AN ABSTRACT OF THE THESIS OF

TINGLIN NIE for the degree of DOCTOR OF PHILOSOPHY

in Mechanical Engineering presented on April 15, 1988

Title: DERIVATION AND AUTOMATIC GENERATION OF KANE'S DYNAMICAL EQUATIONS FOR MECHANICAL MANIPULATORS

Abstract approved: Redacted for privacy

Charles E. Smith

To design and precisely control a manipulator requires developing an efficient dynamic model of the system. The present work demonstrates how this can be accomplished by employing Kane's dynamical equations. First, a detailed manual derivation of the equations of motion for a particular robot is presented in such way that each step in the analysis serves as an example for the derivation of the dynamic equations for serial manipulators in general. Discussions are given regarding the merits of using different sets of generalized speeds. Based on this formulation procedure, an algorithm is presented, which enables computers to automatically generate and integrate Kane's dynamical equations of motion for mechanical manipulators, or to calculate the torques and/or forces required to carry out a user-specified motion of the manipulator. Applicable to nearly any manipulator, the algorithm uses recursive computations of angular velocities, velocities, generalized inertia forces and generalized

active forces. To save computation time, all ingredients for formulating the equations of motion are expressed in scalar forms. The resulting equations contain no approximations beyond the assumptions that the links are rigid and perfectly connected. As an example, simulation results of both manually-derived and automatically-generated equations for the Intelledex 605 Robot arm are reported and it is shown that the algorithm is more efficient than those in the listed literature.

DERIVATION AND AUTOMATIC GENERATION OF KANE'S DYNAMICAL EQUATIONS FOR MECHANICAL MANIPULATORS

BY

TINGLIN NIE

A THESIS

submitted to

Oregon State University

in partial fulfillment of

the requirements for the

degree of

Doctor of Philosophy

Completed April 15, 1988

Commencement June 1988

APPROVED:

Redacted for privacy

Professor of Mechanical Engineering in charge of major

Redacted for privacy

Head of Department of Mechanical Engineering

Redacted for privacy

Dean of Graduate School

Date thesis is presented April 15, 1988

Typed by the author Tinglin Nie

ACKNOWLEDGEMENT

I would like to first express my gratitude to Dr. Charles E. Smith, for serving as my major professor during this study. His excellent instruction in Kane's dynamics has given me a deeper understanding of my research, which I am sure will contribute to my success in the future. His patiently reviewing the manuscript is also highly appreciated.

Appreciation is also extended to the members of my Ph.D committee, Dr. Robert E. Wilson, Dr. Timothy C. Kennedy, Dr. Alan H. Robinson and Dr. Edward H. Piepmeier, as well as to the faculty, staff in the Department of Mechanical Engineering for their advice and help during my years at Oregon State University. To my fellow students and friends, I especially owe an endless debt of thanks for many valuable conversations that helped me guide my research. Their friendship and assistance will always be remembered.

Special thanks are extended to Dr. Thomas R. Kane, Professor of Stanford University who has fathered Kane's dynamics, for his kindly reviewing part of this thesis and giving high comments, which the presented work may not deserve.

I would also like to acknowledge Dr. Gordon M. Reistad, Head of the Department of Mechanical Engineering, and the former Head, Dr. James R. Welty, who provided me graduate teaching assistantship during the last few years of my study at the Oregon State University, without which this work would not have been possible.

I am especially indebted to China, where I am originally from. I never forget the people on the other side of the Pacific Ocean. I hope that I could contribute my knowledge to the Chinese people in my future work.

Finally, I wish to dedicate this dissertation to my wife, Junxi Wang, and my daughter, Molly Nie, whose love, understanding and support always encourage me to achieve my academic goal during the three years that this thesis was being researched and prepared.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

DERIVATION AND AUTOMATIC GENERATION OF KANE'S DYNAMICAL EQUATIONS FOR MECHANICAL MANIPULATORS

I. INTRODUCTION

Development of an efficient mathematical representation of manipulator dynamics is essential for the advanced control and design of manipulator systems. In robot control, dynamical equations are frequently used to compute the forces and torques needed to drive the system to achieve desired motions, a task that may be performed repeatedly and, in most cases, rapidly. As regards manipulator design, dynamical equations are employed to carry out simulations for the purpose of testing the performance of a manipulator. Consequently, constructing the most efficient computational algorithm and finding the most efficient way to approach dynamic equations of motion are of the first importance in the field of robotics. The purpose of the present work is to show how this can be accomplished by using Kane's dynamical equations and to develop a computational algorithm to automatically generate such equations.

Two methods are widely used in deriving the equations of motion for mechanical manipulators, namely, the Newton-Euler formulation and the Lagrangian formulation. The Newton-Euler formulation is derived by direct interpretation of Newton's second law of motion, which describes dynamic systems in terms of force and momentum. The equations incorporate all the forces and moments acting on an individual arm link, including the coupling torques, forces and moments between adjacent links. In the Lagrangian formulation, on the other hand, the system's dynamic behavior is described in terms of work and energy using generalized coordinates. Much effort has been devoted to developing effective procedures to obtain the equations of motion in the fields of spacecraft and robotics by using these two methods.

In 1965, Hooker and Margulies [1] presented an algorithm, based on the Newton-Euler formulation, to derive the dynamical equations for an n-body satellite. This paper is considered to be the first paper describing a quite general derivation and computational algorithm in the field of spacecraft. In the same year, Uicker [2] derived the exact equations of motion for rigid-link spatial mechanical systems, using 4X4 displacement matrices. The results were specially written for open kinematic chains, the most common manipulator configuration, by Kahn in 1969 [3]. The following year, Keat [4] reported on the derivation of dynamical equations of nonrigid satellites by using the Lagrangian formulation. Woo and Freudenstein [5], Yang [6] investigated the use of screw calculus in deriving the equations of motion for spatial mechanisms. The basic objective of these works was to provide a programmable, analytical formulation for studying the dynamics of general, rigid link mechanical systems. A host of additional papers [e.g. 7-20] soon followed during the late of 1960's and the early of 1970's.

The derivations cited above provided a theoretical framework for the study of what is called multibody dynamics. The results were too complicated and the computations too time-consuming to be practical for design or real-time control in the robotic field. In 1974, Bejczy [21] presented an approach to the equations of motion of a robot arm and showed some simplifications of the basic equations. In parallel efforts, to reduce the computation time for evaluating the generalized actuator torques, Whitney [22], Raibert [23], Raibert and Horn [24] considered replacing some calculations by table look-up schemes. This straightforward method, however, requires a very large memory space and is difficult to modify when the mass properties are changed.

The relatively promising methods for solving this analysis task in real-time are the recursive formulation presented in the last few years since Stepanenko and Vukobratovic published their paper [25] in 1976. Orin [26], Luh, Walker and Paul [27] devised the recursive Newton-Euler dynamics computation. The method is recursive in the sense that velocities and accelerations are found sequentially starting from the fixed base link. Then, force or torque balances at each successive joint, starting at the free end of the arm, determine the actuator torques due to the inertial and applied loads. Paul and Luh [28] also gave a more efficient implementation of this method, while Hollerbach [29] developed independently the recursive relations based on the Lagrangian formulation. Hollerbach [30] and Kanade [31] further improved the computation efficiency by customizing the dynamic computations to particular robot structures. These recursive methods form a computationally faster algorithm for calculating the kinematic terms and for constructing the equations of motion. However, they provide no information about the overall structure of the dynamic system needed for analyzing its dynamic behavior.

Derivation of dynamic equations of motion for manipulators is a time-consuming and error-prone process. Awareness of this problem comes into evidence in nearly every paper dealing with the robotic dynamics. In the mid--1970's, dynamicists began to think about what is now called automatic generation, which means computer programs intended to simultaneously generate and integrate the equations of motion numerically for user specified arrangements of connected bodies or mechanical manipulators. As early as in 1973, Dillon [32] presented a program to generate the equations of motion for linkage mechanisms based on the Lagrangian technique. This program was then used to check the correctness of certain derived equations of motion. Two years later, Langrana and Bartel [33] reported an automated method for dynamic analysis of spatial linkages for biomedical application. These early works seem too tedious to be practical if the algorithm was used for six-link robots. Research on this field had been silent for few years until 1981 when Luh and Lin [34] developed an algorithm to automatically simplify the dynamic equations of motion for a manipulator. This algorithm is based on the combination of the Newton-Euler and the Lagrangian formulations. The following year, Thomas and Tesar [35] presented a numerical simulation algorithm and announced that a general computer package based on this algorithm had been written for the static and the dynamic analysis of six-joint manipulators.

All of the works cited above are based on either the Newton-Euler's method, the Lagrangian method, or a combination of these two. The resulting procedures have serious difficulties. The equations obtained from the Newton-Euler's method include the constraint forces acting between two adjacent links. Therefore, additional arithmetic operations are required to eliminate these nonworking terms and to obtain the explicit relation between the joint torques and the resultant motion in terms of joint displacements. The Lagrangian formulation, providing relief of this burden, suffers new problems. The manual labor needed to derive and differentiate the kinetic energy expression can be time-consuming and difficult to accomplish without error. The resulting equations are very difficult to modify after they have been developed, and the significance of individual terms in these expressions is often obscure. Both methods are quite laborious, and, when one attempts to save manual labor by resorting to the use of a computer, one finds frequently that intermediate computations need such large memory spaces that the storage requirements exceed the capacities of the largest available computers, even when the manipulator being analyzed possesses only a modest number of links. Therefore, there is a need to find a new method that is minimally laborious and leads directly to the simplest possible computational algorithm.

In the early 1960's, T. R. Kane developed an approach [36] that reduces the formulation of dynamical equations to a straightforward, deductive procedure instead of the classical formulations. The formulation was called "Lagrangian form of D'Alembert's principle" at that time. In this formulation, the concept of partial velocity was introduced. In accordance with Newton's second law, the formulation was founded by dot-multiplying the active forces and the inertia forces with the partial velocities. In deriving the equations of motion, this method significantly reduces the amount of hand labor, as stated by Peter Radetsky [37], "a growing army of disciples claims that Kane's dynamical equations are so far the most efficient method in dynamics -- and the more difficult the problem, the more valuable it is." In connection with spacecraft dynamics, Kane and Levinson [38] further showed that this method enables one to work systematically with fairly complicated multibody dynamics, to eliminate effortlessly forces and torques that are of no interest, and to produce straightforwardly explicit equations of motion having a computationally sound form. This formulation was formally called Kane's dynamical equations when Kane, Likins and Levinson published the book "Spacecraft Dynamics" [39] in 1983.

The first paper regarding Kane's dynamical equations in the field of robotic dynamics was published by Huston and Kelly [40,41] in 1982. In the next year, Kane and Levinson presented a detailed formulation procedure using Kane's dynamical equations for the Stanford Arm [42]. Simulation results based on the derived equations were reported. A detailed comparison of Kane's dynamical equations with the Lagrangian formulation was given in reference [43]. In this paper, the authors focused their attention on the labor that must be expended in formulating the equations of motion and on the form assumed by these equations, which determines the number of operations required for a numerical solution of the equations. Following this, Kane and Fassler [44] further investigated the derivation of closed-form of dynamic equations for robots and manipulators with the same algorithm, and concluded that the Kane's dynamical equations provided the best basis for the solution of multibody dynamics.

However, since Kane's dynamical equations have been introduced only since 1968, unlike those of classical methods, the literature about it, especially in the robotic field, is sparse, consisting only of the few references cited above. In other words, the theoretical framework is available, but the details of the computational algorithm needs further development, and some aspects need to be further discussed. First, the references do not present a general guiding idea on how to construct the generalized speeds, without which one may not know how to start his work. Another problem is the elimination of the nonworking contact forces. Rather than considering a specific robot, a general expression that fits all robots needs to be derived to provide a better basis for automatic generation. For automatic generation, references [36, 38-44] provide a good basis for constructing the equations of motion, but what they have done so far is primarily based on hand-derived equations rather than letting a computer do it. The procedure described in the most recent paper [45] (Fall, 1986) regarding automatic generation of Kane's dynamical equations avoids writing explicitly the expressions of accelerations and generalized inertia forces, but with this algorithm, one must spend almost the same amount of labor on creating "inertia coefficients" and their derivatives. In general, the algorithms of automatic generation by Kane's formulation reported so far can only avoid writing explicitly the equations of motion, all other ingredients, including kinematic and kinetic quantities and generalized inertia and active forces, are still derived by hand. Thus, the procedure is still very burdensome for the manipulators with six-degrees of freedom. Indeed, when one's ultimate goal is the numerical solution of the equations of motion, one may employ computer codes to handle the derivation of velocities, angular velocities and other kinematic and kinetic ingredients, and to write the dynamical equations. One may then proceed directly to the creation of a computer program that yields simulation results.

To show how these deficiencies may be overcome, the present work derives the dynamic equations for the Intelledex 605 Robot Arm by employing Kane's dynamical equations. The formulation procedure is general for all serial robot arms with detailed discussions on how to select the generalized speeds and how to eliminate the nonworking forces for the general six-link robots. Based on the formulation procedure, an algorithm for automatic generation of Kane's dynamical equations for manipulators is then derived. This algorithm requires analysts to provide only the geometric configuration data of the manipulator being analyzed, i.e., the elements of a set of transformation matrices between links, the derivation of all kinematic and kinetic ingredients and formulation of the equations of motion being left for a computer. Computer programs based on both the hand-derived equations and the automatic algorithm are presented.

The remaining chapters of this work are arranged as follows.

Chapter II mainly deals with the manual derivation of Kane's dynamical equations. The sequel begins with coordinate assignations and transformations for the Intelledex 605 Robot Arm. Then, guidance on how to define generalized speeds is given. Next, the kinematic and kinetic ingredients needed for constructing Kane's dynamical equations are worked out. Thereafter, the equations of motion for the example robot arm are established. Finally, simulation results based on the equations are reported. In Chapter III, the algorithm of automatic generation of Kane's dynamical equations is derived. The symmetry property of the inertia matrix of the system equations is proved. This property is then used in constructing the algorithm to reduce the arithmetic operation in the computer programs. Simulation results based on this algorithm are also reported and compared with those from the hand-derived equations. Discussions and conclusions are presented in the last chapter.

II. DERIVATION

II.1 SPECIFICATIONS AND TRANSFORMATIONS

Figure 2.1 is a schematic representation of the Intelledex 605 Robot Arm, which consists of six links designated by A, B, ..., F. Link A can be rotated in a Newtonian reference frame N about axis Z_0 fixed in N. A supports B, which can be made to rotate relative A about the axis Z_1 fixed in A and B. C is connected to B, D to C and so on in such way that the members of each pair can be made to undergo relative rotation about their common axis, as indicated in Figure 2.1. The quantities q_1, q_2, ..., q_6 are radian measures of the angles of these six relative rotations, which are defined as generalized coordinates. For the configuration depicted in Figure 2.1, q_1, q_2, ..., q_6 are regarded as being equal to zero. A^*, B^*, ..., F^* are the mass centers of the links A, B, ..., F respectively. L_1, L_2, L_{11}, ..., L_{63} are linear measures used to specify the coordinate components of the mass centers A^*, B^*, ..., F^*.

The coordinate frame assigned to each link is shown in Figure 3.2. Namely, axes x_0, y_0, z_0 are fixed in the reference frame N with a set of mutually perpendicular unit vectors $\mathbf{n}_1$, $\mathbf{n}_2$, $\mathbf{n}_3$ in the directions corresponding to x_0, y_0, z_0 respectively. Axes x_1, y_2, z_3 are fixed in link A with the unit vectors $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ in the direction of each of

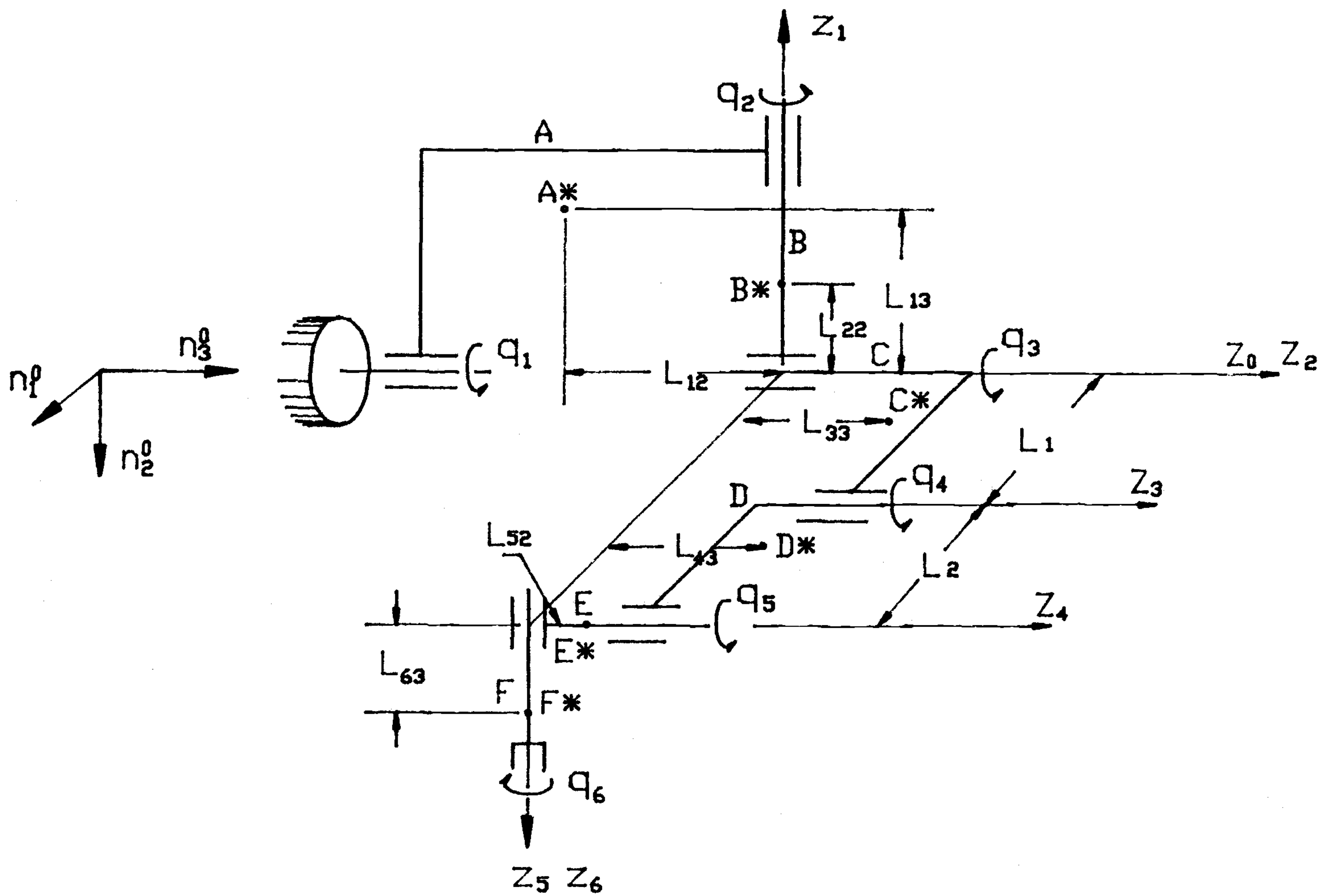

Figure 2.1. Schematic representation of the Intelledex 605 robot, shown for $q_1, \ldots, q_6 = 0$.

the axes respectively, and x_2, y_2, z_2 with $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_3$ are fixed in B, x_3, y_3, z_3 with $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$ fixed in C, and so on through F.

Once the coordinate frames have been assigned, one can begin to establish the transformation matrix A_i (i=1, 2, ..., 6) relating the coordinate frame of link i to the coordinate frame i-1, as described in reference [46]. Specifically, the transformations between adjoining bodies are:

$$A_1 = \begin{bmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_2 & 0 & -s_2 \\ s_2 & 0 & c_2 \\ 0 & -1 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & -s_4 & 0 \\ s_4 & c_4 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} c_5 & 0 & -s_5 \\ s_5 & 0 & c_5 \\ 0 & -1 & 0 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 \\ s_6 & c_6 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where s_i and c_i are the abbreviations of the sine and cosine of angle q_i, that is

$$s_i = \sin(q_i)$$

$$c_i = \cos(q_i)$$

The abbreviations such as

$$s_{ij} = \sin(q_i+q_j)$$

$$c_{ij} = \cos(q_i+q_j)$$

$$s_{ijk} = \sin(q_i+q_j+q_k)$$

$$c_{ijk} = \cos(q_i+q_j+q_k)$$

will also be used in the later part of this work.

The next step to be undertaken is to evaluate the products of the above transformation matrices. These products relate each coordinate frame towards the base coordinate frame N and are historically called T matrices.

$$T_1 = A_1$$

$$T_2 = A_1A_2$$

$$\ldots\ldots$$

$$T_6 = A_1A_2A_3A_4A_5A_6$$

These in turn give

$$\{\mathbf{N}\} = [T_1]\{\mathbf{a}\} = [T_2]\{\mathbf{b}\} = \ldots\ldots = [T_6]\{\mathbf{f}\}$$

Therefore, the unit vectors of the base coordinate frame can be expressed in each coordinate frame by the following transformation relations.

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{Bmatrix} \qquad (2.1)$$

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1c_2 & -s_1 & -c_1s_2 \\ s_1c_2 & c_1 & -s_1s_2 \\ s_2 & 0 & c_2 \end{bmatrix} \begin{Bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \end{Bmatrix} \tag{2.2}$$

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1c_2c_3-s_1s_3 & -c_1c_2s_3-s_1c_3 & -c_1s_2 \\ s_1c_2c_3+c_1s_3 & -s_1c_2s_3+c_1c_3 & -s_1s_2 \\ s_2c_{34} & -s_2s_3 & c_2 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.3}$$

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1c_2c_{34}-s_1s_{34} & -c_1c_2s_{34}-s_1c_{34} & -c_1s_2 \\ s_1c_2c_{34}+c_1s_{34} & -s_1c_2s_{34}+c_1c_{34} & -s_1s_2 \\ s_2c_{345} & -s_2s_{34} & -c_2 \end{bmatrix} \begin{Bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \mathbf{d}_3 \end{Bmatrix} \tag{2.4}$$

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1c_2c_{345}-s_1s_{345} & c_1s_2 & -c_1c_2s_{345}-s_1c_{345} \\ s_1c_2c_{345}+c_1s_{345} & s_1s_2 & -s_1c_2s_{345}+c_1c_{345} \\ s_2c_{345} & -c_2 & -s_2s_{345} \end{bmatrix} \begin{Bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{Bmatrix} \tag{2.5}$$

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} (c_1c_2c_{345}-s_1s_{345})c_6+c_1s_2s_6 & -(c_1c_2c_{345}-s_1s_{345})s_6+c_1s_2c_6 & -c_1c_2s_{345}-s_1c_{345} \\ (s_1c_2c_{345}+c_1s_{345})c_6+s_1s_2s_6 & -(s_1c_2c_{345}+c_1s_{345})s_6+s_1s_2c_6 & -s_1c_2s_{345}+c_1c_{345} \\ s_2c_{345}c_6-c_2s_6 & -s_2c_{345}s_6-c_2c_6 & -s_2s_{345} \end{bmatrix} \begin{Bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{Bmatrix} \tag{2.6}$$

For the reason that will be seen later, the expressions of each set of unit vectors **n**'s, **a**'s, ...,**f**'s in terms of unit vectors $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$ will also be needed. these are:

$[\mathbf{n}] = [A_1][A_2][A_3][\mathbf{c}]$

$$[\mathbf{a}] = [A_2][A_3][\mathbf{c}]$$

$$[\mathbf{b}] = [A_3][\mathbf{c}]$$

$$[\mathbf{d}] = [A_4]^{-1}[\mathbf{c}]$$

$$[\mathbf{f}] = [A_5]^{-1}[A_4]^{-1}[\mathbf{c}]$$

$$[\mathbf{f}] = [A_6]^{-1}[A_5]^{-1}[A_4]^{-1}[\mathbf{c}]$$

i.e.

$$\begin{Bmatrix} \mathbf{n}_1 \\ \mathbf{n}_2 \\ \mathbf{n}_3 \end{Bmatrix} = \begin{bmatrix} c_1c_2c_3-s_1s_3 & -c_1c_2s_3-s_1c_3 & -c_1s_2 \\ s_1c_2c_3+c_1s_3 & -s_1c_2s_3+c_1c_3 & -s_1s_2 \\ s_2c_3 & -s_2s_3 & c_2 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.7}$$

$$\begin{Bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{Bmatrix} = \begin{bmatrix} c_2c_3 & -c_2s_3 & -s_2 \\ s_2c_3 & -s_2s_3 & c_2 \\ -s_3 & -c_3 & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.8}$$

$$\begin{Bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \end{Bmatrix} = \begin{bmatrix} c_3 & -s_3 & 0 \\ s_3 & c_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.9}$$

$$\begin{Bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \mathbf{d}_3 \end{Bmatrix} = \begin{bmatrix} c_4 & s_4 & 0 \\ -s_4 & c_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.10}$$

$$\begin{Bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{Bmatrix} = \begin{bmatrix} c_{45} & s_{45} & 0 \\ 0 & 0 & -1 \\ -s_{45} & c_{45} & 0 \end{bmatrix} \begin{Bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{Bmatrix} \tag{2.11}$$

$$\begin{Bmatrix} f_1 \\ f_2 \\ f_3 \end{Bmatrix} = \begin{bmatrix} c_{45}c_6 & s_{45}c_6 & -s_6 \\ -c_{45}s_6 & -s_{45}s_6 & -c_6 \\ -s_{45} & c_{45} & 0 \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \\ c_3 \end{Bmatrix} \tag{2.12}$$

II.2 GENERALIZED SPEEDS

For a simple nonholonomic system S possessing n-m degrees of freedom, the n-m quantities u_1, $u_2, \ldots, u_{n-m}$, called generalized speeds, are defined [36] as linear combinations of $\dot{q}_1$, $\dot{q}_2$, ..., $\dot{q}_{n-m}$ by means of equations of the form

$$u_r = \sum_{s=1}^{n-m} U_{rs}\dot{q}_s + U_r, \quad (r=1,\ 2,\ \ldots,\ n-m) \tag{2.13}$$

where U_{rs} and U_r are functions of the coordinates q_1, q_2, ..., q_n, and t, and these quantities are chosen in such a way that equations (2.13) can be solved uniquely for $\dot{q}_1$, $\dot{q}_2$, ..., $\dot{q}_{n-m}$.

It then follows that the velocity $\mathbf{V}$ of a typical particle P of S can be expressed uniquely as

$$\mathbf{V} = \sum_{r=1}^{n-m} \mathbf{v}_r u_r + \mathbf{v}_t \tag{2.14}$$

Similarly, the angular velocity of a rigid body of S can be written as

$$\mathbf{W} = \sum_{r=1}^{n-m} \mathbf{w}_r u_r + \mathbf{w}_t \tag{2.15}$$

Where $\mathbf{v}_r$, $\mathbf{w}_r$, $\mathbf{v}_t$ and $\mathbf{w}_t$ are functions of q_1, ..., q_{n-m} and t, and the $\mathbf{v}_r$, $\mathbf{w}_r$ are called the rth partial velocity and the rth partial angular velocity, respectively.

By the definition (2.13), it is evident that generalized speeds are used to specify the motion of a system, rather than its configuration. Generalized speeds can be any linear combinations of $\dot{q}_1, \ldots, \dot{q}_{n-m}$ (subject to the invertability mentioned above). They are not necessarily time derivatives of any functions [36]. Therefore, there is actually an unlimited number of ways to define generalized speeds for a given system. It then comes to the question that what definition provides the best basis for deriving dynamical equations for a manipulator? To help answer this question, a few points are to be discussed.

First, the guiding idea in introducing generalized speeds is to reduce the labor required to derive dynamical equations. To this end, selection of definitions for generalized speeds should be made such that corresponding expressions for partial velocities and partial angular velocities, which are to be dot multiplied with active and inertia forces, be as simple as possible. Thus, with proper selection, simpler expressions of these ingredients can lead to a noticeable simplification of the derivation process and of the resulting equations.

In practice, most six-link manipulators have three or more revolute joints. The velocity of a point P fixed in one of the links is usually found by the following equation.

$$\mathbf{v}^P = \mathbf{v}^Q + \mathbf{w} \times \mathbf{r}$$

This indicates that simpler expressions of angular velocities can lead to simpler expressions of velocities, and thus simpler partial velocities. However, simple forms of velocities cannot usually lead to simple expressions of angular velocities; in fact, they may be even more complicated. Therefore, in selecting the forms of generalized speeds, one should preferably make the expressions of angular velocities as simple as possible.

The next point to be discussed is the choice of the link relative to whose angular velocity the generalized speeds are defined. Consider, first, the manipulators having three or five revolute joints. As shown in Figure 2.2, for the system having three revolute joints, if each of the three components of the angular velocity of the third link is defined as a generalized speed, the angular velocities for link 1, link 2 and link 3, respectively, can be written as

$$\mathbf{w}^1 = (u_1c_3/s_2 - u_2s_3/s_2)\mathbf{a}_2$$

$$\mathbf{w}^2 = (u_1c_3-u_2s_3)\mathbf{b}_1+(u_1s_3+u_2c_3)\mathbf{b}_2+(u_1c_3-u_2s_3)t_2\mathbf{b}_3$$

$$\mathbf{w}_3 = u_1\mathbf{c}_1+u_2\mathbf{c}_2+u_3\mathbf{c}_3$$

where t_2 stands for $\tan(q_2)$. In these expressions, there are 11 multiplications and 4 additions. On the other hand, if the first link is chosen as the base, then,

$$\mathbf{w}^1 = u_1\mathbf{a}_2$$

$$\mathbf{w}^2 = u_2s_1\mathbf{b}_1 - u_2\mathbf{b}_2 + u_1c_2\mathbf{b}_3$$

$$\mathbf{w}^3 = (u_1s_2c_3-u_2s_3)\mathbf{c}_1-(u_1s_2s_3+u_2c_3)\mathbf{c}_2+(u_1c_2+u_3)\mathbf{c}_3$$

There are 9 multiplications and 3 additions in these equations. However, if the second link is chosen as the base, then,

$$\mathbf{w}^1 = s_2u_1\mathbf{a}_2$$

$$\mathbf{w}^2 = u_1\mathbf{b}_1+u_2\mathbf{b}_2+u_3\mathbf{b}_3$$

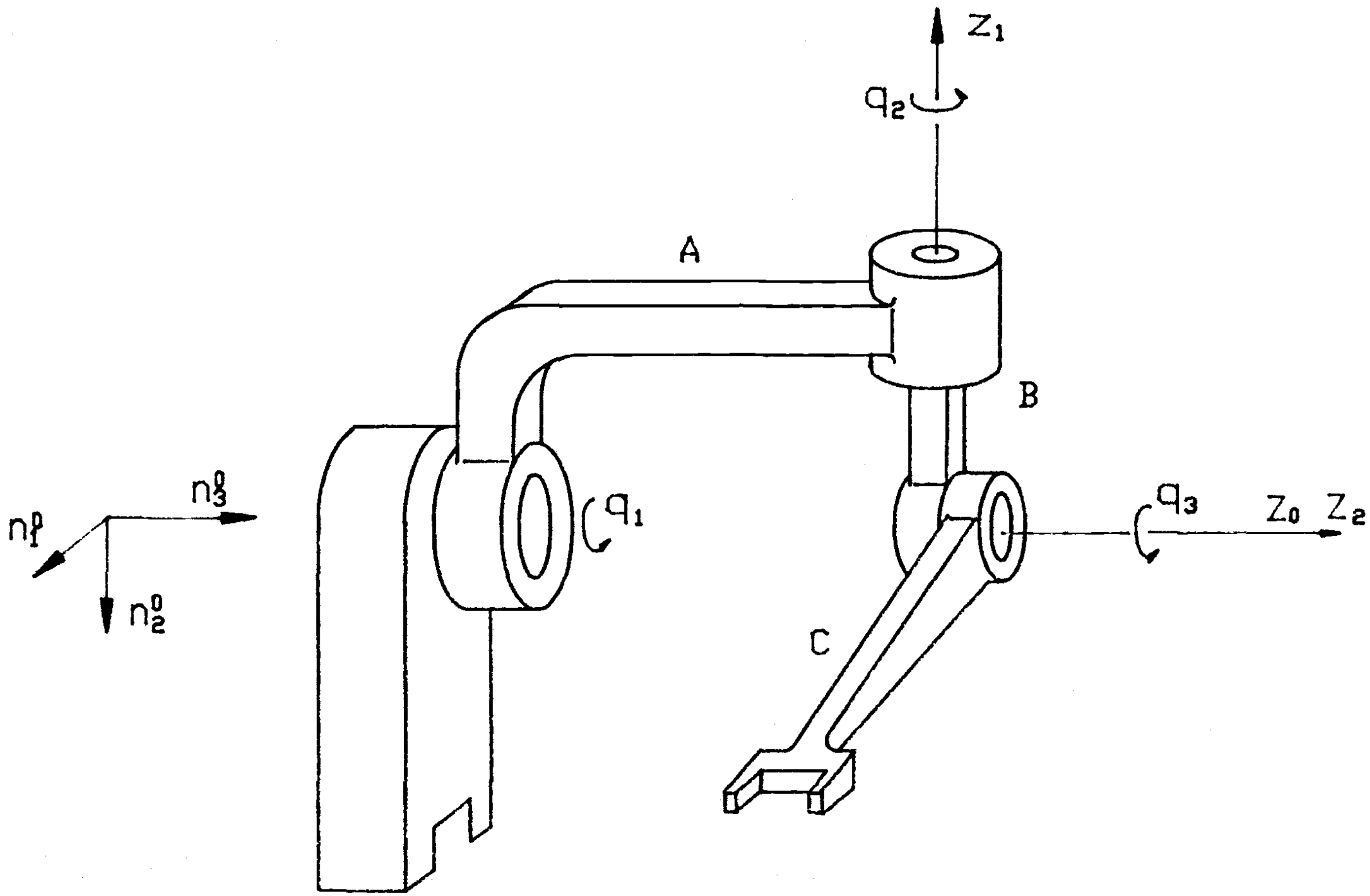

Figure 2.2. Three rotational link manipulator.

$$\mathbf{w}^3 = (u_1c_3+u_2s_3)\mathbf{c}_1+(-u_1s_3+u_2c_3)\mathbf{c}_2+(u_1t_2+u_3)\mathbf{c}_3$$

These expressions only need 7 multiplications and 3 additions. Especially, for the manipulators having five revolute joints, choosing the middle link as the base can significantly reduce the number of arithmetical operations. This is because the number of matrix transformations from the middle link to other links is 2X(1+2)=6, while the number from the first link or from the last one to others is 1+2+3+4+5=15. For manipulators having six revolute joints, either the third link or the fourth link can be defined as the base. However, since the kinematic ingredients of the former link are more frequently used than those of the latter, it is better to make the angular velocity of the third link have the simplest form, that is, the third link is taken as the base. For the same reason, the second link is taken as the base for a manipulators having four revolute joints.

