IER = ISYS (ISYS_RUNTM,IAC0,IAC1,IAC2)
SEC = FLOAT (ICPUMS) / T1000
RETURN
END
117

```
SUBROUTINE DINPUT

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(7,7),Q2(7,7),Q3(7,7),Q4(7,7),ANGLE(7,7)
COMMON/BLOC2/QD1(7,7),QD2(7,7),QD3(7,7),QD4(7,7),VEL(7,7)
COMMON/BLOC3/QDD1(7,7),QDD2(7,7),QDD3(7,7),QDD4(7,7)
+ ,ACCEL(7,7)
COMMON/BLOC4/R1(7,7),R2(7,7),R3(7,7),R4(7,7)
COMMON/TRANS/QQ1(7,7),QQD1(7,7),QQDD1(7,7),RR2(7,7)
+ ,RR4(7,7)
COMMON/BLOC5/T0,TE,DT,T
COMMON/BLOC6/PHI1(7,7),PHI2(7,7),PHI3(7,7),PHI4(7,7)
COMMON/BLOC7/XX1(7,7),YY1(7,7),XX4(7,7),YY4(7,7),
+ XX2(7,7),YY2(7,7),XX3(7,7),YY3(7,7)
COMMON/BLOC8/ICV(20),ILN(20),ITYPE(20),ILINK(20),
+ ISIDE(20),IDRIVER(20),ICB(20),IL(20)
COMMON/BLOC9/BETA(7,7),NB,NL(20),NPOS,NVEL,NACC
COMMON/BLOC10/PHI1D(7,7),PHI2D(7,7),PHI3D(7,7),PHI4D(7,7),
+ PHI1DD(7,7),PHI2DD(7,7),PHI3DD(7,7),PHI4DD(7,7)
COMMON/BLOC11/FL1(5,5),FL4(5,5),S1(5,5),S4(5,5)
READ(2,*)NB
WRITE(3,51)NB
51 FORMAT(/5X,'NO. OF BRANCHES IN THE SYSTEM '/3X,I3)
DO 20 N = 1,NB
READ(2,*)IBN,ICV(N),ILN(N),ITYPE(N),ILINK(N),ISIDE(N)
WRITE(3,61)N,IBN,ICV(N),ILN(N),ITYPE(N),ILINK(N),ISIDE(N)
61 FORMAT(5X,'CONNECTIVITY DETAILS (BRANCH ',13,')',
      /5X,'IBN, ICV, ILN, ITYPE, ILINK, ISIDE'/6(3X,13))
READ(2,*)NL(N)
WRITE(3,71)N,NL(N)
71 FORMAT(5X,'NO. OF LINKAGES IN BRANCH ',13/3X,I3)
DO 30 K = 1,NL(N)
READ(2,*)XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
WRITE(3,81)N,K,XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
81 FORMAT(5X,'BRANCH # ',13)
WRITE(3,91)K,XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
91 FORMAT('/5X,'LOCAL COORDINATES FOR GROUND LINK',
      '+ ,XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K))
IF(ILINK(N).EQ.3) THEN
  WRITE(3,82) N,K,XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
82 FORMAT(5X,'THE 1ST LINKAGE CONNECTS TO THE LINK3 OF '
      '+ ,LINKAGE ',13,','/3X/4(3X,13))
ELSE
FL1(N,K) = SQRT(XX1(N,K)*XX1(N,K)+YY1(N,K)*YY1(N,K))
FL4(N,K) = SQRT(XX4(N,K)*XX4(N,K)+YY4(N,K)*YY4(N,K))
IF (FL1(N,K) .EQ. 0.0) THEN
  S1(N,K) = 0.0
ELSE
```
$S_1(N,K) = \text{ATAN2}(Y_{I}(N,K), X_{I}(N,K))$

ENDIF

IF (FL4(N,K) .EQ. 0.0) THEN
    $S_4(N,K) = 0.0$
ELSE
    $S_4(N,K) = \text{ATAN2}(Y_{4}(N,K), X_{4}(N,K))$
ENDIF

IF(K.EQ.1) THEN
    READ(2,*) IDRIVER(N), ANGLE(N,1), VEL(N,1), ACCEL(N,1)
    WRITE(3,112) IDRIVER(N), ANGLE(N,1), VEL(N,1), ACCEL(N,1)
112 FORMAT(/5X,'INITIAL GLOBAL ANGLE AND (GLOBAL:0, ',
        + 'RELATIVE:1 OR DEPENDENT:2) VELOCITY AND ',
        + 'ACCEL. OF CRANK'/2X,I3,3F14.4)
    IF(IDRIVER(N) .EQ. 2) THEN
        READ(2,*) ICB(N), IL(N)
        WRITE(3,115) ICB(N), IL(N)
115 FORMAT(/5X,'THE INPUT ANGLE,VEL. AND ACCEL. ',
        + 'DEPEND ON THE LINK4 OF LINKAGE',I3,'OF BRANCH',I3)
    ENDIF
ENDIF
ELSE
    C....INITIALIZE THE ANGULAR VEL. AND ACCEL. OF GROUND LINK
    PHILD(N,K) = 0.0
    PHILDD(N,K) = 0.0
ENDIF

READ(2,*) R2(N,K), R3(N,K), R4(N,K), BETA(N,K)
WRITE(3,101) K, R2(N,K), R3(N,K), R4(N,K), BETA(N,K)
101 FORMAT(/5X,'LENGTHS OF OTHER LINKS AND ADJACENT ANGLE',
        + ' /4(2X,F10.4))
C
C....TRANSFER INPUT DATA OF LINK
IF(ISIDE(N) .EQ. 1) THEN
    RR4(N,K) = R2(N,K)
    RR2(N,K) = R4(N,K)
ELSE
    RR2(N,K) = R2(N,K)
    RR4(N,K) = R4(N,K)
ENDIF

N2 = N
K2 = K
IF(N.GT.1) GOTO 30
IF(K.GT.1) GOTO 30
READ(2,*) ANGLE(1,1), VEL(1,1), ACCEL(1,1)
WRITE(3,111) ANGLE(1,1), VEL(1,1), ACCEL(1,1)
111 FORMAT(/5X,'INITIAL GLOBAL ANGLE AND VEL. AND',
        + 'ACCEL. OF CRANK /3(2X,F12.4))
    READ(2,*) T0, TE, DT
    WRITE(3,121)
121 FORMAT(5X,'START TIME, FINISH TIME AND TIME INCREMENT')
    WRITE(3,131) T0, TE, DT
131 FORMAT(/3(F10.4,4X))
CONTINUE
CONTINUE
READ(2,*)NPOS,NVEL,NACC
IF((NPOS .NE. 1) .AND. (NPOS .NE. 0)) NPOS=1
IF((NVEL .NE. 1) .AND. (NVEL .NE. 0)) NVEL=1
IF((NACC .NE. 1) .AND. (NACC .NE. 0)) NACC=1
WRITE(3,141)NPOS,NVEL,NACC
141 FORMAT(/5X,'THE FOLLOWING INDICATE THE CODES FOR PRINT'
+, ' REQUEST',/5X,'"1" INDICATES "PRINT". "0" INDICATES'
+, ' "DO NOT PRINT",/5X,'(DEFAULT VALUE IS "1")',/5X,
+, 'NPOSITION =',I2,'NVELOCITY =',I2,' NACCEL. =',I2/)
RETURN
END
SUBROUTINE KINEMAT

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(7,7),Q2(7,7),Q3(7,7),Q4(7,7),ANGLE(7,7)
COMMON/BLOC2/QD1(7,7),QD2(7,7),QD3(7,7),QD4(7,7),VEL(7,7)
COMMON/BLOC3/QDD1(7,7),QDD2(7,7),QDD3(7,7),QDD4(7,7)
COMMON/BLOC4/R1(7,7),R2(7,7),R3(7,7),R4(7,7)
COMMON/TRANS/QQ1(7,7),QQD1(7,7),QQDD1(7,7),RR2(7,7),RR4(7,7)
COMMON/BLOC5/T0,TE,DT,T
COMMON/BLOC6/PHI1(7,7),PHI2(7,7),PHI3(7,7),PHI4(7,7)
COMMON/BLOC7/XX1(7,7),YY1(7,7),XX4(7,7),YY4(7,7)
COMMON/BLOC8/ICV(20),ILN(20),ITYPE(20),ILINK(20)
COMMON/BLOC9/BETA(7,7),NB,NL(20),NPOS,NVEL,NACC
COMMON/BLOC10/PHI2D(7,7),PHI3D(7,7),PHI4D(7,7),
PHI1DD(7,7),PHI2DD(7,7),PHI3DD(7,7),PHI4DD(7,7)
COMMON/BLOC11/FL1(5,5),FL4(5,5),S1(5,5),S4(5,5)
DIMENSION A(30),QQ2(7,7),QQ3(7,7),QQ4(7,7),QQD2(7,7)
PI = 3.141592653
PI2 = 6.283185307
QQ1(1,1) = ANGLE(1,1)
QQD1(1,1) = VEL(1,1)
QQDD1(1,1) = ACCEL(1,1)

C

DO 50 I = 1,NB
IF(I.GT.1) THEN
IF(ILINK(I) - 3) 10,30,20
10 QQ1(I,1)=Q1(ICV(I),ILN(I))+PHI1(ICV(I),ILN(I))
   QQD1(I,1)=QD1(ICV(I),ILN(I))
   QQDD1(I,1) = QDD1(ICV(I),ILN(I))
   GO TO 35
20 QQ1(I,1)=PI-Q2(ICV(I),ILN(I))+PHI1(ICV(I),ILN(I))
   QQD1(I,1) = -QD2(ICV(I),ILN(I))
   QQDD1(I,1) = -QDD2(ICV(I),ILN(I))
   GO TO 35
C
30 QQ1(I,1) = ANGLE(I,1)
IF(ISIDE(I).EQ.ISIDE(ICV(I))) THEN
   ISIGN = 1
ELSE
   ISIGN = -1
ENDIF
DO 37 IP = 1,NL(I)
   IQ = IP - 1
   IC = ILN(I)+IQ*ISIGN
   SS1 = S1(I,IP)+PHI3(ICV(I),IC)
   SS4 = S4(I,IP)+PHI3(ICV(I),IC)
C

PHIID(I, IP) = PHI3D(ICV(I), IC)

37

PHI1DD(I, IP) = PHI3DD(ICV(I), IC)

IF(IDRIVER(I) - 1) 2, 3, 4

2

QQD1(1, 1) = VEL(1, 1) - PHIID(1, 1)

QQDD1(1, 1) = ACCEL(1, 1) - PHI1DD(1, 1)

GO TO 35

3

QQD1(I, 1) = VEL(I, 1)

QQDD1(I, 1) = ACCEL(I, 1)

GO TO 35

4

QQ1(I, 1) = PI - Q2(ICB(I), IL(I)) + PHI1(ICB(I), IL(I))

QQD1(I, 1) = -QD2(ICB(I), IL(I))

QQDD1(I, 1) = -QDD2(ICB(I), IL(I))

35

IF(ISIDE(I) .EQ. 1) THEN

QQ1(I, 1) = PI - QQ1(I, 1)

QQD1(I, 1) = -QQD1(I, 1)

QQDD1(I, 1) = -QQDD1(I, 1)

ENDIF

ENDIF

C

DO 60 J = 1, NL(I)

Y41 = YY4(I, J) - YY1(I, J)

X41 = XX4(I, J) - XX1(I, J)

R1(I, J) = SQRT(X41*X41 + Y41*Y41)

C ...CHECK GRASHOF'S LAW

IF(T.NE.T0) GOTO 33 ! In some cases of applications

IF(ILINK(I).EQ.3) THEN ! DCHECK can be neglected.

C

IF((J.EQ.1) .AND. (IDRIVER(I) .NE. 2)) CALL DCHECK(I, J)

ELSE

IF((I+J) .EQ. 2) CALL DCHECK(I, J)

ENDIF

C

33

PHI1(I, J) = ATAN2(Y41, X41)

IF(ISIDE(I) .EQ. 1) THEN

PHIIJ = -PHI1(I, J)

ELSE

PHIIJ = PHI1(I, J)

ENDIF

C

QQ1(I, J) = QQ1(I, J) - BETA(I, J) - PHIIJ

C

...ANGULAR POSITION ANALYSIS ......

SIN1 = SIN(QQ1(I, J))

COS1 = COS(QQ1(I, J))

A(1) = (R3(I, J)*R3(I, J) + RR4(I, J)*RR4(I, J) - R1(I, J)*R1(I, J)
\[ A(2) = \frac{R1(I,J)}{R3(I,J) * RR4(I,J)} \]
\[ A(3) = RR2(I,J) * SIN1 \]
\[ A(4) = RR2(I,J) * COS1 \]
\[ A(5) = ACOS(A(1) + A(2) * A(4)) \]

\[
\begin{align*}
    & \text{IF}( \text{R1}(I,J) = \text{R3}(I,J) ) \text{ AND } \text{R2}(I,J) = \text{R4}(I,J) \text{ THEN} \\
    & \quad A(8) = 0.0 \\
    & \quad A(9) = QQ1(I,J) \\
    & \text{ELSE} \\
    & \quad A(6) = \text{SIN}(A(5)) \\
    & \quad A(7) = \text{COS}(A(5)) \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{RR}4A6 = RR4(I,J) * A(6) \\
    & \text{R3A6} = R3(I,J) * A(6) \\
    & \text{PI22} = PI2 \\
    & \text{IF}( \text{ITYPE}(I) - 2 = 0 \text{ OR } 55 \text{ OR } 40 \text{ OR } 100, 55, 40 \\
    & \quad \text{IF}( \text{QQ1}(I,J) \leq \text{PI} ) \text{ GO TO 100} \\
    & \quad \text{IF}( \text{QQ1}(I,J) > \text{PI2} ) \text{ GO TO 100} \\
    & \quad \text{RR}4A6 = -\text{RR}4A6 \\
    & \quad \text{R}3A6 = -\text{R}3A6 \\
    & \quad \text{PI22} = 0.0 \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{AA8} = -A(3) + \text{RR}4A6 \\
    & \text{BB8} = -A(4) + R3(I,J) + R1(I,J) - \text{RR}4(I,J) * A(7) \\
    & \text{AA9} = A(3) - R3A6 \\
    & \text{BB9} = A(4) + \text{RR}4(I,J) - R1(I,J) - R3(I,J) * A(7) \\
    & A(8) = 2 * \text{ATAN2}(\text{AA8}, \text{BB8}) + \text{PI22} \\
    & A(9) = 2 * \text{ATAN2}(\text{AA9}, \text{BB9}) + \text{PI22} \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{QQ}3(I,J) = A(5) \\
    & \text{QQ}4(I,J) = \text{PI} - \text{QQ}1(I,J) + A(8) \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{IF}( \text{ISIDE}(I) \text{.EQ. 1) THEN} \\
    & \quad \text{PHI}2(I,J) = \text{QQ}2(I,J) + \text{PHI}1(I,J) \\
    & \quad \text{PHI}3(I,J) = -A(8) + \text{PHI}1(I,J) + \text{PI}2 \\
    & \quad \text{PHI}4(I,J) = \text{PI} - \text{QQ}1(I,J) + \text{PHI}1(I,J) \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{ELSE} \\
    & \quad \text{PHI}2(I,J) = \text{QQ}1(I,J) + \text{PHI}1(I,J) \\
    & \quad \text{PHI}3(I,J) = A(8) + \text{PHI}1(I,J) \\
    & \quad \text{PHI}4(I,J) = A(9) + \text{PHI}1(I,J) \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{ENDIF} \\
    \end{align*}
\]

\[
\begin{align*}
    & \text{CALCULATE THE COORD. OF EACH END OF LINK} \\
    & \text{XX}2(I,J) = \text{XX}1(I,J) + R2(I,J) * \text{COS} \text{(PHI}2(I,J)) \\
    & \text{YY}2(I,J) = \text{YY}1(I,J) + R2(I,J) * \text{SIN} \text{(PHI}2(I,J)) \\
    & \text{XX}3(I,J) = \text{XX}4(I,J) + R4(I,J) * \text{COS} \text{(PHI}4(I,J)) \\
    & \text{YY}3(I,J) = \text{YY}4(I,J) + R4(I,J) * \text{SIN} \text{(PHI}4(I,J)) \\
\end{align*}
\]
C ...COMPUTE THE ENTRIES OF JACOBIAN MATRIX
SIN2 = SIN(QQ2(I,J))
SIN4 = SIN(QQ4(I,J))
COS2 = COS(QQ2(I,J))
COS4 = COS(QQ4(I,J))
SIN12 = SIN1*COS2+COS1*SIN2
SIN14 = SIN1*COS4+COS1*SIN4
COS12 = COS1*COS2-SIN1*SIN2
COS14 = COS1*COS4-SIN1*SIN4