In summary, the guiding idea for introducing generalized speeds is to simplify the expressions of angular velocities, velocities, partial angular velocities and partial velocities, more importantly to obtain the simplest forms of angular velocities. For the manipulators having three or four revolute joints, it is best to choose the second link as the base. For five or six revolute joint manipulators, the third links should be chosen as the bases. Once the base is chosen, define each of three mutually perpendicular components of its angular velocity as a generalized speed.

For the Intelledex 605 Robot, link C is taken as the base, the angular velocity of which can be found by

$$\mathbf{w}^C = \dot{q}_1\mathbf{n}_3 + \dot{q}_2\mathbf{a}_3 + \dot{q}_3\mathbf{b}_3$$

In terms of unit vectors $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$ (refer to equations (2.7-9)), this can be expressed as

$$\mathbf{w}^C = (s_2c_3\dot{q}_1-s_3\dot{q}_2)\mathbf{c}_1-(s_2s_3\dot{q}_1+c_3\dot{q}_2)\mathbf{c}_2+(c_2\dot{q}_1+\dot{q}_3)\mathbf{c}_3 \quad (2.17)$$

The generalized speeds $u_1, \ldots, u_6$ are defined as

$$u_i = \mathbf{w}^C\cdot\mathbf{c}_i \qquad (i=1,\ 2,\ 3)$$

$$u_i = \dot{q}_i \qquad (i=4,\ 5,\ 6)$$

That is,

$$\begin{aligned} u_1 &= s_2c_3\dot{q}_1-s_3\dot{q}_2 \\ u_2 &= -s_2s_3\dot{q}_1-c_3\dot{q}_2 \\ u_3 &= c_2\dot{q}_1+\dot{q}_3 \\ u_4 &= \dot{q}_4 \\ u_5 &= \dot{q}_5 \\ u_6 &= \dot{q}_6 \end{aligned} \qquad (2.18)$$

If the Intelledex 605 Robot Arm is not operated in the vicinity of $q_2=0°$ or $q_2=180°$, equations (2.18) can be solved uniquely for

$$\begin{aligned} \dot{q}_1 &= (c_3u_1-s_3u_2)/s_2 \\ \dot{q}_2 &= -s_3u_1-c_3u_2 \\ \dot{q}_3 &= u_3+(s_3u_2-c_3u_1)c_2/s_2 \\ \dot{q}_4 &= u_4 \\ \dot{q}_5 &= u_5 \\ \dot{q}_6 &= u_6 \end{aligned} \qquad (2.19)$$

For convenience, define

$$z_1 = \dot{q}_1 = (c_3u_1-s_3u_2)/s_2 \tag{2.20}$$

$$z_2 = \dot{q}_2 = -s_3u_1-c_3u_2$$

$$z_3 = \dot{q}_3 = u_3+(s_3u_2-c_3u_1)c_2/s_2$$

II.3 KINEMATIC INGREDIENTS

Angular velocities and velocities are to be expressed in two different forms, one involving the generalized speeds explicitly, another implicitly. The explicit form must be used when one tries to find partial velocities and partial angular velocities by inspecting the coefficients of the generalized speeds and to form expressions of accelerations (partly) and angular accelerations by differentiating the available velocities or angular velocities. On the other hand, when one tries to determine the cross products or dot products of angular velocities, velocities with some other vectors, the implicit forms should be used for this can considerably save hand labor.

Wiht reference to Figure 2.1, the angular velocity of link A in the base reference frame can be expressed in form of

$$\mathbf{w}^A = \dot{q}_1\mathbf{n}_3$$

In terms of the vector basis fixed in link A, this can also be written as

$$\mathbf{w}^A = \dot{q}_1\mathbf{a}_2$$

In view of equation(2.20), in the implicit form,

$$\mathbf{w}^A = z_1\mathbf{a}_2$$

Alternatively, in the explicit form,

$$\mathbf{w}^A = (Z_4 u_1 + Z_5 u_2)\mathbf{a}_2$$

where

$$Z_4 = c_3/s_2$$

$$Z_5 = -s_3/s_2$$

For the angular velocity of link B in N, one finds,

$$\mathbf{w}^B = \mathbf{w}^A + \dot{q}_2 \mathbf{a}_3$$

In terms of the vector basis fixed in the link B,

$$\mathbf{w}^B = Z_6 \mathbf{b}_1 + Z_7 \mathbf{b}_2 + Z_{10} \mathbf{b}_3 \tag{2.21}$$

$$\mathbf{w}^B = (u_1 c_3 - u_2 s_3)\mathbf{b}_1 + (u_1 s_3 + u_2 c_3)\mathbf{b}_2 + (Z_8 u_1 + Z_9 u_2)\mathbf{b}_3$$

with

$$Z_6 = u_1 c_3 - u_2 s_3$$

$$Z_7 = -Z_2$$

$$Z_8 = Z_4 c_2$$

$$Z_9 = Z_5 c_2$$

$$Z_{10} = Z_8 u_1 + Z_9 u_2$$

The angular velocity of link C can be directly found from equations (2.17) and (2.18) to be

$$\mathbf{w}^C = u_1 \mathbf{c}_1 + u_2 \mathbf{c}_2 + u_3 \mathbf{c}_3 \tag{2.22}$$

Based on $\mathbf{w}^C$, one can express $\mathbf{w}^D$, the angular velocity of link D, as

$$\mathbf{w}^D = \mathbf{w}^C + \dot{q}_4 \mathbf{c}_3$$

In terms of the vector basis that is fixed on link D,

$$\mathbf{w}^D = (u_1 c_4 + u_2 s_4)\mathbf{d}_1 + (-u_1 s_4 + u_2 c_4)\mathbf{d}_2 + (u_3 + u_4)\mathbf{d}_3 \tag{2.23}$$

$$\mathbf{w}^D = Z_{11} \mathbf{d}_1 + Z_{12} \mathbf{d}_2 + Z_{13} \mathbf{d}_3$$

where

$$Z_{11} = u_1 c_4 + u_2 s_4$$

$$Z_{12}=-u_1s_4+u_2c_4$$

$$Z_{13}=u_3+u_4$$

Similarly, the angular velocities of link E and link F are found to be,

$$\mathbf{w}^E = Z_{17}\mathbf{e}_1+Z_{18}\mathbf{e}_2+Z_{19}\mathbf{e}_3 \tag{2.24}$$

$$\mathbf{w}^E = (u_1Z_{14}+u_2Z_{15})\mathbf{e}_1-(u_3+u_4+u_5)\mathbf{e}_2+(u_1Z_{16}+u_2Z_{14})\mathbf{e}_3$$

where

$$Z_{14}=c_{45}$$

$$Z_{15}=s_{45}$$

$$Z_{16}=-Z_{15}$$

$$Z_{17}=u_1Z_{14}+u_2Z_{15}$$

$$Z_{18}=-Z_{13}-u_5$$

$$Z_{19}=u_1Z_{16}+u_2Z_{14}$$

and

$$\mathbf{w}^F = Z_{24}\mathbf{f}_1+Z_{25}\mathbf{f}_2+Z_{26}\mathbf{f}_3 \tag{2.25}$$

$$\mathbf{w}^F = (Z_{20}u_1+Z_{21}u_2+Z_{18}s_6)\mathbf{f}_1-(Z_{22}u_1+Z_{23}u_2-Z_{18}c_6)\mathbf{f}_2$$
$$+(Z_{16}u_1+Z_{14}u_2+u_6)\mathbf{f}_3$$

with

$$Z_{20}=Z_{14}c_6$$

$$Z_{21}=Z_{15}c_6$$

$$Z_{22}=Z_{14}s_6$$

$$Z_{23}=Z_{15}s_6$$

$$Z_{24}=Z_{20}u_1+Z_{21}u_2+Z_{18}s_6$$

$$Z_{25}=-Z_{22}u_1-Z_{23}u_2+Z_{18}c_6$$

$$Z_{26}=Z_{16}u_1+Z_{14}u_2+u_6$$

As for the velocities of links A, B and C, one notices that point O is fixed in the base reference frame N. With the vector from the point O to the mass center of a link denoted as $\mathbf{r}$, the angular velocity of this link denoted as $\mathbf{W}$,

$$\mathbf{V} = \mathbf{W} \times \mathbf{r} \tag{2.26}$$

For link A,

$$\mathbf{r} = OA^* = -L_{12}\mathbf{a}_2+L_{13}\mathbf{a}_3$$

Substitution of $\mathbf{r}$ and $\mathbf{W}^A$ into equation (2.26) and introduction of

$$Z_{27}=L_{13}Z_4$$

$$Z_{28}=L_{13}Z_5$$

$$Z_{29}=Z_{27}u_1+Z_{28}u_2$$

lead to

$$\mathbf{V}^{A*} = (Z_{27}u_1+Z_{28}u_2)\mathbf{a}_1 \tag{2.27}$$

$$\mathbf{V}^{A*} = Z_{29}\mathbf{a}_1$$

Similarly, the velocities of the mass centers of link B and C are found to be

$$\mathbf{V}^{B*} = (Z_{30}u_1+Z_{31}u_2)\mathbf{b}_1+(Z_{32}u_1+Z_{33}u_2)\mathbf{b}_3 \tag{2.28}$$

$$\mathbf{V}^{B*} = Z_{34}\mathbf{b}_1+Z_{35}\mathbf{b}_3$$

where

$$Z_{30}=L_{22}Z_8$$

$$Z_{31}=L_{22}Z_9$$

$$Z_{32}=-L_{22}c_3$$

$$Z_{33}=L_{22}s_3$$

$$Z_{34}=Z_{30}u_1+Z_{31}u_2$$

$$Z_{35}=Z_{32}u_1+Z_{33}u_2$$

and

$$\mathbf{v}^{C*} = L_{33}u_2\mathbf{c}_1+(L_{31}u_3-L_{33}u_1)\mathbf{c}_2-L_{31}u_2\mathbf{c}_3$$

$$\mathbf{v}^{C*} = Z_{36}\mathbf{c}_1+Z_{37}\mathbf{c}_2+Z_{38}\mathbf{c}_3$$

where

$$Z_{36}=L_{33}u_2$$

$$Z_{37}=L_{31}u_3-L_{33}u_1$$

$$Z_{38}=-L_{31}u_2$$

The velocity of the mass center of link D is determined by applying the following equation

$$\mathbf{v}^{D*} = \mathbf{v}^P+\mathbf{w}^D X PD^* \qquad (2.30)$$

where $\mathbf{v}^P$ stands for the velocity of point P which is fixed in link C. $\mathbf{v}^P$ can be obtained by simply replacing $L_{31}u_2\mathbf{c}_3$ in the expression of $\mathbf{v}^{C*}$ with $L_1u_2\mathbf{c}_3$; that is,

$$\mathbf{v}^P = L_{33}u_2\mathbf{c}_1+(L_{31}u_3-L_{33}u_1)\mathbf{c}_2-L_1u_2\mathbf{c}_3$$

Notice that

$$PD^*= L_{41}\mathbf{d}_1+L_{43}\mathbf{d}_3$$

Substituting these expressions and $\mathbf{w}^D$ into equation(2.30), one obtains

$$\mathbf{v}^{D*}=(-Z_{39}u_1+Z_{40}u_2+Z_{41}u_3)\mathbf{d}_1+(-Z_{40}u_1-Z_{39}u_2+Z_{42}u_3+Z_{41}u_4)\mathbf{d}_2$$
$$+(Z_{43}u_1-Z_{44}u_2)\mathbf{d}_3$$

$$\mathbf{v}^{D*} = Z_{45}\mathbf{d}_1+Z_{46}\mathbf{d}_2+Z_{47}\mathbf{d}_3 \qquad (2.31)$$

with

$$Z_{39}=(L_{33}+L_{43})s_4$$

$$Z_{40}=(L_{33}+L_{43})c_4$$

$$Z_{41}=L_1s_4$$

$$Z_{42}=L_1c_4+L_{41}$$

$$Z_{43}=L_{41}s_4$$

$$Z_{44}=L_1+L_{41}c_4$$

$$Z_{45}=-Z_{39}u_1+Z_{40}u_2+Z_{41}u_3$$

$$Z_{46}=-Z_{40}u_1-Z_{39}u_2+Z_{42}u_3+L_{41}u_4$$

$$Z_{47}=Z_{43}u_1-Z_{44}u_2$$

Following exactly the same procedure, $\mathbf{v}^{E*}$ and $\mathbf{v}^{F*}$ are found to be

$$\mathbf{v}^{E*} = (Z_{54}u_1+Z_{55}u_2+Z_{56}u_3+Z_{57}u_4)\mathbf{e}_1+(-Z_{49}u_1+Z_{50}u_2)\mathbf{e}_2$$
$$+(-Z_{55}u_1+Z_{54}u_2+Z_{58}u_3+Z_{59}u_4)\mathbf{e}_3$$

$$\mathbf{v}^{E*} = Z_{60}\mathbf{e}_1+Z_{61}\mathbf{e}_2+Z_{62}\mathbf{e}_3$$

with

$$Z_{48}=L_1c_4+L_2$$

$$Z_{49}=L_2s_4$$

$$Z_{50}=L_1+L_2c_4$$

$$Z_{51}=L_{33}+L_{43}-L_{52}$$

$$Z_{52}=Z_{39}c_5+Z_{40}s_5$$

$$Z_{53}=Z_{40}c_5-Z_{39}s_5$$

$$Z_{54}=Z_{15}Z_{51}-Z_{52}$$

$$Z_{55}=Z_{53}-Z_{14}Z_{51}$$

$$Z_{56}=Z_{41}c_5+Z_{48}s_5$$

$$Z_{57}=L_2s_5$$

$$Z_{58}=Z_{48}c_5-Z_{41}s_5$$

$$Z_{59}=L_2c_5$$

$$Z_{60}=Z_{54}u_1+Z_{55}u_2+Z_{56}u_3+Z_{57}u_4$$

$$Z_{61}=-Z_{49}u_1+Z_{50}u_2$$

$$Z_{62}=-Z_{55}u_1+Z_{54}u_2+Z_{58}u_3+Z_{59}u_4$$

and

$$\mathbf{v}^{F*} = (Z_{64}u_1+Z_{65}u_2+Z_{66}u_3+Z_{67}u_4-L_{63}u_5)\mathbf{e}_1+(Z_{68}u_1+Z_{69}u_2)\mathbf{e}_2$$
$$+(-Z_{65}u_1+Z_{64}u_2+Z_{58}u_3+Z_{59}u_4)\mathbf{e}_3$$
$$\mathbf{v}^{F*} = Z_{70}\mathbf{e}_1+Z_{71}\mathbf{e}_2+Z_{72}\mathbf{e}_3 \qquad (2.32)$$

with

$Z_{63}=L_{33}+L_{43}$

$Z_{64}=Z_{15}Z_{63}-Z_{52}$

$Z_{65}=Z_{53}-Z_{14}Z_{63}$

$Z_{66}=Z_{56}-L_{63}$

$Z_{67}=Z_{57}-L_{63}$

$Z_{68}=-Z_{49}-L_{63}Z_{14}$

$Z_{69}=Z_{50}-L_{63}Z_{15}$

$Z_{70}=Z_{64}u_1+Z_{65}u_2+Z_{66}u_3+Z_{67}u_4-L_{63}u_5$

$Z_{71}=Z_{68}u_1+Z_{69}u_2$

$Z_{72}=-Z_{65}u_1+Z_{64}u_2+Z_{58}u_3+Z_{59}u_4$

The partial velocities and the partial angular velocities can be obtained by simply inspecting the coefficients of the generalized speeds in the expressions of velocities and angular velocities. For the robot under consideration, the partial angular velocities and the partial velocities are listed in Table 1 and Table 2 respectively. In the tables, the capital letters of the first columns identify the link or point in question and the number of the first rows stand for the generalized speed in question. For example, $\mathbf{w}_3^D$, the third partial angular velocity of link D can be found by checking the element on the third column and row D in Table 1, i.e.

$$\mathbf{w}_3^D = \mathbf{d}_3$$

TABLE 1. PARTIAL ANGULAR VELOCITIES ($\mathbf{w}_r^R$)

	1	2	3	4	5	6
A	$Z_4\mathbf{a}_2$	$Z_5\mathbf{a}_2$	0	0	0	0
B	$c_3\mathbf{b}_1+s_3\mathbf{b}_2+Z_8\mathbf{b}_3$	$-s_3\mathbf{b}_1+c_3\mathbf{b}_2+Z_9\mathbf{b}_3$	0	0	0	0
C	$\mathbf{c}_1$	$\mathbf{c}_2$	$\mathbf{c}_3$	0	0	0
D	$c_4\mathbf{d}_1-s_4\mathbf{d}_2$	$s_4\mathbf{d}_1+c_4\mathbf{d}_2$	$\mathbf{d}_3$	$\mathbf{d}_3$	0	0
E	$Z_{14}\mathbf{e}_1-Z_{16}\mathbf{e}_3$	$Z_{15}\mathbf{e}_1+Z_{14}\mathbf{e}_3$	$-\mathbf{e}_2$	$-\mathbf{e}_2$	$-\mathbf{e}_2$	0
F	$Z_{20}\mathbf{f}_1-Z_{22}\mathbf{f}_2+Z_{16}\mathbf{f}_3$	$Z_{21}\mathbf{f}_1-Z_{23}\mathbf{f}_2+Z_{14}\mathbf{f}_3$	$-s_6\mathbf{f}_1-c_6\mathbf{f}_2$	$-s_6\mathbf{f}_1-c_6\mathbf{f}_2$	$-s_6\mathbf{f}_1-c_6\mathbf{f}_2$	$\mathbf{f}_3$

TABLE 2. PARTIAL VELOCITIES ($\mathbf{v}_r^R$)

	1	2	3	4	5	6
A*	$Z_{27}\mathbf{a}_1$	$Z_{28}\mathbf{a}_1$	0	0	0	0
B*	$Z_{30}\mathbf{b}_1+Z_{31}\mathbf{b}_3+Z_8\mathbf{b}_3$	$Z_{31}\mathbf{b}_1+Z_{33}\mathbf{b}_3+Z_9\mathbf{b}_3$	0	0	0	0
C*	$-L_{33}\mathbf{c}_1$	$L_{33}\mathbf{c}_1-L_{31}\mathbf{c}_3$	$L_{31}\mathbf{c}_2$	0	0	0
D*	$-Z_{39}\mathbf{d}_1-Z_{40}\mathbf{d}_2+Z_{43}\mathbf{d}_3$	$Z_{40}\mathbf{d}_1-Z_{39}\mathbf{d}_2-Z_{44}\mathbf{d}_3$	$Z_{41}\mathbf{d}_1+Z_{42}\mathbf{d}_2$	$L_{41}\mathbf{d}_2$	0	0
E*	$Z_{54}\mathbf{e}_1-Z_{49}\mathbf{e}_2-Z_{55}\mathbf{e}_3$	$Z_{55}\mathbf{e}_1+Z_{50}\mathbf{e}_2+Z_{54}\mathbf{e}_3$	$Z_{56}\mathbf{e}_1+Z_{58}\mathbf{e}_3$	$Z_{57}\mathbf{e}_1+Z_{59}\mathbf{e}_3$	0	0
F*	$Z_{64}\mathbf{e}_1+Z_{68}\mathbf{e}_2-Z_{55}\mathbf{e}_3$	$Z_{65}\mathbf{e}_1+Z_{69}\mathbf{e}_2+Z_{54}\mathbf{e}_3$	$Z_{66}\mathbf{e}_1+Z_{58}\mathbf{e}_3$	$Z_{67}\mathbf{e}_1+Z_{59}\mathbf{e}_3$	$-L_{63}\mathbf{e}_1$	0

Next, the angular acceleration for each link can be obtained by differentiating the corresponding angular velocity with respect to time t. It should be noted that, in these expressions, the derivatives of the generalized speeds should be brought into evidence explicitly. For example, $\boldsymbol{\alpha}^A$, the angular acceleration of link A, can be found by differentiating equation (2.21),

$$\boldsymbol{\alpha}^A = (Z_4\dot{u}_1+\dot{Z}_4u_1+Z_5\dot{u}_2+\dot{Z}_5u_2)\mathbf{a}_2$$

With Z_{73}, Z_{74} and Z_{75} defined as in Appendix 1, this can be written as

$$\boldsymbol{\alpha}^A = (Z_4\dot{u}_1+Z_5\dot{u}_2+Z_{75})\mathbf{a}_2$$

Similarly, differentiation of equations (2.22-2.26) and introduction of the quantities $Z_{76},\ldots,Z_{93}$ as defined in Appendix 1, lead to

$$\boldsymbol{\alpha}^B = (c_3\dot{u}_1-s_3\dot{u}_2+Z_{78})\mathbf{b}_1+(s_3\dot{u}_1+c_3\dot{u}_2+Z_{79})\mathbf{b}_2 + (Z_8\dot{u}_1+Z_9\dot{u}_2+Z_{80})\mathbf{b}_3 \tag{2.34}$$

$$\boldsymbol{\alpha}^C = \dot{u}_1\mathbf{c}_1+\dot{u}_2\mathbf{c}_2+\dot{u}_3\mathbf{c}_3 \tag{2.35}$$

$$\boldsymbol{\alpha}^D = (c_4\dot{u}_1+s_4\dot{u}_2+Z_{81})\mathbf{d}_1+(-s_4\dot{u}_1+c_4\dot{u}_2+Z_{82})\mathbf{d}_2 + (\dot{u}_3+\dot{u}_4)\mathbf{d}_3 \tag{2.36}$$

$$\boldsymbol{\alpha}^E = (Z_{14}\dot{u}_1+Z_{15}\dot{u}_2+Z_{85})\mathbf{e}_1-(\dot{u}_3+\dot{u}_4+\dot{u}_5)\mathbf{e}_2 + (Z_{16}\dot{u}_1+Z_{14}\dot{u}_2+Z_{86})\mathbf{e}_3 \tag{2.37}$$

$$\boldsymbol{\alpha}^F = (Z_{20}\dot{u}_1+Z_{21}\dot{u}_2-s_6(\dot{u}_3+\dot{u}_4+\dot{u}_5))\mathbf{f}_1-(Z_{22}\dot{u}_1+Z_{23}\dot{u}_2 + c_6(\dot{u}_3+\dot{u}_4+\dot{u}_5))\mathbf{f}_2 + (s_3\dot{u}_1+c_3\dot{u}_2+Z_{79})\mathbf{b}_2 \tag{2.38}$$

The last group of kinematic ingredients, the accelerations of the mass centers for each link, can be found by employing the following equation

$$^{N}\mathbf{a}^{P} = \frac{^{R}d\,^{N}\mathbf{v}^{P}}{dt} + {}^{N}\mathbf{w}^{R} \times {}^{N}\mathbf{v}^{P}$$

As with angular accelerations, the time derivatives of the generalized speeds should be brought into evidence in the expressions of the accelerations. Therefore, the explicit forms of velocities should be used in determining the first term of the above equation, and the implicit forms for both the velocities and the angular velocities are to be used to carry out the second term in the equation. For example, to evaluate $\mathbf{a}^{A*}$, the acceleration of the mass center of link A, one can first differentiate equation (2.28),

$$d\mathbf{v}^{A*}/dt = (Z_{27}\dot{u}_1+Z_{28}\dot{u}_2+\dot{Z}_{27}u_1+\dot{Z}_{28}u_2)\mathbf{a}_1,$$

then determine the cross product

$$\mathbf{w}^{A}X\mathbf{v}^{A*} = -Z_1Z_{29}\mathbf{a}_3,$$

add them together and define Z_{94} and Z_{95} as in Appendix 1. The result is

$$\mathbf{a}^{A*} = (Z_{27}\dot{u}_1+Z_{28}\dot{u}_2+Z_{94})\mathbf{a}_1+Z_{95}\mathbf{a}_3 \tag{2.40}$$

Similarly, $\mathbf{a}^{B*}$, ..., $\mathbf{a}^{F*}$ are found to be

$$\mathbf{a}^{B*}=(Z_{30}\dot{u}_1+Z_{31}\dot{u}_2+Z_{100})\mathbf{b}_1+Z_{101}\mathbf{b}_2+(Z_{32}\dot{u}_1+Z_{33}\dot{u}_2+Z_{102})\mathbf{b}_3 \tag{2.41}$$

$$\mathbf{a}^{C*}=(L_{33}\dot{u}_2+Z_{102})\mathbf{c}_1+(-L_{33}\dot{u}_1+L_{31}\dot{u}_3+Z_{104})\mathbf{c}_2+(-L_{31}\dot{u}_1+Z_{105})\mathbf{c}_3 \tag{2.42}$$

$$\mathbf{a}^{D*}=(-Z_{39}\dot{u}_1+Z_{40}\dot{u}_2+Z_{41}\dot{u}_3+Z_{112})\mathbf{d}_1$$
$$+(-Z_{40}\dot{u}_1-Z_{39}\dot{u}_2+Z_{42}\dot{u}_3+L_{41}\dot{u}_4+Z_{113})\mathbf{d}_2+(Z_{43}\dot{u}_1-Z_{44}\dot{u}_2+Z_{114})\mathbf{d}_3 \tag{2.43}$$

$$\mathbf{a}^{E*}=(Z_{54}\dot{u}_1+Z_{55}\dot{u}_2+Z_{56}\dot{u}_3+Z_{57}\dot{u}_4+Z_{125})\mathbf{e}_1+(-Z_{49}\dot{u}_1+Z_{50}\dot{u}_2+Z_{126})\mathbf{e}_2$$
$$+(-Z_{55}\dot{u}_1+Z_{54}\dot{u}_2+Z_{58}\dot{u}_3+Z_{59}\dot{u}_4+Z_{127})\mathbf{e}_3 \tag{2.44}$$

$$\mathbf{a}^{F*}=(Z_{64}\dot{u}_1+Z_{65}\dot{u}_2+Z_{66}\dot{u}_3+Z_{67}\dot{u}_4-L_{63}\dot{u}_5+Z_{135})\mathbf{e}_1$$
$$+(Z_{68}\dot{u}_1+Z_{69}\dot{u}_2+Z_{136})\mathbf{e}_2+(-Z_{65}\dot{u}_1+Z_{64}\dot{u}_2+Z_{58}\dot{u}_3+Z_{59}\dot{u}_4+Z_{137})\mathbf{e}_3 \tag{2.45}$$

II.4 GENERALIZED INERTIA FORCES

Based on the kinematic analysis performed in the previous section, generalized inertia forces are now to be determined. The definition for the generalized inertia forces can be found in references [36,39] as follows.

If S is a simple nonholonomic system possessing n-m degrees of freedom in a reference frame N, n-m quantities k_1^*, ..., k_{n-m}^*, called generalized inertia forces for S in N, are defined as

$$k_r^* = \sum_{i=1}^{n} \mathbf{v}_r^{Pi} \cdot \mathbf{R}_i^* \qquad (r=1,\ 2,\ \ldots,\ n\text{-}m) \tag{2.46}$$

where n is the number of particles comprising S, P_i is a typical particle, $\mathbf{v}_r^{Pi}$ is the rth partial velocity of P_i and $\mathbf{R}_i^*$, the inertia force for P_i in N, is given by

$$\mathbf{R}_i^* = -m_i \mathbf{a}_i \tag{2.47}$$

Furthermore, the contribution to the rth generalized inertia force made by the particles of a rigid body R belonging to S, denoted by $(k_r^*)_R$, is given by

$$(k_r^*)_R = \mathbf{w}_r \cdot \mathbf{T}^* + \mathbf{v}_r \cdot \mathbf{R}^* \qquad (r=1,2,\ldots,n) \tag{2.48}$$

where $\mathbf{w}_r$ and $\mathbf{v}_r$ are, respectively, the rth partial angular velocity and the rth partial velocity of mass center of R in N, and

$$\mathbf{T}^* = -\mathbf{I} \cdot \boldsymbol{\alpha} - \mathbf{W} \mathbf{X} \mathbf{I} \cdot \mathbf{W} \tag{2.49}$$

For the example robot in discussion, assume that the unit vectors $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ are parallel to central principal axes of inertia of link A, and A_1, A_2, A_3 are the central principal moments of the inertia, so that $\mathbf{I}^A$, the central inertia dyadic of A, can be expressed as

$$\mathbf{I}^A = A_1\mathbf{a}_1\mathbf{a}_1+A_2\mathbf{a}_2\mathbf{a}_2+A_3\mathbf{a}_3\mathbf{a}_3$$

Similarly, the central inertia dyadic of the links B,C,...,F are respectively defined as

$$\mathbf{I}^B = B_1\mathbf{b}_1\mathbf{b}_1+B_2\mathbf{b}_2\mathbf{b}_2+B_3\mathbf{b}_3\mathbf{b}_3$$

$$\mathbf{I}^C = C_1\mathbf{c}_1\mathbf{c}_1+C_2\mathbf{c}_2\mathbf{c}_2+C_3\mathbf{c}_3\mathbf{c}_3$$

$$\mathbf{I}^D = D_1\mathbf{d}_1\mathbf{d}_1+D_2\mathbf{d}_2\mathbf{d}_2+D_3\mathbf{d}_3\mathbf{d}_3$$

$$\mathbf{I}^E = E_1\mathbf{e}_1\mathbf{e}_1+E_2\mathbf{e}_2\mathbf{e}_2+E_3\mathbf{e}_3\mathbf{e}_3$$

$$\mathbf{I}^F = F_1\mathbf{f}_1\mathbf{f}_1+F_2\mathbf{f}_2\mathbf{f}_2+F_3\mathbf{f}_3\mathbf{f}_3$$

where B_1, B_2, ..., F_2, F_3 denote the central principal moments of inertia of the links B, C, ..., F respectively.