IF((R1(I,J).EQ.R3(I,J)) .AND. (R2(I,J).EQ.R4(I,J))) THEN
    QQD2(I,J) = -QQD1(I,J)
    QQD3(I,J) = QQD1(I,J)
    QQD4(I,J) = -QQD1(I,J)
ELSE
    A(10) = 2*RR2(I,J)*RR4(I,J)
    A(11) = 2*R1(I,J)*R3(I,J)
    A(12) = 2*RR2(I,J)*R3(I,J)
    A(13) = 2*R1(I,J)*RR2(I,J)
    A(14) = 2*R1(I,J)*RR4(I,J)
    A(15) = -A(10)*SIN12+A(13)*SIN1
    A(16) = A(13)*SIN1
    A(17) = -A(11)*SIN14+A(13)*SIN1
    A(18) = -A(10)*SIN12+A(14)*SIN2
    A(19) = -2*R3(I,J)*RR4(I,J)*SIN(QQ3(I,J))
    A(20) = -A(11)*SIN14+A(12)*SIN4
ENDIF

C ...VELOCITY COMPUTATION
QQD1(I,J) = QQD1(I,J)
QQD2(I,J) = -A(15)*QQD1(I,J)/A(18)
QQD3(I,J) = -A(16)*QQD1(I,J)/A(19)
QQD4(I,J) = -A(17)*QQD1(I,J)/A(20)

ENDIF

C ...ACCELERATION CALCULATION
QQDD1(I,J) = QQDD1(I,J)
QQDD2(I,J) = (-A(23)*QQD1(I,J)-A(26)*QQD2(I,J)-A(15)
    +A(27)*QQD3(I,J)-A(20)*QQD4(I,J))/A(18)
QQDD3(I,J) = (-A(24)*QQD1(I,J)-A(27)*QQD3(I,J)-A(16)
    +A(26)*QQD2(I,J)-A(23)*QQD1(I,J))/A(18)
\[
\begin{align*}
\text{QQDD4}(I,J) &= \frac{\text{QQDD1}(I,J) - A(20) \times \text{QQD4}(I,J) - A(17) \times \text{QQDD1}(I,J)}{A(20)} \\
\end{align*}
\]

ENDIF

C

C...TRANSFER QQ (INTERNAL VARIABLES) TO Q (EXTERNAL VARIABLES)

IF(ISIDE(I) .EQ. 1) THEN

Q2(I,J) = QQ1(I,J)
Q1(I,J) = QQ2(I,J)
Q4(I,J) = QQ3(I,J)
Q3(I,J) = QQ4(I,J)
QD2(I,J) = QQD1(I,J)
QD1(I,J) = QQD2(I,J)
QD4(I,J) = QQD3(I,J)
QD3(I,J) = QQD4(I,J)
QDD2(I,J) = QQDD1(I,J)
QDD1(I,J) = QQDD2(I,J)
QDD4(I,J) = QQDD3(I,J)
QDD3(I,J) = QQDD4(I,J)

ELSE

Q1(I,J) = QQ1(I,J)
Q2(I,J) = QQ2(I,J)
Q3(I,J) = QQ3(I,J)
Q4(I,J) = QQ4(I,J)
QD1(I,J) = QQD1(I,J)
QD2(I,J) = QQD2(I,J)
QD3(I,J) = QQD3(I,J)
QD4(I,J) = QQD4(I,J)
QDD1(I,J) = QQDD1(I,J)
QDD2(I,J) = QQDD2(I,J)
QDD3(I,J) = QQDD3(I,J)
QDD4(I,J) = QQDD4(I,J)

ENDIF

C...TRANSFER VEL. & ACC. FROM RELATIVE COORD. TO GLOBAL COORD.

PHI2D(I,J) = QD1(I,J) + PHI1D(I,J)
PHI3D(I,J) = QD1(I,J) + QD4(I,J) + PHI1D(I,J)
PHI4D(I,J) = -QD2(I,J) + PHI1D(I,J)
PHI2DD(I,J) = QDD1(I,J) + PHI1DD(I,J)
PHI3DD(I,J) = QDD1(I,J) + QDD4(I,J) + PHI1DD(I,J)
PHI4DD(I,J) = -QDD2(I,J) + PHI1DD(I,J)

C

QQ1(I,J+1) = PI - QQ2(I,J) + PHIIJ
QQD1(I,J+1) = -QQD2(I,J)
QQDD1(I,J+1) = -QQDD2(I,J)

60 CONTINUE
50 CONTINUE
RETURN
END
SUBROUTINE DCHECK(N2, K2)

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC4/R1(7,7),R2(7,7),R3(7,7),R4(7,7)
COMMON/TRANS/QQ1(7,7),QQD1(7,7),QQDD1(7,7),RR2(7,7)
 + ,RR4(7,7)
DIMENSION X(5)
TOLENCE = 0.0000001
X(1) = R1(N2,K2)
X(2) = RR2(N2,K2)
X(3) = R3(N2,K2)
X(4) = RR4(N2,K2)

C

Z = X(1)+X(2)+X(3)+X(4)
BIG = X(1)
SMALL = X(1)
DO 40 I = 1,3
   I2 = I+1
   IF(X(I2).GT.BIG) BIG = X(I2)
   IF(X(I2).LT.SMALL) SMALL = X(I2)
40 CONTINUE
Y = BIG + SMALL
W = Z - Y + TOLENCE
IF(Y.GT.W) CALL ERROR(N2,K2)
RETURN
END

SUBROUTINE ERROR(N3,K3)

C

WRITE(5,151)
151 FORMAT(/,80('**')//)
WRITE(5,*)' *** ERROR IN INPUT DATA ? ***'
WRITE(5,*)' LINK LENGTHS DO NOT SATISFY GRASHOF'S LAW'
WRITE(5,161) N3,K3
161 FORMAT('LINK LENGTHS OF BRANCH',I2,' LINKAGE ',I2,
 + ' ARE INCORRECT')
WRITE(5,*)
WRITE(5,*)'PLEASE VERIFY THE DATA OR MODIFY THE PROGRAM.'
WRITE(5,*)
STOP
END
SUBROUTINE DOUTPUT

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(7,7),Q2(7,7),Q3(7,7),Q4(7,7),ANGLE(7,7)
COMMON/BLOC2/QD1(7,7),QD2(7,7),QD3(7,7),QD4(7,7),VEL(7,7)
COMMON/BLOC3/QDD1(7,7),QDD2(7,7),QDD3(7,7),QDD4(7,7),
+ ACCEL(7,7)
COMMON/BLOC4/R1(7,7),R2(7,7),R3(7,7),R4(7,7)
COMMON/BLOC5/T0,TE,DT,T
COMMON/BLOC6/PHI1(7,7),PHI2(7,7),PHI3(7,7),PHI4(7,7)
COMMON/BLOC7/XX1(7,7),YY1(7,7),XX4(7,7),YY4(7,7),
+ XX2(7,7),YY2(7,7),XX3(7,7),YY3(7,7)
COMMON/BLOC8/ICV(20),ILN(20),ITYPE(20),ILINK(20),
+ ISIDE(20),IDRIVER(20),ICB(20),IL(20)
COMMON/BLOC9/BETA(7,7),NB,NL(20),NPOS,NVEL,NACC
COMMON/BLOC10/PHIID(7,7),PHI2D(7,7),PHI3D(7,7),PHI4D(7,7),
+ PHI1DD(7,7),PHI2DD(7,7),PHI3DD(7,7),PHI4DD(7,7)
IF(T.GT.T0) GOTO 5
WRITE(3,171)
171 FORMAT(5X,'LINK LENGTHS')
WRITE(3,181)
181 FORMAT(/5X,'BRANCH #',5X,'LINKAGE #',7X,'R1',7X,'R2',
+ 7X,'R3',7X,'R4',9X,'BETA ')
DO 70 M = 1,NB
DO 80 L = 1,NL(M)
IF(L.GT.L) GOTO 15
IF(M.GT.L) GOTO 16
16 WRITE(3,201)L,R1(M,L),R2 (M,L) ,R3(M,L) ,R4 (M,L) ,BETA(M,L)
201 FORMAT(19X,I3,8X,F7.3,3(2X,F7.3),6X,F7.4)
GOTO 80
15 WRITE(3,191)M,L,R1 (M,L),R2 (M,L),R3 (M,L),R4 (M,L),BETA (M,L)
191 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
GOTO 80
70 CONTINUE
5 WRITE(3,311) T
311 FORMAT(/5X,'** TIME =',1PE12.4,', **','/5X,'-----'/)
IF(NPOS.EQ.0) GOTO 6
WRITE(3,221)
221 FORMAT(5X,'POSITION ANGLES OF INPUT AND OUTPUT LINKS')
WRITE(3,231)
231 FORMAT(/5X,'BRANCH #',5X,'LINKAGE #',4X,'PHI1',6X,'PHI2',
+ 6X,'PHI3',6X,'PHI4',10X,'Q1',10X,'Q4',10X,'Q3',10X,'Q2 ')
DO 90 M1 = 1,NB
DO 100 L1 = 1,NL(M1)
IF(L1.GT.L1) GOTO 39
90 WRITE(3,241)M1,L1,PHI1(M1,L1),PHI2(M1,L1),PHI3(M1,L1),
+ PH14(M1,L1),Q1(M1,L1),Q4(M1,L1),Q3(M1,L1),Q2(M1,L1)
241 FORMAT(6X,13,10X,13,3X,8(1PE12.4))
WRITE(3,504)XX1(M1,L1),XX2(M1,L1),XX3(M1,L1),XX4(M1,L1),
+ YY1(M1,L1),YY2(M1,L1),YY3(M1,L1),YY4(M1,L1)
GOTO 100
WRITE(3,251)L1,PHI1(M1,L1),PHI2(M1,L1),PHI3(M1,L1),
+ PHI4(M1,L1),Q1(M1,L1),Q4(M1,L1),Q3(M1,L1),Q2(M1,L1)
251 FORMAT(19X,I3,3X,8(1PE12.4))
WRITE(3,504)XX1(M1,L1),XX2(M1,L1),XX3(M1,L1),XX4(M1,L1),
+ YY1(M1,L1),YY2(M1,L1),YY3(M1,L1),YY4(M1,L1)
504 FORMAT(/33X,'XI =',1PE12.4,8X,'X2 =',1PE12.4,8X,'X3 =',
+ 1PE12.4,8X,'X4 =',1PE12.4,/33X,'Y1 =',1PE12.4,8X,
+ 'Y2 =',1PE12.4,8X,'Y3 =',1PE12.4,8X,'Y4 =',1PE12.4/)
100 CONTINUE
90 CONTINUE
6 IF(NVEL.EQ.0) GOTO 7
WRITE(3,261)
261 FORMAT(/5X,'ANGULAR VEL. OF INPUT AND OUTPUT LINKS'/)
WRITE(3,271)
271 FORMAT(5X,'BRANCH #',5X,'LINKAGE #',3X,'PHI1D',5X,
+ 'PHI2D',5X,'PHI3D',5X,'PHI4D',9X,'QD1',9X,
+ 'QD4',9X,'QD3',9X,'QD2')
DO 200 M2 = 1,NB
DO 300 L2 = 1,NL(M2)
IF(L2.GT.1) GOTO 38
WRITE(3,281)M2,L2,PHIID(M2,L2),PHI2D(M2,L2),PHI3D(M2,L2),
+ PHI4D(M2,L2),QD1(M2,L2),QD4(M2,L2),QD3(M2,L2),QD2(M2,L2)
281 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
GOTO 300
38 WRITE(3,291)L2,PHI1D(M2,L2),PHI2D(M2,L2),PHI3D(M2,L2),
+ PHI4D(M2,L2),QD1(M2,L2),QD4(M2,L2),QD3(M2,L2),QD2(M2,L2)
291 FORMAT(19X,I3,3X,8(1PE12.4))
300 CONTINUE
200 CONTINUE
7 IF(NACC.EQ.0) GOTO 69
WRITE(3,301)
301 FORMAT(/5X,'ANGULAR ACCEL. OF INPUT AND OUTPUT LINKS'/)
WRITE(3,311)
311 FORMAT(5X,'BRANCH #',5X,'LINKAGE #',2X,'PHI1DD',4X,
+ 'PHI2DD',4X,'PHI3DD',4X,'PHI4DD',8X,'QDD1',8X,
+ 'QDD4',8X,'QDD3',8X,'QDD2')
DO 400 M3 = 1,NB
DO 500 L3 = 1,NL(M3)
IF(L3.GT.1) GOTO 36
WRITE(3,321)M3,L3,PHIIDD(M3,L3),PHI2DD(M3,L3),
+ PHI3DD(M3,L3),PHI4DD(M3,L3),QDD1(M3,L3),
+ QDD4(M3,L3),QDD3(M3,L3),QDD2(M3,L3)
321 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
GOTO 500
36 WRITE(3,331)L3,PHIIDD(M3,L3),PHI2DD(M3,L3),
+ PHI3DD(M3,L3),PHI4DD(M3,L3),QDD1(M3,L3),
+ QDD4(M3,L3),QDD3(M3,L3),QDD2(M3,L3)
331 FORMAT(19X,I3,3X,8(1PE12.4))
500 CONTINUE
400 CONTINUE
69 RETURN
END
APPENDIX B

LISTING OF DYNAMIC ANALYSIS PROGRAM
**DYNAMIC ANALYSIS OF FOUR-BAR LINKAGES**

THIS PROGRAM ANALYSES CASCADES OF PLANAR FOUR-BAR MECHANISMS HORIZONTALLY AND VERTICALLY BY GIVEN INERTIAL PROPERTIES AND LENGTHS OF LINKS AND INITIAL ANGLES AND ANGULAR VELOCITIES OF THE INPUT LINKS (CRANKS).

---

### DICTIONARY OF VARIABLES:

<table>
<thead>
<tr>
<th>NAME</th>
<th>TYPE</th>
<th>USAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1</td>
<td>REAL</td>
<td>LENGTH OF THE GROUND LINK (LINK1)</td>
</tr>
<tr>
<td>R2, RR2</td>
<td>REAL</td>
<td>LENGTH OF THE INPUT LINK (LINK2)</td>
</tr>
<tr>
<td>R3</td>
<td>REAL</td>
<td>LENGTH OF THE COUPLER LINK (LINK3)</td>
</tr>
<tr>
<td>R4, RR4</td>
<td>REAL</td>
<td>LENGTH OF THE OUTPUT LINK (LINK4)</td>
</tr>
<tr>
<td>X, Y</td>
<td>REAL</td>
<td>GLOBAL COORDINATES OF MASS CENTER</td>
</tr>
<tr>
<td>XX, YY</td>
<td>REAL</td>
<td>GLOBAL COORD. OF THE END OF LINK</td>
</tr>
<tr>
<td>PHI1</td>
<td>REAL</td>
<td>GLOBAL ANGLE OF THE GROUND LINK</td>
</tr>
<tr>
<td>PHI2</td>
<td>REAL</td>
<td>GLOBAL ANGLE OF THE INPUT LINK</td>
</tr>
<tr>
<td>PHI3</td>
<td>REAL</td>
<td>GLOBAL ANGLE OF THE COUPLER LINK</td>
</tr>
<tr>
<td>PHI4</td>
<td>REAL</td>
<td>GLOBAL ANGLE OF THE OUTPUT LINK</td>
</tr>
<tr>
<td>BETA</td>
<td>REAL</td>
<td>ANGLE BETWEEN TWO ADJACENT LINKAGES</td>
</tr>
<tr>
<td>Q1, QQ1</td>
<td>REAL</td>
<td>RELATIVE ANGLE OF CRANK</td>
</tr>
<tr>
<td>QD1, QQD1</td>
<td>REAL</td>
<td>RELATIVE ANGULAR VELOCITY OF CRANK</td>
</tr>
<tr>
<td>QDD1, QQDD1</td>
<td>REAL</td>
<td>RELATIVE ANGULAR ACCEL. OF CRANK</td>
</tr>
<tr>
<td>T0</td>
<td>REAL</td>
<td>STARTING TIME</td>
</tr>
<tr>
<td>DTIME</td>
<td>REAL</td>
<td>TIME STEP</td>
</tr>
<tr>
<td>T1</td>
<td>REAL</td>
<td>FINISH TIME</td>
</tr>
<tr>
<td>NB</td>
<td>INTEGER</td>
<td>NUMBER OF BRANCHES IN A SYSTEM</td>
</tr>
<tr>
<td>NL</td>
<td>INTEGER</td>
<td>NUMBER OF LINKAGES IN EACH BRANCH</td>
</tr>
<tr>
<td>IBN</td>
<td>INTEGER</td>
<td>ADDITIONAL BRANCH NUMBER</td>
</tr>
<tr>
<td>ICV</td>
<td>INTEGER</td>
<td>BRANCH NUMBER TO BE CONNECTED</td>
</tr>
<tr>
<td>ILN</td>
<td>INTEGER</td>
<td>LINKAGE NUMBER TO BE CONNECTED</td>
</tr>
<tr>
<td>ITYPE</td>
<td>INTEGER</td>
<td>TYPE OF LINKAGE</td>
</tr>
<tr>
<td>(1: LEADING, 2: LAGGING, 3: R1=R2=R3=R4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ILINK</td>
<td>INTEGER</td>
<td>CONNECTED LINK NUMBER</td>
</tr>
<tr>
<td>(2: LINK2, 3: LINK3, 4: LINK4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ISIDE</td>
<td>INTEGER</td>
<td>CONNECTED SIDE</td>
</tr>
<tr>
<td>(1: LEFT, 2: RIGHT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDRIVER</td>
<td>INTEGER</td>
<td>FLAG OF ANGULAR VELOCITY AND ACCEL.</td>
</tr>
<tr>
<td>(0: GLOBAL, 1: RELATIVE, 2: DEPENDENT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SM2</td>
<td>REAL</td>
<td>MASS OF INPUT LINK (LINK2)</td>
</tr>
<tr>
<td>U2</td>
<td>REAL</td>
<td>MOMENT OF INERTIA OF INPUT LINK</td>
</tr>
<tr>
<td>SJ2</td>
<td>REAL</td>
<td>GENERALIZED MOMENT OF INERTIA OF LINK2</td>
</tr>
<tr>
<td>DJ2</td>
<td>REAL</td>
<td>PARTIAL DERIVATIVE OF &quot;SJ2&quot; W.R.T. Q2</td>
</tr>
<tr>
<td>QPJACO</td>
<td>REAL</td>
<td>JACOBIAN Q</td>
</tr>
<tr>
<td>PJACO</td>
<td>REAL</td>
<td>JACOBIAN P</td>
</tr>
<tr>
<td>PGAMA</td>
<td>REAL</td>
<td>RIGHT SIDE OF CONSTRAINT EQUATION</td>
</tr>
<tr>
<td>TOQ</td>
<td>REAL</td>
<td>APPLIED TORQUE</td>
</tr>
</tbody>
</table>
GENFORCE REAL GENERALIZED FORCE
SKINEN REAL KINETIC ENERGY
POTEN REAL POTENTIAL ENERGY
TOTALEN REAL MECHANICAL ENERGY

*********************************************************
C MAIN PROGRAM
C EXTERNAL SUBROUTINES:
C DINPUT — FOR ACQUIRING INPUT DATA FOR ALL LINKAGES
C DCHECK — CHECKING OF LINK LENGTHS (GRASHOF'S LAW)
C ERROR — ERROR MESSAGE
C KINEAMT — KINEMATIC ANALYSIS
C DOUTPUT — FOR DATA OUTPUT
C CPUITM — MEASUREMENT OF THE CPU TIME
C INITL — SETTING INITIAL CONDITION FOR INTEGRATION
C RUNGK4 — THE FOURTH-ORDER RUNGE-KUTTA ALGORITHM
C DIFEQN — COMPUTING FIRST ORDER DIFFERENTIAL EQUATIONS
C RSD — FUNCTION OF ROTATIONAL SPRING-DAMPER ACTUATOR
C TRIG — COMPUTING TRIGONOMETRIC TERMS
C CENTER — COMPUTING VEL. AND ACCEL. OF MASS CENTER
C LOWERLINK — COMPUTING EQ. OF MOTION OF LOWER LINKAGE
C UPERLINK — COMPUTING EQ. OF MOTION OF UPPER LINKAGE
C CONSTRAIN — COMPUTING CONSTRAINT DIFFERENTIAL EQUATIONS
C MATRIX — SOLVING THE SYSTEM EQUATIONS OF MOTION
C SOLVE — ARRANGING THE ENTRIES OF MASS MATRIX
C LINEAR — COMPUTING THE SOLUTION OF EQUATION AX=C
C LU — L-U FACTORIZATION ALGORITHM
C ENERGY — KINETIC, POTENTIAL AND MECHANICAL ENERGIES
C *********************************************************
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/Q1D(5,5),Q2D(5,5),Q3D(5,5),Q4D(5,5),VEL(5,5)
COMMON/BLOC3/Q1DD(5,5),Q2DD(5,5),Q3DD(5,5),Q4DD(5,5),ACCEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/TRANS/QQ1(5,5),QQ1D(5,5),QQ1DD(5,5),RR2(5,5),RR4(5,5)
COMMON/BLOC5/T0,TE,DTIME,T
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/BLOC7/XX1(5,5),XX4(5,5),YY1(5,5),YY4(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC10/SM1(5,5),SM2(5,5),SM3(5,5),SM4(5,5)
COMMON/TORQUE/TOQ1(5,5),TOQ2(5,5),TOQ3(5,5),TOQ4(5,5)
COMMON/TORQUEIN/TOQ1IN(5,5),TOQ2IN(5,5),TOQ3IN(5,5),TOQ4IN(5,5)
COMMON/BLOC11/R23,R24,R123,R124,R2423,TR23,TR24,SIN2,
* DYNAMIC ANALYSIS OF FOUR-BAR LINKAGES *

WRITE(5,11)

11 FORMAT(/' * DYNAMIC ANALYSIS OF FOUR-BAR LINKAGES *',
+ /3X,37(' -')//)

CALL CPUTM(SEC)
TIMEB = SEC

C....READ DATA FROM INPUT FILE ......
CALL DINPUT

C DATA
T=T0
H=DTIME
NSTEP=INT((TE-T0)/DTIME)+1
N = 4
C... FOR ONE BRANCH
   IF(NB .EQ. 1) N = 2
   NN = N/2
C -------- POINTERS --------
   N1=1
   N2=N1+N
   N3=N2+N
   N4=N3+N
   N5=N4+N
   N6=N5+N
   N7=N6+N
   N8=N7+N
C ---- INITIAL CONDITION ----
   CALL INITL(NB,NN,N,D(N1))
C ------- INTEGRATION -------
   CALL RUNGK4(T,H,NSTEP,NN,N,D(N1),D(N2),D(N3),D(N4),
                 D(N5),D(N6),D(N7))
   CALL CPUTM(SEC)
   TIMEE = SEC
   TEND = TIMEE-TIMEB
   WRITE(5,*)' TOTAL CPU TIME = ',TEND,' SECS'
   WRITE(3,41)TEND
41 FORMAT(/5X,'TOTAL CPU TIME = ',F14.6,' SECS')
   CLOSE(2)
   CLOSE(3)
   STOP
END
SUBROUTINE DINPUT

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)

COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/TRANS/QQ1(5,5),QQD1(5,5),QQDD1(5,5),RR2(5,5),
+ RR4(5,5)
COMMON/BLOC5/T0,TE,DTIME,T
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/BLOC7/XX1(5,5),XX4(5,5),YY1(5,5),YY4(5,5),
+ XX2(5,5),XX3(5,5),YY2(5,5),YY3(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5),
+ IDRIVER(5),ICB(5),IL(5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC10/S1(5,5),S2(5,5),S3(5,5),S4(5,5),
+ Ul(5,5),U2(5,5),U3(5,5),U4(5,5)
COMMON/TORQUE/TOQ1(5,5),TOQ2(5,5),TOQ3(5,5),TOQ4(5,5)
COMMON/TORQUEIN/TOQ1IN(5,5),TOQ2IN(5,5),TOQ3IN(5,5),
+ TOQ4IN(5,5)
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5)
+ ,PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC20/NRSD,IRSD(20),IRSDL(20),IRSDI(20),
+ IRSDJ(20),RS(20),DC(20),ZTE(20)
COMMON/BLOC21/FL1(5,5),FL4(5,5),S1(5,5),S4(5,5)
READ(2,*),NB
WRITE(3,51)NB

51 FORMAT(/5X,'NO. OF BRANCHES IN THE SYSTEM '/3X,I3)
   DO 20 N = 1,NB
      READ(2,*)IBN,ICV(N),ILN(N),ITYPE(N),ILINK(N),ISIDE(N)
      WRITE(3,61)N,IBN,ICV(N),ILN(N),ITYPE(N),ILINK(N),ISIDE(N)
   61 FORMAT(5X,'CONNECTIVITY DETAILS ....(BRANCH ',13,' )',
      + /5X,'IBN, ICV,ILN, ITYPE, ILINK, ISIDE',
      + /6(3X,I3))
      READ(2,*)NL(N)
      WRITE(3,71)N,NL(N)