In accordance with equation (2.47), the inertia force of link A, can be written

$$\mathbf{R}_A^* = -m_A(Z_{27}\dot{u}_1+Z_{28}\dot{u}_2+Z_{94})\mathbf{a}_1-m_AZ_{95}\mathbf{a}_3$$

where m_A is the mass of link A. Meanwhile, equation (2.49) indicates that the inertia torque of A, is given by

$$\mathbf{T}_A^* = -\mathbf{I}^A\cdot\boldsymbol{\alpha}^A - \mathbf{w}^A\mathrm{x}\mathbf{I}^A\cdot\mathbf{w}^A$$

This gives

$$\mathbf{T}_A^* = -A_2(Z_4\dot{u}_1+Z_5\dot{u}_2+Z_{75})\mathbf{a}_2$$

On the other hand, equation (2.48) gives

$$(k_r^*)_A = \mathbf{w}_r^A\cdot\mathbf{T}_A^* + \mathbf{v}_r^{A*}\cdot\mathbf{R}_A^* \qquad (r=1,\ 2,\ ...,\ 6)$$

Substitution $\mathbf{T}_A^*$, $\mathbf{R}_A^*$ and the corresponding partial angular velocities $\mathbf{w}_r^{\mathbf{A}}$, partial velocities $\mathbf{v}_r^{\mathbf{A}*}$ (which can be found from Table 1 and Table 2) into the above equation for r=1, ..., 6 results in

$$(k_1^*)_A = -A_2Z_4(Z_4\dot{u}_1+Z_5\dot{u}_2+Z_{75})-m_AZ_{27}(Z_{27}\dot{u}_1+Z_{28}\dot{u}_2+Z_{94})$$
$$(k_2^*)_A = -A_2Z_5(Z_4\dot{u}_1+Z_5\dot{u}_2+Z_{75})-m_AZ_{28}(Z_{27}\dot{u}_1+Z_{28}\dot{u}_2+Z_{94})$$
$$(k_r^*)_A = 0 \qquad (r=3, \ldots, 6)$$

After introducing the quantities Z_{152} through Z_{161}, as defined in Appendix 1, the contributions to the generalized inertia forces made by link A then turn out to be

$$(k_1^*)_A = Z_{154}\dot{u}_1+Z_{155}\dot{u}_2+Z_{156}$$
$$(k_2^*)_A = Z_{159}\dot{u}_1+Z_{160}\dot{u}_2+Z_{161}$$
$$(k_r^*)_A = 0 \qquad (r=3, \ldots, 6)$$

The contributions to the generalized inertia forces made by other links are determined in the same way as indicated above. Quantities Z_{162} through Z_{242}, involved in the expressions of the contributions are as defined in Appendix 1. The expressions for these contributions are listed below.

Contributions made by link B:

$$(k_1^*)_B = Z_{165}\dot{u}_1+Z_{166}\dot{u}_2+Z_{167}$$
$$(k_2^*)_B = Z_{168}\dot{u}_1+Z_{169}\dot{u}_2+Z_{170}$$
$$(k_r^*)_B = 0 \qquad (r=3, 4, \ldots, 6)$$

Contributions made by link C:

$$(k_1^*)_C = Z_{174}\dot{u}_1+Z_{175}\dot{u}_2+Z_{176}$$
$$(k_2^*)_C = Z_{178}\dot{u}_2+Z_{179}$$
$$(k_3^*)_C = Z_{180}\dot{u}_1+Z_{181}\dot{u}_2+Z_{182}$$

$(k_r^*)_C = 0 \qquad (r=4,\ 5,\ 6)$

Contributions made by link F:

$(k_1^*)_D = Z_{186}\dot{u}_1+Z_{187}\dot{u}_2+Z_{188}\dot{u}_3+Z_{194}\dot{u}_4+Z_{189}$

$(k_2^*)_D = Z_{187}\dot{u}_1+Z_{189A}\dot{u}_2+Z_{190}\dot{u}_3+Z_{195}\dot{u}_4+Z_{191}$

$(k_3^*)_D = Z_{188}\dot{u}_1+Z_{190}\dot{u}_2+Z_{192}\dot{u}_3+Z_{196}\dot{u}_4+Z_{193}$

$(k_4^*)_D = Z_{194}\dot{u}_1+Z_{195}\dot{u}_2+Z_{196}\dot{u}_3+Z_{197}\dot{u}_4+Z_{198}$

$(k_r^*)_D = 0 \qquad (r=5,\ 6)$

Contributions made by link F:

$(k_1^*)_E = Z_{202}\dot{u}_1+Z_{203}\dot{u}_2+Z_{204}\dot{u}_3+Z_{205}\dot{u}_4+Z_{206}$

$(k_2^*)_E = Z_{203}\dot{u}_1+Z_{207}\dot{u}_2+Z_{208}\dot{u}_3+Z_{097}\dot{u}_4+Z_{210}$

$(k_3^*)_E = Z_{204}\dot{u}_1+Z_{208}\dot{u}_2+Z_{211}\dot{u}_3+Z_{212}\dot{u}_4-E_2\dot{u}_5+Z_{213}$

$(k_4^*)_E = Z_{205}\dot{u}_1+Z_{209}\dot{u}_2+Z_{212}\dot{u}_3+Z_{214}\dot{u}_4-E_2\dot{u}_5+Z_{215}$

$(k_5^*)_E = -E_2(\dot{u}_3+\dot{u}_4+\dot{u}_5)+Z_{200}$

$(k_6^*)_E = 0$

Contributions made by link F:

$(k_1^*)_F = Z_{219}\dot{u}_1+Z_{220}\dot{u}_2+Z_{221}\dot{u}_3+Z_{222}\dot{u}_4+Z_{223}\dot{u}_5+Z_{240}\dot{u}_6+Z_{224}$

$(k_2^*)_F = Z_{220}\dot{u}_1+Z_{225}\dot{u}_2+Z_{226}\dot{u}_3+Z_{227}\dot{u}_4+Z_{228}\dot{u}_5+Z_{241}\dot{u}_6+Z_{229}$

$(k_3^*)_F = Z_{221}\dot{u}_1+Z_{226}\dot{u}_2+Z_{230}\dot{u}_3+Z_{231}\dot{u}_4+Z_{232}\dot{u}_5+Z_{234}$

$(k_4^*)_F = Z_{222}\dot{u}_1+Z_{227}\dot{u}_2+Z_{231}\dot{u}_3+Z_{235}\dot{u}_4+Z_{226}\dot{u}_5+Z_{237}$

$(k_5^*)_F = Z_{223}\dot{u}_1+Z_{228}\dot{u}_2+Z_{232}\dot{u}_3+Z_{236}\dot{u}_4+Z_{238}\dot{u}_5+Z_{239}$

$(k_6^*)_F = Z_{240}\dot{u}_1+Z_{241}\dot{u}_2-F_3\dot{u}_6+Z_{242}$

Finally, the generalized inertia forces are constructed by summing the corresponding contributions of each link, i.e.

$$\begin{aligned} k_r^* &= (k_r^*)_A+(k_r^*)_B+\ldots+(k_r^*)_F \\ &= \sum_{s=1}^{6} X_{rs}\dot{u}_s+Z_{242+r} \qquad (r=1,\ \ldots,\ 6) \qquad (2.50) \end{aligned}$$

i.e.

$$
\begin{aligned}
k_1^* &= X_{11}\dot{u}_1+X_{12}\dot{u}_2+X_{13}\dot{u}_3+X_{14}\dot{u}_4+X_{15}\dot{u}_5+X_{16}\dot{u}_6+Z_{243} \\
k_2^* &= X_{21}\dot{u}_1+X_{22}\dot{u}_2+X_{23}\dot{u}_3+X_{24}\dot{u}_4+X_{25}\dot{u}_5+X_{26}\dot{u}_6+Z_{244} \\
k_3^* &= X_{31}\dot{u}_1+X_{32}\dot{u}_2+X_{33}\dot{u}_3+X_{34}\dot{u}_4+X_{35}\dot{u}_5+X_{36}\dot{u}_6+Z_{245} \qquad (2.51) \\
k_4^* &= X_{41}\dot{u}_1+X_{42}\dot{u}_2+X_{43}\dot{u}_3+X_{44}\dot{u}_4+X_{45}\dot{u}_5+X_{46}\dot{u}_6+Z_{246} \\
k_5^* &= X_{51}\dot{u}_1+X_{52}\dot{u}_2+X_{53}\dot{u}_3+X_{54}\dot{u}_4+X_{55}\dot{u}_5+X_{56}\dot{u}_6+Z_{247} \\
k_6^* &= X_{61}\dot{u}_1+X_{62}\dot{u}_2+X_{63}\dot{u}_3+X_{64}\dot{u}_4+X_{65}\dot{u}_5+X_{66}\dot{u}_6+Z_{248}
\end{aligned}
$$

where Z_{243}, ..., Z_{248} and X_{11}, X_{12}, through X_{66} are listed in Appendix 1.

II.5. GENERALIZED ACTIVE FORCES AND DYNAMIC EQUATIONS

Introduction generalized active forces, according to Kane's method, results in considerable advantage over the Newton-Euler's formulation, because the process eliminates nonworking contact forces. This occurs because many forces that contribute to the resultant acting on a body make no contributions to the generalized active forces.

If S is a simple nonholonomic system possessing n-m degrees of freedom in a reference frame N, n-m quantities $k_1,\ldots,k_{n-m}$, called generalized active forces for S in N, are defined as [36]

$$k_r = \sum_{i=1}^{n} \mathbf{v}_r^{Pi}\cdot\mathbf{R}_i \qquad (r=1,\ 2,\ \ldots,\ n-m) \qquad (2.53)$$

where n is the number of particles comprising S, P_i is a typical particle, $\mathbf{v}_r^{Pi}$ is the rth partial velocity of P_i in N and $\mathbf{R}_i$ is the resultant of all contact and body forces acting on P_i. Furthermore, if a set of contact and body forces acting on a rigid body B is equivalent to a couple of torque $\mathbf{T}$ together with a force $\mathbf{R}$ applied at a point Q of B, then, $(k_r)_B$, the contribution of this set of forces to the rth generalized active force k_r, is given by [36]

$$(k_r)_B = \mathbf{w}_r \cdot \mathbf{T} + \mathbf{v}_r^{Q} \cdot \mathbf{R} \qquad (r=1,\ 2,\ \ldots,\ n) \qquad (2.54)$$

where $\mathbf{w}_r$ and $\mathbf{v}_r^Q$ are, respectively, the rth partial angular velocity of B in N and the rth partial velocity of Q in N.

In the case of the Intelledex 605, there are two kinds of forces that contribute to the generalized active forces, namely, contact forces applied in order to drive the links A,B,...,F, and gravitational forces exerted on each link by the earth.

Consider first the gravitational forces denoted by $\mathbf{G}_A$, $\mathbf{G}_B$, ..., $\mathbf{G}_F$, respectively. In the base coordinate frame, these forces turn out to be

$$\mathbf{G}_A = m_A g \mathbf{n}_2$$
$$\mathbf{G}_B = m_B g \mathbf{n}_2$$
$$\mathbf{G}_C = m_C g \mathbf{n}_2$$
$$\mathbf{G}_D = m_D g \mathbf{n}_2$$
$$\mathbf{G}_E = m_E g \mathbf{n}_2$$
$$\mathbf{G}_F = m_F g \mathbf{n}_2$$

Since $\mathbf{G}_A$, $\mathbf{G}_B$, ..., $\mathbf{G}_F$ are to be dot-multiplied by the partial velocities $\mathbf{v}_r^{A*}$, ..., $\mathbf{v}_r^{F*}$ respectively, it is convenient to express each $\mathbf{G}$ in terms of the coordinate vectors in which the corresponding partial velocities are expressed. With reference to equations (2.1) to (2.6), one can express these in the form

$$\mathbf{G}_A = m_A g(s_1 \mathbf{a}_1 - c_1 \mathbf{a}_3)$$

$$\mathbf{G}_B = m_B g(s_1 c_2 \mathbf{b}_1 + c_1 \mathbf{b}_2 - s_1 s_2 \mathbf{b}_3)$$

$$\mathbf{G}_C = m_C g\{(s_1 c_2 c_3 + c_1 s_3)\mathbf{c}_1 + (c_1 c_3 - s_1 c_2)\mathbf{c}_2 - s_1 s_2 \mathbf{c}_2\}$$

$$\mathbf{G}_D = m_D g\{(s_1 c_2 c_{34} + c_1 s_{34})\mathbf{d}_1 + (c_1 c_{34} - s_1 c_2 s_{34})\mathbf{d}_2 - s_1 s_2 \mathbf{d}_3\}$$

$$\mathbf{G}_E = m_E g\{(s_1 c_2 c_{345} + c_1 s_{345})\mathbf{e}_1 + s_1 s_2 \mathbf{e}_2 + (c_1 c_{345} - s_1 c_2 s_{345})\mathbf{e}_3\}$$

$$\mathbf{G}_F = m_F g\{(s_1 c_2 c_{345} + c_1 s_{345})\mathbf{e}_1 + s_1 s_2 \mathbf{e}_2 + (c_1 c_{345} - s_1 c_2 s_{345})\mathbf{e}_3\}$$

Having accounted for the gravitational forces, contact forces are now to be considered. The set of such forces transmitted from the robot base to the first link A is replaced with a couple of torque $\mathbf{T}^{N/A}$ together with a force $\mathbf{R}^{N/A}$ applied to the link at the mass center A^*. Similarly, the set of contact forces applied to link A by link B can also be replaced with a couple of torque $\mathbf{T}^{B/A}$ together with a force $\mathbf{R}^{B/A}$ applied to A at $\underline{B}^*$ which coincides with B^* and is fixed in link A. By Newton's Third Law, it is known that the set of contact forces transmitted from A to B (and likewise from B to C and so on) is equivalent to a couple of torque $-\mathbf{T}^{B/A}$ together with a force $-\mathbf{R}^{B/A}$ applied to B at the mass center B^* of link B. Likewise, the set of contact forces exerted on link B by link C is replaced with a couple of torque $\mathbf{T}^{C/B}$ together with a force $\mathbf{R}^{C/B}$ applied to link B at $\underline{C}^*$ which coincides with C^* and is fixed in B. Consequently, the set of contact forces exerted on C by B is equivalent to a couple of torque $-\mathbf{T}^{C/B}$ together with a force $-\mathbf{R}^{C/B}$ applied to C at the mass center C^*. Similarly, $\mathbf{T}^{D/C}$, $\mathbf{T}^{E/D}$, $\mathbf{T}^{F/E}$ and $\mathbf{R}^{D/C}$, $\mathbf{R}^{E/D}$, $\mathbf{R}^{F/E}$ and defined as the torques and the forces in connection with the interactions of link C and link D, link D and link E, link E and link F.

In accordance with equation (2.54), the contributions to the generalized active force k_r of all forces acting on each link can be expressed as

$$(k_r)_A = w_r^{A}\cdot(T^{N/A}+T^{B/A})+v_r^{A*}\cdot(R^{N/A}+G_A)+v_r^{B*}\cdot R^{B/A}$$

$$(k_r)_B = w_r^{B}\cdot(-T^{B/A}+T^{C/B})+v_r^{B*}\cdot(-R^{B/A}+G_B)+v_r^{C*}\cdot R^{C/B}$$

$$(k_r)_C = w_r^{C}\cdot(-T^{C/B}+T^{D/C})+v_r^{C*}\cdot(-R^{C/B}+G_C)+v_r^{D*}\cdot R^{D/C}$$

$$(k_r)_D = w_r^{D}\cdot(-T^{D/C}+T^{E/D})+v_r^{D*}\cdot(-R^{D/C}+G_D)+v_r^{E*}\cdot R^{E/D}$$

$$(k_r)_E = w_r^{E}\cdot(-T^{E/D}+T^{F/E})+v_r^{E*}\cdot(-R^{E/D}+G_E)+v_r^{F*}\cdot R^{F/E}$$

$$(k_r)_F = w_r^{F}\cdot(-T^{F/E})+v_r^{F*}\cdot(-R^{F/E}+G_F)$$

$$(r=1,\ 2,\ \ldots,\ 6)$$

The generalized active force k_r is then formed by summing its contributions, i.e.

$$k_r = (k_r)_A+(k_r)_B+\ \ldots\ +(k_r)_F \quad (r=1,\ \ldots,\ 6)$$

Hence,

$$\begin{aligned} k_r = {} & w_r^{A}\cdot T^{N/A}+(w_r^{A}-w_r^{B})\cdot T^{B/A}+(w_r^{B}-w_r^{C})\cdot T^{C/B}+(w_r^{C}-w_r^{D})\cdot T^{D/C} \\ & +(w_r^{D}-w_r^{E})\cdot T^{E/D}+(w_r^{E}-w_r^{F})\cdot T^{F/E}+v_r^{A*}\cdot G_A \\ & +v_r^{B*}\cdot G_B+v_r^{C*}\cdot G_C+v_r^{D*}\cdot G_D+v_r^{E*}\cdot G_E+v_r^{F*}\cdot G_F \end{aligned}$$

$$(r=1,\ 2,\ \ldots,\ 6) \qquad (2.55)$$

Notice that as the contributions made by different links are summed together, all terms involving the nonworking contact forces $R^{B/A}$, $R^{C/B}$, $R^{D/C}$, $R^{E/D}$, $R^{F/E}$ are out, and the term $v_r^{A*}\cdot R^{N/A}$ vanishes, but all the working torques and the gravitational forces remain in evidence. This feature facilitates the task of deriving the equations of motion by Kane's method for the contact forces making no contributions to the generalized forces k_r's.

Next, define

$$\tau_1 = \mathbf{n}_3 \cdot \mathbf{T}^{N/A}$$
$$\tau_2 = -\mathbf{a}_3 \cdot \mathbf{T}^{B/A}$$
$$\tau_3 = -\mathbf{b}_3 \cdot \mathbf{T}^{C/B}$$
$$\tau_4 = -\mathbf{c}_3 \cdot \mathbf{T}^{D/C}$$
$$\tau_5 = -\mathbf{d}_3 \cdot \mathbf{T}^{E/D}$$
$$\tau_6 = -\mathbf{e}_3 \cdot \mathbf{T}^{F/E}$$

To determine the first term of equation (2.55), one notices that

$$\mathbf{w}^A = \dot{q}_1 \mathbf{n}_3 = (Z_4 u_1 + Z_5 u_2) \mathbf{n}_3$$

so that

$$\mathbf{w}_1{}^A \cdot \mathbf{T}^{N/A} = Z_4 \tau_1$$
$$\mathbf{w}_2{}^A \cdot \mathbf{T}^{N/A} = Z_5 \tau_1$$
$$\mathbf{w}_r{}^A \cdot \mathbf{T}^{N/A} = 0 \qquad (r=3, \ldots, 6)$$

As for the second term of the equation, first notice that

$$\mathbf{w}^A - \mathbf{w}^B = -\dot{q}_2 \mathbf{a}_3 = (u_1 s_3 + u_2 c_3) \mathbf{a}_3$$

Then, with the introduction of

$$\mathbf{w}_1{}^A - \mathbf{w}_1{}^B = s_3 \mathbf{a}_3$$
$$\mathbf{w}_2{}^A - \mathbf{w}_2{}^B = c_3 \mathbf{a}_3$$
$$\mathbf{w}_r{}^A - \mathbf{w}_r{}^B = 0 \qquad (r=3, \ldots, 6)$$

there results

$$(\mathbf{w}_1{}^A - \mathbf{w}_1{}^B) \cdot \mathbf{T}^{B/A} = -\tau_2 s_3$$
$$(\mathbf{w}_2{}^A - \mathbf{w}_2{}^B) \cdot \mathbf{T}^{B/A} = -\tau_2 c_3$$
$$(\mathbf{w}_r{}^A - \mathbf{w}_r{}^B) \cdot \mathbf{T}^{B/A} = 0 \qquad (r=3, \ldots, 6)$$

Similarly, one obtains

$$(\mathbf{w}_1{}^B - \mathbf{w}_1{}^C) \cdot \mathbf{T}^{C/B} = -\tau_3 Z_8$$
$$(\mathbf{w}_2{}^B - \mathbf{w}_2{}^C) \cdot \mathbf{T}^{C/B} = -\tau_3 Z_9$$
$$(\mathbf{w}_3{}^B - \mathbf{w}_3{}^C) \cdot \mathbf{T}^{C/B} = \tau_3$$

$$(\mathbf{w}_r{}^B-\mathbf{w}_r{}^C)\cdot\mathbf{T}^{C/B} = 0 \qquad (r=4,\ 5,\ 6)$$
$$(\mathbf{w}_4{}^C-\mathbf{w}_4{}^D)\cdot\mathbf{T}^{D/C} = \tau_4$$
$$(\mathbf{w}_r{}^A-\mathbf{w}_r{}^B)\cdot\mathbf{T}^{D/C} = 0 \qquad (r=1,\ 2,\ 3,\ 5,\ 6)$$
$$(\mathbf{w}_5{}^D-\mathbf{w}_5{}^E)\cdot\mathbf{T}^{E/D} = \tau_5$$
$$(\mathbf{w}_r{}^D-\mathbf{w}_r{}^E)\cdot\mathbf{T}^{E/D} = 0 \qquad (r=1,\ \ldots,\ 4,\ 6)$$
$$(\mathbf{w}_6{}^E-\mathbf{w}_6{}^F)\cdot\mathbf{T}^{E/F} = \tau_6$$
$$(\mathbf{w}_r{}^E-\mathbf{w}_r{}^F)\cdot\mathbf{T}^{E/F} = 0 \qquad (r=1,\ \ldots,\ 5)$$

The last six terms in equation (2.55) are determined by dot-multiplying each gravitational force with the partial velocity of the corresponding mass center and introducing the quantities Z_{253}, Z_{254}, ..., Z_{280} as defined in Appendix 1. These turn out to be

$$\mathbf{v}_1{}^{A*}\cdot\mathbf{G}_A = gZ_{252}$$
$$\mathbf{v}_2{}^{A*}\cdot\mathbf{G}_A = gZ_{253}$$
$$\mathbf{v}_r{}^{A*}\cdot\mathbf{G}_A = 0 \qquad (r=3,\ \ldots,\ 6)$$
$$\mathbf{v}_1{}^{B*}\cdot\mathbf{G}_B = gZ_{256}$$
$$\mathbf{v}_2{}^{B*}\cdot\mathbf{G}_B = gZ_{257}$$
$$\mathbf{v}_r{}^{B*}\cdot\mathbf{G}_B = 0 \qquad (r=3,\ \ldots,\ 6)$$
$$\mathbf{v}_1{}^{C*}\cdot\mathbf{G}_C = gZ_{259}$$
$$\mathbf{v}_2{}^{C*}\cdot\mathbf{G}_C = gZ_{260}$$
$$\mathbf{v}_3{}^{C*}\cdot\mathbf{G}_C = gZ_{261}$$
$$\mathbf{v}_r{}^{C*}\cdot\mathbf{G}_C = 0 \qquad (r=4,\ 5,\ 6)$$
$$\mathbf{v}_1{}^{D*}\cdot\mathbf{G}_D = gZ_{265}$$
$$\mathbf{v}_2{}^{D*}\cdot\mathbf{G}_D = gZ_{266}$$
$$\mathbf{v}_3{}^{D*}\cdot\mathbf{G}_D = gZ_{267}$$
$$\mathbf{v}_4{}^{D*}\cdot\mathbf{G}_D = gZ_{268}$$
$$\mathbf{v}_r{}^{D*}\cdot\mathbf{G}_D = 0 \qquad (r=5,\ 6)$$

$$v_1^{E*} \cdot G_E = gZ_{272}$$
$$v_2^{E*} \cdot G_E = gZ_{273}$$
$$v_3^{E*} \cdot G_E = gZ_{274}$$
$$v_4^{E*} \cdot G_E = gZ_{275}$$
$$v_r^{E*} \cdot G_E = 0 \qquad (r=5,\ 6)$$
$$v_1^{F*} \cdot G_F = gZ_{276}$$
$$v_2^{F*} \cdot G_F = gZ_{277}$$
$$v_3^{F*} \cdot G_F = gZ_{278}$$
$$v_4^{F*} \cdot G_F = gZ_{279}$$
$$v_5^{F*} \cdot G_F = gZ_{280}$$
$$v_6^{F*} \cdot G_F = 0$$

As this point, all the necessary ingredients for equation (2.55) are at hand. Setting the subscript r in the equation equal to 1, 2, ..., 6 respectively, one obtains six equations. Substitution of each group of corresponding ingredients into the equations results in the following expressions for the generalized active forces

$$\begin{aligned} k_1 &= Z_4\tau_1 - s_3\tau_2 - Z_8\tau_3 + Z_{281} \\ k_2 &= Z_5\tau_1 - c_3\tau_2 - Z_9\tau_3 + Z_{282} \\ k_3 &= \tau_3 + Z_{283} \\ k_4 &= \tau_4 + Z_{284} \\ k_5 &= \tau_5 + Z_{285} \\ k_6 &= \tau_6 \end{aligned} \tag{2.56}$$

where Z_{281}, $Z_{282}, \ldots,$ Z_{285} are as defined in Appendix 1.

Now, one is in the position to write the equations of motion for the Intelledex 605 Robot Arm by employing Kane's dynamical equations

$$k_r^* + k_r = 0 \qquad (r=1,\ \ldots,\ 6) \tag{2.57}$$

Substitution of equations (2.52) and (2.56) into the above equations with r=1, 2, ..., 6 gives

$$\begin{aligned}
\sum_{s=1}^{6} X_{1s}\dot{u}_s &= -Z_4\tau_1+s_3\tau_2+Z_8\tau_3-Z_{281}-Z_{243} \\
\sum_{s=1}^{6} X_{2s}\dot{u}_s &= -Z_5\tau_1+c_3\tau_2+Z_9\tau_3-Z_{282}-Z_{244} \\
\sum_{s=1}^{6} X_{3s}\dot{u}_s &= -\tau_3-Z_{283}-Z_{245} \\
\sum_{s=1}^{6} X_{4s}\dot{u}_s &= -\tau_4-Z_{284}-Z_{246} \\
\sum_{s=1}^{6} X_{5s}\dot{u}_s &= -\tau_5-Z_{285}-Z_{247} \\
\sum_{s=1}^{6} X_{6s}\dot{u}_s &= -\tau_6-Z_{241}
\end{aligned} \tag{2.58}$$

These six equations together with equations(2.19) constitute a set of twelve equations with twelve unknowns, which are nonlinear. In matrix form, they can be written as

$$\left[\begin{array}{c|c} X & 0 \\ \hline 0 & U \end{array}\right] \left\{\begin{array}{c} \dot{u} \\ \hline \dot{q} \end{array}\right\} = \left\{\begin{array}{c} \tau \\ \hline u \end{array}\right\} \tag{2.59}$$

The matrix on the left side of this equation is called inertia matrix. The inertia matrix derived by Kane's method is symmetric. This property will be proven in the next section.

To test the validity of the underlying equations, two programs in FORTRAN language are furnished. The first program, called TORQUE (See Appendix 2.), is a straightforward calculation routine. Taking a specified motion as input, this program calculates the torques needed for carrying out the specified motion. In the testing case, the generalized coordinates are designated by the following equations.

$$q_r = \pi\{t-(T/2\pi)\sin(2\pi t/T)\}/3T \qquad (r=1,\ 3,\ \ldots,\ 6)$$

$$q_2 = \pi/2-\pi\{t-(T/2\pi)\sin(2\pi t/T)\}/6T$$

where T is the time span for running the program. these equations (i.e. the input) are plotted in Figure 2.3 and Figure 2.4. The parameters used in connection with the example robot are listed in Table 3. The output of the program TORQUE, the torques needed to carry out the designated motion, are plotted in Figures (2.5-2.10).

Another program ANGLE (Appendix 3) is used to solve the twelve simultaneous, nonlinear differential equations to give the coordinates as functions of time in terms of given driving torques. The core of this program is the subroutine QSOLVE, which is based on the quasi-Newton [47] method. The major expense of this method is to derive an approximation to the Jacobian of the system equations by evaluating the equations of motion at two successive points of the generalized coordinate q. The torques generated by the first program TORQUE are the input and the generalized coordinates are outputs. When the outputs are plotted versus time t, it turns out that the resulting curves are identical to those in Figure 2.3 and Figure 2.4

respectively. This is a good indication that the derived equations of motion are free of errors.

TABLE 3. PARAMETERS CHARACTERIZING THE INTELLEDEX 605 ROBOT

Quantity	Value	Units
m_1	18.1440	kg
m_2	12.2472	kg
m_3	8.6184	kg
m_4	4.9896	kg
m_5	1.3608	kg
m_6	0.9072	kg
L_1	0.2794	m
L_2	0.3226	m
L_{12}	0.1397	m
L_{13}	0.1588	m
L_{22}	0.0127	m
L_{31}	0.1880	m
L_{41}	0.1727	m
$I^1{}_1$	0.1800	kg m^2
$I^1{}_2$	0.0450	kg m^2
$I^1{}_3$	0.1350	kg m^2
$I^2{}_1$	0.0300	kg m^2
$I^2{}_2$	0.0300	kg m^2
$I^2{}_3$	0.0232	kg m^2
$I^3{}_1$	0.1041	kg m^2
$I^3{}_2$	0.1041	kg m^2
$I^3{}_3$	0.0260	kg m^2
$I^4{}_1$	0.0605	kg m^2
$I^4{}_2$	0.0605	kg m^2
$I^4{}_3$	0.0260	kg m^2
$I^5{}_1$	0.0048	kg m^2
$I^5{}_2$	0.0048	kg m^2
$I^5{}_3$	0.0035	kg m^2
$I^6{}_1$	0.0040	kg m^2
$I^6{}_2$	0.0040	kg m^2
$I^6{}_3$	0.0030	kg m^2

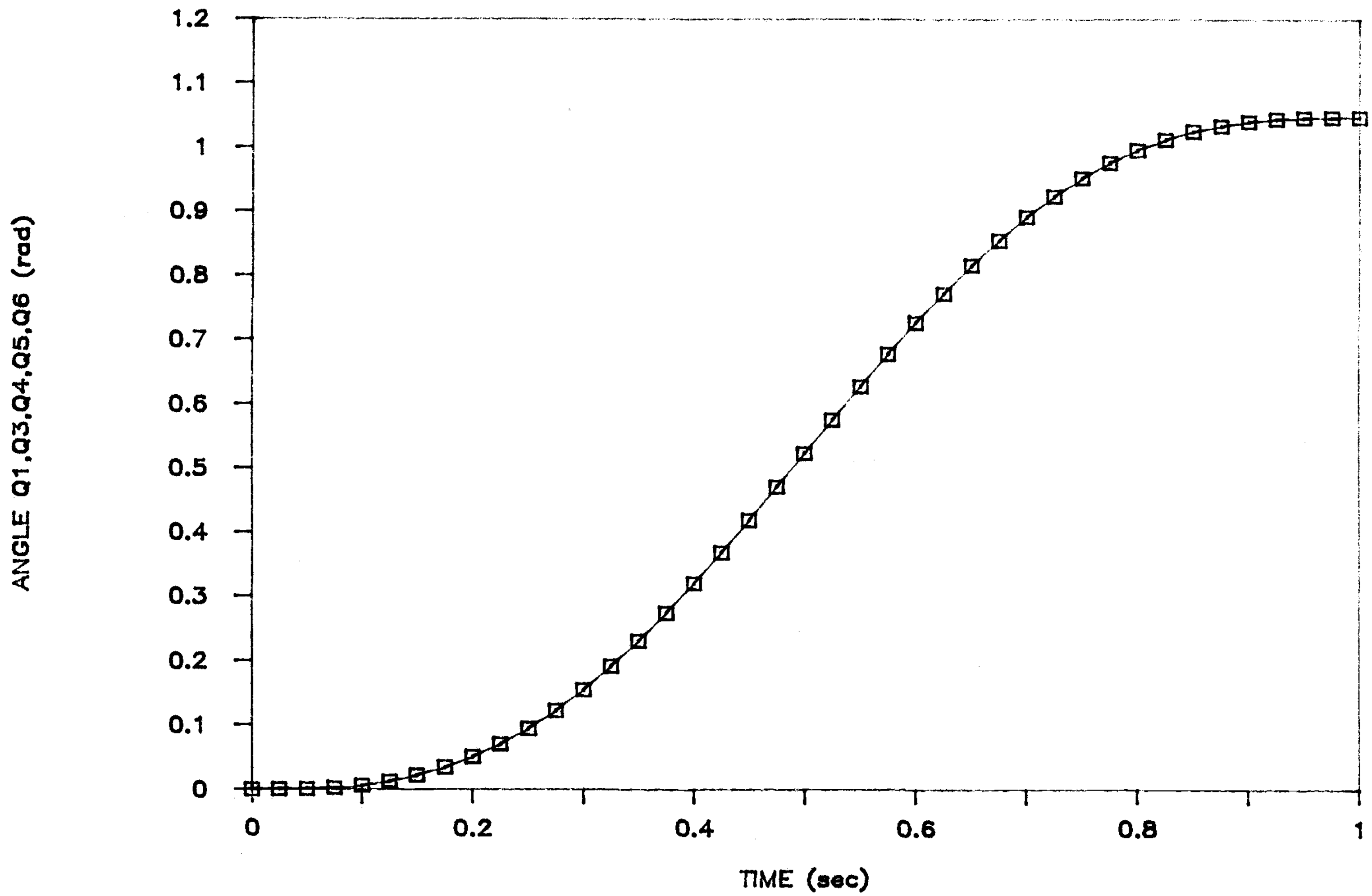

Figure 2.3. Coordinates q_1, q_3, q_4, q_5, q_6 vs time (the specified motion).