71 FORMAT(5X,'NO. OF LINKAGES IN BRANCH ',I3/3X,I3)
   DO 30 K = 1,NL(N)
      READ(2,*)XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
      WRITE(3,81)N
   81 FORMAT(5X,'BRANCH # ',I3)
      WRITE(3,91)K,XX1(N,K),YY1(N,K),XX4(N,K),YY4(N,K)
   91 FORMAT(5X,'LOCAL COORDINATES FOR GROUND LINK',
      + '....(LINKAGE ',I3,') '/4(2X,F10.4))
      IF(ILINK(N).EQ.3) THEN
         WRITE(3,82)
      82 FORMAT(5X,'THE 1ST LINKAGE CONNECTS TO THE LINK3 OF',
         + ' LINKAGE ',I3,' OF BRANCH',I3)
C
FL1(N,K) = SQRT(XX1(N,K) * XX1(N,K) + YY1(N,K) * YY1(N,K))
FL4(N,K) = SQRT(XX4(N,K) * XX4(N,K) + YY4(N,K) * YY4(N,K))
IF(FL1(N,K) .EQ. 0.0) THEN
  S1(N,K) = 0.0
ELSE
  S1(N,K) = ATAN2(YY1(N,K), XX1(N,K))
ENDIF
IF(FL4(N,K) .EQ. 0.0) THEN
  S4(N,K) = 0.0
ELSE
  S4(N,K) = ATAN2(YY4(N,K), XX4(N,K))
ENDIF
IF(K.EQ.1) THEN
  READ(2,*) IDRIVER(N), ANGLE(N,1), VEL(N,1)
  WRITE(3,112) IDRIVER(N), ANGLE(N,1), VEL(N,1)
  IF(IDRIVER(N) .EQ. 2) THEN
    READ(2,*) ICB(N), IL(N)
    WRITE(3,115) IL(N), ICB(N)
  ENDIF
ENDIF
ELSE
  C .. INITIALIZE THE ANGULAR VEL. AND ACCEL. OF GROUND LINK
  PHI1D(N,K) = 0.0
  PHI1DD(N,K) = 0.0
ENDIF
READ(2,*) R2(N,K), R3(N,K), R4(N,K), BETA(N,K)
WRITE(3,101) K, R2(N,K), R3(N,K), R4(N,K), BETA(N,K)
READ(2,*) SM1(N,K), SM2(N,K), SM3(N,K), SM4(N,K)
WRITE(3,103) SM1(N,K), SM2(N,K), SM3(N,K), SM4(N,K)
READ(2,*) U1(N,K), U2(N,K), U3(N,K), U4(N,K)
WRITE(3,105) U1(N,K), U2(N,K), U3(N,K), U4(N,K)
READ(2,*) TOQ1(N,K), TOQ2(N,K), TOQ3(N,K), TOQ4(N,K)
WRITE(3,108) TOQ1(N,K), TOQ2(N,K), TOQ3(N,K), TOQ4(N,K)
C .... TRANSFER TOQ INTO TOQIN
  TOQ1IN(N,K) = TOQ1(N,K)
  TOQ2IN(N,K) = TOQ2(N,K)
  TOQ3IN(N,K) = TOQ3(N,K)
  TOQ4IN(N,K) = TOQ4(N,K)
C .... TRANSFER INPUT DATA OF LINK
  IF(ISIDE(N) .EQ. 1) THEN
RR4(N,K) = R2(N,K)
RR2(N,K) = R4(N,K)
ELSE
RR2(N,K) = R2(N,K)
RR4(N,K) = R4(N,K)
ENDIF
N2 = N
K2 = K
IF(N.GT.l) GOTO 30
IF(K.GT.l) GOTO 30
READ(2,* ) ANGLE(1,1),VEL(1,1)
WRITE(3,111) ANGLE(1,1),VEL(1,1)
111 FORMAT(/5X,'INITIAL GLOBAL ANGLE AND VELOCITY OF CRANK +'/2(2X,F12.4))
READ(2,* )T0,TE,DTIME
WRITE(3,121)
121 FORMAT(5X,'START TIME, FINISH TIME AND TIME INCREMENT')
WRITE(3,131)T0,TE,DTIME
131 FORMAT(/3(F10.4,4X))
30 CONTINUE
20 CONTINUE
READ(2,* ) NRSD
WRITE(3,145) NRSD
145 FORMAT(/5X,'NO. OF ROTATIONAL SPRING-DAMPER ACTUATORS',I3)
IF(NRSD .GE. 1) THEN
WRITE(3,147)
147 FORMAT(/5X,'BRANCH #,LINKAGE #,LINK I, LINK J', +'/ ',SP. STIFF.,DAMP. COEF.,ZERO TORQUE ANGLE')
DO 150 NSD = 1,NRSD
READ(2,* )IRSDB(NSD),IRSDL(NSD),IRSDI(NSD),IRSDJ(NSD), + RS(NSD),DC(NSD),ZTE(NSD)
WRITE(3,155) IRSDB(NSD),IRSDL(NSD),IRSDI(NSD), + IRSDJ(NSD),RS(NSD),DC(NSD),ZTE(NSD)
155 FORMAT(/5X,4I7,1X,3F12.4)
150 CONTINUE
ENDIF
READ(2,* )NPOS,NVEL,NACC
IF((NPOS .NE. 1) .AND. (NPOS .NE. 0)) NPOS=1
IF((NVEL .NE. 1) .AND. (NVEL .NE. 0)) NVEL=1
IF((NACC .NE. 1) .AND. (NACC .NE. 0)) NACC=1
WRITE(3,141) NPOS,NVEL,NACC
141 FORMAT(/5X,'THE FOLLOWING INDICATE THE CODES FOR PRINT', +' REQUEST'/5X,'"1" INDICATES "PRINT". "0" INDICATES ', +' "DO NOT PRINT"'/5X,'(DEFAULT VALUE IS "1")'/5X, +' NPOSITION =','I2,' NVELOCITY =','I2,' NACCEL. =','I2//)
1 RETURN
END
SUBROUTINE KINEMAT

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC3/QDD1(5,5),QDD2(5,5),QDD3(5,5),QDD4(5,5),ACCEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/TRANS/QQ1(5,5),QQD1(5,5),QQDD1(5,5),RR2(5,5),RR4(5,5)
COMMON/BLOC5/T0,TE,DTIME,T
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/BLOC7/XX1(5,5),XX4(5,5),YY1(5,5),YY4(5,5),XX2(5,5),XX3(5,5),YY2(5,5),YY3(5,5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5)
PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC21/FL1(5,5),FL4(5,5),S1(5,5),S4(5,5)
DIMENSION A(30),QQ2(7,7),QQ3(7,7),QQ4(7,7),QQD2(7,7)
+ QQDD2(7,7),QQDD3(7,7),QQDD4(7,7)

PI = 3.141592653
PI2 = 6.283185307
QQ1(1,1) = ANGLE(1,1)
DO 50 I = 1,NB
IF(I.GT.1) THEN
  IF(ILINK(I) - 3) 10,30,20
10 QQ1(I,1)=Q1(ICV(I),ILN(I))+PHI1(ICV(I),ILN(I))
   QQD1(I,1)=QD1(ICV(I),ILN(I))
   QQDD1(I,1)=QDD1(ICV(I),ILN(I))
   GO TO 35
20 QQ1(I,1)=PI-Q2(ICV(I),ILN(I))+PHI1(ICV(I),ILN(I))
   QQD1(I,1) = -QD2(ICV(I),ILN(I))
   QQDD1(I,1)=-QDD2(ICV(I),ILN(I))
   GO TO 35
C
30 QQ1(I,1) = ANGLE(1,1)
   IF(ISIDE(I).EQ.ISIDE(ICV(I))) THEN
     ISIGN = 1
   ELSE
     ISIGN = -1
   ENDFI
   DO 37 IP = 1,NL(I)
     IQ = IP - 1
     IC = ILN(I)+IQ*ISIGN
     SS1= S1(I,IP)+PHI3(ICV(I),IC)
     SS4= S4(I,IP)+PHI3(ICV(I),IC)
     XX1(I,IP)=XX2(ICV(I),IC)+FL1(I,IP)*COS(SS1)
     YY1(I,IP)=YY2(ICV(I),IC)+FL1(I,IP)*SIN(SS1)
$XX4(I,IP) = XX3(ICV(I),IC) + FL4(I,IP) \times \cos(SS4)$

$YY4(I,IP) = YY3(ICV(I),IC) + FL4(I,IP) \times \sin(SS4)$

$\text{PHI1D}(I,IP) = \text{PHI3D}(ICV(I),IC)$

$\text{PHI1DD}(I,IP) = \text{PHI3DD}(ICV(I),IC)$

$\text{IF}(\text{IDRIVER}(I) - 1) = \{2,3,4\}$

$QQD1(1,1) = VEL(1,1) - \text{PHI1D}(1,1)$

$QQDD1(1,1) = VEL(1,1) - \text{PHI1DD}(1,1)$

$\text{GO TO 35}$

$QQD1(1,1) = VEL(1,1)$

$QQDD1(1,1) = VEL(1,1)$

$\text{GO TO 35}$

$QQ1(1,1) = \pi - Q2(ICB(I),IL(I)) + \text{PHI1}(ICB(I),IL(I))$

$QQD1(1,1) = -QD2(ICB(I),IL(I))$

$QQDD1(1,1) = -QDD2(ICB(I),IL(I))$

$\text{IF}(\text{ISIDE}(I) . \text{EQ.} 1) \text{ THEN}$

$QQ1(I,1) = \pi - QQ1(I,1)$

$QQD1(I,1) = -QQD1(I,1)$

$QQDD1(I,1) = -QQDD1(I,1)$

$\text{ENDIF}$

$\text{ENDIF}$

$\text{DO } J = 1,\text{NL}(I)$

$Y41 = YY4(I,J) - YY1(I,J)$

$X41 = XX4(I,J) - XX1(I,J)$

$R1(I,J) = \sqrt{X41^2 + Y41^2}$

$\text{CHECK GRASHOF'S LAW}$

$\text{IF}(T . \text{NE.} T0) \text{ GOTO 33} \quad ! \text{In some cases of applications}$

$\text{IF}(ILINK(I). \text{EQ.} 3) \text{ THEN} \quad ! \text{DCHECK can be neglected.}$

$\text{IF}((J . \text{EQ.} 1) . \text{AND.} (\text{IDRIVER}(I) . \text{NE.} 2)) \text{ CALL DCHECK}(I,J)$

$\text{ELSE}$

$\text{IF}((I+J) . \text{EQ.} 2) \text{ CALL DCHECK}(I,J)$

$\text{ENDIF}$

$\text{ENDIF}$

$\text{PHI1}(I,J) = \text{ATAN2}(Y41,X41)$

$\text{IF}(\text{ISIDE}(I) . \text{EQ.} 1) \text{ THEN}$

$\text{PHIIJ} = -\text{PHI1}(I,J)$

$\text{ELSE}$

$\text{PHIIJ} = \text{PHI1}(I,J)$

$\text{ENDIF}$

$\text{QQ1}(I,J) = \text{QQ1}(I,J) - \text{BETA}(I,J) - \text{PHIIJ}$

$\text{ANGULAR POSITION ANALYSIS} \ldots$\ldots$

$\text{SIN1} = \text{SIN}(\text{QQ1}(I,J))$

$\text{COS1} = \text{COS}(\text{QQ1}(I,J))$

$A(1) = (R3(I,J) \times R3(I,J) + R4(I,J) \times R4(I,J) - R1(I,J) \times R1(I,J) + R2(I,J) \times R2(I,J)) / (2 \times R3(I,J) \times R4(I,J))$

$A(2) = R1(I,J) / (R3(I,J) \times R4(I,J))$

$A(3) = R2(I,J) \times \text{SIN1}$
A(4) = RR2(I,J)*COS1
A(5) = ACOS(A(1)+A(2)*A(4))
C
IF((R1(I,J).EQ.R3(I,J)).AND.(R2(I,J).EQ.R4(I,J))) THEN
A(8) = 0.0
ELSE
A(6) = SIN(A(5))
A(7) = COS(A(5))
C
RR4A6 = RR4(I,J)*A(6)
C R3A6 = R3(I,J)*A(6)
P122 = 0.0
IF(ITYPE(I) - 2) 100,55,40
40 IF(QQ1(1,1) .LE. PI ) GO TO 100
IF(QQ1(1,1) .GT. PI2) GO TO 100
55 RR4A6 = -RR4A6
C R3A6 = -R3A6
P122 = PI2
100 AA8 = -A(3)+RR4A6
BB8 = -A(4)+R3(I,J)+R1(I,J)-RR4(I,J)*A(7)
C AA9 = A(3)-R3A6
C BB9 = A(4)+RR4(I,J)-R1(I,J)-R3(I,J)*A(7)
A(8) = 2*ATAN2(AA8,BB8)+PI22
C A(9) = 2*ATAN2(AA9,BB9)+PI22
ENDIF
C
QQ3(I,J) = A(5)
QQ4(I,J) = PI - QQ1(I,J) +A(8)
C....FOR THE SPECIAL CASES OF R1=R4 AND R2=R3, etc.
A(9) = QQ1(I,J)+QQ3(I,J)+QQ4(I,J)-PI+PI22
QQ2(I,J) = PI - A(9)
IF(ISIDE(I) .EQ. 1) THEN
PHI2(I,J) = QQ2(I,J)+PHI1(I,J)
PHI3(I,J) = -A(8)+PHI1(I,J) + PI2
PHI4(I,J) = PI-QQ1(I,J)+PHI1(I,J)
ELSE
PHI2(I,J) = QQ1(I,J)+PHI1(I,J)
PHI3(I,J) = A(8)+PHI1(I,J)
PHI4(I,J) = A(9)+PHI1(I,J)
ENDIF
C....CALCULATE THE COORD. OF EACH END OF LINK
XX2(I,J) = XX1(I,J)+R2(I,J)*COS(PHI2(I,J))
YY2(I,J) = YY1(I,J)+R2(I,J)*SIN(PHI2(I,J))
XX3(I,J) = XX4(I,J)+R4(I,J)*COS(PHI4(I,J))
YY3(I,J) = YY4(I,J)+R4(I,J)*SIN(PHI4(I,J))
C...COMPUTE THE ENTRIES OF JACOBIAN MATRIX
SIN2 = SIN(QQ2(I,J))
SIN4 = SIN(QQ4(I,J))
COS2 = COS(QQ2(I,J))
COS4 = COS(QQ4(I,J))
SIN12 = SIN1*COS2+COS1*SIN2
\[ \sin 14 = \sin 1 \cdot \cos 4 + \cos 1 \cdot \sin 4 \]
\[ \cos 12 = \cos 1 \cdot \cos 2 - \sin 1 \cdot \sin 2 \]
\[ \cos 14 = \cos 1 \cdot \cos 4 - \sin 1 \cdot \sin 4 \]

C

\[ \text{IF} ((\text{R1}(I,J) \cdot \text{EQ.} \cdot \text{R3}(I,J)) \cdot \text{AND.} \cdot (\text{R2}(I,J) \cdot \text{EQ.} \cdot \text{R4}(I,J))) \text{ THEN} \]
\[ \text{QQD2}(I,J) = -\text{QQD1}(I,J) \]
\[ \text{QQD3}(I,J) = \text{QQD1}(I,J) \]
\[ \text{QQD4}(I,J) = -\text{QQD1}(I,J) \]

ELSE
\[ A(10) = 2 \cdot \text{RR2}(I,J) \cdot \text{RR4}(I,J) \]
\[ A(11) = 2 \cdot \text{R1}(I,J) \cdot \text{R3}(I,J) \]
\[ A(12) = 2 \cdot \text{RR2}(I,J) \cdot \text{R3}(I,J) \]
\[ A(13) = 2 \cdot \text{R1}(I,J) \cdot \text{RR2}(I,J) \]
\[ A(14) = 2 \cdot \text{R1}(I,J) \cdot \text{RR4}(I,J) \]
\[ A(15) = -A(10) \cdot \sin 12 + A(13) \cdot \sin 1 \]
\[ A(16) = A(13) \cdot \sin 1 \]
\[ A(17) = -A(11) \cdot \sin 14 + A(13) \cdot \sin 1 \]
\[ A(18) = -A(10) \cdot \sin 12 + A(14) \cdot \sin 2 \]
\[ A(19) = -2 \cdot \text{R3}(I,J) \cdot \text{RR4}(I,J) \cdot \sin (\text{QQ3}(I,J)) \]
\[ A(20) = -A(11) \cdot \sin 14 + A(12) \cdot \sin 4 \]