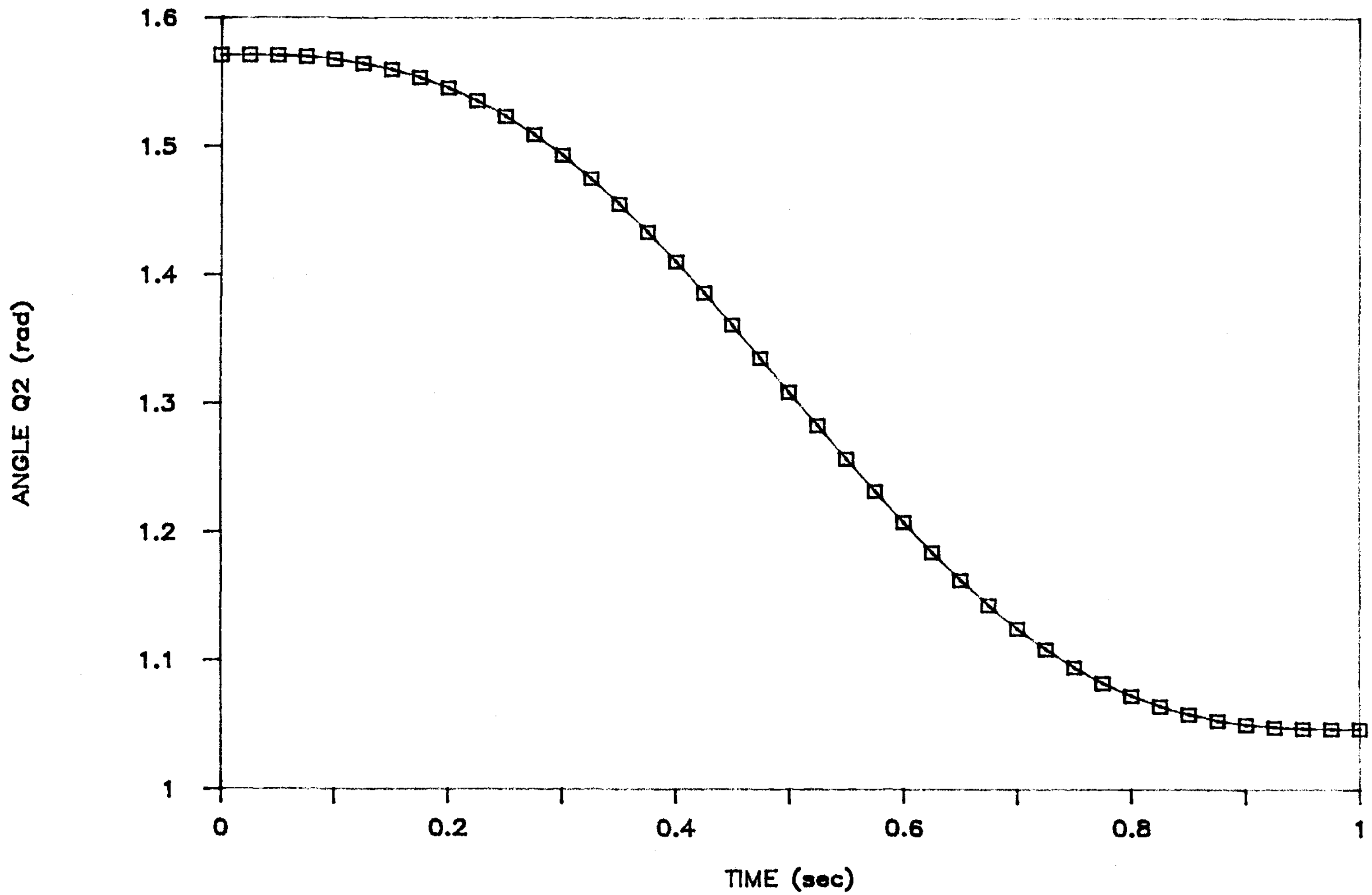

Figure 2.4. Coordinate q_2 vs time (the specified motion).

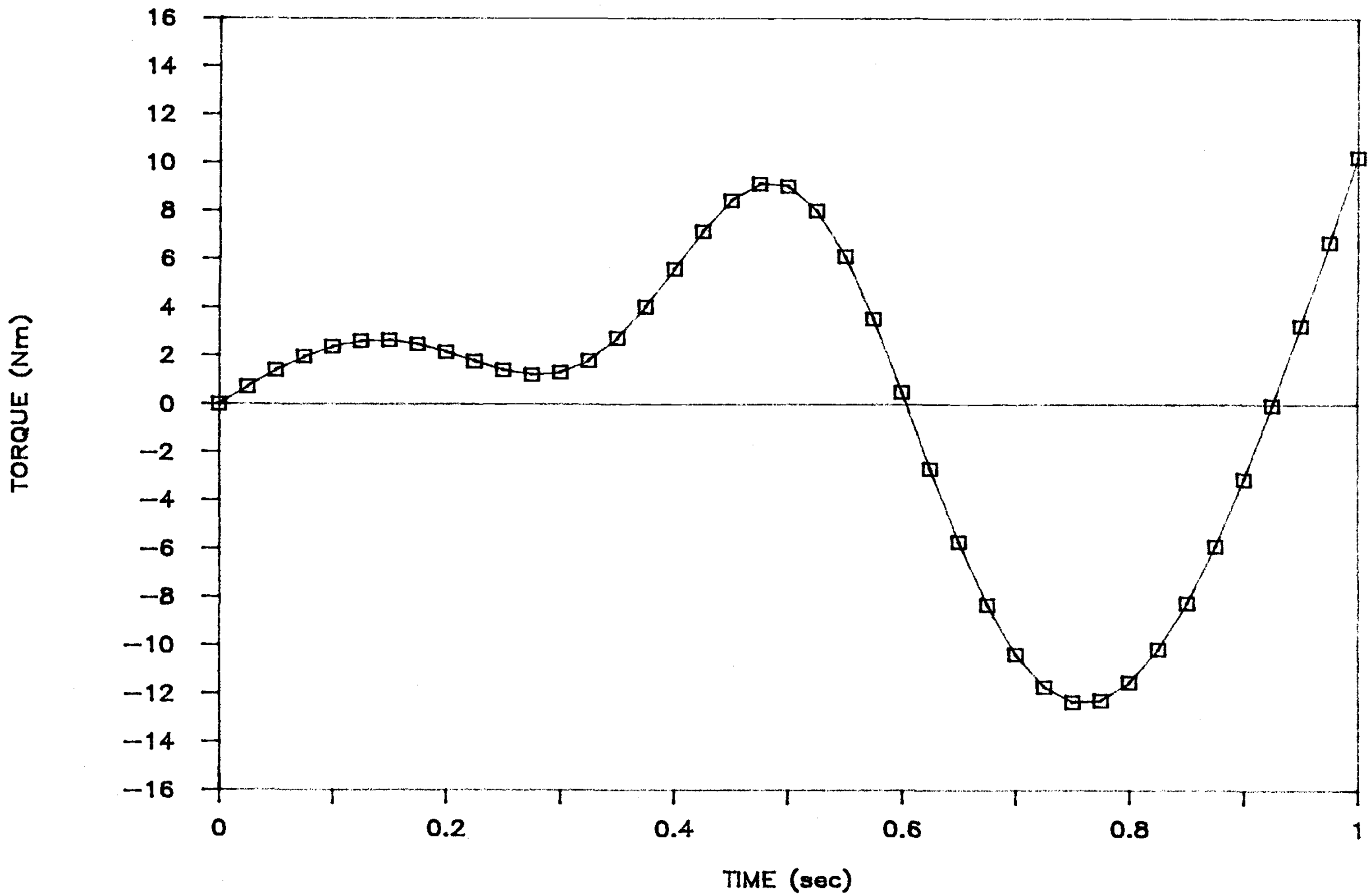

Figure 2.5. Torque τ_1 vs time (output of TORQUE).

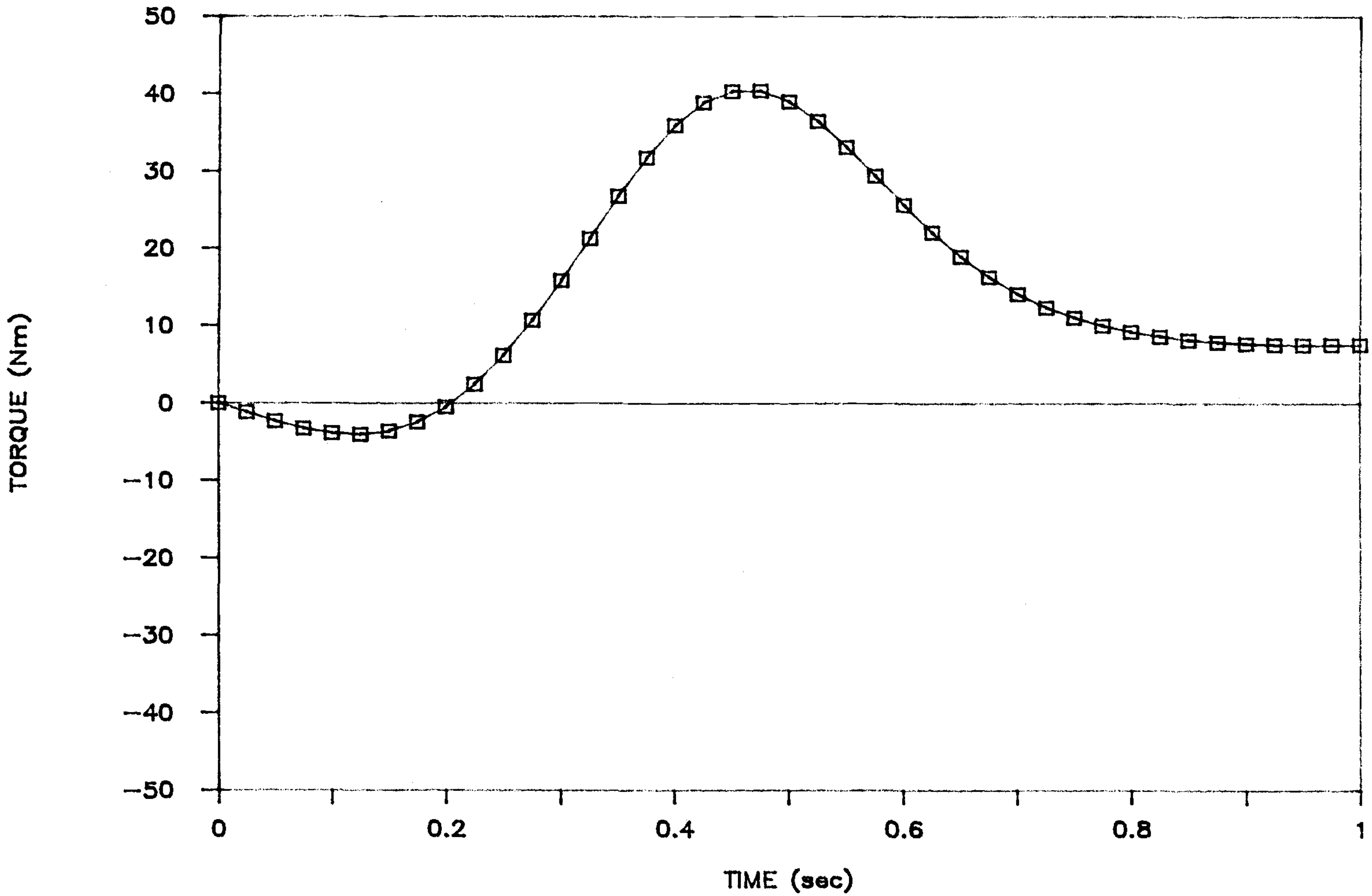

Figure 2.6. Torque τ_2 vs time (output of TORQUE).

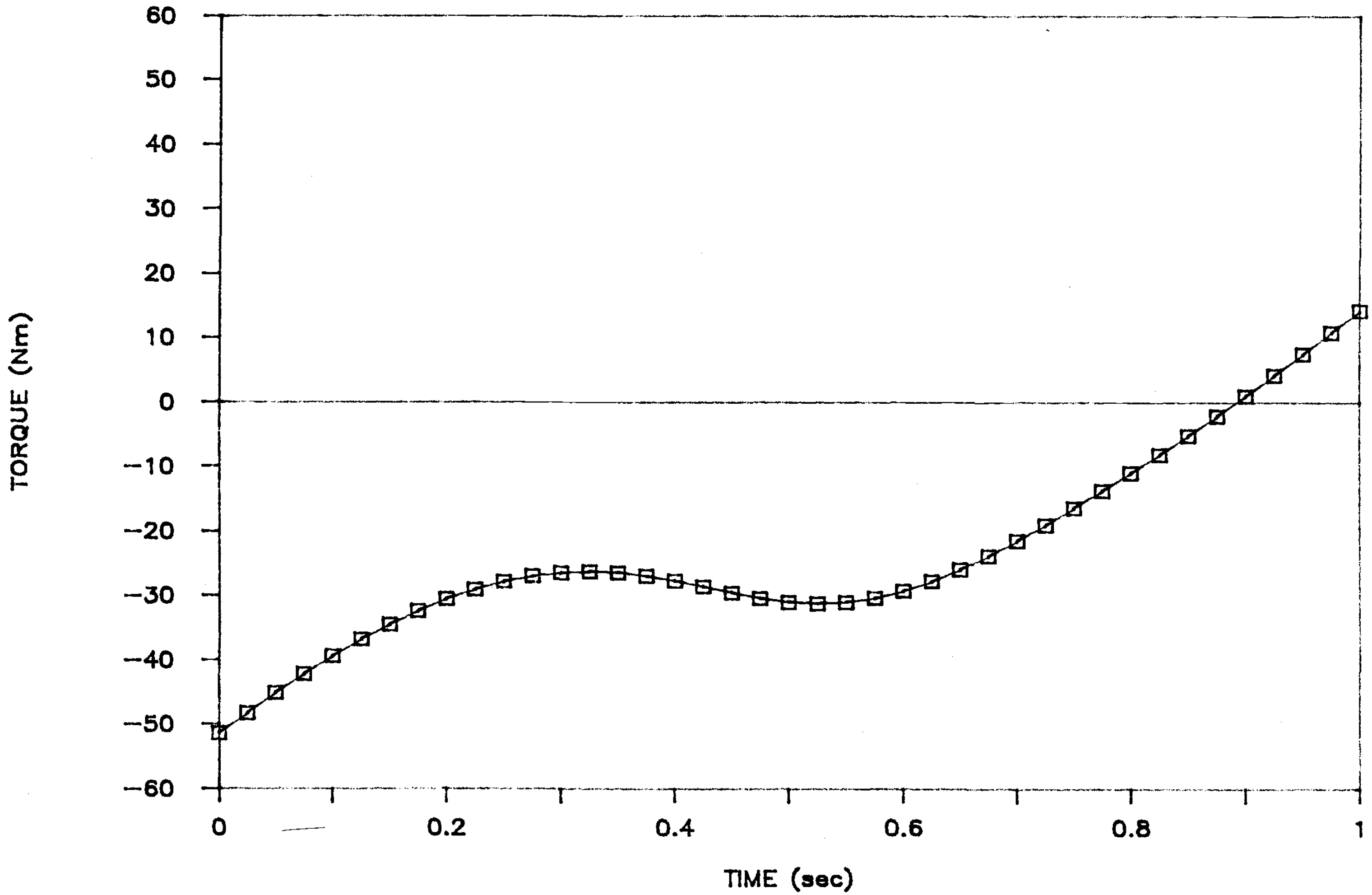

Figure 2.7. Torque τ_3 vs time (output of TORQUE).

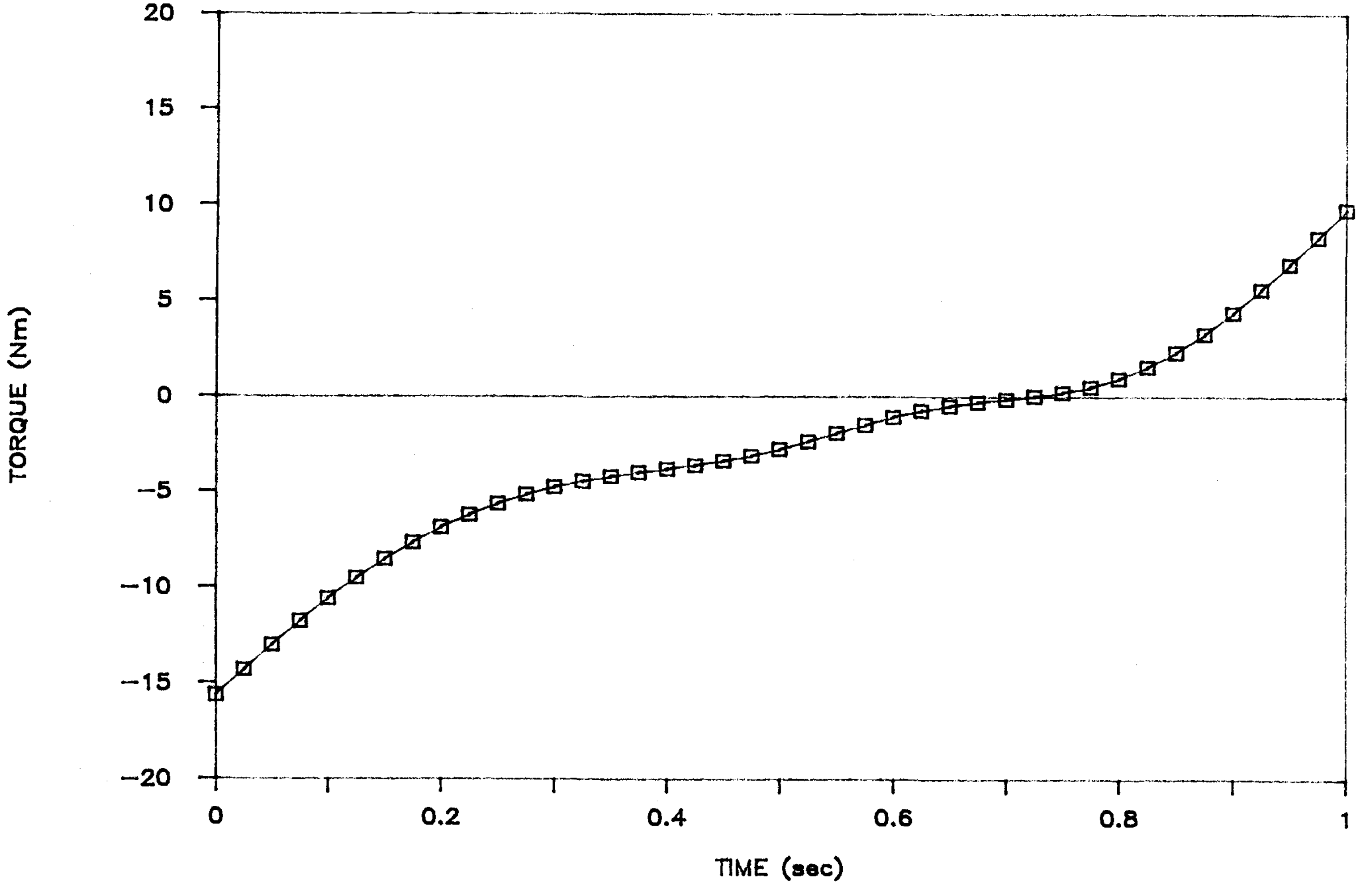

Figure 2.8. Torque τ_4 vs time (output of TORQUE).

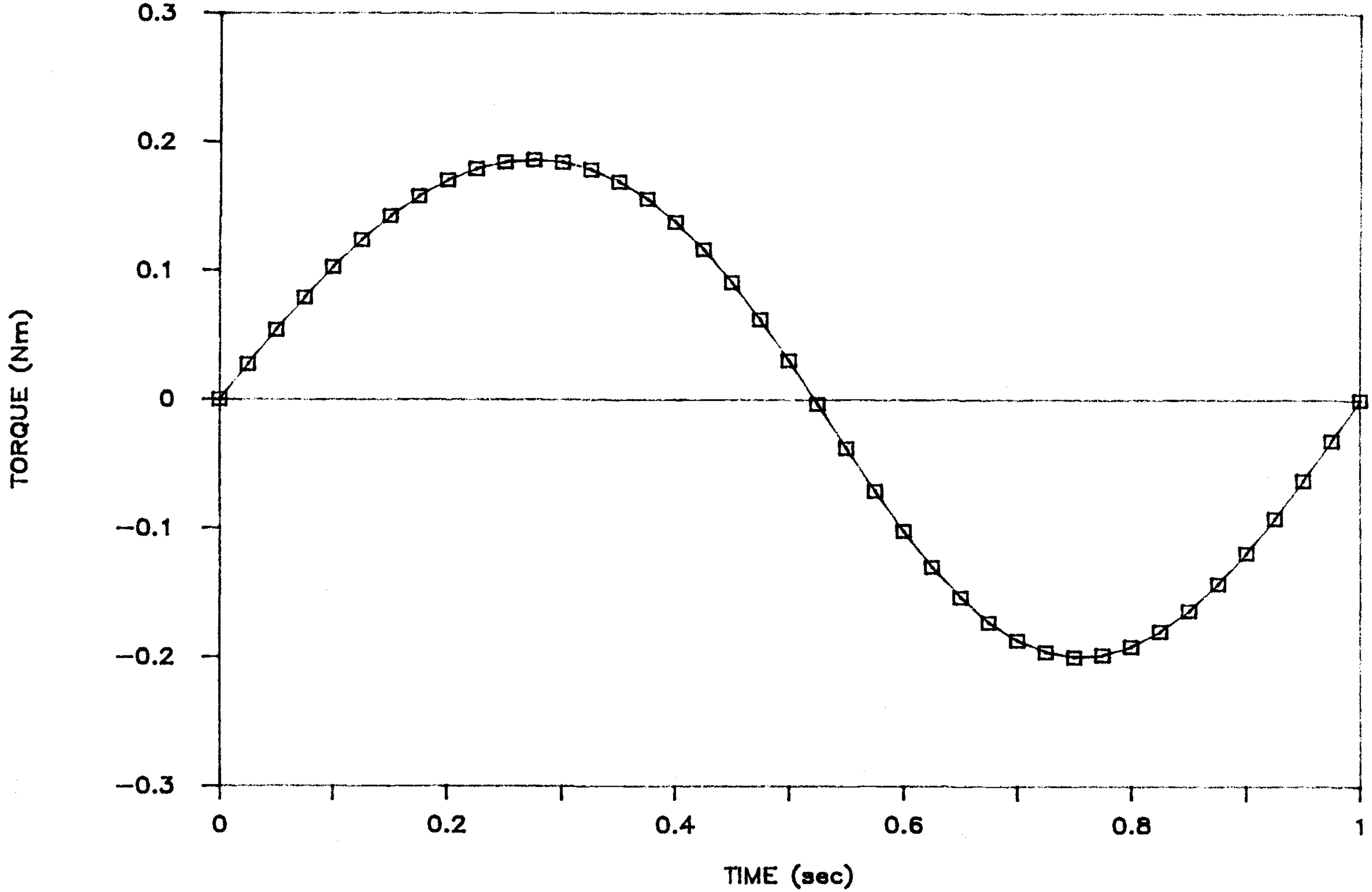

Figure 2.9. Torque τ_5 vs time (output of TORQUE).

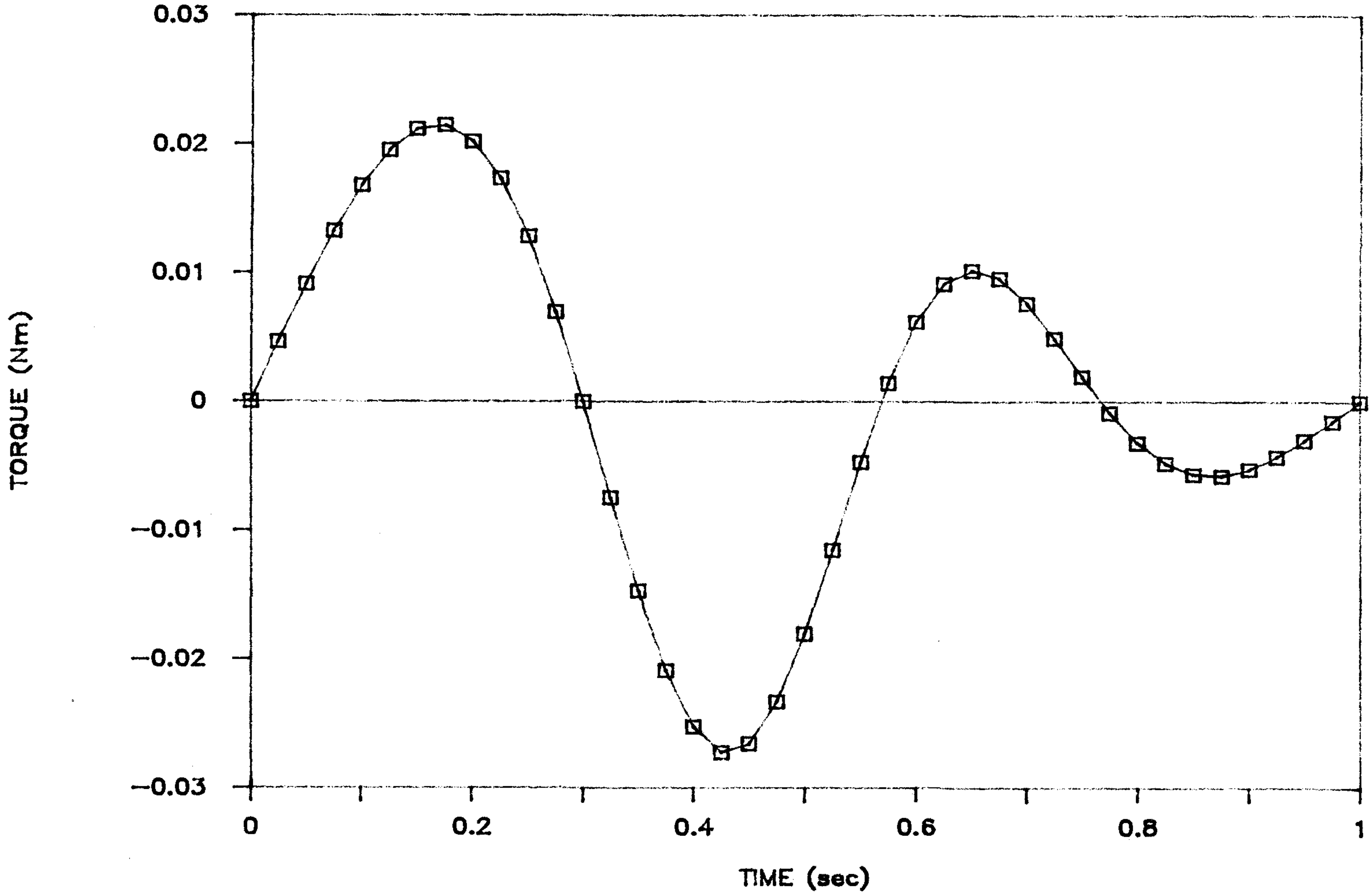

Figure 2.10. Torque τ_6 vs time (output of TORQUE).

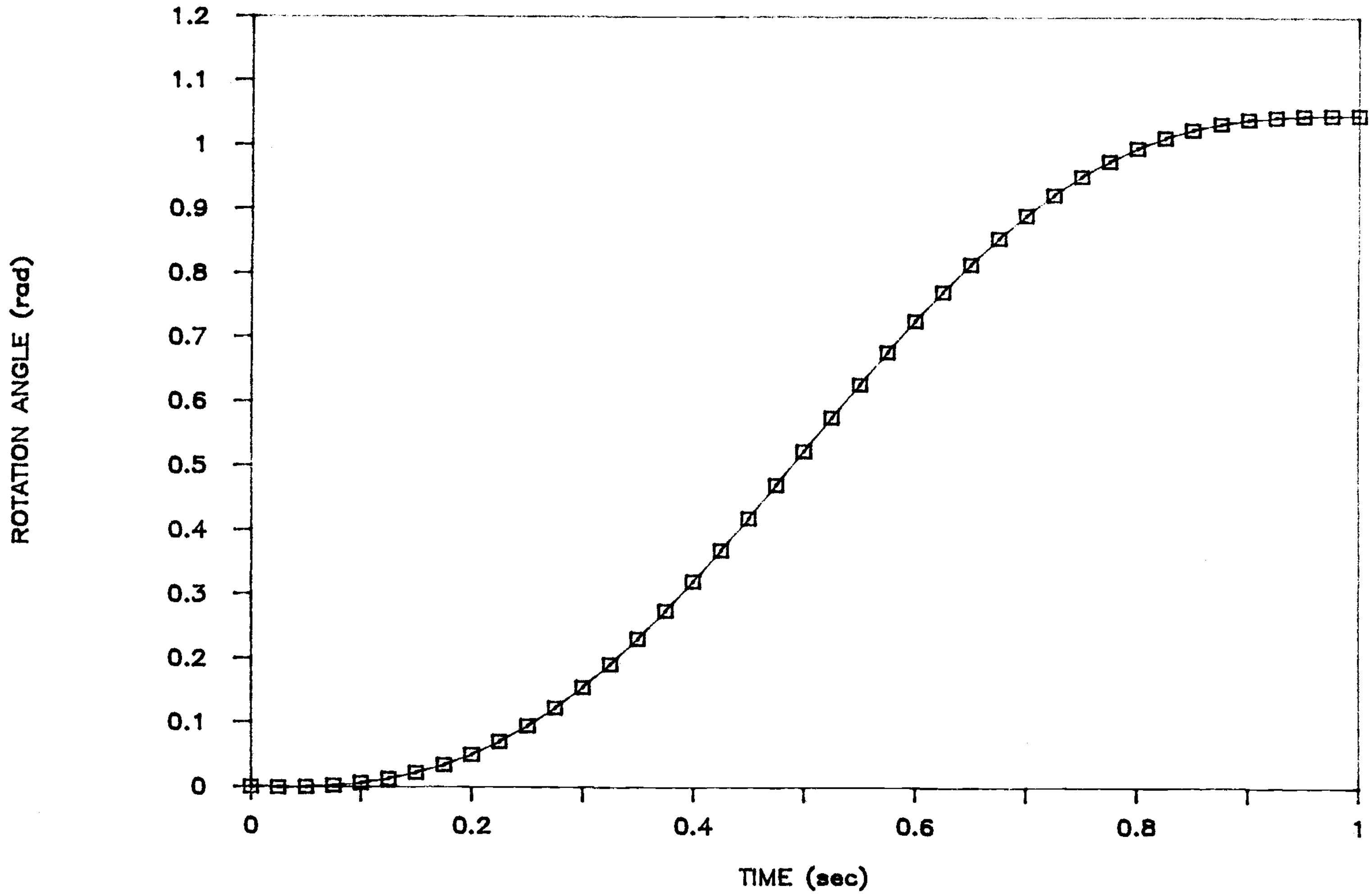

Figure 2.11. Coordinate q_1 vs time (output of ANGLE).

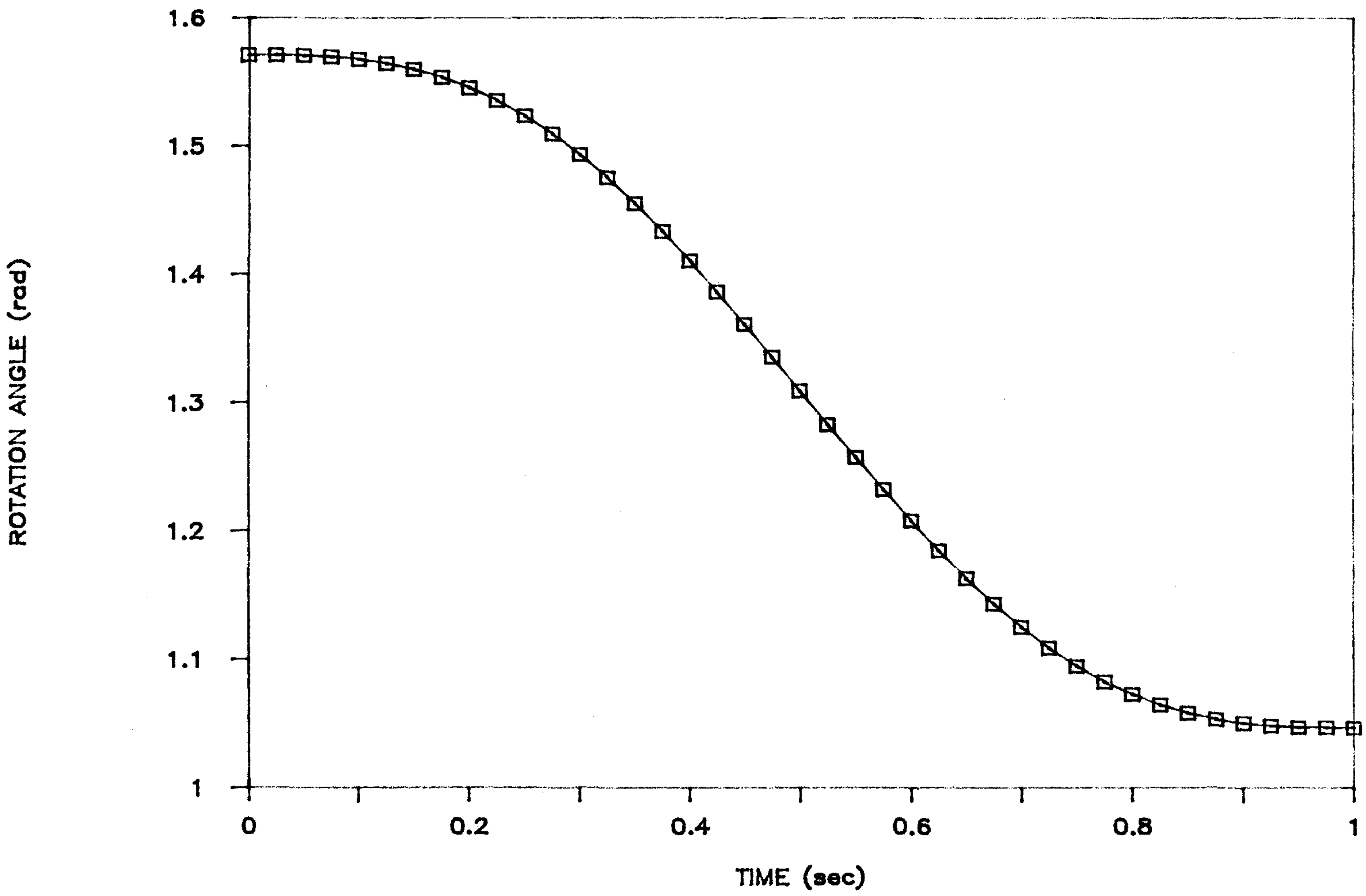

Figure 2.12. Coordinate q_2 vs time (output of ANGLE).

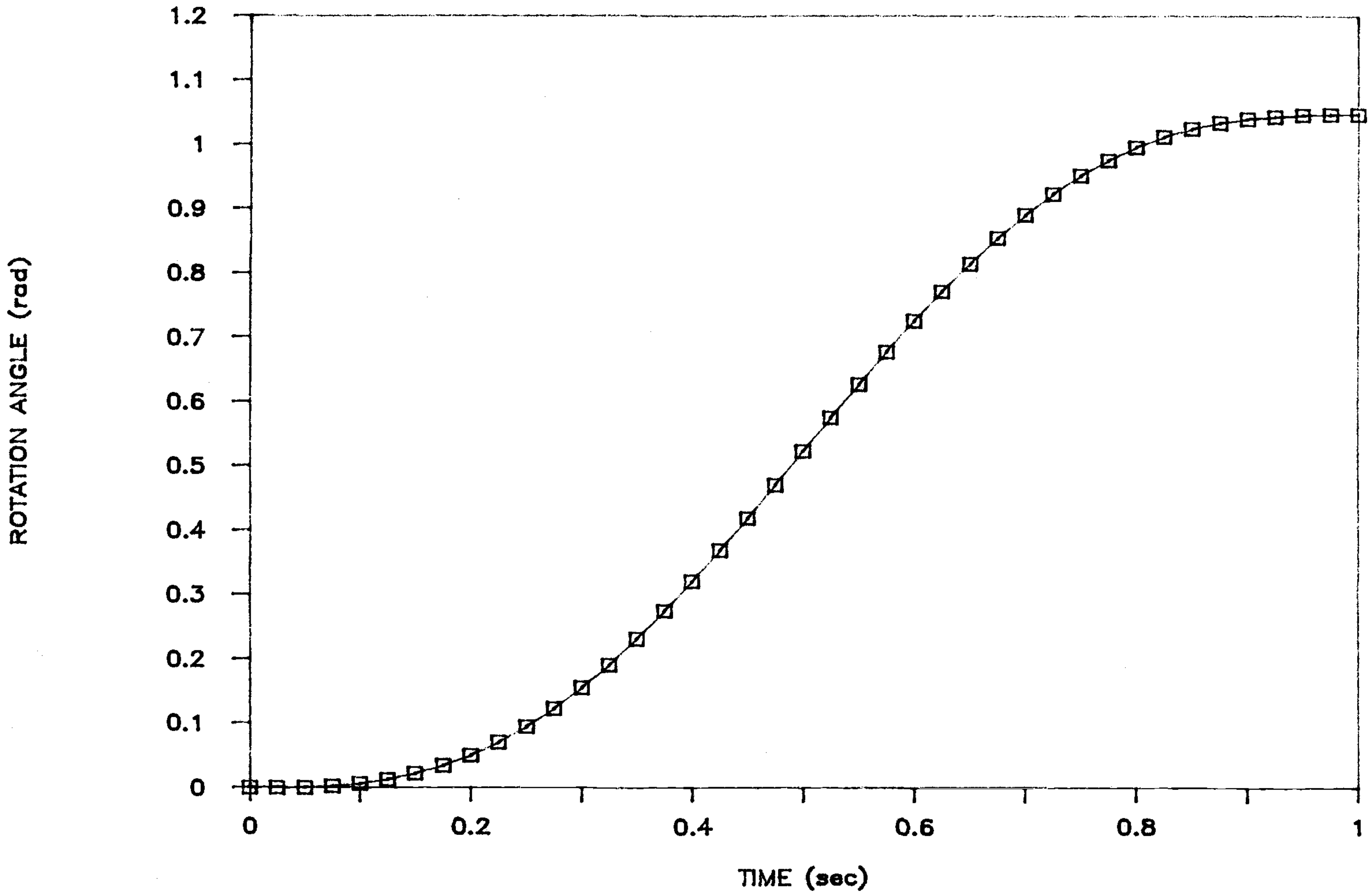

Figure 2.13. Coordinate q_3 vs time (output of ANGLE).

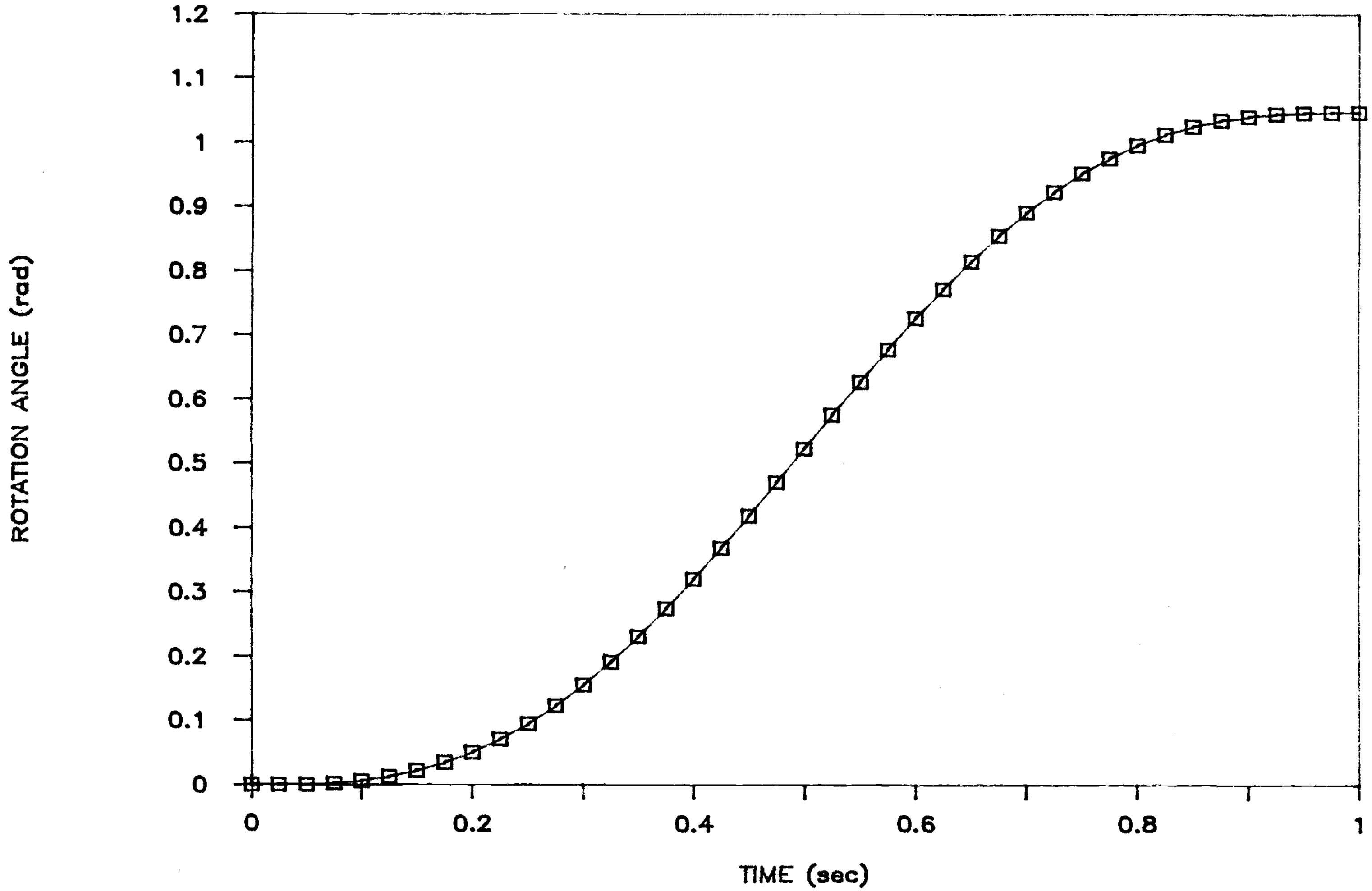

Figure 2.14. Coordinate q_4 vs time (output of ANGLE).

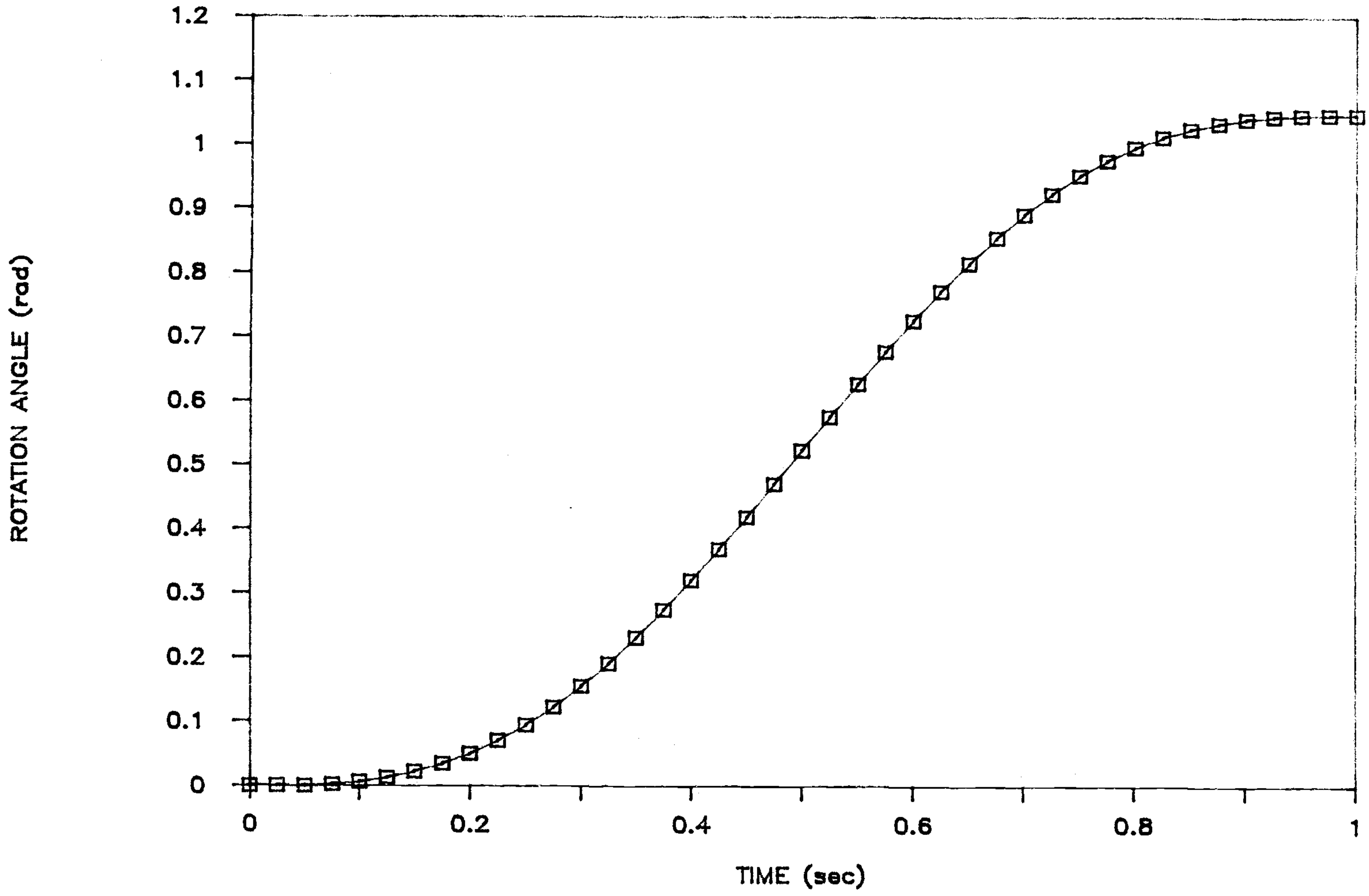

Figure 2.15. Coordinate q_5 vs time (output of ANGLE).

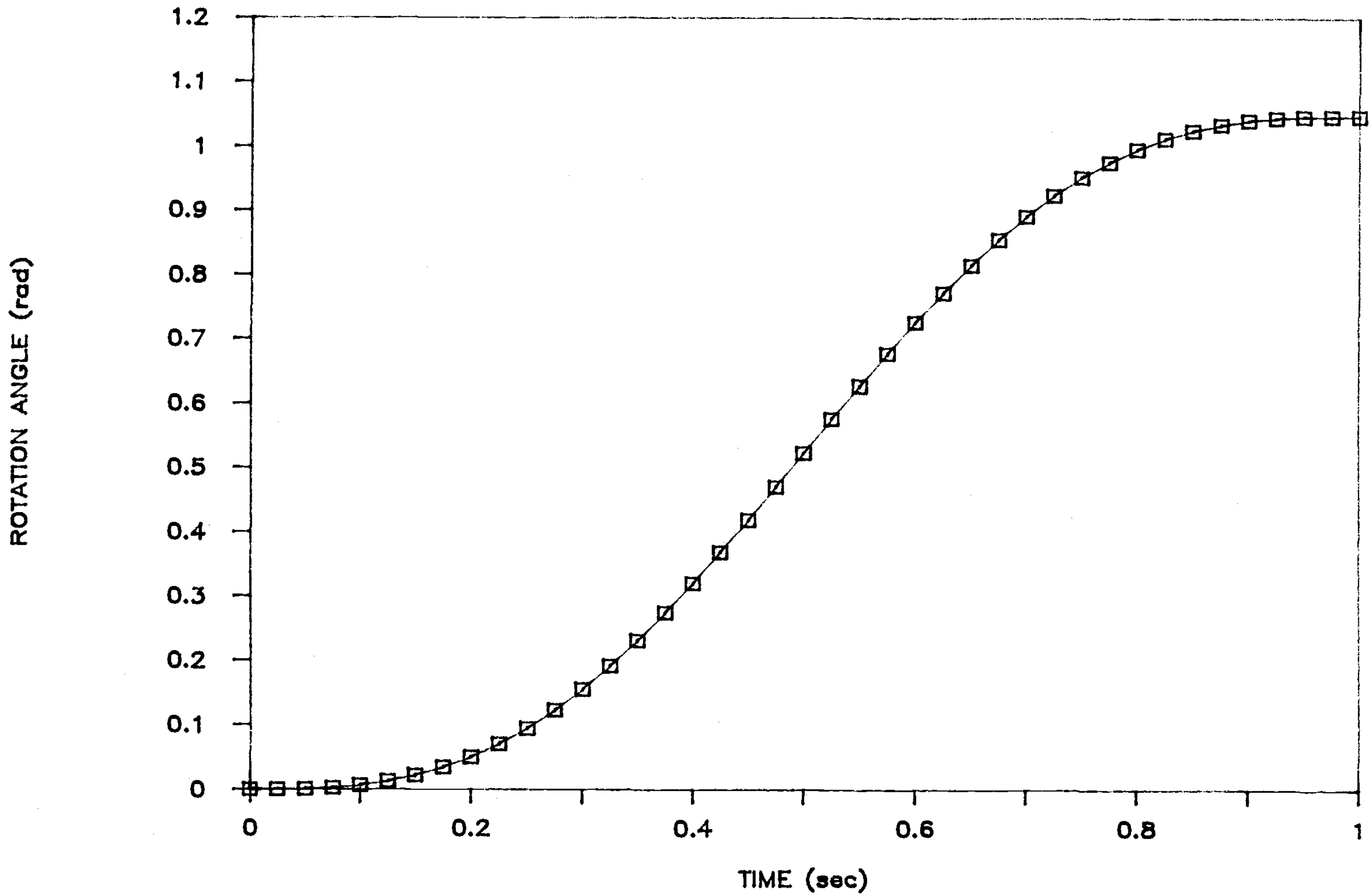

Figure 2.16. Coordinate q_6 vs time (output of ANGLE).

III. AUTOMATIC GENERATION

III.1 GENERAL

It can be seen from the derivation procedure demonstrated in the previous chapter that, although the labor required to formulate the dynamic equations by Kane's method is significantly reduced from those based on classical methods, it is still very burdensome when a system consists of more than a few rigid bodies. For a six-link manipulator, the derivation is difficult, time-consuming, and error-prone; in fact, it may take even longer to locate and remove errors in the derived equations than to derive them.

As mentioned in Chapter I, the best way to avoid such problems is to resort to computers to simultaneously generate and integrate the equations of motion. This is called automatic generation. There are, at present time, two research directions in the field of automatic generation. One is called computer symbolic manipulation; another is recursive computation. For the first category, Rosenthal and Sherman [48] reported a detailed computer code manipulation procedure by using Kane's formulation. The idea used there is to divide the generation process into two stages. First, the symbol-manipulation code is used along with a general-purpose multibody program to create a special-purpose simulation code for a particular configuration. These simplified equations are then converted into a FORTRAN subroutine. In the second stage, this subroutine is incorporated with a main simulation program. The major advantage of this approach is that the form of the equations of motion obtained by this method are nearly the same as those derived by hand. However, the discontinuity in the process will limit the efficiency of the simulation. Each time the system configuration is changed, one must begin with generating symbolic equations and complete the interfacing of these equations with control input and output files.

The algorithm being developed in this chapter belongs to the second category, the recursive computation. Based on Kane's formulation, this algorithm enables one to bypass all the manual derivations of the ingredients to formulate the equations. It requires that one input only structural data (describing component inertia properties, interconnections, and so forth) and either driving forces or a specified motion. Then, depending on which of the latter is specified, the program generates the variation with time of configuration and speed, or of required driving forces.

Kane's Method allows one to form scalar expressions of kinematic ingredients at a very early stage in the formulation process. Instead of expressing the kinematic and kinetic ingredients in vector forms, this paper expresses all these ingredients into algebraic expressions that only contain additions and multiplications of scalar elements. This avoids repeatedly calling of subroutines to calculate dot products and cross products, thus saving computation time.

The remainder of this chapter includes two sections. Section III.2 is comprised of the derivations of the algorithm. Section III.3 outlines two systematic procedures for using the algorithm and discusses its validity by giving the simulation results for the Intelledex 605 robot.

III.2 FORMULATION

Refer to Figure 3.1. To specify a general configuration of a mechanical manipulator, a coordinate frame is assigned to each link. In doing so, Paul's recommendation [46] is employed in the present work, with additional emphasis on the following points. First, for the base coordinate frame, one of the three unit vectors must be aligned with gravitational forces. For revolute joints, in which the rotation angle q_k is defined as a generalized coordinate, the kth set of three mutually perpendicular unit vectors ($\mathbf{n}_1^k$, $\mathbf{n}_2^k$, $\mathbf{n}_3^k$) is fixed on the link k, and the axis of the rotation is aligned with the unit vector $\mathbf{n}_3^{k-1}$. The origin of the coordinate frame k is set to be at the intersection of the common normal between the axis of link k-1 and link k and the axis of joint k. In the case of intersecting joint axes, the origin is set at the point of intersection of the joint axes. In the case of a prismatic joint, the distance q_k the link moves from its origin is

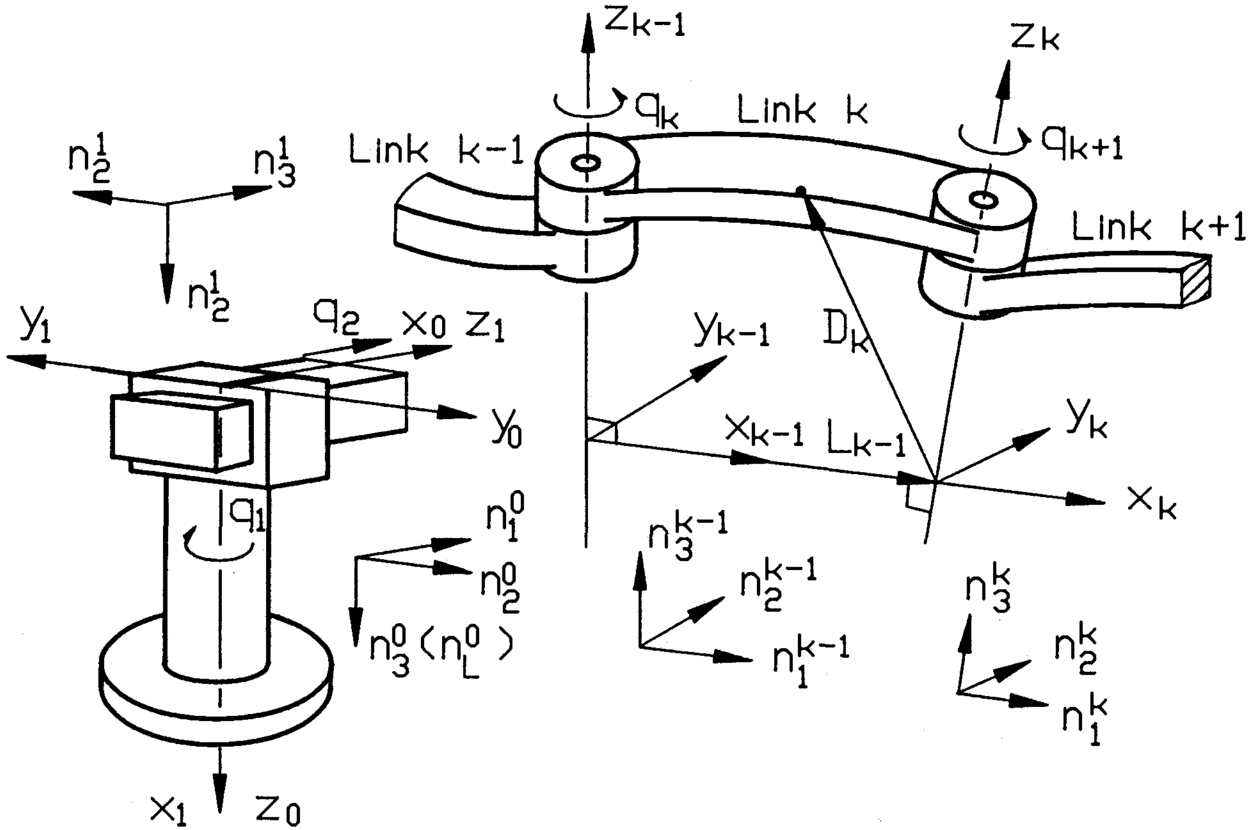

Figure 3.1. Numbering of coordinate frames.

defined as the generalized coordinate. The direction of q_k must be aligned with the unit vector $\mathbf{n}_3{}^{k-1}$.

Having assigned coordinate frames to all links, one can establish transformation matrices. These transformation matrices are 3X3 matrices and are historically called A matrices. The matrix A^k specifies the orientation of link k with respect to link k-1. That is, the elements of the matrix

$$A^k = \begin{bmatrix} A^k{}_{11} & A^k{}_{12} & A^k{}_{13} \\ A^k{}_{21} & A^k{}_{22} & A^k{}_{23} \\ A^k{}_{31} & A^k{}_{32} & A^k{}_{33} \end{bmatrix}$$

are the direction cosines between the unit vectors attached to the two links:

$$A^k{}_{ij} = \mathbf{n}^k{}_i \cdot \mathbf{n}^{k-1}{}_j$$

The interested reader may refer to [46] for greater detail.

Next, $\mathbf{D}^K$ is defined as the vector from the origin of the kth coordinate frame to the mass center of link k, and $\mathbf{L}^K$ as the vector from the same origin to the next coordinate origin. These may be expressed as

$$\mathbf{D}^K = d^k{}_1\mathbf{n}_1 + d^k{}_2\mathbf{n}_2 + d^k{}_3\mathbf{n}_3 \qquad (3.1)$$

$$\mathbf{L}^K = L^k{}_1\mathbf{n}_1 + L^k{}_2\mathbf{n}_2 + L^k{}_3\mathbf{n}_3 \qquad (3.2)$$

In this formulation, the generalized speeds are simply defined as

$$u_k = \dot{q}_k \qquad \text{(for } k=1,\ldots,N) \qquad (3.3)$$

where N is the number of links that constitute the mechanism. The reason for defining the generalized speeds in this way will be seen later.

Based on the scheme just described, a set of expressions of all kinematic and kinetic ingredients is to be constructed. First, to derive a general expression for angular velocities, consider the adjacent links k-1 and k. The angular velocity of link k can be written as

$$\mathbf{w}^k = \mathbf{w}^{k-1} + \dot{q}_k \mathbf{n}_3{}^{k-1} \tag{3.4}$$

or

$$\mathbf{w}^k = \mathbf{w}^{k-1} + u_k \mathbf{n}_3{}^{k-1}$$
$$= W_1{}^{k-1}\mathbf{n}_1{}^{k-1} + W_2{}^{k-1}\mathbf{n}_2{}^{k-1} + (W_3{}^{k-1}+u_k)\,\mathbf{n}_3{}^{k-1}$$

In the kth coordinate frame, this can be expressed as

$$\mathbf{w}^k = (\sum_{i=1}^{3} W_i{}^{k-1}A^k{}_{i1}+u_k A^k{}_{31})\,\mathbf{n}_1{}^k$$
$$+ (\sum_{i=1}^{3} W_i{}^{k-1}A^k{}_{i2}+u_k A^k{}_{32})\,\mathbf{n}_2{}^k$$
$$+ (\sum_{i=1}^{3} W_i{}^{k-1}A^k{}_{i3}+u_k A^k{}_{33})\,\mathbf{n}_3{}^k$$

or

$$\mathbf{w}^k = \sum_{j=1}^{3}(\sum_{i=1}^{3} W_i{}^{k-1}A_{ij}{}^k+u_k A_{3j}{}^k)\,\mathbf{n}_j{}^k \tag{3.5}$$

On the other hand, since

$$\mathbf{w}^k = \sum_{j=1}^{3} W_j{}^k \mathbf{n}_j{}^k, \tag{3.6}$$

the component of $\mathbf{w}^k$ in the direction of $\mathbf{n}_j{}^k$ is

$$W_j{}^k = \sum_{i=1}^{3} W_i{}^{k-1}A_{ij}{}^k+u_k A_{3j}{}^k \quad (j=1,2,3.\ k=1,\ldots,N) \tag{3.7}$$

Equation (3.7) is true for all links, including the first rotational link. To verify this, observe first that the angular velocities of all links up to the first rotational link are zero, so that if k designates the first rotational link,

$$W_j^{k-1} = 0 \qquad (j=1,\ 2,\ 3)$$

In this case, equation (3.7) reduces to

$$W_j^{k} = u_k A^k{}_{3j} \qquad (j=1,\ 2,\ 3) \qquad (3.8)$$

On the other hand, the angular velocity of the first rotational link can be expressed as

$$\begin{aligned} \mathbf{w}^k &= \dot{q}_k \mathbf{n}_3{}^{k-1} \\ &= u_k \mathbf{n}_3{}^{k-1} \qquad (j=1,2,3) \end{aligned}$$

This can be written in terms of the unit vectors of the kth coordinate frame as

$$\mathbf{w}^k = u_k (A^k{}_{31}\mathbf{n}_1{}^k + A^k{}_{32}\mathbf{n}_2{}^k + A^k{}_{33}\mathbf{n}_3{}^k)$$

which implies that

$$W_j^{k} = u_k A^k{}_{3j}$$

which is exactly the same as equation (3.8).