C ... VELOCITY COMPUTATION
\[ \text{QQD1}(I,J) = \text{QQD1}(I,J) \]
\[ \text{QQD2}(I,J) = -A(15) \cdot \text{QQD1}(I,J) / A(18) \]
\[ \text{QQD3}(I,J) = -A(16) \cdot \text{QQD1}(I,J) / A(19) \]
\[ \text{QQD4}(I,J) = -A(17) \cdot \text{QQD1}(I,J) / A(20) \]

ENDIF

C ... TRANSFER QQ (INTERNAL VARIABLES) TO Q (EXTERNAL VARIABLES)
\[ \text{IF} (\text{ISIDE}(I) \cdot \text{EQ.} \cdot 1) \text{ THEN} \]
\[ Q2(I,J) = \text{QQ1}(I,J) \]
\[ Q1(I,J) = \text{QQ2}(I,J) \]
\[ Q4(I,J) = \text{QQ3}(I,J) \]
\[ Q3(I,J) = \text{QQ4}(I,J) \]
\[ QD2(I,J) = \text{QQD1}(I,J) \]
\[ QD1(I,J) = \text{QQD2}(I,J) \]
\[ QD4(I,J) = \text{QQD3}(I,J) \]
\[ QD3(I,J) = \text{QQD4}(I,J) \]

ELSE
\[ Q1(I,J) = \text{QQ1}(I,J) \]
\[ Q2(I,J) = \text{QQ2}(I,J) \]
\[ Q3(I,J) = \text{QQ3}(I,J) \]
\[ Q4(I,J) = \text{QQ4}(I,J) \]
\[ QD1(I,J) = \text{QQD1}(I,J) \]
\[ QD2(I,J) = \text{QQD2}(I,J) \]
\[ QD3(I,J) = \text{QQD3}(I,J) \]
\[ QD4(I,J) = \text{QQD4}(I,J) \]

ENDIF

C ... TRANSFER VELOCITY FROM RELATIVE COORD. TO GLOBAL COORD.
\[ \text{PHI2D}(I,J) = \text{QD1}(I,J) + \text{PHI1D}(I,J) \]
\[ \text{PHI3D}(I,J) = \text{QD1}(I,J) + \text{QD4}(I,J) + \text{PHI1D}(I,J) \]
\[ \text{PHI4D}(I,J) = -\text{QD2}(I,J) + \text{PHI1D}(I,J) \]
IF(NFLAG .EQ. 2) THEN
C .. TIME DERIVATIVES OF THE JACOBIAN MATRIX
  IF((R1(I,J) .EQ. R3(I,J)) .AND. R2(I,J) .EQ. R4(I,J)) THEN
    QQDD2(I,J) = -QQDDI(I,J)
    QQDD3(I,J) = QQDD1(I,J)
    QQDD4(I,J) = -QQDD1(I,J)
  ELSE
    A(21) = QQD1(I,J) + QQD2(I,J)
    A(22) = QQD1(I,J) + QQD4(I,J)
    A(23) = -A(10)*A(21)*COS12+A(13)*QQD1(I,J)*COS1
    A(24) = A(13)*QQD1(I,J)*COS1
    A(25) = -A(11)*A(22)*COS14+A(13)*QQD1(I,J)*COS1
    A(26) = -A(11)*A(22)*COS12+A(14)*QQD2(I,J)*COS2
    A(27) = -2*R3(I,J)*RR4(I,J)*QQD3(I,J)*COS(QQ3(I,J))
    A(28) = -A(11)*A(22)*COS14+A(12)*QQD4(I,J)*COS4
  ENDIF
C .. ACCELERATION CALCULATION
  QQDD1(I,J) = QQDD1(I,J)
  QQDD2(I,J) = (-A(23)*QQD1(I,J)-A(26)*QQD2(I,J)-A(15)+
                 *QQDD1(I,J))/A(18)
  QQDD3(I,J) = (-A(24)*QQD1(I,J)-A(27)*QQD3(I,J)-A(16)+
                 *QQDD1(I,J))/A(19)
  QQDD4(I,J) = (-A(25)*QQD1(I,J)-A(28)*QQD4(I,J)-A(17)+
                 *QQDD1(I,J))/A(20)
ENDIF
C .. TRANSFER QQ (INTERNAL VARIABLES) TO Q (EXTERNAL VARIABLES)
  IF(ISIDE(I) .EQ. 1) THEN
    QQ2(I,J) = QQDD2(I,J)
    QQ3(I,J) = QQDD3(I,J)
    QQ4(I,J) = QQDD4(I,J)
  ELSE
    QQ1(I,J) = QQDD1(I,J)
    QQ2(I,J) = QQDD2(I,J)
    QQ3(I,J) = QQDD3(I,J)
    QQ4(I,J) = QQDD4(I,J)
  ENDIF
C .. TRANSFER ACCEL. FROM RELATIVE TO GLOBAL COORD.
  PHI2DD(I,J) = QQD1(I,J)+PHI1DD(I,J)
  PHI3DD(I,J) = QQD1(I,J)+QQD4(I,J)+PHI1DD(I,J)
  PHI4DD(I,J) = -QQD2(I,J)+PHI1DD(I,J)
C
  QQDD1(I,J+1) = -QQDD2(I,J)
ENDIF
  QQL(I,J+1) = PI - QQ2(I,J) + PHIJ
  QQD1(I,J+1) = -QQD2(I,J)
60 CONTINUE
50 CONTINUE
RETURN
END
SUBROUTINE DCHECK(N2, K2)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/TRANS/QQ1(5,5),QQD1(5,5),QQDD1(5,5),RR2(5,5),
+ RR4(5,5)
DIMENSION X(5)
TOLENCE = 0.001
X(1) = R1(N2,K2)
X(2) = RR2(N2,K2)
X(3) = R3(N2,K2)
X(4) = RR4(N2,K2)

C

Z = X(1)+X(2)+X(3)+X(4)
BIG = X(1)
SMALL = X(1)
DO 40 I = 1,3
   I2 = I+1
   IF(X(I2) .GT. BIG) BIG = X(I2)
   IF(X(I2) .LT. SMALL) SMALL = X(I2)
40 CONTINUE
Y = BIG + SMALL
W = Z - Y + TOLENCE
IF(Y.GT.W) CALL ERROR(N2,K2)
RETURN
END

SUBROUTINE ERROR(N3,K3)

C

WRITE(5,151)
151 FORMAT(//,80('*')//)
WRITE(5,*).' *** ERROR IN INPUT DATA ? ***'
WRITE(5,*)' LINK LENGTHS DO NOT SATISFY GRASHOF'S LAW'
WRITE(5,161) N3,K3
161 FORMAT('LINK LENGTHS OF BRANCH ',I2,' LINKAGE ',I2,
+ ' ARE INCORRECT')
WRITE(5,*)'PLEASE VERIFY THE DATA OR MODIFY THE PROGRAM'
WRITE(5,*)
STOP
END
SUBROUTINE DOUTPUT

IMPLICIT DOUBLE PRECISION(A-H,O-Z)

COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC3/QDD1(5,5),QDD2(5,5),QDD3(5,5),QDD4(5,5),
+ ACCEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/BLOC5/T0,TE,DTIME,T
+ COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
+ COMMON/BLOC7/XX1(5,5),XX4(5,5),YY1(5,5),YY4(5,5),
+ XX2(5,5),XX3(5,5),YY2(5,5),YY3(5,5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC10/SM1(5,5),SM2(5,5),SM3(5,5),SM4(5,5),
+ U1(5,5),U2(5,5),U3(5,5),U4(5,5)
COMMON/BLOC14/SKINEN,POTEN,TOTALEN,VARIEN,CONSTEN
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5),
+ PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)

IF(T.GT.T0) GOTO 5
WRITE(3,171)
171 FORMAT(5X,'LINK LENGTHS, ADJACENT ANGLE, MASS AND',
+ ' MOMENT OF INERTIA')
WRITE(3,181)
181 FORMAT(/5X,'BRANCH #',2X,'LINKAGE #',4X,'R1',6X,'R2',6X,'R3',6X,
+ 'R4',6X,'M1',6X,'M2',6X,'M3',6X,'M4',6X,
+ '/28X,'U1',6X,'U2',6X,'U3',6X,'U4'/)
DO 70 M = 1,NB
DO 80 L = 1,NL(M)
IF(L.GT.L) GOTO 15

15 WRITE(3,201)M,
+ L,R1(M,L) ,R2(M,L) ,R3(M,L) ,R4(M,L) ,
+ BETA(M,L),SM1(M,L),SM2(M,L),SM3(M,L),SM4(M,L),
+ U1(M,L),U2(M,L),U3(M,L),U4(M,L)
201 FORMAT(16X,I3,5X,F8.3,4(F8.3),2(/24X,4(F8.3))/)
GOTO 80
80 CONTINUE
70 CONTINUE

IF(NPOS.EQ.0) GOTO 6
WRITE(3,221)
221 FORMAT(5X,'POSITION ANGLES OF INPUT AND OUTPUT LINKS')
WRITE(3,231)
231 FORMAT(/5X,'BRANCH #',5X,'LINKAGE #',4X,'Q1',6X,'Q2',6X,
+ 'Q3',6X,'Q4',10X,'Q5',10X,'Q6',10X,'Q7',10X,'Q8')
DO 90 M1 = 1,NB
DO 100 L1 = 1,NL(M1)
90 CONTINUE
100 CONTINUE
5 WRITE(3,211) T
211 FORMAT(/5X,'*** TIME = ',(1PE12.4),'/5X','-----'/)
IF(NPOS.EQ.0) GOTO 6
WRITE(3,221)
221 FORMAT(5X,'POSITION ANGLES OF INPUT AND OUTPUT LINKS')
WRITE(3,231)
231 FORMAT(/5X,'BRANCH #',5X,'LINKAGE #',4X,'Q1',6X,'Q2',6X,
+ 'Q3',6X,'Q4',10X,'Q5',10X,'Q6',10X,'Q7',10X,'Q8')
DO 90 M1 = 1,NB
DO 100 L1 = 1,NL(M1)
90 CONTINUE
100 CONTINUE
IF(L1.GT.1) GOTO 39
WRITE(3,241)M1,L1,PHI1(M1,L1),PHI2(M1,L1),PHI3(M1,L1),
+ PHI4(M1,L1),Q1(M1,L1),Q2(M1,L1)
+ Q4(M1,L1),Q3(M1,L1)
241 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
WRITE(3,504)XX1(M1,L1),XX2(M1,L1),XX3(M1,L1),XX4(M1,L1),
+ YY1(M1,L1),YY2(M1,L1),YY3(M1,L1),YY4(M1,L1)
WRITE(3,251)L1,PHI1(M1,L1),PHI2(M1,L1),PHI3(M1,L1),
+ PHI4(M1,L1),Q1(M1,L1),Q4(M1,L1),Q3(M1,L1)
+ Q2(M1,L1)
251 FORMAT(19X,I3,3X,8(1PE12.4))
WRITE(3,504)XX1(M1,L1),XX2(M1,L1),XX3(M1,L1),XX4(M1,L1),
+ YY1(M1,L1),YY2(M1,L1),YY3(M1,L1),YY4(M1,L1)
GOTO 100
39 WRITE(3,251)L1,PHI1(M1,L1),PHI2(M1,L1),PHI3(M1,L1),
+ PHI4(M1,L1),Q1(M1,L1),Q4(M1,L1),Q3(M1,L1)
+ Q2(M1,L1)
504 FORMAT(/33X,'XI = ',1PE12.4,8X,'X2 = ',1PE12.4,8X,'X3 = ',1PE12.4,8X,
+ 'X4 = ',1PE12.4,8X,'Y1 = ',1PE12.4,8X,'Y2 = ',1PE12.4,8X,'Y3 = ',1PE12.4,8X,
+ 'Y4 = ',1PE12.4/)}
100 CONTINUE
90 CONTINUE
6 IF(NVEL.EQ.0) GOTO 7
WRITE(3,261)
261 FORMAT(/5X,'ANGULAR VEL. OF INPUT AND OUTPUT LINKS')
WRITE(3,271)
271 FORMAT(5X,'BRANCH #',5X,'LINKAGE #',3X,'PHI1D',5X,
+ 'PHI2D',5X,'PHI3D',5X,'PHI4D',9X,'QD1',9X,'QD4',
+ 9X,'QD3',9X,'QD2')
DO 200 M2 = 1,NB
DO 300 L2 = 1,NL(M2)
IF(L2.GT.1) GOTO 38
WRITE(3,281)M2,L2,PHI1D(M2,L2),PHI2D(M2,L2),PHI3D(M2,L2),
+ PHI4D(M2,L2),QD1(M2,L2),QD4(M2,L2),QD3(M2,L2),QD2(M2,L2)
281 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
GOTO 300
38 WRITE(3,291)L2,PHI1D(M2,L2),PHI2D(M2,L2),PHI3D(M2,L2),
+ PHI4D(M2,L2),QD1(M2,L2),QD4(M2,L2),QD3(M2,L2),QD2(M2,L2)
291 FORMAT(19X,I3,3X,8(1PE12.4))
300 CONTINUE
200 CONTINUE
7 IF(NACC.EQ.0) GOTO 69
WRITE(3,301)
301 FORMAT(/5X,'ANGULAR ACCEL. OF INPUT AND OUTPUT LINKS')
WRITE(3,311)
311 FORMAT(5X,'BRANCH #',5X,'LINKAGE #',2X,'PHI1DD',4X,
+ 'PHI2DD',4X,'PHI3DD',4X,'PHI4DD',8X,'QDD1',8X,
+ 'QDD4',8X,'QDD3',8X,'QDD2')
DO 400 M3 = 1,NB
DO 500 L3 = 1,NL(M3)
IF(L3.GT.1) GOTO 36
WRITE(3,321)M3,L3,PHI1DD(M3,L3),PHI2DD(M3,L3),
+ PHI3DD(M3,L3),PHI4DD(M3,L3),QDD1(M3,L3),
+ QDD4(M3,L3),QDD3(M3,L3),QDD2(M3,L3)
321 FORMAT(6X,I3,10X,I3,3X,8(1PE12.4))
GOTO 500
36 WRITE(3,331)L3,PHI1DD(M3,L3),PHI2DD(M3,L3),
+ PHI3DD(M3,L3),PHI4DD(M3,L3),QDD1(M3,L3),
+ QDD4(M3,L3),QDD3(M3,L3),QDD2(M3,L3)
331 FORMAT(19X,I3,3X,8(1PE12.4))
500 CONTINUE
400 CONTINUE
WRITE(3,710)
710 FORMAT(/5X,'KINETIC ENERGY',3X,'POTENTIAL ENERGY',3X,
+ 'MECHANICAL ENERGY',3X,'ENERGY VARIATION')
WRITE(3,720) SKINEN,POTEN,TOTALEN,VARIEN
720 FORMAT(/5X,F12.5,5X,F12.5,8X,F12.5,6X,F12.5//)
69 RETURN
END
SUBROUTINE CPUTM (SEC)
-----------------------------------
C....RETURN CPU TIME IN SECONDS
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DATA T1000 /1000.00/
C
%INCLUDE "QSYM.F77"
INTEGER*2 IBUF(0:ISYS_GRLTH)
EQUIVALENCE (ICPUMS,IBUF(ISYS_GRCH))
IAC0 = -1
IAC1 = 0
IAC2 = WORDADDR (IBUF)
IER = ISYS (ISYS_RUNTM,IAC0,IAC1,IAC2)
SEC = FLOAT (ICPUMS) / T1000
RETURN
END