If the kth link is translational, its angular velocity is equal to that of the preceding link, i.e.

$$\begin{aligned} \mathbf{w}^k &= \mathbf{w}^{k-1} \\ &= \sum_{j=1}^{3} W_j^{k-1} \mathbf{n}_j{}^{k-1} \end{aligned}$$

In the kth coordinate frame, this can be expressed in the form

$$\mathbf{w}^k = \sum_{j=1}^{3}\sum_{i=1}^{3} W_i^{k-1} A_{ij}{}^k \mathbf{n}_j{}^k$$

So that,

$$W_j^{k} = \sum_{i=1}^{3} W_i^{k-1} A_{ij}{}^k \qquad (j=1,\ 2,\ 3) \qquad (3.9)$$

On the other hand, if the kth link has no prescribed motion, W_j^{k-1} can be expressed in the form

$$W_j^{k-1} = \sum_{r=1}^{k-1} w_{jr}^{k-1} u_r \qquad (j=1,\ 2,\ 3) \tag{3.10}$$

where w_{jr}^{k-1} is the jth component of the rth partial angular velocity of link k-1. Substitutions of equation (3.10) into equations (3.7) and (3.9) give

$$W_j^k = \begin{cases} \sum_{i=1}^{3} \sum_{r=1}^{k-1} w_{ir}^{k-1} A_{ij}^k u_r + A_{3j}^k u_k & \text{(k is rotational)} \\ \sum_{i=1}^{3} \sum_{r=1}^{k-1} w_{ir}^{k-1} A_{ij}^k u_r & \text{(k is translational)} \end{cases} \tag{3.11}$$

Taking the partial derivative with respect to u_r in the above equation, one is left with the recursive expressions for the jth component of the rth partial angular velocity for link k

$$w_{jr}^k = \begin{cases} \sum_{i=1}^{3} w_{ir}^{k-1} A_{ij}^k & \text{(for r=1,...,k-1)} \\ A_{3j}^k & \text{(for r=k)} \\ 0 & \text{(for r>k)} \end{cases} \tag{3.12}$$

Then, $\dot{w}_{jr}^k$, the time derivative of w_{jr}^k can be found by simply taking the time derivative of both sides of the above equations. That is,

$$\dot{w}_{jr}^k = \begin{cases} \sum_{i=1}^{3} (\dot{w}_{ir}^{k-1} A_{ij}^k + w_{ir}^{k-1} \dot{A}_{ij}^k) & \text{(for r=1,...,k-1)} \\ \dot{A}_{3j}^k & \text{(for r=k)} \\ 0 & \text{(for r>k)} \end{cases} \tag{3.13}$$

For rotational links, the velocity $\mathbf{V}^k$ of the kth mass center and the velocity $\mathbf{S}^{k+1}$ of the k+1 th coordinate origin are related by

$$\mathbf{v}^k = \mathbf{s}^k + \mathbf{w}^k \mathrm{x} \mathbf{D}^k \tag{3.14}$$

$$\mathbf{s}^{k+1} = \mathbf{s}^k + \mathbf{w}^k \mathrm{x} \mathbf{L}^k \tag{3.15}$$

where $\mathbf{D}^k$ and $\mathbf{L}^k$ are as defined previously. $\mathbf{s}^k$, the velocity of the kth coordinate origin, may be expressed in the form

$$\mathbf{s}^k = S_1{}^k \mathbf{n}_1{}^{k-1} + S_2{}^k \mathbf{n}_2{}^{k-1} S_3{}^k \mathbf{n}_3{}^{k-1} \tag{3.16}$$

Substituting equations (3.1), (3.6) and (3.16) into equation (3.14) and rearranging it, one finds

$$\begin{aligned} \mathbf{v}^k = & \left(\sum_{i=1}^{3} S_i{}^k A^k{}_{i1} + W_2{}^k d_3{}^k - W_3{}^k d_2{}^k\right) \mathbf{n}_1{}^k \\ & + \left(\sum_{i=1}^{3} S_i{}^k A^k{}_{i2} + W_3{}^k d_1{}^k - W_1{}^k d_3{}^k\right) \mathbf{n}_2{}^k \\ & + \left(\sum_{i=1}^{3} S_i{}^k A^k{}_{i3} + W_1{}^k d_2{}^k - W_2{}^k d_1{}^k\right) \mathbf{n}_3{}^k \end{aligned} \tag{3.17}$$

so that the velocity components of the kth mass center in the kth coordinate frame are found to be

$$\left\{ \begin{aligned} V_1{}^k &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i1} + W_2{}^k d_3{}^k - W_3{}^k d_2{}^k \\ V_2{}^k &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i2} + W_3{}^k d_1{}^k - W_1{}^k d_3{}^k \\ V_3{}^k &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i3} + W_1{}^k d_2{}^k - W_2{}^k d_1{}^k \qquad (k=1,\ldots N) \end{aligned} \right. \tag{3.18}$$

Similarly, the velocity components of the k+1 th coordinate origin turn out to be

$$\left\{ \begin{aligned} S_1{}^{k+1} &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i1} + W_2{}^k L_3{}^k - W_3{}^k L_2{}^k \\ S_2{}^{k+1} &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i2} + W_3{}^k L_1{}^k - W_1{}^k L_3{}^k \\ S_3{}^{k+1} &= \sum_{i=1}^{3} S_i{}^k A^k{}_{i3} + W_1{}^k L_2{}^k - W_2{}^k L_1{}^k \end{aligned} \right. \tag{3.19}$$

In the case when the kth link is translational, the relationships that replace (3.14) and (3.15) are

$$\mathbf{v}^k = \mathbf{s}^k + \mathbf{w}^k \times \mathbf{D}^k + u_k \mathbf{n}_3^{k-1} \tag{3.20}$$

$$\mathbf{s}^{k+1} = \mathbf{s}^k + \mathbf{w}^k \times \mathbf{L}^k + u_k \mathbf{n}_3^{k-1} \tag{3.21}$$

Notice that, in this case,

$$\mathbf{D}^k = d_1{}^k \mathbf{n}_1{}^k + d_2{}^k \mathbf{n}_2{}^k + d_3{}^k \mathbf{n}_3{}^k + q_k \mathbf{n}^{k-1} \tag{3.22}$$

$$\mathbf{L}^k = L_1{}^k \mathbf{n}_1{}^k + L_2{}^k \mathbf{n}_2{}^k + L_3{}^k \mathbf{n}_3{}^k + q_k \mathbf{n}^{k-1} \tag{3.23}$$

Substituting these expressions and the other corresponding terms into equation (3.20) and (3.21), respectively, resolving them into the kth coordinate frame and rearranging them, one can find the following relationships that, for the translational link, replace (3.18) and (3.19):

$$\begin{cases} v_1{}^k = \sum_{i=1}^{3} s_i{}^k A^k{}_{i1} + A^k{}_{31} u_k + W_2{}^k (d_3{}^k + A^k{}_{33} q_k) - W_3{}^k (d_2{}^k + A^k{}_{32} q_k) \\ v_2{}^k = \sum_{i=1}^{3} s_i{}^k A^k{}_{i2} + A^k{}_{32} u_k + W_3{}^k (d_1{}^k + A^k{}_{31} q_k) - W_1{}^k (d_3{}^k + A^k{}_{33} q_k) \\ v_3{}^k = \sum_{i=1}^{3} s_i{}^k A^k{}_{i3} + A^k{}_{33} u_k + W_1{}^k (d_2{}^k + A^k{}_{32} q_k) - W_2{}^k (d_1{}^k + A^k{}_{31} q_k) \end{cases}$$

$$(k=1, \ldots, N) \tag{3.24}$$

and

$$\begin{cases} s_1{}^{k+1} = \sum_{i=1}^{3} s_i{}^k A^k{}_{i1} + A^k{}_{31} u_k + W_2{}^k (L_3{}^k + A^k{}_{33} q_k) - W_3{}^k (L_2{}^k + A^k{}_{32} q_k) \\ s_2{}^{k+1} = \sum_{i=1}^{3} s_i{}^k A^k{}_{i2} + A^k{}_{32} u_k + W_3{}^k (L_1{}^k + A^k{}_{31} q_k) - W_1{}^k (L_3{}^k + A^k{}_{33} q_k) \\ s_3{}^{k+1} = \sum_{i=1}^{3} s_i{}^k A^k{}_{i3} + A^k{}_{33} u_k + W_1{}^k (L_2{}^k + A^k{}_{32} q_k) - W_2{}^k (L_1{}^k + A^k{}_{31} q_k) \end{cases}$$

$$(k=0, \ldots, N-1) \tag{3.25}$$

As with the angular velocities, if there is no prescribed motion for the mechanism under consideration, the velocities $\mathbf{v}^k$ and $\mathbf{s}^k$ can be expressed in terms of the partial velocities and the generalized speeds. These are

$$\mathbf{v}^k = \sum_{r=1}^{k} \mathbf{v}_r{}^k u_r$$

$$\mathbf{s}^k = \sum_{r=1}^{k-1} \mathbf{s}_r{}^k u_r$$

Alternatively,

$$v_j{}^k = \sum_{r=1}^{k} v_{jr}{}^k u_r \qquad (j=1,\ 2,\ 3)$$

$$s_j{}^k = \sum_{r=1}^{k-1} s_{jr}{}^k u_r \qquad (j=1,\ 2,\ 3)$$

where $v_{jr}{}^k$ and $s_{jr}{}^k$ are the jth component of the rth partial velocity of, respectively, the kth mass center and the kth coordinate origin. Substituting the above expressions into equations (3.24), (3.25) respectively and performing the same operations as those for the partial angular velocities, one obtains, for rotational joints:

$$\begin{cases} v_{1r}{}^k = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i1} + w_{2r}{}^k d_3{}^k - w_{3r}{}^k d_2{}^k \\ v_{2r}{}^k = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i2} + w_{3r}{}^k d_1{}^k - w_{1r}{}^k d_3{}^k \\ v_{3r}{}^k = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i3} + w_{1r}{}^k d_2{}^k - w_{2r}{}^k d_1{}^k \end{cases} \qquad (r \le k) \qquad (3.26)$$

$$\begin{cases} s_{1r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i1} + w_{2r}{}^k L_3{}^k - w_{3r}{}^k L_2{}^k \\ s_{2r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i2} + w_{3r}{}^k L_1{}^k - w_{1r}{}^k L_3{}^k \\ s_{3r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^k A^k{}_{i3} + w_{1r}{}^k L_2{}^k - w_{2r}{}^k L_1{}^k \end{cases} \qquad (r \le k) \qquad (3.27)$$

and for translational joints:

$$\begin{cases} v_{1r}{}^{k} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i1} + w_{2r}{}^{k}(d_3{}^{k} + A^{k}{}_{33}q_k) - w_{3r}{}^{k}(d_2{}^{k} + A^{k}{}_{32}q_k) \\ v_{2r}{}^{k} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i2} + w_{3r}{}^{k}(d_1{}^{k} + A^{k}{}_{31}q_k) - w_{1r}{}^{k}(d_3{}^{k} + A^{k}{}_{33}q_k) \\ v_{3r}{}^{k} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i3} + w_{1r}{}^{k}(d_2{}^{k} + A^{k}{}_{32}q_k) - w_{2r}{}^{k}(d_1{}^{k} + A^{k}{}_{31}q_k) \\ \qquad\qquad\qquad\qquad\qquad\qquad (\text{for } r<k) \\ v_{jr}{}^{k} = A^{k}{}_{3j} \qquad\qquad (\text{for } r=k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.28}$$

$$\begin{cases} s_{1r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i1} + w_{2r}{}^{k}(L_3{}^{k} + A^{k}{}_{33}q_k) - w_{3r}{}^{k}(L_2{}^{k} + A^{k}{}_{32}q_k) \\ s_{2r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i2} + w_{3r}{}^{k}(L_1{}^{k} + A^{k}{}_{31}q_k) - w_{1r}{}^{k}(L_3{}^{k} + A^{k}{}_{33}q_k) \\ s_{3r}{}^{k+1} = \sum_{i=1}^{3} s_{ir}{}^{k}A^{k}{}_{i3} + w_{1r}{}^{k}(L_2{}^{k} + A^{k}{}_{32}q_k) - w_{2r}{}^{k}(L_1{}^{k} + A^{k}{}_{31}q_k) \\ \qquad\qquad\qquad\qquad\qquad\qquad (\text{for } r<k) \\ s_{jr}{}^{k+1} = A^{k}{}_{3j} \qquad\qquad (\text{for } r=k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.29}$$

Taking the derivative with respect to time of each member of the above equations, and defining

$$P_{jr}{}^{k} = \sum_{i=1}^{3} (\dot{s}_{ir}{}^{k}A_{ij}{}^{k} + s_{ir}{}^{k}\dot{A}_{ij}{}^{k}) \tag{3.30}$$

one is left with the following expressions, for rotational joints:

$$\begin{cases} \dot{v}_{1r}{}^{k} = P_{ir}{}^{k} + \dot{w}_{2r}{}^{k}d_3{}^{k} - \dot{w}_{3r}{}^{k}d_2{}^{k} \\ \dot{v}_{2r}{}^{k} = P_{ir}{}^{k} + \dot{w}_{3r}{}^{k}d_1{}^{k} - \dot{w}_{1r}{}^{k}d_3{}^{k} \qquad (r\leq k) \\ \dot{v}_{3r}{}^{k} = P_{ir}{}^{k} + \dot{w}_{1r}{}^{k}d_2{}^{k} - \dot{w}_{2r}{}^{k}d_1{}^{k} \\ \dot{v}_{jr}{}^{k} = 0 \qquad\qquad (r>k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.31}$$

$$\begin{cases} \dot{s}_{1r}{}^{k+1} = P_{ir}{}^{k} + \dot{w}_{2r}{}^{k}L_3{}^{k} - \dot{w}_{3r}{}^{k}L_2{}^{k} \\ \dot{s}_{2r}{}^{k+1} = P_{ir}{}^{k} + \dot{w}_{3r}{}^{k}L_1{}^{k} - \dot{w}_{1r}{}^{k}L_3{}^{k} \qquad (r\leq k) \\ \dot{s}_{3r}{}^{k+1} = P_{ir}{}^{k} + \dot{w}_{1r}{}^{k}L_2{}^{k} - \dot{w}_{2r}{}^{k}L_1{}^{k} \\ \dot{s}_{jr}{}^{k+1} = 0 \qquad\qquad (r>k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.32}$$

and for translational joints:

$$\begin{cases} \dot{v}_{1r}{}^{k} = P_{1r}{}^{k}+\dot{w}_{2r}{}^{k}(d_3{}^{k}+A^{k}{}_{33}q_k)-w_{2r}{}^{k}(\dot{A}^{k}{}_{33}q_2{}^{k}+A^{k}{}_{33}u_k) \\ \qquad -\dot{w}_{3r}{}^{k}(d_2{}^{k}+A^{k}{}_{32}q_k)-w_{3r}{}^{k}(\dot{A}^{k}{}_{32}q_k+A^{k}{}_{32}u_k) \\ \dot{v}_{2r}{}^{k} = P_{2r}{}^{k}+\dot{w}_{3r}{}^{k}(d_1{}^{k}+A^{k}{}_{31}q_k)-w_{3r}{}^{k}(\dot{A}^{k}{}_{31}q_2{}^{k}+A^{k}{}_{31}u_k) \\ \qquad -\dot{w}_{1r}{}^{k}(d_3{}^{k}+A^{k}{}_{33}q_k)-w_{1r}{}^{k}(\dot{A}^{k}{}_{33}q_k+A^{k}{}_{33}u_k) \qquad (r<k) \\ \dot{v}_{3r}{}^{k} = P_{3r}{}^{k}+\dot{w}_{1r}{}^{k}(d_2{}^{k}+A^{k}{}_{32}q_k)-w_{1r}{}^{k}(\dot{A}^{k}{}_{32}q_2{}^{k}+A^{k}{}_{32}u_k) \\ \qquad -\dot{w}_{2r}{}^{k}(d_1{}^{k}+A^{k}{}_{31}q_k)-w_{2r}{}^{k}(\dot{A}^{k}{}_{31}q_k+A^{k}{}_{31}u_k) \\ \dot{v}_{jr}{}^{k} = \dot{A}^{k}{}_{3j} \qquad (r=k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.33}$$

$$\begin{cases} \dot{s}_{1r}{}^{k+1} = P_{1r}{}^{k}+\dot{w}_{2r}{}^{k}(L_3{}^{k}+A^{k}{}_{33}q_k)-w_{2r}{}^{k}(\dot{A}^{k}{}_{33}q_2{}^{k}+A^{k}{}_{33}u_k) \\ \qquad -\dot{w}_{3r}{}^{k}(L_2{}^{k}+A^{k}{}_{32}q_k)-w_{3r}{}^{k}(\dot{A}^{k}{}_{32}q_k+A^{k}{}_{32}u_k) \\ \dot{s}_{2r}{}^{k+1} = P_{2r}{}^{k}+\dot{w}_{3r}{}^{k}(L_1{}^{k}+A^{k}{}_{31}q_k)-w_{3r}{}^{k}(\dot{A}^{k}{}_{31}q_2{}^{k}+A^{k}{}_{31}u_k) \\ \qquad -\dot{w}_{1r}{}^{k}(L_3{}^{k}+A^{k}{}_{33}q_k)-w_{1r}{}^{k}(\dot{A}^{k}{}_{33}q_k+A^{k}{}_{33}u_k) \qquad (r<k) \\ \dot{s}_{3r}{}^{k+1} = P_{3r}{}^{k}+\dot{w}_{1r}{}^{k}(L_2{}^{k}+A^{k}{}_{32}q_k)-w_{1r}{}^{k}(\dot{A}^{k}{}_{32}q_2{}^{k}+A^{k}{}_{32}u_k) \\ \qquad -\dot{w}_{2r}{}^{k}(L_1{}^{k}+A^{k}{}_{31}q_k)-w_{2r}{}^{k}(\dot{A}^{k}{}_{31}q_k+A^{k}{}_{31}u_k) \\ \dot{s}_{jr}{}^{k} = \dot{A}^{k}{}_{3j} \qquad (r=k,\ j=1,\ 2,\ 3) \end{cases} \tag{3.34}$$

Next, the expressions for the generalized inertia forces and the generalized active forces also need to be carried out. For the system under consideration, the acceleration of the kth mass center can be obtained by

$$\mathbf{a}^{k} = \frac{{}^{k}d\mathbf{v}^{k}}{dt} + \mathbf{w}^{k} \times \mathbf{v}^{k} \tag{3.35}$$

The first term of the right side of the above equation can be proved to be

$$\frac{{}^{k}d\mathbf{v}^{k}}{dt} = \sum_{j=1}^{3}\left(\sum_{r=1}^{k} v_{jr}{}^{k}\dot{u}_r + \dot{v}_{jr}{}^{k}u_r\right)\mathbf{n}_j{}^{k}$$

To obtain scalar expressions for the second term, define

$$\mathbf{B}^k = \mathbf{w}^k \times \mathbf{v}^k$$

then,

$$B_1{}^k = w_2{}^k v_3{}^k - w_3{}^k v_2{}^k$$
$$B_2{}^k = w_3{}^k v_1{}^k - w_1{}^k v_3{}^k \qquad (3.36)$$
$$B_3{}^k = w_1{}^k v_2{}^k - w_2{}^k v_1{}^k$$

So that the equation (3.35) can be rewritten

$$\mathbf{a}^k = \sum_{j=1}^{3} \left(\sum_{r=1}^{k} v_{jr}{}^k \dot{u}_r + \dot{v}_{jr}{}^k u_r + B_j{}^k \right) \mathbf{n}_j{}^k$$

To further simplify the above expression, define

$$D_j{}^k = \sum_{r=1}^{k} \dot{v}_{jr}{}^k u_r + B_j{}^k \qquad (j=1,\ 2,\ 3) \qquad (3.37)$$

Then the scalar expression of the acceleration for the kth mass center turns out to be

$$\mathbf{a}^k = \sum_{j=1}^{3} \left(\sum_{r=1}^{k} v_{jr}{}^k \dot{u}_r + D_j{}^k \right) \mathbf{n}_j{}^k \qquad (3.38)$$

Now, let m_k be the mass of the kth link; then the inertia force acting on the mass center is

$$\mathbf{R}^{*k} = -m_k \mathbf{a}^k$$

Substitution of the equation (3.38) into the above equation gives

$$\mathbf{R}^{*k} = -\sum_{j=1}^{3} \left(\sum_{r=1}^{k} v_{jr}{}^k \dot{u}_r + D_j{}^k \right) m_k \mathbf{n}_j{}^k \qquad (3.39)$$

The inertia torque of the kth link about its mass center can be obtained from the following formula [36]

$$\mathbf{T}^{*k} = -\mathbf{I}^k \cdot \boldsymbol{\alpha}^k - \mathbf{w}^k \times \mathbf{I}^k \cdot \mathbf{w}^k \qquad (3.40)$$

where $\boldsymbol{\alpha}^k$ is the angular acceleration of the link and can be obtained by taking time derivative to each side of the equation (3.10) and then summing the three components as

$$\boldsymbol{\alpha}^k = \sum_{j=1}^{3} \sum_{r=1}^{k} (w_{jr}{}^k \dot{u}_r + \dot{w}_{jr}{}^k u_r)\mathbf{n}_j{}^k \tag{3.41}$$

or

$$\boldsymbol{\alpha}^k = \sum_{j=1}^{3} \sum_{r=1}^{k} (w_{jr}{}^k \dot{u}_r + E_j{}^k)\mathbf{n}_j{}^k \tag{3.42}$$

with $E_j{}^k$ defined as

$$E_j{}^k = \sum_{r=1}^{k} \dot{w}_{jr}{}^k u_r \tag{3.43}$$

$\mathbf{I}^k$ is the central inertia dyadic of link k. In general, the central principal axes of the inertia of the kth link are not parallel to the kth set of coordinate vectors for an arbitrary robot configuration. Therefore, the inertia dyadic $\mathbf{I}^k$ generally consists of nine elements including three moments of inertia and six products of inertia. To derive the expression of the generalized inertia torques for the general cases, it is nothing more than just representing the inertia dyadic with a double summation and plugging it into the equation (3.40). However, in order to keep the assumptions the same as those for the methods mentioned in Chapter I so that the comparison can be based on the same bases, it is simply assumed in the present work that the three unit vectors $\mathbf{n}^k{}_1$, $\mathbf{n}^k{}_2$, $\mathbf{n}^k{}_3$ fixed on link k are parallel to the central principal axes of the inertia of the link, so that $\mathbf{I}^k$ can be expressed as

$$\mathbf{I}^k = I_1{}^k\, \mathbf{n}_1{}^k \mathbf{n}_1{}^k + I_2{}^k\, \mathbf{n}_2{}^k \mathbf{n}_2{}^k + I_3{}^k\, \mathbf{n}_3{}^k \mathbf{n}_3{}^k$$

Now, substituting α^k, $\mathbf{W}^k$ and $\mathbf{I}^k$ into equation (3.40), one can obtain

$$\mathbf{T}^{*k} = -\sum_{j=1}^{3} \left(\sum_{r=1}^{k} I_j{}^k w_{jr}{}^k \dot{u}_r + H_j{}^k \right)\mathbf{n}_j{}^k \tag{3.44}$$

where

$$H_1^k = I_1^k E_1^k + W_2^k W_3^k (I_3^k - I_2^k)$$
$$H_2^k = I_2^k E_2^k + W_3^k W_1^k (I_1^k - I_3^k) \quad (3.45)$$
$$H_3^k = I_3^k E_3^k + W_1^k W_2^k (I_2^k - I_1^k)$$

With $\mathbf{R}^{*k}$, $\mathbf{T}^{*k}$, $\mathbf{W}^k$, and $\mathbf{V}^k$ in hand, one is ready to construct the expressions for generalized inertia forces. The contribution from link k to the generalized inertia force K_r^* is denoted by $(K_r^*)_k$ and can be evaluated from

$$(K_r^*)_k = \mathbf{w}_r^k \cdot \mathbf{T}^{*k} + \mathbf{v}_r^k \cdot \mathbf{R}^{*k} \quad (3.46)$$

Substituting equations (3.39), (3.44) and

$$\mathbf{w}_r^k = \sum_{j=1}^{3} w_{jr}{}^k \mathbf{n}_j{}^k$$

$$\mathbf{v}_r^k = \sum_{j=1}^{3} v_{jr}{}^k \mathbf{n}_j{}^k$$

into equation (3.46) and rearranging one has

$$(K_r^*)_k = -\sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k)\dot{u}_m$$
$$-\sum_{j=1}^{3} (w_{jr}{}^k H_j{}^k + m_k v_{jr}{}^k D_j{}^k)$$

The generalized inertia force is the sum of contributions evaluated as above:

$$K^*{}_r = \sum_{k=1}^{N} (K^*{}_r)_k$$
$$= -\sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k)\dot{u}_m$$
$$-\sum_{k=1}^{N} \sum_{j=1}^{3} (w_{jr}{}^k H_j{}^k + m_k v_{jr}{}^k D_j{}^k) \quad (r=1, \ldots, N) \quad (3.47)$$

In addition to the generalized inertia forces, the expressions for the generalized active forces need to be constructed. To this end, let $\mathbf{F}^{k-1/k}$ be the resultant of forces from link k-1 acting on link k, $\mathbf{T}^{k-1/k}$ be the resultant moment about the mass center of link k of these forces, G_k be the gravitational force acting on the mass center, and $v^{\bar{k}}$ denote the velocity of the point of link k-1 coinciding with the mass center of the link k. Then, with the same reasoning that led to equation (2.55) in Chapter II, one can express the rth generalized active force as

$$
\begin{aligned}
K_r = & (\mathbf{w}_r^1-\mathbf{w}_r^0)\cdot\mathbf{T}^{0/1}+(\mathbf{w}_r^2-\mathbf{w}_r^1)\cdot\mathbf{T}^{1/2}+\ldots \\
& +(\mathbf{w}_r^N-\mathbf{w}_r^{N-1})\cdot\mathbf{T}^{N-1/N}+(\mathbf{v}_r^1-\mathbf{v}_r^{\bar{1}})\cdot\mathbf{F}^{0/1}+(\mathbf{v}_r^2-\mathbf{v}_r^{\bar{2}})\cdot\mathbf{F}^{1/2} \\
& +\ldots+(\mathbf{v}_r^N-\mathbf{v}_r^{\bar{N}})\cdot\mathbf{F}^{N-1/N}+\mathbf{v}_r^1\cdot\mathbf{G}_1+\mathbf{v}_r^2\cdot\mathbf{G}_2+\ldots+\mathbf{v}_r^N\cdot\mathbf{G}_N
\end{aligned}
\tag{3.48}
$$

Recalling the definitions of the generalized speeds and the angular velocities and the specifications for coordinate frames, the differences of the angular velocities between two adjacent links may be expressed simply as

$$\mathbf{w}^k - \mathbf{w}^{k-1} = u_k\mathbf{n}_3^{k-1}$$

that is

$$
\begin{aligned}
\mathbf{w}^1 - \mathbf{w}^0 &= u_1\mathbf{n}_3^0 \\
\mathbf{w}^2 - \mathbf{w}^1 &= u_2\mathbf{n}_3^1 \\
&\cdots\cdots \\
\mathbf{w}^N - \mathbf{w}^{N-1} &= u_k\mathbf{n}_3^{N-1}
\end{aligned}
$$

From these equations, the general expression for the difference of the partial angular velocities between two adjacent links follows as

$$
\mathbf{w}_r^k-\mathbf{w}_r^{k-1} = \begin{cases} \mathbf{n}_3^{k-1} & \text{(for } r=k) \\ 0 & \text{(for } r\neq k) \end{cases}
$$

The same is true for the difference of the partial velocities between two adjacent links with translational connection, thus

$$\mathbf{v}_r^{\,k}-\mathbf{v}_r^{\,\bar{k}} = \begin{cases} \mathbf{n}_3^{\,k-1} & (\text{for } r=k) \\ 0 & (\text{for } r\neq k) \end{cases}$$

Consequently, equation (3.48) can be rewritten as

$$K_r = (\mathbf{w}_r^{\,r}-\mathbf{w}_r^{\,r-1})\cdot\mathbf{T}^{r-1/r} + (\mathbf{v}_r^{\,r}-\mathbf{v}_r^{\,\bar{r}})\cdot\mathbf{F}^{r-1/r} + \mathbf{v}_r^{\,1}\cdot\mathbf{G}_1 + \mathbf{v}_r^{\,2}\cdot\mathbf{G}_2 + \ldots + \mathbf{v}_r^{\,N}\cdot\mathbf{G}_N$$

or

$$K_r=\sum_{k=1}^{N}(\mathbf{v}_r^{\,k}\cdot\mathbf{G}_k) + (\mathbf{w}_r^{\,r}-\mathbf{w}_r^{\,r-1})\cdot\mathbf{T}^{r-1/r}+(\mathbf{v}_r^{\,r}-\mathbf{v}_r^{\,\bar{r}})\cdot\mathbf{F}^{r-1/r} \tag{3.49}$$

Notice that, for the rth and r-1 th links, either

$$\begin{cases} \mathbf{w}_r^{\,r}-\mathbf{w}_r^{\,r-1} = \mathbf{n}_3^{\,r-1} \\ \mathbf{v}_r^{\,r}-\mathbf{v}_r^{\,\bar{r}} = 0 \end{cases} \quad (\text{for the rotational joint})$$

or

$$\begin{cases} \mathbf{w}_r^{\,r}-\mathbf{w}_r^{\,r-1} = 0 \\ \mathbf{v}_r^{\,r}-\mathbf{v}_r^{\,\bar{r}} = \mathbf{n}_3^{\,r-1} \end{cases} \quad (\text{for the translational joint})$$

must be satisfied. Therefore, one can introduce f_r as

$$f_r = \begin{cases} (\mathbf{w}_r^{\,r}-\mathbf{w}_r^{\,r-1})\cdot\mathbf{T}^{r-1/r} & (\text{when joint k is rotational}) \\ (\mathbf{v}_r^{\,r}-\mathbf{v}_r^{\,\bar{r}})\cdot\mathbf{F}^{r-1/r} & (\text{when joint k is translational}) \end{cases}$$

i.e.

$$f_r = \begin{cases} \mathbf{n}_3^{\,r-1}\cdot\mathbf{T}^{r-1/r} & (\text{when joint k is rotational}) \\ \mathbf{n}_3^{\,r-1}\cdot\mathbf{F}^{r-1/r} & (\text{when joint k is translational}) \end{cases}$$

Substituting this into equation (3.49) gives

$$K_r = \sum_{k=1}^{N}(\mathbf{v}_r^{\,k}\cdot\mathbf{G}_k) + f_r \tag{3.50}$$

Now, consider $\mathbf{G}_k$, the gravitational force. In the base coordinate,

$$\mathbf{G}_k = m_k g \mathbf{n}_L{}^0 \tag{3.51}$$

Where $\mathbf{n}_L{}^0$ is the unit vector of the base coordinate frame that coincides with the direction of the gravitational force. In terms of the quantities

$$Y^k{}_j = \begin{cases} A^1{}_{Lj} & (k=1) \\ \sum_{i=1}^{3} Y_i{}^{k-1} A^k{}_{ij} & (k=2,3,\ldots,N) \end{cases} \tag{3.52}$$

this may be expressed as

$$\mathbf{n}_L{}^0 = \sum_{j=1}^{3} Y_j{}^{k-1} \mathbf{n}^k{}_j \tag{3.53}$$

so that

$$K_r = \sum_{k=1}^{N} \sum_{j=1}^{3} m_k g Y_j{}^k v_{jr}{}^k + f_r \tag{3.54}$$

Finally, the dynamical equations are given by substituting the available expressions into the Kane's dynamical equations [36]

$$K_r + K_r{}^* = 0 \qquad (r=1,\ldots,N) \tag{3.55}$$

That is

$$\sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k)\dot{u}_m = K_r - Z_r \qquad (r=1,2,\ldots,N) \tag{3.56}$$

where

$$Z_r = \sum_{k=1}^{N} \sum_{j=1}^{3} (w_{jr}{}^k H_j{}^k + m_k v_{jr}{}^k D_j{}^k) \qquad (r=1,2,\ldots,N) \tag{3.57}$$

These expressions, together with

$$\dot{q}_r = u_r \qquad (r=1,2,\ldots,N),$$

furnish a set of 2N system equations. In matrix form, these may be expressed as

$$\left[\begin{array}{c|c} F & 0 \\ \hline 0 & I \end{array}\right] \left\{\begin{array}{c} \dot{U} \\ \hline \dot{q} \end{array}\right\} = \left\{\begin{array}{c} K-Z \\ \hline U \end{array}\right\} \tag{3.58}$$

where F is called the inertia matrix. Observe that the element in the mth row and rth column of the inertia matrix F, denoted by F_{mr}, is equal to the element in the rth row and the mth column,

$$\begin{aligned} F_{rm} &= \sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k) \\ &= \sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jr}{}^k w_{jm}{}^k + m_k v_{jr}{}^k v_{jm}{}^k) \\ &= F_{mr} \end{aligned}$$

i.e.

$$F_{rm} = F_{mr} \qquad (3.59)$$

This symmetry property enables one to reduce, from N^2 to $N(N+1)/2$, the number of the arithmetic operations for calculating the coefficients.

III.3 COMPUTATIONAL PROCEDURE

The preceding algorithm is derived with dual purposes; one is to determine the values of the generalized coordinates (i.e. desired motions) with driving torques and forces given; another involves determining the driving torques and/or forces needed to carry out a user-specified motion.

The first category involves integration of a set of nonlinear differential equations. The generalized coordinates are output. Users are to be asked to specify driving torques and forces either in the form of a set of numerical values or in the form of functions. Also, initial values of the generalized coordinates and their time derivatives must be provided.

In writing a computer program to integrate the set of differential equations, one may reorder the calculation procedure and modify some of the expressions derived in the last section to reduce memory space and to avoid repeated calculations. In summary, the following calculation procedure is suggested.

(1) Calculate all partial angular velocities by equation (3.12), carry out their time derivatives by equation (3.13), then use the following equation to find all the components of each angular velocity.

$$w_j{}^k = \begin{cases} \sum_{r=1}^{k} w_{jr}{}^k u_r & \text{(link k is rotational)} \\ \sum_{r=1}^{k-1} w_{jr}{}^k u_r & \text{(link k is translational)} \end{cases}$$

(2) Determine partial velocities for each mass center and for each coordinate origin using equations (3.26) or (3,.8) and (3.27) or (3.29) respectively, then calculate the components of each velocity by

$$v_j{}^k = \sum_{r=1}^{k} v_{jr}{}^k u_r \qquad (j=1,2,3.\ k=1,\ldots,N)$$

$$s_j{}^{k+1} = \sum_{r=1}^{k} s_{jr}{}^{k+1} u_r \qquad (j=1,2,3.\ k=0,\ldots,N-1)$$

(3) Calculate $P_{jr}{}^k$ by equation (3.30), then determine the time derivatives of each partial velocity by equations (3.31) or (3.33) and (3.32) or (3.34).

(4) Determine the intermediate variables in the order of $B_j{}^k$ by equation (3.36), $D_j{}^k$ by equation (3.37), $E_j{}^k$ by equation (3.43), $H_j{}^k$ by equation (3.45) and Z_r by equation (3.57).

(5) Find $Y_j{}^k$ by equation (3.52), then build N generalized active forces by equation (3.54).

(6) Calculate inertia coefficients by the equation

$$F_{rm} = \sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k)$$

taking advantage of the symmetry property to reduce the arithmetic operations.

(7) Substitute the expressions into Kane's equations to obtain

$$\sum_{m=1}^{N} F_{rm} \dot{u}_m = K_r - Z_r \qquad (r=1,\ldots,N)$$

together with

$$\dot{q}_r = u_r \qquad (r=1,\ldots,N)$$

(8) Finally, with the given initial conditions and the specified torques and/or forces for a particular system, one can perform integration of the dynamic and kinematic equations derived in step (7) to find the generalized coordinates for each link at all time steps.

As for the second category, the dynamic inverse of the first category, users are asked to specify a desired motion for the system. In other words, a set of numerical values or functions specifying generalized coordinates and their derivatives are given as input, and the program provides driving torques and forces. This is a straightforward calculation to evaluate the driving forces f_r, which appear in equation (3.54). For convenience, introduce K'_r as

$$K'_r = \sum_{k=1}^{N} \sum_{j=1}^{3} m_k g \; Y_j{}^k v_{jr}{}^k \tag{3.60}$$

So that the equation (3.54) can be written as

$$K_r = f_r + K'_r \tag{3.61}$$

Substituting of equation (3.61) into the Kane's dynamical equation and rearranging it give

$$f_r = -K^*_r - K'_r \tag{3.62}$$

A complete calculation procedure for the dynamic inverse is outlined in the following

(1) Calculate all partial angular velocities by equation (3.12), carry out their time derivatives by equation (3.13), then use the following equation to find all the components of each angular velocity.

$$w_j{}^k = \begin{cases} \sum_{r=1}^{k} w_{jr}{}^k u_r & \text{(link k is rotational)} \\ \sum_{r=1}^{k-1} w_{jr}{}^k u_r & \text{(link k is translational)} \end{cases}$$

(2) Determine partial velocities for each mass center and for each coordinate origin using equations (3.26) or (3.28) and (3.27) or (3.29) respectively, then calculate the components of each velocity by

$$v_j{}^k = \sum_{r=1}^{k} v_{jr}{}^k u_r \qquad (j=1,2,3.\ k=1,\ldots,N)$$

$$s_j{}^{k+1} = \sum_{r=1}^{k} s_{jr}{}^{k+1} u_r \qquad (j=1,2,3.\ k=0,\ldots,N-1)$$

(3) Calculate P_{jr}^{k} by equation (3.30), then determine the time derivatives of each partial velocity by equations (3.31) or (3.33) and (3.32) or (3.34).

(4) Determine the intermediate variables in the order of B_j^k by equation (3.36), D_j^k by equation (3.37), E_j^k by equation (3.43), H_j^k by equation (3.45) and Z_r by equation (3.57).

(5) Find Y_j^k by equation (3.52), then evaluate quantity K'_r by the equation (3.60)

(6) Calculate inertia coefficients by the equation

$$F_{rm} = \sum_{k=1}^{N} \sum_{j=1}^{3} \sum_{m=1}^{k} (I_j{}^k w_{jm}{}^k w_{jr}{}^k + m_k v_{jm}{}^k v_{jr}{}^k)$$

taking advantage of the symmetry property to reduce the arithmetic operations.

(7) Determine the generalized inertia forces by

$$K^*{}_r = \sum_{m=1}^{N} F_{rm}\dot{u}_m + Z_r \qquad (r=1,\ldots,N)$$

(8) Finally, calculate the active forces and/or torques by

$$f_r = -K^*{}_r - K'{}_r$$

In both cases, mass properties and structural data describing the geometric relations between links must be specified. For the structural data, users can use equation (2.37) in reference [36] to enlist a computer to automatically generate each element of the N transformation matrices and their derivatives; or, to save memory space and computation time, one may simply write those elements into a computer program.

To test the validity of the underlying algorithm, two programs, AUTOTF (Appendix 4) and AUTOQ (Appendix 5), have been written in FORTRAN language. AUTOTF is to automatically generate driving torques and/or forces with a specified motion. AUTOQ is to simultaneously generate and solve the equations of motion with driving torques and/or forces given. The algorithm for numerically solving the nonlinear differential equations is the same as that for the the program ANGLE in Chapter II.

For comparison, the Intelledex 605 robot with the same parameters used for programs TORQUE and ANGLE in Chapter II is again simulated with both AUTOTF and AUTOQ. The specifications of the coordinate frames are as shown in Figure 3.2.

For the example use of AUTOTF, the rotation angles $q_1(t),\ldots,q_6(t)$ have been specified as

$$q_r = \pi(t-T\sin(2\pi t/T)/2\pi)/3T \qquad (r=1,\ 3,\ \ldots,\ 6)$$

$$q_2 = \pi/2-\pi(t-T\sin(2\pi t/T)/2\pi)/6T \qquad (60)$$

with T=2s, the same as that for program TORQUE plotted in Figures 2.3 and 2.4. The outputs (see Appendix 4), driving torques on each of the six links, are plotted in Figures 3.3-3.8. The values of these torques are precisely the same as those generated by program TORQUE at every time step. This is an indication that both the automatic generation algorithm and the hand-derived equations are correct.

As an example inverse process use of the AUTOQ, the output file of the above-described run of AUTOTF was used as input torques. The outputs, plotted in Figures 3.9-3.14, are consistent with those shown in Figures 2.3-2.4. This further indicates the algorithm is correct.

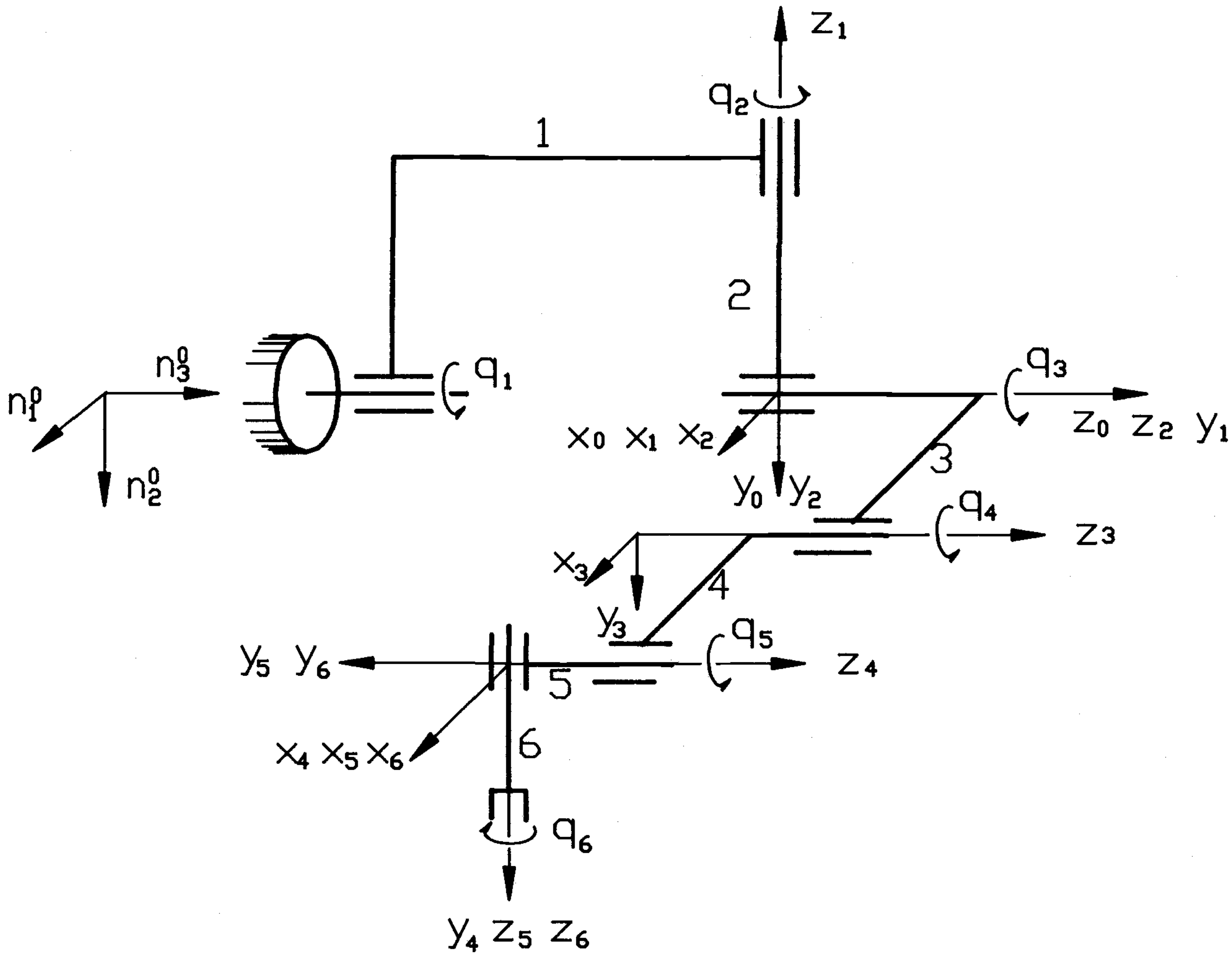

Figure 3.2. Coordinate frames for the Intelledex 605 arm.

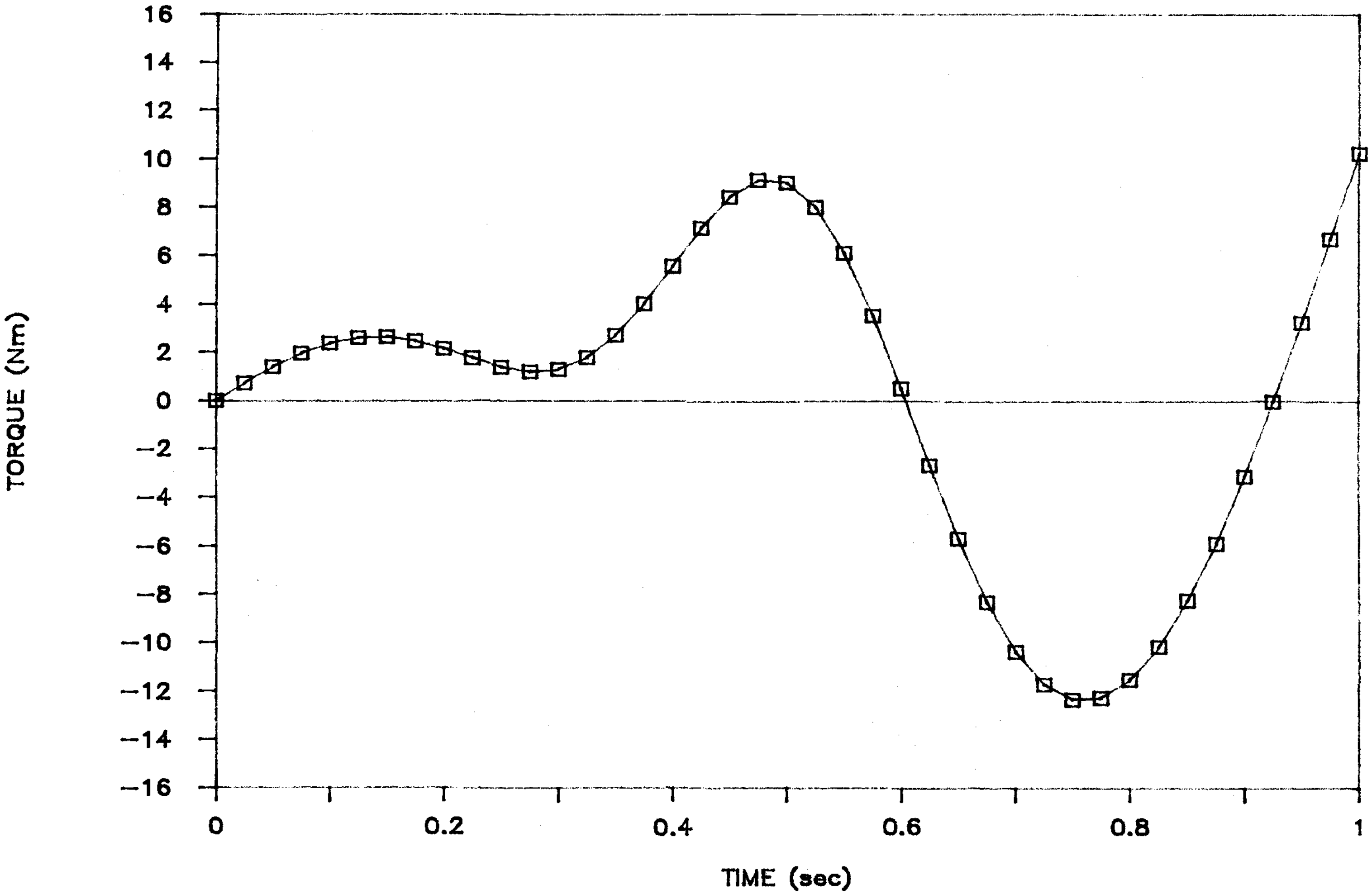

Figure 3.3. Torque τ_1 vs time (output of AUTOTF).

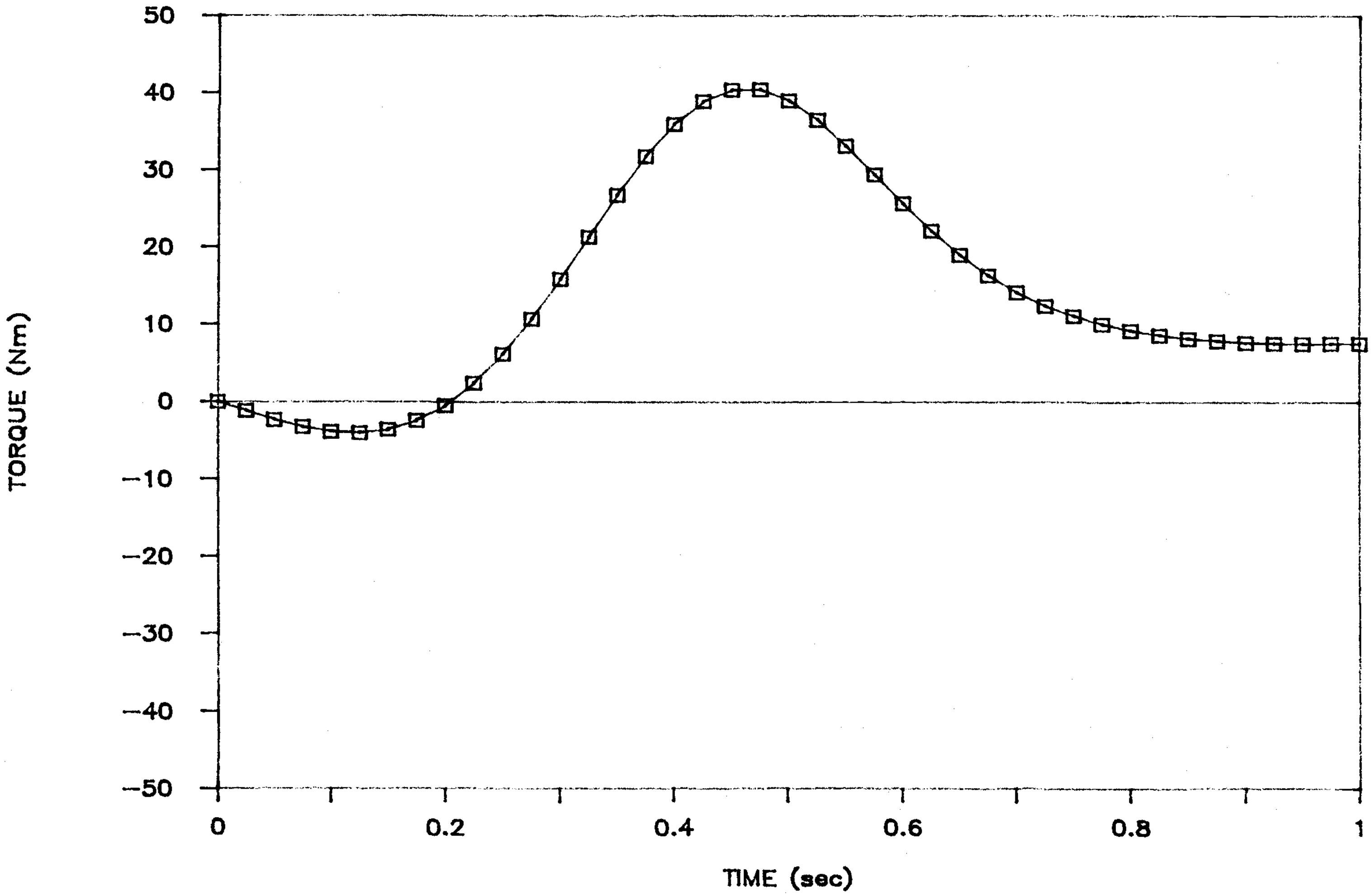

Figure 3.4. Torque τ_2 vs time (output of AUTOTF).

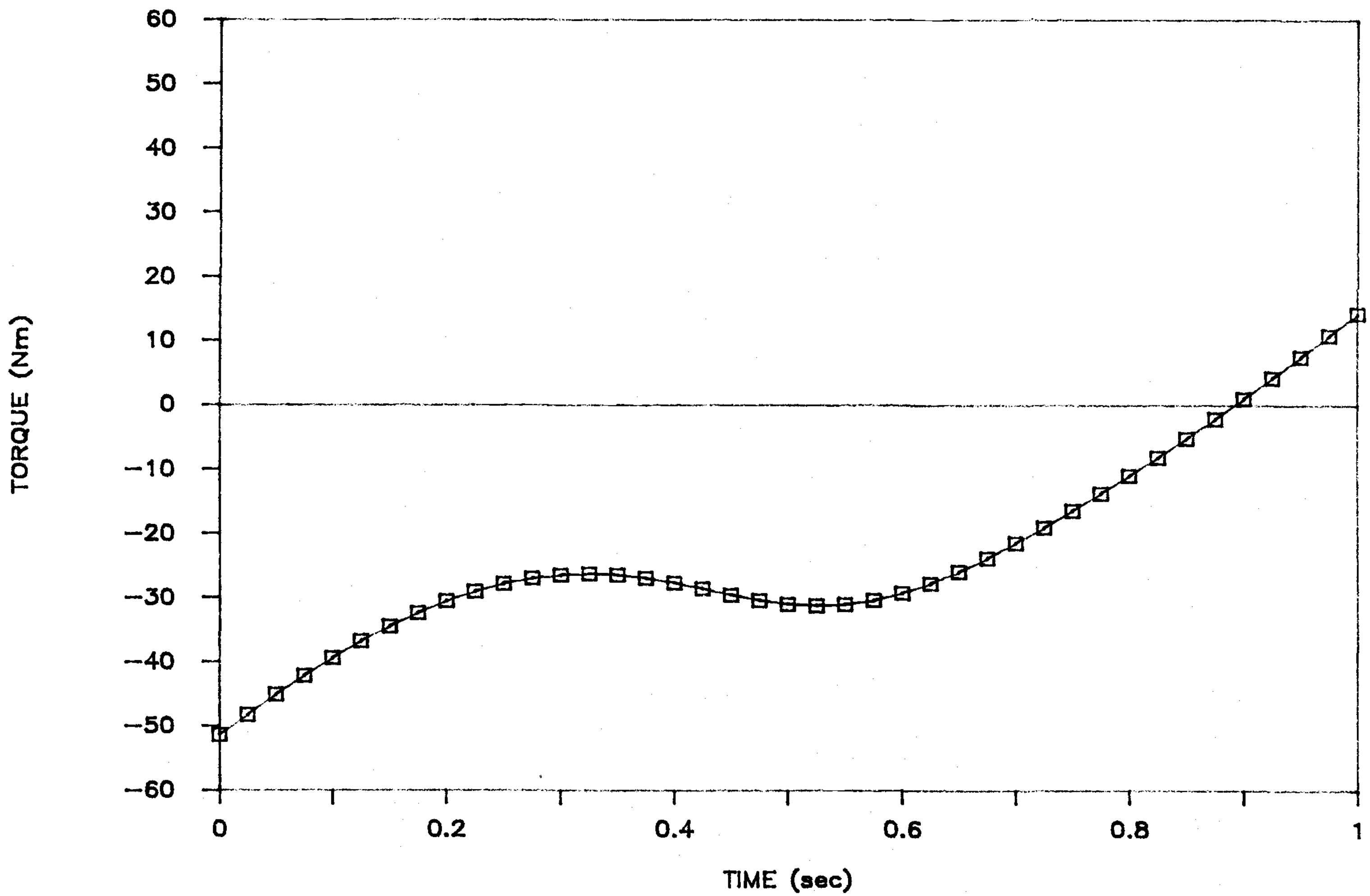

Figure 3.5. Torque τ_3 vs time (output of AUTOTF).

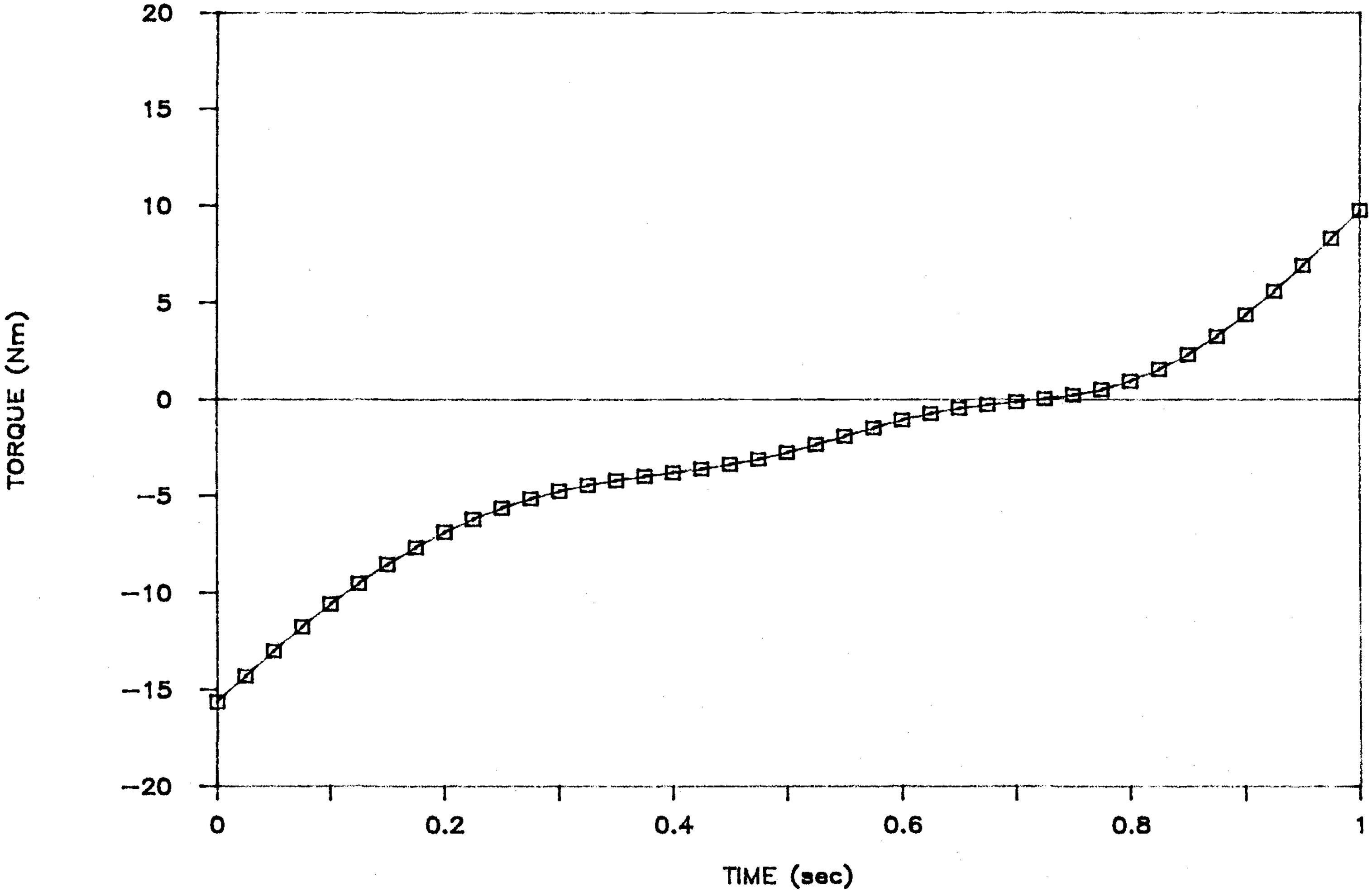

Figure 3.6. Torque τ_4 vs time (output of AUTOTF).

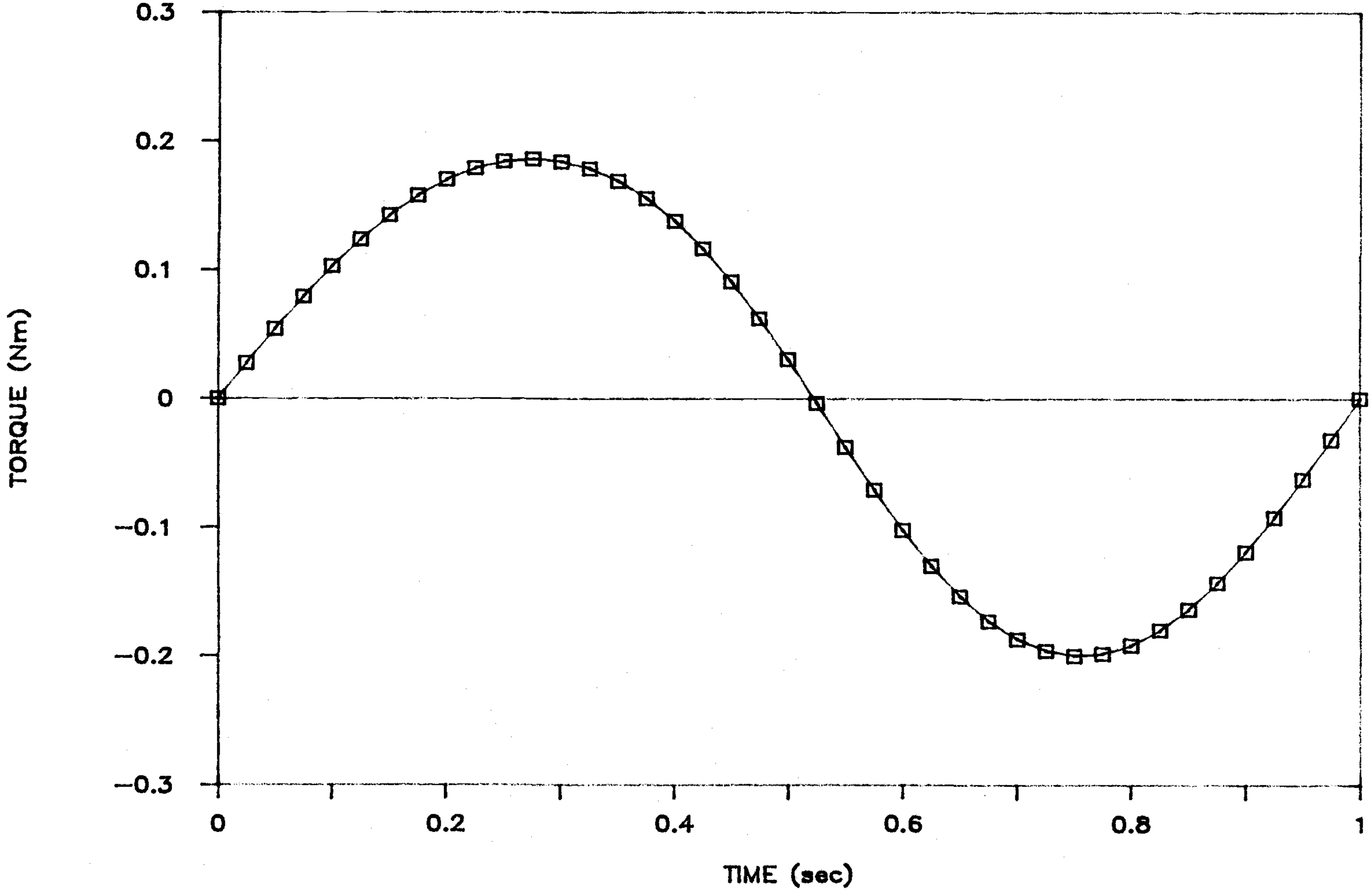

Figure 3.7. Torque τ_5 vs time (output of AUTOTF).

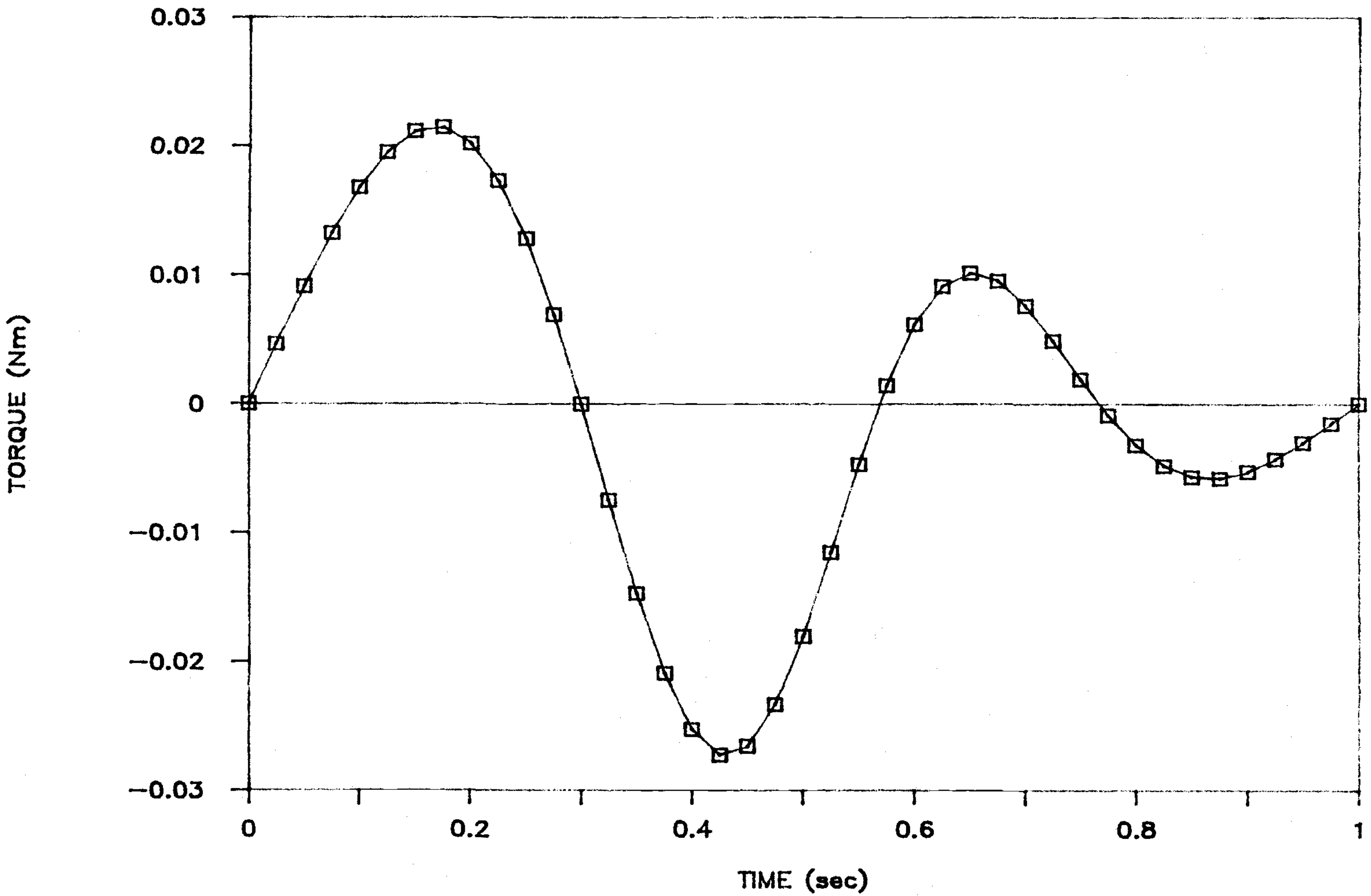

Figure 3.8. Torque τ_6 vs time (output of AUTOTF).