SUBROUTINE INITL(NB,NN,N,Y)
-----------------------------
C
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5),
+    IDRIVER(5),ICB(5),IL(5)
DIMENSION Y(N)
Y(1)=ANGLE(1,1)
Y(NN+1) = VEL(1,1)
DO 58 I = 1,NB
   IF(ILINK(I) .EQ. 3) THEN
      II = I+NN
      Y(I) = ANGLE(I,1)
      Y(II) = VEL(I,1)
   ENDIF
58 CONTINUE
RETURN
END
SUBROUTINE RUNGK4(T,H,NSTEP,NN,N,Y,F,F1,F2,F3,F4,YY)

C IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
DIMENSION Y(N),F(N),F1(N),F2(N),F3(N),F4(N),YY(N)
HH=0.5*H
TS=T
DO 100 I=1,NSTEP
  NFLAG = 2
  CALL DIFEQN(T,NN,N,Y,F)
  CALL ENERGY(T)
  CALL DOUTPUT
  IF(I .EQ. NSTEP) RETURN
  NFLAG=0
  DO 10 J=1,N
    F1(J)=H*F(J)
    TT=T+HH
    DO 20 J=1,N
      YY(J)=Y(J)+0.5*F1(J)
      CALL DIFEQN(TT,NN,N,YY,F)
    DO 30 J=1,N
      F2(J)=H*F(J)
      YY(J)=Y(J)+0.5*F2(J)
      CALL DIFEQN(TT,NN,N,YY,F)
      TT=T+H
      DO 40 J=1,N
        F3(J)=H*F(J)
        YY(J)=Y(J)+F3(J)
        CALL DIFEQN(TT,NN,N,YY,F)
      T=TS+H*FLOAT(I)
      DO 50 J=1,N
        F4(J)=H*F(J)
      Y(J)=Y(J)+(F1(J)+2.0*F2(J)+2.0*F3(J)+F4(J))/6.0
  100 CONTINUE
END
SUBROUTINE DIFEQN(T,NN,N,Y,F)

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC3/QDD1(5,5),QDD2(5,5),QDD3(5,5),QDD4(5,5),
+ ACCEL(5,5)
COMMON/TRANS/QQ1(5,5),QQD1(5,5),QQDD1(5,5),RR2(5,5),
+ RR4(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5),
+ IDRIVER(5),ICB(5),IL(5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC20/NRSD,IRSDB(20),IRSDL(20),IRSDI(20),
+ IRSDJ(20),RS(20),DC(20),ZTE(20)
DIMENSION Y(N),F(N),DD(20)

C......TRANSFER Y TO QQ1 AND QQD1 OF THE DRIVING LINK ......
ANGLE(1,1)=Y(1)
QQD1(1,1)=Y(NN+1)
DO 30 K=1,NB
   IF(ILINK(K) .EQ. 3) THEN
      KK=K+NN
      ANGLE(K,1)=Y(K)
      VEL(K,1)=Y(KK)
   ENDIF
30 CONTINUE
C
IF(NRSD .GT. 0) CALL RSD
DO 10 I=1,NB
   DO 20 J=1,NL(I)
C
   CALL TRIG(I,J)
   CALL CENTER(I,J)
C
   IF(ILINK(I) .NE. 3) THEN
      CALL LOWERLINK(I,J)
   ELSE
      CALL UPPERLINK(I,J)
   ENDIF
20 CONTINUE
10 CONTINUE
C
CALL KINEMAT
C
CALL MATRIX(DD)
F(1)=Y(NN+1)
C
FOR ONE BRANCH
IF(NB .EQ. 1) THEN
   F(NN+1) = DD(1)
ELSE
   F(NN+1) = DD(5)
DO 40 L=1,NB
IF(ILINK(L) .EQ. 3) THEN
   LL=L+NN
   F(L)=Y(LL)
   F(LL)=DD(3)
ENDIF
40    CONTINUE
   ENDIF
C
   IF(NFLAG .EQ. 0) RETURN
C.....TRANSFER F TO QQDD AND CALCULATE OTHER REAL ACCELERATIONS
   QQDD1(1,1) = F(NN+1)
   DO 50 M=1,NB
      IF(ILINK(M) .EQ. 3) THEN
         MM=M + NN
         ACCEL(M,1)=F(MM)
      ENDIF
   50    CONTINUE
C
   CALL KINEMAT
C
   CALL INTEREST(X,Y,DDX2,DDY2,DDX3,DDY3,DDX4,DDY4)
   RETURN
END
SUBROUTINE RSD

C IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/TORQUE/TOQ1(5,5),TOQ2(5,5),TOQ3(5,5),TOQ4(5,5)
COMMON/TORQUEIN/TOQ1IN(5,5),TOQ2IN(5,5),TOQ3IN(5,5),
   TOQ4IN(5,5)
+ COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5)
+ ,PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC20/NRSD,IRSDB(20),IRSDL(20),IRSDI(20),
   IRSDJ(20),RS(20),DC(20),ZTE(20)
DIMENSION TETA(4),TETAD(4),TQ(4)

DO 60 LL=1,NRSD

C KBR = IRSDB(LL)
KLI = IRSDL(LL)
IB = IRSDI(LL)
JB = IRSDJ(LL)

C TETA(1) = PHI1(KBR,KLI)
TETA(2) = PHI2(KBR,KLI)
TETA(3) = PHI3(KBR,KLI)
TETA(4) = PHI4(KBR,KLI)

C TETAD(1) = PHI1D(KBR,KLI)
TETAD(2) = PHI2D(KBR,KLI)
TETAD(3) = PHI3D(KBR,KLI)
TETAD(4) = PHI4D(KBR,KLI)

C TQ(1) = TOQ1IN(KBR,KLI)
TQ(2) = TOQ2IN(KBR,KLI)
TQ(3) = TOQ3IN(KBR,KLI)
TQ(4) = TOQ4IN(KBR,KLI)

C DTE = TETA(IB) - ZTE(LL) - TETA(JB) ! ZTE is zero torque angle.
DTED = TETAD(IB) - TETAD(JB) ! This angle depends on the
DTOR = RS(LL) * DTE + DC(LL) * DTED ! directions of vectors I & J.
TQ(IB) = TQ(IB) - DTOR
TQ(JB) = TQ(JB) + DTOR

C TOQ1(KBR,KLI) = TQ(1)
TOQ2(KBR,KLI) = TQ(2)
TOQ3(KBR,KLI) = TQ(3)
TOQ4(KBR,KLI) = TQ(4)

60 CONTINUE
RETURN
END
SUBROUTINE TRIG(I,J)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5),
+ IDRIVER(5),ICB(5),IL(5)
COMMON/BLOC11/R23,R24,R123,R124,R2423,TR23,TR24,SIN2,COS2,
+ SIN3,COS3,SIN4,COS4,SIN24,COS24,SIN23,COS23,SIN43,COS43
T24 = PHI2(I,J)-PHI4(I,J)
T23 = PHI2(I,J)-PHI3(I,J)
T43 = PHI4(I,J)-PHI3(I,J)
SIN2= SIN(PHI2(I,J))
COS2= COS(PHI2(I,J))
SIN3= SIN(PHI3(I,J))
COS3 = COS(PHI3(I,J))
SIN4 = SIN(PHI4(I,J))
COS4 = COS(PHI4(I,J))
SIN24=SIN(T24)
COS24=COS(T24)
SIN23=SIN(T23)
COS23=COS(T23)
SIN43=SIN(T43)
COS43=COS(T43)
SIN4343=SIN43*SIN43
R23=R2(I,J)/R3(I,J)*SIN24/SIN43
R24=R2(I,J)/R4(I,J)*SIN23/SIN43
R123=1-R24
R123=1-R23
R2423=R24-R23
PR23=R2(I,J)*(COS24*SIN43*R124-SIN24*COS43*R2423)/
+ (R3(I,J)*SIN4343)
TR23 = QD1(I,J)*PR23
IF(ILINK(I) .EQ. 3) THEN
  PR24=R2(I,J)*(COS23*SIN43*R123-SIN23*COS43*R2423)/
  + (R4(I,J)*SIN4343)
  TR24 = QD1(I,J)*PR24
ENDIF
RETURN
END
SUBROUTINE CENTER(I,J)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMM/NBLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5),
+QCEL(5,5)
COMMON/BLOC3/QDD1(5,5),QDD2(5,5),QDD3(5,5),QDD4(5,5),ACCEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/BLOC6/PH1(5,5),PH2(5,5),PH3(5,5),PH4(5,5)
COMM/BlOC7/XX1(5,5),XX4(5,5),YY1(5,5),YY4(5,5),
+XX2(5,5),XX3(5,5),YY2(5,5),YY3(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILNK(5),ISIDE(5),
+IDRIVER(5),ICB(5),IL(5)
C....FOR ONE BRANCH
COMM/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMM/BLOCl1/R23,R24,R123,R124,R2423,TR23,TR24,SIN2,
+COS2,SIN3,COS3,SIN4,COS4,SIN24,COS24,
+SIN23,COS23,SIN43,COS43
COMM/BLOC15/PH1D(5,5),PH2D(5,5),PH3D(5,5),PH4D(5,5)
+PH1DD(5,5),PH2DD(5,5),PH3DD(5,5),PH4DD(5,5)
COMMON/BLOC16/Y1(5,5),Y2(5,5),Y3(5,5),Y4(5,5),DX1(5,5),
+DY1(5,5),DX2(5,5),DY2(5,5),DX3(5,5),DY3(5,5),DX4(5,5),
+DY4(5,5),DDX1(5,5),DDY1(5,5),DDX3(5,5),DDY3(5,5)
COMMON/BLOC17/R1S1,R1C1,R1S2,R1C2,R1S3,R1C3,R1S4,R1C4,
+R2S2,R2C2,R3S3,R3C3,R4S4,R4C4,R2S12,R2C12
COMMON/BLOC19/DX3DT1(5,5),DX3DT2(5,5),DDX3T(5,5),
+DY3DT1(5,5),DY3DT2(5,5),DDY3T(5,5)
IF(ILNK(I) .NE. 3) THEN
  DX1(I,J) = 0.0
  DY1(I,J) = 0.0
  DDX1(I,J) = 0.0
  DDY1(I,J) = 0.0
ELSE
  IF(ISIDE(I) .EQ. ISIDE(1C1(I))) THEN
    ISIGN = 1
  ELSE
    ISIGN = -1
  ENDF
  IQ = J-1
  IC = ILN(I)+IQ*ISIGN
  Y1(I,J) = Y3(1C1(I),IC)
  DX1(I,J) = DX3(1C1(I),IC)
  DY1(I,J) = DY3(1C1(I),IC)
  DDX1(I,J) = DDX3(1C1(I),IC)
  DDX1(I,J) = DDX3(1C1(I),IC)
ENDIF
C
RLS1 = 0.5*RL(I,J)*SIN(PHI1(I,J))
R1C1 = 0.5*RL(I,J)*COS(PHI1(I,J))
RLS2 = 0.5*RL(I,J)*SIN2
$R_1 C_2 = 0.5 R_1(I,J) \cos^2$
$R_1 S_3 = 0.5 R_1(I,J) \sin^3$
$R_1 C_3 = 0.5 R_1(I,J) \cos^3$
$R_1 S_4 = 0.5 R_1(I,J) \sin^4$
$R_2 S_2 = 0.5 R_2(I,J) \sin^2$
$R_2 C_2 = 0.5 R_2(I,J) \cos^2$
$R_3 S_3 = 0.5 R_3(I,J) \sin^3$
$R_3 C_3 = 0.5 R_3(I,J) \cos^3$
$R_4 S_4 = 0.5 R_4(I,J) \sin^4$
$R_4 C_4 = 0.5 R_4(I,J) \cos^4$

$R_2 S_{12} = 2 R_2 S_2$
$R_2 C_{12} = 2 R_2 C_2$

$Y_2(I,J) = Y_1(I,J) - R_1 S_1 + R_2 S_2$
$Y_3(I,J) = Y_1(I,J) - R_1 S_1 + R_2 S_{12} + R_3 S_3$
$Y_4(I,J) = Y_1(I,J) + R_1 S_1 + R_4 S_4$

FOR ONE BRANCH

IF (NB .EQ. 1) RETURN

$DX_2(I,J) = DX_1(I,J) + R_1 S_1 \phi_{1D}(I,J) - R_2 S_2 \phi_{2D}(I,J)$
$DY_2(I,J) = DY_1(I,J) - R_1 C_1 \phi_{1D}(I,J) + R_2 C_2 \phi_{2D}(I,J)$
$DX_3(I,J) = DX_1(I,J) + R_1 S_1 \phi_{1D}(I,J) - R_2 S_{12} \phi_{2D}(I,J) + R_3 S_3 \phi_{3D}(I,J)$
$DY_3(I,J) = DY_1(I,J) - R_1 C_1 \phi_{1D}(I,J) + R_2 C_{12} \phi_{2D}(I,J) + R_3 C_3 \phi_{3D}(I,J)$
$DX_4(I,J) = DX_1(I,J) - R_1 S_1 \phi_{1D}(I,J) - R_4 S_4 \phi_{4D}(I,J)$
$DY_4(I,J) = DY_1(I,J) + R_1 C_1 \phi_{1D}(I,J) + R_4 C_4 \phi_{4D}(I,J)$

$DX_{3D1}(I,J) = R_1 S_1 - R_2 S_{12} - R_3 S_3$
$DX_{3D2}(I,J) = -R_2 S_{12} - R_3 S_3 - R_2$3
$DDX_3(T,J) = -R_3 S_3 * TR_{23} * QD_1(I,J) + R_1 C_1 \phi_{1D}(I,J) + \phi_{1D}(I,J) - R_2 C_{12} \phi_{2D}(I,J) * \phi_{2D}(I,J) + R_3 C_3 \phi_{3D}(I,J) * \phi_{3D}(I,J)$
$DY_{3D1}(I,J) = -R_1 C_1 + R_2 C_{12} + R_3 C_3$
$DY_{3D2}(I,J) = R_2 C_{12} + R_3 C_3 * R_{23}$
$DDY_3(T,J) = R_3 C_3 * TR_{23} * QD_1(I,J) + R_1 S_1 \phi_{1D}(I,J) + \phi_{1D}(I,J) - R_2 S_{12} \phi_{2D}(I,J) * \phi_{2D}(I,J) + R_3 S_3 \phi_{3D}(I,J) * \phi_{3D}(I,J)$

$DDX_3(I,J) = DDX_1(I,J) + DDX_{3D1}(I,J) * \phi_{1DD}(I,J) + DDX_{3D2}(I,J) * \phi_{1DD}(I,J) + QD_1(I,J) + DDX_3(T,I,J)$
$DDY_3(I,J) = DDY_1(I,J) + DDX_{3D1}(I,J) * \phi_{1DD}(I,J) + DDX_{3D2}(I,J) * \phi_{1DD}(I,J) + QD_1(I,J) + DDY_3(T,I,J)$
SUBROUTINE LOWERLINK(I,J)

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/BLOC10/SM1(5,5),SM2(5,5),SM3(5,5),SM4(5,5),
  U1(5,5),U2(5,5),U3(5,5),U4(5,5)
COMMON/TORQUE/TOQ1(5,5),TOQ2(5,5),TOQ3(5,5),TOQ4(5,5)
COMMON/BLOC11/R23,R24,R123,R124,R2423,TR23,TR24,SIN2,COS2,
  SIN3,COS3,SIN4,COS4,SIN24,COS24,SIN23,COS23,SIN43,COS43
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5),
  PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC18/AX(5,5),AY(5,5),AT(5,5),AQ(5,5),AF(5,5),
  BX(5,5),BY(5,5),BT(5,5),BQ(5,5),BF(5,5),
  CX(5,5),CY(5,5),CT(5,5),CQ(5,5),CF(5,5),
  DX(5,5),DY(5,5),DT(5,5),DQ(5,5),DF(5,5),
  SJ(5,5),QF(5,5)
DATA G/9.81/
C....CALCULATE GENERALIZED MOMENT OF INERTIA (SJ=SJ2+SJ3+SJ4)
C....CALCULATE DJ = DJ2 + DJ3 + DJ4 (WHERE DJ2 = 0.0)
SJ2=SM2(I,J)*R2(I,J)*R2(I,J)/4.+U2(I,J)
E1=(-R2(I,J)*SIN2-R3(I,J)/2.0*R23*SIN3
E2=R2(I,J)*COS2+R3(I,J)/2.0*R23*COS3
SJ3=SM3(I,J)*E1*E1+E2*E2+U3(I,J)*R23+R23
SJ(I,J)=SJ2+SJ3+SJ4
C
E3=(SIN43*COS24*R124-SIN24*COS43*R2423)/(SIN43*SIN43)
DJ3=2*SM3(I,J)*E1*(-R2(I,J)*COS2-R3(I,J)/2.0*R23*COS3*+
  R23-R2(I,J)/2.0*SIN3*E3)+2*SM3(I,J)*E2*(-R2(I,J)*+
  SIN2-R3(I,J)/2.0*R23*SIN3*R23+R2(I,J)/2.0*COS3*E3)+
  2*U3(I,J)*R23*R2(I,J)/R3(I,J)*E3
E4=(SIN43*COS24*R123-SIN24*COS43*R2423)/(SIN43*SIN43)
  R4(I,J)*R24*E4
DJ = DJ3+DJ4
TORQUES=TOQ2(I,J)+TOQ3(I,J)*R23+TOQ4(I,J)*R24
GENFORCE = -G*((SM2(I,J)/2.+SM3(I,J))*R2(I,J)*COS2+
  R3(I,J)*R3(I,J)/2.*COS3*R2+SM3(I,J)*R3(I,J)/2.*COS3*R23+SM4(I,J)/+
  2.*R4(I,J)*COS3*R24)+TORQUES
QF(I,J) = GENFORCE-DJ*PHI2D(I,J)*PHI2D(I,J)/2.
RETURN
END
SUBROUTINE UPPERLINK(I,J)

C IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC10/SM1(5,5),SM2(5,5),SM3(5,5),SM4(5,5),
 + U1(5,5),U2(5,5),U3(5,5),U4(5,5)
COMMON/TORQUE/TOQ1(5,5),TOQ2(5,5),TOQ3(5,5),TOQ4(5,5)
COMMON/BLOC11/R23,R24,R123,R124,R2423,TR23,TR24,SIN2,COS2,
 + SIN3,COS3,SIN4,COS4,SIN24,COS24,SIN23,COS23,SIN43,COS43
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5),
 + PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC17/R1S1,R1C1,R1S2,R1C2,R1S3,R1C3,R1S4,R1C4,
 + R2S2,R2C2,R3S3,R3C3,R4S4,R4C4,R2S12,R2C12
COMMON/BLOC18/AX(5,5),AY(5,5),AT(5,5),AQ(5,5),AF(5,5),
 + BX(5,5),BY(5,5),BT(5,5),BQ(5,5),BF(5,5),
 + CX(5,5),CY(5,5),CT(5,5),CQ(5,5),CF(5,5),
 + DX(5,5),DY(5,5),DT(5,5),DQ(5,5),DF(5,5),
 + SJ(5,5),QF(5,5)
COMMON/BLOC19/DX3DT1(5,5),DX3DT2(5,5),DDX3T(5,5),
 + DY3DT1(5,5),DY3DT2(5,5),DDY3T(5,5)

DATA G/9.81/
DX2DT2 =-R2S2
DX2DT2 = R2C2
DX4DT2 =-R4S4*R24
DY4DT2 = R4C4*R24

C DX2DT1 = R1S1-R2S2
DY2DT1 =-R1C1+R2C2
DX4DT1 =-R1S1-R4S4
DY4DT1 = R1C1+R4C4

C DDX2T1 =DX2DT1
DDX2T2 =DX2DT2
DDX2T =R1C1*PHI1D(I,J)*PHI1D(I,J)-R2C2*PHI2D(I,J)*
 + PHI2D(I,J)
DDY2T1 =DY2DT1
DDY2T2 =DY2DT2
DDY2T =R1S1*PHI1D(I,J)*PHI1D(I,J)-R2S2*PHI2D(I,J)*
 + PHI2D(I,J)
DDX3T1 =DX3DT1(I,J)
DDX3T2 =DX3DT2(I,J)
DDX3T =-R4S4*TR24*QD1(I,J)-R1C1*PHI1D(I,J)*
 + PHI1D(I,J)-R4C4*PHI4D(I,J)*PHI4D(I,J)
DDY4T1 =DY4DT1
DDY4T2 =DY4DT2
DDY4T =R4C4*TR24*QD1(I,J)-R1S1*PHI1D(I,J)*
 + PHI1D(I,J)-R4S4*PHI4D(I,J)*PHI4D(I,J)
Y2T2 = R2C2
Y3T2 = R3C3*R23+R2C12
Y4T2 = R4C4*R24

C

POWER1=(TOQ2(I,J)+TOQ3(I,J)*R23+TOQ4(I,J)*R24)*QD1(I,J)

POWER2=(TOQ1(I,J)+TOQ2(I,J)+TOQ3(I,J)+TOQ4(I,J))

+ PHI1D(I,J)

C...CALCULATE THE EQUATION OF MOTION W.R.T. Q1

CX(I,J) =SM2(I,J)*DX2DT2+SM3(I,J)*DX3DT2(I,J)+SM4(I,J)*

+ DX4DT2

CY(I,J) =SM2(I,J)*DY2DT2+SM3(I,J)*DY3DT2(I,J)+SM4(I,J)*

+ DY4DT2

CT(I,J) =SM2(I,J)*(DX2DT2*DDX2T2+DY2DT2*DDY2T2)+SM3(I,J)*

+ (DX3DT2(I,J)*DDX3T1+DY3DT2(I,J)*DDY3T1)+SM4(I,J)*

+ *(DX4DT2*DDX4T1+DY4DT2*DDY4T1))+U2(I,J)+U3(I,J)*

+ R23+U4(I,J)*R24

CQ(I,J) =SM2(I,J)*(DX2DT2*DDX2T2+DY2DT2*DDY2T2)+SM3(I,J)*

+ (DX3DT2(I,J)*DDX3T2+DY3DT2(I,J)*DDY3T2)+SM4(I,J)*

+ (DX4DT2*DDX4T2+DY4DT2*DDY4T2)+U2(I,J)+U3(I,J)*

+ R23+R24

CF(I,J) =-SM2(I,J)*(DX2DT2*DDX2T1+DY2DT2*DDY2T1)-SM3(I,J)*

+ (DX3DT2(I,J)*DDX3T1+DY3DT2(I,J)*DDY3T1)+SM4(I,J)*

+ *(DX4DT2*DDX4T1+DY4DT2*DDY4T1)+U2(I,J)+U3(I,J)*

+ U3(I,J)*TR23+U4(I,J)*TR24+PHI1D(I,J)

+ PHI4D(I,J)-QD1(I,J)*U3(I,J)*R23+PHI3D(I,J)*U3(I,J)*R23

+ G*(SM2(I,J)*Y2T2+SM3(I,J)*Y3T2+

+ SM4(I,J)*Y4T2)+POWER1

C

Y2T1 =-R1C1+R2C2
Y3T1 =-R1C1+R2C12+R3C3
Y4T1 =R1C1+R4C4

C...CALCULATE THE EQUATION OF MOTION W.R.T. PHI1

BX(I,J) =SM2(I,J)*DX2DT1+SM3(I,J)*DX3DT1(I,J)+SM4(I,J)*

+ DX4DT1

BY(I,J) =SM2(I,J)*DY2DT1+SM3(I,J)*DY3DT1(I,J)+SM4(I,J)*

+ DY4DT1

BT(I,J) =SM2(I,J)*(DX2DT1*DDX2T1+DY2DT1*DDY2T1)+SM3(I,J)*

+ *(DX3DT1(I,J)*DDX3T1+DY3DT1(I,J)*DDY3T1)+SM4(I,J)*

+ *(DX4DT1*DDX4T1+DY4DT1*DDY4T1)+U1(I,J)+U2(I,J)+

+ U3(I,J)*TR23+U4(I,J)

BQ(I,J) =CT(I,J)

BF(I,J) =-SM2(I,J)*(DX2DT1*DDX2T+DY2DT1*DDY2T)-SM3(I,J)*

+ (DX3DT1(I,J)*DDX3T(I,J)+DY3DT1(I,J)*DDY3T(I,J))-

+ SM4(I,J)*(DX4DT1*DDX4T+DY4DT1*DDY4T)-QD1(I,J)*

+ *(U3(I,J)*TR23+U4(I,J)*TR24)-G*(SM2(I,J)*Y2T1+

+ SM3(I,J)*Y3T1+SM4(I,J)*Y4T1)+POWER2

C...CALCULATE THE EQUATION OF MOTION W.R.T. XI

AX(I,J) =SM1(I,J)+SM2(I,J)+SM3(I,J)+SM4(I,J)

AY(I,J) =0.0

AT(I,J) =BX(I,J)

AQ(I,J) =CX(I,J)
AF(I,J) = -SM2(I,J)*DDX2T - SM3(I,J)*DDX3T(I,J) - SM4(I,J)*DDX4T

C ... CALCULATE THE EQUATION OF MOTION W.R.T. Y1

DX(I,J) = AX(I,J) + BY(I,J) + CY(I,J)

ddf(I,J) = -SM2(I,J)*DDY2T - SM3(I,J)*DDY3T(I,J) - SM4(I,J)*DDY4T - G*DY(I,J)

RETURN

END
SUBROUTINE CONSTRAIN(I,J)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC1/Q1(5,5),Q2(5,5),Q3(5,5),Q4(5,5),ANGLE(5,5)
COMMON/BLOC2/QD1(5,5),QD2(5,5),QD3(5,5),QD4(5,5),VEL(5,5)
COMMON/BLOC4/R1(5,5),R2(5,5),R3(5,5),R4(5,5)
COMMON/BLOC6/PHI1(5,5),PHI2(5,5),PHI3(5,5),PHI4(5,5)
COMMON/BLOC8/ICV(5),ILN(5),ITYPE(5),ILINK(5),ISIDE(5),
+ IDRIVER(5),ICB(5),IL(5)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC13/QTJACO(5,5),TJACO(5,5),TGAMA(5,5),
+ QXJACO(5,5),XJACO(5,5),XGAMA(5,5),
+ QYJACO(5,5),YJACO(5,5),YGAMA(5,5),
+ QPJACO(5,5),PJACO(5,5),PGAMA(5,5)
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5)
+ PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC16/Y1(5,5),Y2(5,5),Y3(5,5),Y4(5,5),DX1(5,5),
+ DY1(5,5),DX2(5,5),DY2(5,5),DX3(5,5),DY3(5,5),DX4(5,5),
+ DY4(5,5),DDX1(5,5),DDY1(5,5),DDX3(5,5),DDY3(5,5)
COMMON/BLOC19/DX3DT1(5,5),DX3DT2(5,5),DDX3T(5,5),
+ DY3DT1(5,5),DY3DT2(5,5),DDY3T(5,5)

IF(ILINK(I) .EQ. 3) THEN
  C....CONSTRAINED EQUATION BETWEEN UPPER AND LOWER LINKAGES
  IF(ISIDE(I) .EQ. ISIDE(ICV(I))) THEN
    ISIGN = 1
  ELSE
    ISIGN = -1
  ENDF
  IQ = J-1
  IC = ILN(I)+IQ*ISIGN
  R2R3=R2(ICV(I),IC)*R3(ICV(I),IC)
  R1R2=R1(ICV(I),IC)*R2(ICV(I),IC)
  R1R3=R1(ICV(I),IC)*R3(ICV(I),IC)
  T23=PHI2(ICV(I),IC)-PHI3(ICV(I),IC)
  V23=PHI2D(ICV(I),IC)—PHI3D(ICV(I),IC)
  T3=PHI3(ICV(I),IC)-PHI1(ICV(I),IC)
  V3=PHI3D(ICV(I),IC)-PHI1D(ICV(I),IC)
  R23S=R2R3*SIN(T23)
  QTJACO(I,J)=R1R2*SIN(Q1(ICV(I),IC))—R23S
  TJACO(I,J)=R23S+R1R3*SIN(T3)
  TGAMA(I,J)=-R1R2*COS(Q1(ICV(I),IC)))*QD1(ICV(I),IC)*
+ QD1(ICV(I),IC)—R1R3*COS(T3)*V3*V3+R2R3*
+ COS(T23)*V23*V23+(R23S+R1R3*SIN(T3))*
+ PHI1DD(ICV(I),IC)
  QXJACO(I,J)=DX3DT2(ICV(I),IC)
  XJACO(I,J)=—1
  XGAMA(I,J)=-DDX3T(ICV(I),IC)—DX3DT1(ICV(I),IC)*
+ PHI1DD(ICV(I),IC)—DDX1(ICV(I),IC)
  QYJACO(I,J)=DY3DT2(ICV(I),IC)
  YJACO(I,J)=—1
  YGAMA(I,J)=-DDY3T(ICV(I),IC)—DY3DT1(ICV(I),IC)*
+ PH1DD(ICV(I),IC) - DDY1(ICV(I),IC)

ENDIF

C CONSTRAINT EQUATION BETWEEN ADJACENT LINKAGES

IF(J .EQ. 1) THEN
  IF(ILINK(I) .EQ. 3) THEN
    IF(IDRIVER(I) .EQ. 2) THEN
      IA=ICB(I)
      K =ILN(I)
    ELSE
      GO TO 10
    ENDIF
  ELSE
    IF(I .GT. 1) THEN
      IA=ICV(I)
      K =ILN(I)
    ELSE
      GO TO 10
    ENDIF
  ENDIF
ELSE
  IA=I
  K =J-1
ENDIF

R1R2=R1(IA,K)*R2(IA,K)
R1R4=R1(IA,K)*R4(IA,K)
R2R4=R2(IA,K)*R4(IA,K)
ABL=-PHI2(IA,K)+PHI2(IA,J)+BETA(IA,J)
BLA=PHI2(IA,J)+BETA(IA,J)-PHI1(IA,K)
SABL=SIN(ABL)*R2R4
SBLA=SIN(BLA)*R1R4
VBA1=PHI2D(IA,J)-PHI1D(IA,K)
VBA2=PHI2D(IA,J)-PHI2D(IA,K)
DBA=PHI1DD(IA,J)-PHI1DD(IA,K)
QPJACO(I,J)=R1R2*SIN(QI(IA,K))-SABL
PJACO(I,J)=-SBLA+SABL
PGAMA(I,J)=R1R4*COS(BLA)*VBA1*VBA1+SBLA*DBA-R2R4*
             COS(ABL)*VBA2*VBA2+SBLA*DBA-R1R2*
             + COS(Q1(IA,K))*QD1(IA,K)*QD1(IA,K)
10 RETURN
END
SUBROUTINE MATRIX(DD)

C
C          -------------------
C          IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C          FOR ONE BRANCH
C          COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
C          DIMENSION B(260),I(15),DD(15)
C          DATA.....
C          DATA EPS/0.00001/,N/8/
C
C          IF MORE THAN EIGHT EQUATIONS ARE TO BE SOLVED THEN THIS
C          SUBROUTINE MUST BE MODIFIED, WHERE N IS THE MAXIMUM
C          NUMBER OF EQUATIONS TO BE SOLVED. THE DIMENSIONS OF
C          ARRAY B AND ARRAY I ARE EQUAL TO N(N+2) AND N,
C          RESPECTIVELY.
C
C          POINTERS FOR SUBARRAYS.
C          FOR ONE BRANCH
C          IF(NB .EQ. 1) N=3
C          N1=1
C          N2=N1+N*N
C          N3=N2+N
C          NUSED=N3+N-1
C
C          PERFORM L-U FACTORIZATION: ILU=1
C          ILU=1
C          CALL SOLVE(B(N1),B(N2),B(N3),I,N,ILU,EPS)
C          DO 10 II=1,N
C          III=II-1
C          10 DD(II)=B(N2+III)
C          RETURN
C          END
SUBROUTINE SOLVE(AA,C,W,ICOL,N,ILU,EPS)
C
IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C
FOR ONE BRANCH
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
C
COMMON/BLOC13/QTJACO(5,5),TJACO(5,5),TGAMA(5,5),
+ QXJACO(5,5),XJACO(5,5),XGAMA(5,5),
+ QYJACO(5,5),YJACO(5,5),YGAMA(5,5),
+ QPJACO(5,5),PJACO(5,5),PGAMA(5,5)
C
COMMON/BLOC18/AX(5,5),AY(5,5),AT(5,5),AQ(5,5),AF(5,5),
+ BX(5,5),BY(5,5),BT(5,5),BQ(5,5),BF(5,5),
+ CX(5,5),CY(5,5),CT(5,5),CQ(5,5),CF(5,5),
+ DX(5,5),DY(5,5),DT(5,5),DQ(5,5),DF(5,5),
+ SJ(5,5),QF(5,5)
DIMENSION AA(N,N),C(N),W(N),ICOL(N)
C
THE ENTRIES OF MASS MATRIX AA MUST BE REARRANGED BY
THE USER IF THE SYSTEM EQUATIONS OF MOTION ARE CHANGED.
C
FOR ONE BRANCH
C
DO 10 JJ=1,N
DO 10 11=1,N
AA(JJ,II)= 0.0
FOR ONE BRANCH
C
IF(NB .NE. 1) THEN
C
AA(1,1)=AX(2,1)
AA(1,2)=AT(2,1)
AA(1,3)=AQ(2,1)
AA(1,4)=AY(2,1)
AA(1,5)=XJACO(2,1)
AA(2,1)=BX(2,1)
AA(2,2)=BT(2,1)
AA(2,3)=BQ(2,1)
AA(2,4)=BY(2,1)
AA(2,5)=TJACO(2,1)
AA(3,1)=CX(2,1)
AA(3,2)=CT(2,1)
AA(3,3)=CQ(2,1)
AA(3,4)=CY(2,1)
AA(4,1)=DX(2,1)
AA(4,2)=DT(2,1)
AA(4,3)=DQ(2,1)
AA(4,4)=DY(2,1)
AA(4,5)=YJACO(2,1)
AA(5,1)=SJ(1,1)
AA(5,2)=QXJACO(2,1)
AA(5,3)=QTJACO(2,1)
AA(5,4)=QYJACO(2,1)
\[ \begin{align*}
AA(6,1) &= AA(1,6) \\
AA(6,5) &= AA(5,6) \\
AA(7,2) &= AA(2,7) \\
AA(7,5) &= AA(5,7) \\
AA(8,4) &= AA(4,8) \\
AA(8,5) &= AA(5,8) \\
\end{align*} \]

C....INPUT THE RIGHT SIDE OF AX=C (i.e.,VECTOR C)
\[ \begin{align*}
C(1) &= AF(2,1) \\
C(2) &= BF(2,1) \\
C(3) &= CF(2,1) \\
C(4) &= DF(2,1) \\
C(5) &= QF(1,1) \\
C(6) &= XGAMA(2,1) \\
C(7) &= TGAMA(2,1) \\
C(8) &= YGAMA(2,1) \\
\end{align*} \]

ELSE
\[ \begin{align*}
AA(1,1) &= SJ(1,1) \\
AA(1,3) &= QPJACO(1,2) \\
AA(2,2) &= SJ(1,2) \\
AA(2,3) &= PJACO(1,2) \\
AA(3,1) &= AA(1,3) \\
AA(3,2) &= AA(2,3) \\
C(1) &= QF(1,1) \\
C(2) &= QF(1,2) \\
C(3) &= PGAMA(1,2) \\
\end{align*} \]

ENDIF

C....SOLVE AX=C
CALL LINEAR(AA,C,W,ICOL,N,ILU,EPS)
RETURN
END
SUBROUTINE LINEAR (A,C,W,ICOL,N,ILU,EPS)

C

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DIMENSION A(N,N),C(N),W(N),ICOL(N)
IF (ILU.GT.0) CALL LU (A,ICOL,N,EPS)
DO 10 J=1,N
 10 W(J)=C(J)

C....PERFORM FORWARD ELIMINATION STEP. LY=C
DO 30 J=2,N
  SUM=W(J)
  JM1=J-1
  DO 20 K=1,JM1
    SUM=SUM-A(J/K)*W(K)
  20 W(J)=SUM

C....PERFORM BACK SUBTITION STEP. UX=Y
  W(N)=W(N)/A(N,N)
  NP1=N+1
  DO 50 J=2,N
    I=NP1-J
    SUM=W(I)
    IP1=I+1
    DO 40 K=IP1,N
      SUM=SUM-A(I,K)*W(K)
    40 W(I)=SUM/A(I,I)

C....PERMUTE THE SOLUTION VECTOR TO ITS ORIGINAL FORM
DO 60 J=1,N
  C(ICOL(J))=W(J)
RETURN
END
SUBROUTINE LU (A, ICOL, N, EPS)

C IMPLICIT DOUBLE PRECISION(A-H,O-Z)
DIMENSION A(N,N), ICOL(N)

DO 10 K = 1, N
10 ICOL(K) = K

NML = N - 1
DO 50 I = 1, NML
   PIV = ABS(A(I,I))
   IPIV = I
   IP1 = I + 1
   DO 20 K = IP1, N
      TEMP = ABS(A(I,K))
      IF (TEMP .LE. PIV) GO TO 20
      PIV = TEMP
      IPIV = K
   20 CONTINUE

   IF (PIV .LT. EPS) GO TO 60
   IF (IPIV .EQ. I) GO TO 40
   II = ICOL(I)
   ICOL(I) = ICOL(IPIV)
   ICOL(IPIV) = II

   DO 30 J = 1, N
      TEMP = A(J,I)
      A(J,I) = A(J,IPIV)
   30 A(J,IPIV) = TEMP

   DO 40 J = IP1, N
      A(J,I) = A(J,I) / A(I,I)
   40 DO 50 K = IP1, N
      A(J,K) = A(J,K) - A(J,I) * A(I,K)
   50 CONTINUE

RETURN

60 WRITE (3, 200)
200 FORMAT (5X, '***SINGULAR MATRIX***')
STOP
END
SUBROUTINE ENERGY(T)

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
COMMON/BLOC9/BETA(5,5),NB,NL(5),NFLAG,NPOS,NVEL,NACC
COMMON/BLOC10/SM1(5,5),SM2(5,5),SM3(5,5),SM4(5,5), + U1(5,5),U2(5,5),U3(5,5),U4(5,5)
COMMON/BLOC14/SKINEN,POTEN,TOTALEN,VARIEN,CONSTEN
COMMON/BLOC15/PHI1D(5,5),PHI2D(5,5),PHI3D(5,5),PHI4D(5,5), + PHI1DD(5,5),PHI2DD(5,5),PHI3DD(5,5),PHI4DD(5,5)
COMMON/BLOC18/AX(5,5),AY(5,5),AT(5,5),AQ(5,5),AF(5,5), + BX(5,5),BY(5,5),BT(5,5),BQ(5,5),BF(5,5), + CX(5,5),CY(5,5),CT(5,5),CQ(5,5),CF(5,5), + DX(5,5),DY(5,5),DT(5,5),DQ(5,5),DF(5,5), + SJ(5,5),QF(5,5)
DATA G/9.81/
POTEN=0.0
SKINEN=0.0
DO 10 I=1,NB
DO 20 J=1,NL(I)
  IF(ILINK(I) .NE. 3) THEN
    SKINEN=0.5*SJ(I,J)*PHI2D(I,J)*PHI2D(I,J)+SKINEN
  ELSE
  ENDIF
POTEN=G*(SM1(I,J)*Y1(I,J)+SM2(I,J)*Y2(I,J)+SM3(I,J)* + + Y3(I,J)+SM4(I,J)*Y4(I,J))+POTEN
  CONTINUE
10 CONTINUE
TOTALEN=SKINEN+POTEN
IF(T .EQ. 0.0) CONSTEN=TOTALEN
VARIEN=TOTALEN-CONSTEN
RETURN
END
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>Rotational transformation matrix.</td>
</tr>
<tr>
<td>$d$</td>
<td>Damping coefficient.</td>
</tr>
<tr>
<td>$F_x$</td>
<td>Applied force in X-direction.</td>
</tr>
<tr>
<td>$F_y$</td>
<td>Applied force in Y-direction.</td>
</tr>
<tr>
<td>$g$</td>
<td>Gravitational constant.</td>
</tr>
<tr>
<td>$g$</td>
<td>A set of constraint forces.</td>
</tr>
<tr>
<td>$J, J^*$</td>
<td>Generalized inertia.</td>
</tr>
<tr>
<td>$k$</td>
<td>Spring stiffness.</td>
</tr>
<tr>
<td>$L$</td>
<td>Lagrangian function.</td>
</tr>
<tr>
<td>$m_i$</td>
<td>Mass of each link.</td>
</tr>
<tr>
<td>$N_f$</td>
<td>Number of degrees of freedom.</td>
</tr>
<tr>
<td>$N_c$</td>
<td>Number of coordinates.</td>
</tr>
<tr>
<td>$N_h$</td>
<td>Number of holonomic constraints.</td>
</tr>
<tr>
<td>$P$</td>
<td>Power input.</td>
</tr>
<tr>
<td>$q_i, q_i^*$</td>
<td>Relative angles.</td>
</tr>
<tr>
<td>$\mathbf{q}$</td>
<td>A set of independent coordinates.</td>
</tr>
<tr>
<td>$Q_i$</td>
<td>Generalized forces.</td>
</tr>
<tr>
<td>$Q_i^*$</td>
<td>Nonconservative forces.</td>
</tr>
<tr>
<td>$r_i$</td>
<td>Length of each link.</td>
</tr>
<tr>
<td>$\mathbf{r}$</td>
<td>Global position vector.</td>
</tr>
<tr>
<td>$s'$</td>
<td>Local position vector.</td>
</tr>
<tr>
<td>$t$</td>
<td>Time.</td>
</tr>
<tr>
<td>$T$</td>
<td>Kinetic energy.</td>
</tr>
</tbody>
</table>
\( T_\theta \)  
Applied torque.

\( u_i \)  
Moment of inertia.

\( V \)  
Potential energy.

\( X, Y \)  
Global coordinate system.

\( \phi \)  
Deformed angle of spring.

\( \phi^* \)  
Undeformed angle of spring.

\( \phi_i, \phi^*_i \)  
Global angles relative to the X-axis.

\( \dot{\theta}_i \)  
Absolute angles relative to the ground link.

\( \Phi \)  
A set of independent constraint equations.

\( \Phi_{ij} \)  
The constraint Jacobian matrix.

\( \alpha \)  
Angular acceleration of the driving link.

\( \beta \)  
Adjacent angle between two linkages.

\( \gamma \)  
Transmission angle of a four-bar linkage.

\( \gamma \)  
The right side of the acceleration equation.

\( \nu \)  
The right side of the velocity equation.

\( \lambda \)  
Lagrange multiplier.

\( \xi, \eta \)  
Local body fixed coordinate system.
REFERENCES