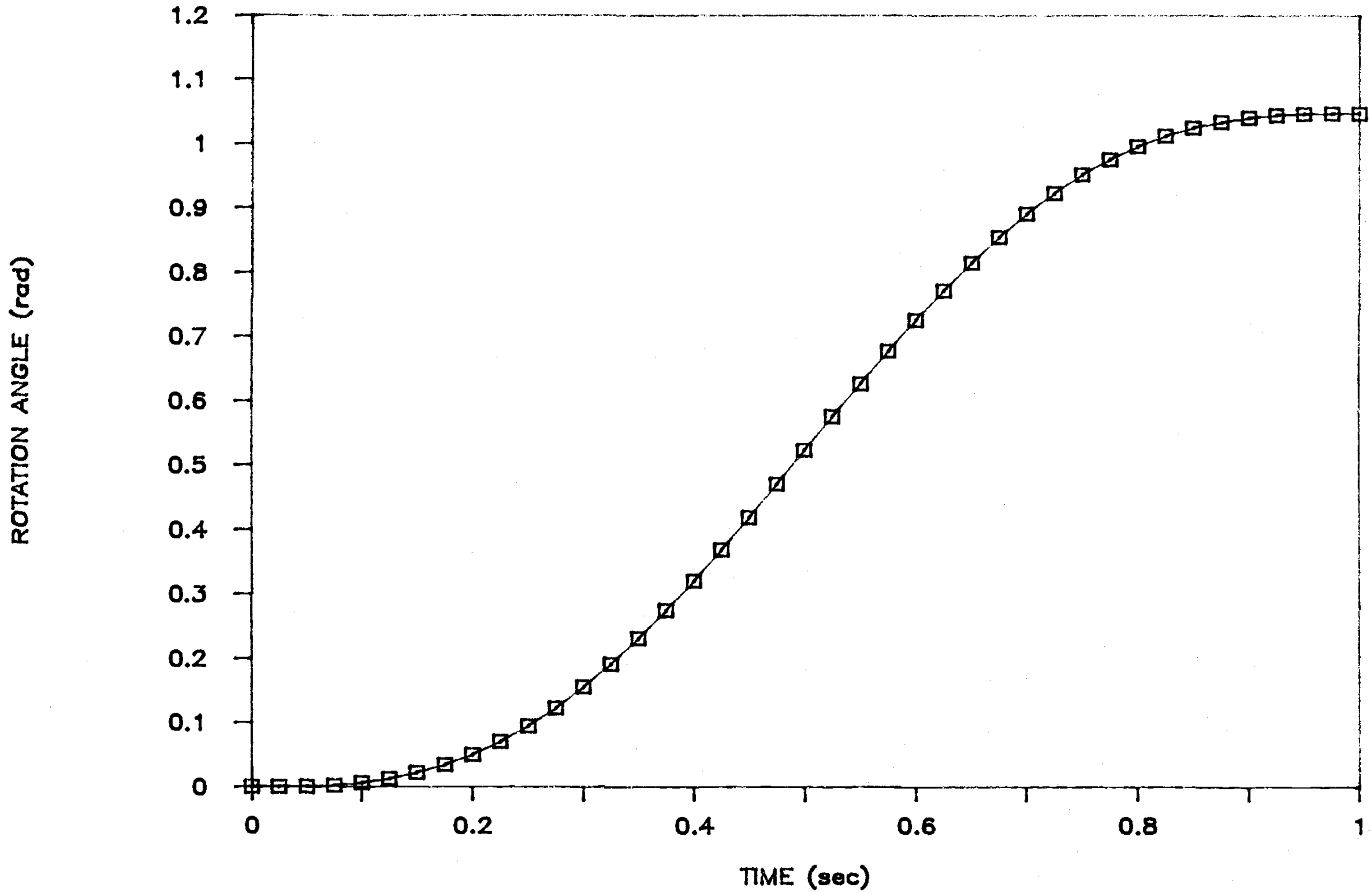

Figure 3.9. Coordinate q_1 vs time (output of AUTOQ).

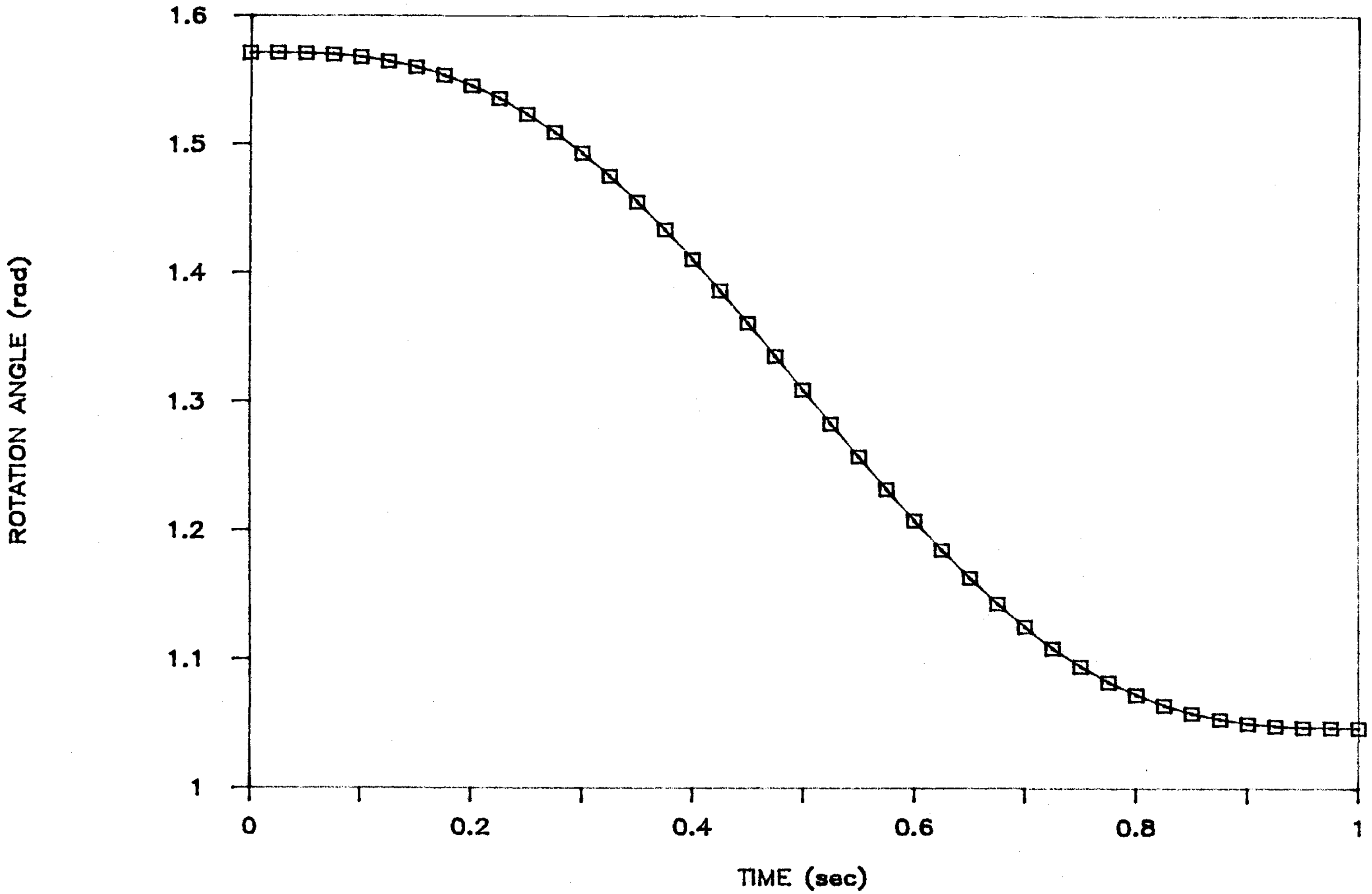

Figure 3.10. Coordinate q_2 vs time (output of AUTOQ).

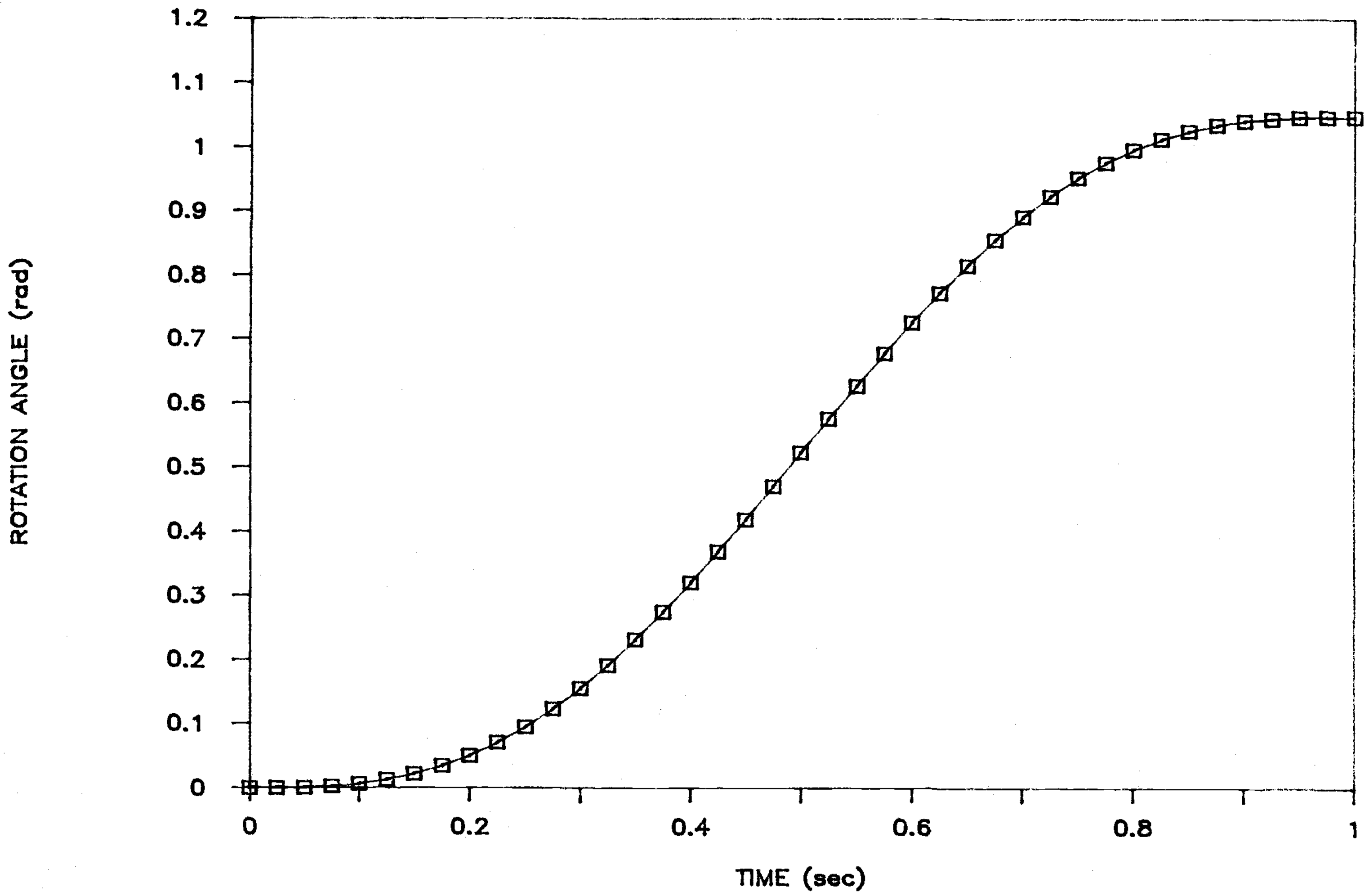

Figure 3.11. Coordinate q_3 vs time (output of AUTOQ).

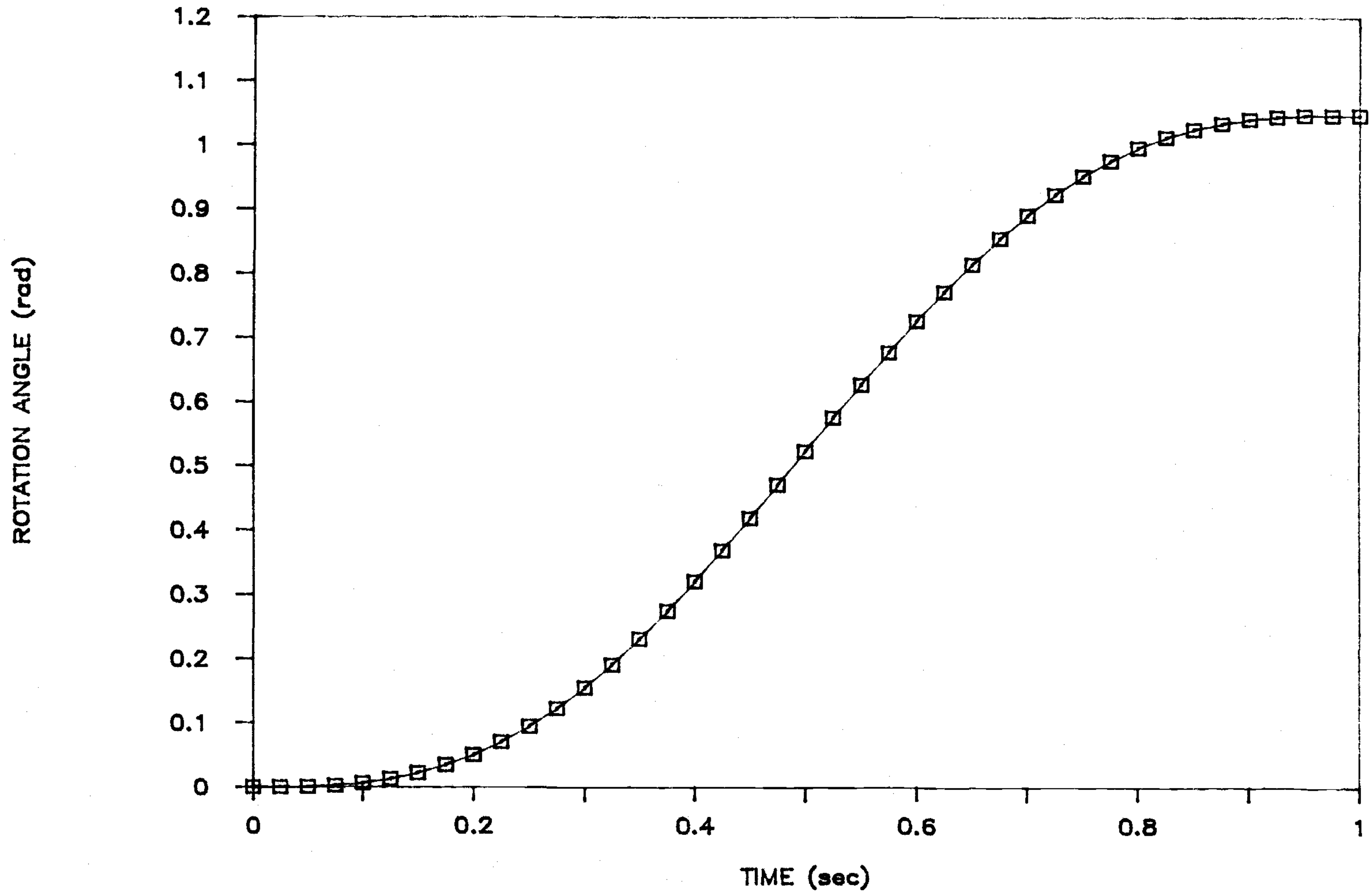

Figure 3.12. Coordinate q_4 vs time (output of AUTOQ).

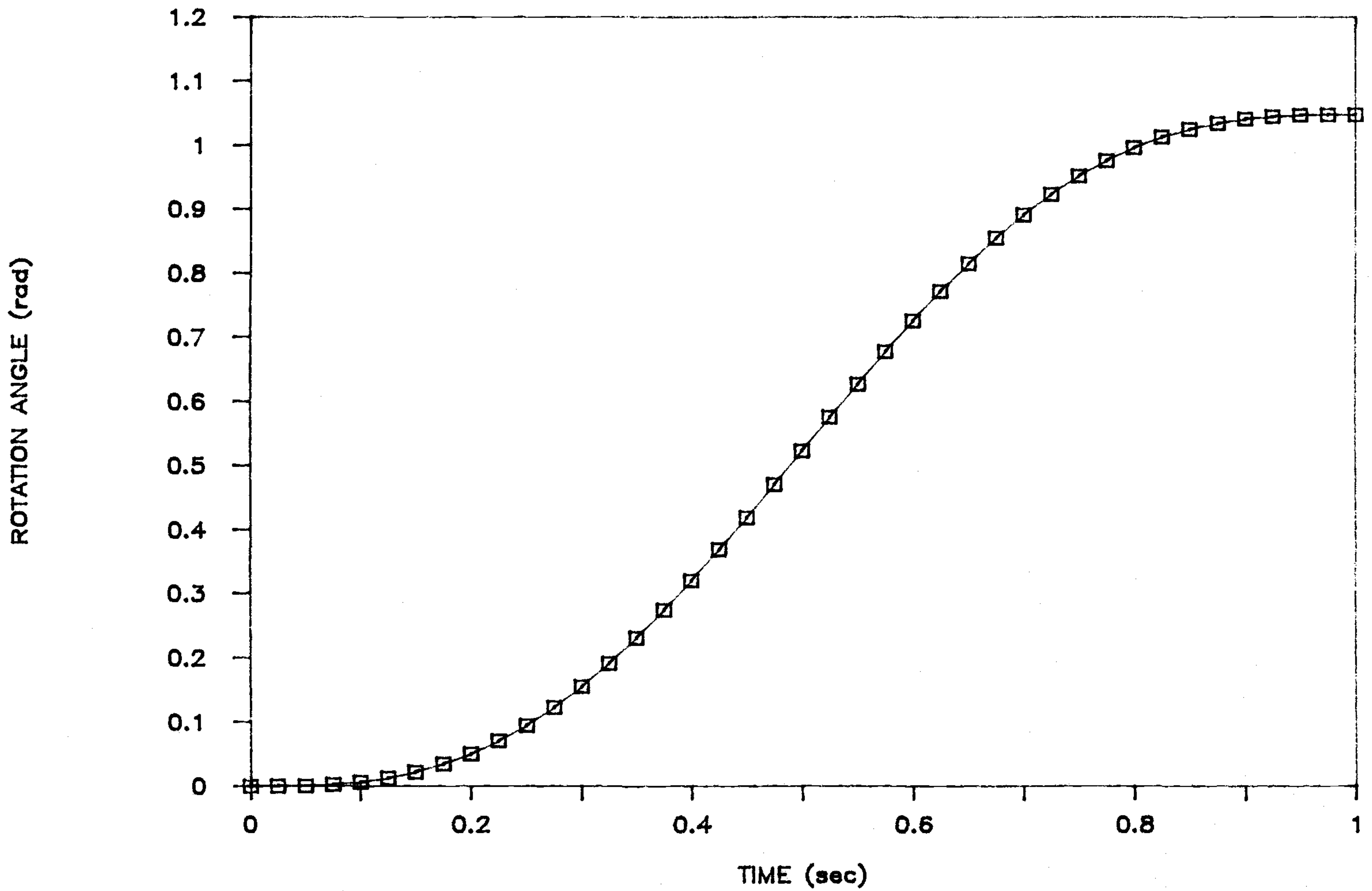

Figure 3.13. Coordinate q_5 vs time (output of AUTOQ).

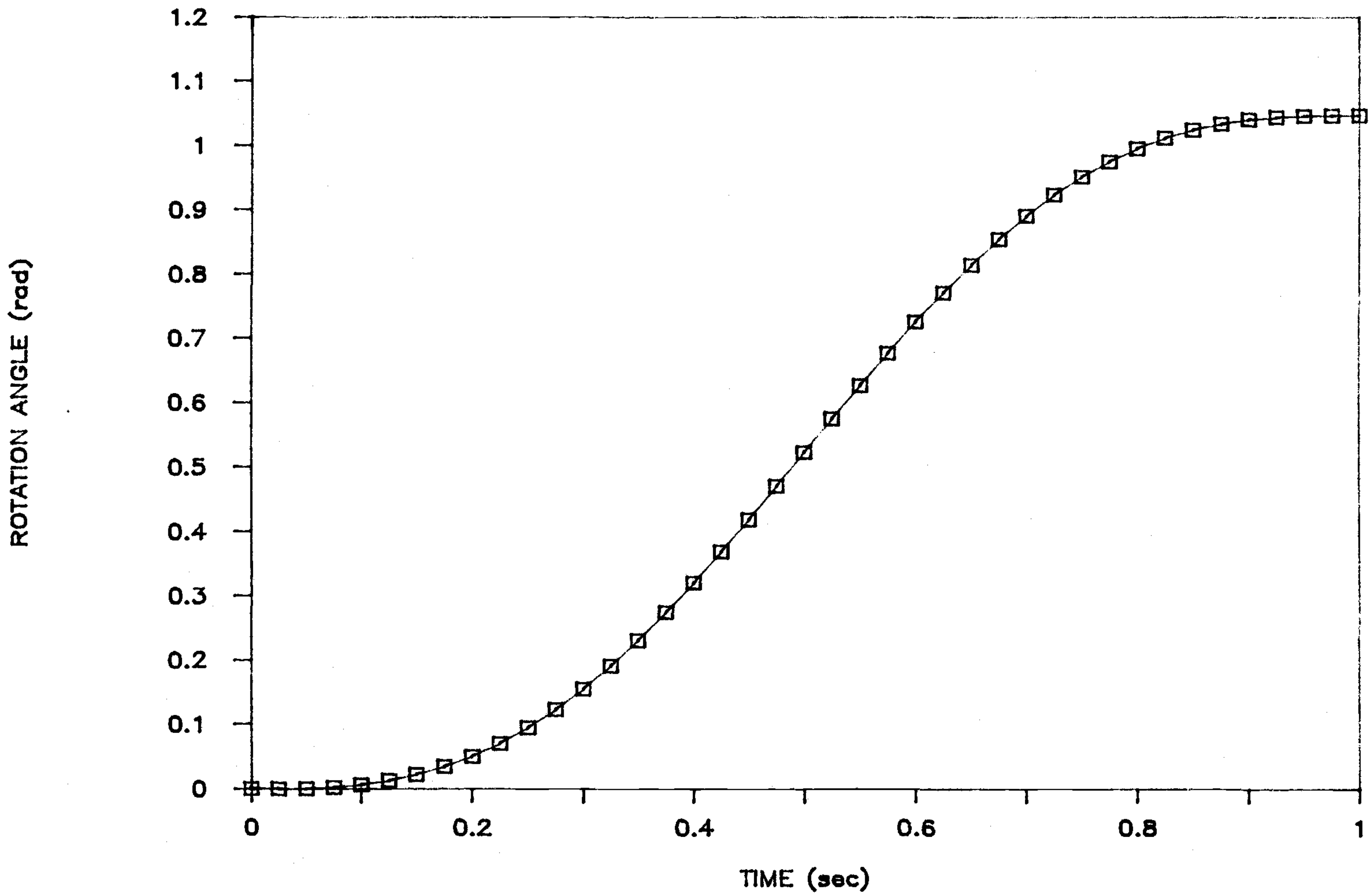

Figure 3.14. Coordinate q_6 vs time (output of AUTOQ).

IV. DISCUSSIONS AND CONCLUSIONS

1. In the field of manipulator dynamics, one of the principal concerns of dynamicists is computational efficiency. The derivation procedure of Kane's dynamical equations presented in this work apparently avoids such problems intrinsic to the use of Newton-Euler or Lagrange equations as: (1) introducing and subsequently eliminating large numbers of nonworking contact forces between rigid bodies, as required in the Newton-Euler formulation; (2) extensive effort required to differentiate the kinetic expression and large number of unnecessary arithmetic operations, which are inevitable in Lagrangian formulation; (3) repeatedly evaluating N(N-1)/2 inertia coefficients which are determined by symmetric property of the inertia matrix. Accordingly, the use of Kane's dynamical equations proposed here may be expected to lead to computational algorithms involving fewer arithmetic operations than algorithms generated by employing the best available Lagrangian and Newton-Euler approaches.

2. A comparison of the numbers of arithmetic operations indicates a computational efficiency, for this algorithm applied to the Intelledex 605, similar to that shown by Kane and Levinson [42] for the Stanford arm, who say,

"According to Hollerbach [29], when one resorts to the

> Lagrangian approach to determine quantities equivalent to our τ_1, ..., τ_5 and F_6 for an instant at which q_1, ..., q_6, q_1, ..., q_6, q_1, ..., q_6 are specified, one must perform 2195 multiplications and 1719 additions. These numbers are reduced to 1541 and 1196, respectively, if a Newton-Euler technique method reported by walker and Orin [49] is employed, and they become 852 and 738, respectively, when either the Newton-Euler technique discussed by Hollerbach or Silver's Lagrangian formulation [50] is used."

With Kane's method, however, Kane and Levinson continue,

> "one needs but 646 multiplications and 394 additions to accomplish the same task."

In the present work, 672 multiplications and 404 additions are needed. The difference between these numbers and those stated by Kane and Levinson is because of different robot configurations. The Intelledex 605 robot consists of six rotational links, while the Stanford Arm discussed by Kane and Levinson consists of five rotational links and one translational link. Moreover, the numbers discussed in the present work could be reduced if one is willing to spend time on working with the trigonometric identities to find the simplest expressions for the kinematic terms.

3. Based on Kane's dynamical equations, a fully automatic generation algorithm of the equations of motion has been derived. This algorithm bypasses the manual derivations of all kinematic and kinetic ingredients,

leading in a straightforward way to the equations of motion, reducing the amount of hand labor to a minimum. The equations of motion derived by this algorithm is "exact" and explicit. The algorithm can furnish computer programs to simultaneously generate and integrate the equations of motion or to carry out the inverse dynamics. On an IBM AT microcomputer, the program AUTOTF takes about 0.28 second on average to compute six torques. Although this means that it is not feasible to perform on-line computation on an IBM AT, it would certainly be faster than that reported by Luh, Walker and Paul [27] if the same computer (PDP11/45) were used, since the number of arithmetic operations required by Kane's method is about 15% less than for Newton-Euler's method employed by Luh Walker and Paul.

4. The automatic generation algorithm presented in this work can be used for any type of linkage mechanism. This also implies that it is highly likely that a computer can be employed to deal with any given multibody dynamics problem. However, due to the fact that it is extremely difficult for a computer to recognize algebraic simplifications, such as trigonometric identities or terms that cancel each other, it is by no means true that it is always best to use a general purpose multibody computer program to carry out the requisite calculations. Indeed, when a particular multibody configuration is under consideration and the on-line computational efficiency is more important than amount of manual labor, the opposite is

true: it is better to formulate the necessary expressions by hand than to attempt to work with an available multibody program. When more than a few multibody configurations are to be investigated, it is definitely better to apply an automatic generation method rather than to work with them by hand for it can save considerable time for the analysts.

5. Generalized speeds play a central role in the formulation procedure of Kane's dynamical equations. Properly choosing the forms of generalized speeds can significantly reduce the number of arithmetic operations. In doing so, selection of the definitions for generalized speeds should be made such that corresponding expressions for partial velocities and partial angular velocities be as simple as possible. Human analysts can accomplish this by following the guiding idea summarized in page 21. But for automatic generation, selecting the definitions of generalized speeds in this way seems to make generalized active forces appear in a more complicated set of equations, thus additional arithmetic operations need to be accommodated. In an effort to avoid such problems, the first order time derivatives of the generalized coordinates are simply defined as the generalized speeds in the automatic generation algorithm; this in turn brings more arithmetic operations that the author would not like to have. Further study on this aspect is therefore suggested.

6. In performing dynamic inverse or integrating the equations of motion, the appropriate form of the dynamic

equations should consist of equations described in terms of all independent position variables and forces/torques that are explicitly involved in the dynamic equations. In other words, the dynamic equations are expected to have a closed-form. Deriving explicit input-output dynamic equations is very time-consuming if either the Newton-Euler or the Lagrangian method is employed. By contrast, the use Kane's dynamical equations leads directly to explicit equations of motion. Employing the automatic generation algorithm presented in this work can bring position variables, input forces and/or torques and inertia coefficients all in evidence in the dynamic equations. Therefore, the dynamic equations generated by this algorithm are particularly useful for either integration or to perform dynamic inverse.

REFERENCES

1. Hooker, W. W. and Margulies, G., "The Dynamical Attitude Equations for an n-Body Satellite," The Journal of the Astronautical sciences, Vol.XII, No.4, 1965.

2. Uicker, J. J., "On the Dynamic Analysis of Spatial Linkages Using 4X4 Matrices," Doctoral Dissertation, Department of Mechanical Engineering and Astronautical Sciences, Northwestern University, 1965.

3. Kahn, T. R., "The Near-Minimum-Time Control of Open-Loop Artificial Kinematic Chains," Artificial Intelligence Laboratory, Stanford University, 106, 1969.

4. Keat, J. E., "Dynamical Equations of Nonrigid Satellites," AIAA Journal, Vol.8, No.7, 1970.

5. Woo, L. S. and Freudenstein, F., "Dynamic Analysis of Mechanisms Using Screw Coordinates," ASME Journal of Engineering for Industry, Feb., 1971.

6. Yang, A. T., "Inertia Force Analysis of Spatial Mechanisms," ASME Journal of Engineering for Industry, Feb. 1971.

7. Roberson, Robert E. and Wittenburg, Jens, "A Dynamical Formulism for an Arbitrary Number of Interconnected Rigid Bodies, With Reference to the Problem of Satellite Attitude Control," Proceeding of the Third Congress of the International Federation of Automatic Control, London, Paper 46.D, June 24, 1966.

8. Fleischer, Gerald E., "Multi-Rigid-Body Attitude Dynamics Simulation," JPL Technical Report 32-1516, Feb. 15, 1971.

9. Chace, M. A., "Analysis of the Time-Dependence of Multi-Freedom Mechanical Systems in Relative Coordinates," ASME Journal of Engineering for Industry, Vol.89, 1967.

10. Uicker, J. J., "Dynamic Behavior of Spatial Linkages," ASME Journal of Engineering for Industry, Vol.91, No.1, Feb. 1969.

11. Chace, M. A. and Bayazitoglu, Y. O., "Development and Application of a Generalized d'Alembert Force Multifreedom Mechanical Systems," ASME Journal of

Engineering for Industry, Vol.93, 1971.

12. Gupta, V. K., "Dynamic Analysis of Multi-Rigid-Body Systems," ASME Journal of Engineering for Industry, Vol. 9,1974.

13. Likins, Peter W., "Dynamic Analysis of a System of Hinge-Connected Rigid Bodies with Nonrigid Appendages," International Journal of Solids and Structures, Vol.9, 1973.

14. Huston, R. L. and Passerello, C. E., "On the Dynamics of Chain Systems," ASME Paper 74-WA/Aut-11, 1974.

15. Larson, V., "State Equations for an n-Body Spacecraft," The Journal of the Astronautical Sciences, Vol.XXII, No.1, July-Sept. 1974.

16. Paul, B., "Analytical Dynamics of Mechanisms--A Computer Oriented Overview," Mechanism and Machine Theory, Vol. 10, 1975.

17. Hughes, P. C., "Dynamics of a Chain of Flexible Bodies," The Journal of the Astronautical Sciences, Vol.XXVII. No.4, Oct.-Dec. 1979.

18. Frisch, Harold P.,"A Vector-Dyadic Development of the Equations of Motion for N-Coupled Rigid Bodies and Point Masses," NASA TN D-7767, Oct. 1974.

19. Ho, J. Y. L., "Direct Path Method for Flexible Multibody Spacecraft Dynamics," Journal of Spacecraft and Rockets, Vol.14, Feb. 1977.

20. Hooker, W. W., "Equations of Motion for Interconnected Rigid and Elastic Bodies," Celestial Mechanics, Vol.11, No.3, May 1975.

21. Bejczy, A. K., "Robot Arm Dynamics and Control," NASA Technical Memorandum 33-669, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, Feb., 1974.

22. Whitney, D. E., "State-Space Models of Remote Manipulation Tasks," Transactions of the IEEE, Journal of Automatic Controls, Vol.14, Dec. 1969.

23. Raibert, M. H., "Mechanical Arm Control Using a State-Space Memory," Society of Manufacturing Engineers Technical Paper MS-77-742, 1977.

24. Raibert, M. H. and Horn, B. K. P., "Manipulator Control Using the Configuration Space Method," Industrial Robot, Vol.5, No.2, 1978.

25. Stepanenko, Y. and Vukobratovic, M., "Dynamics of Articulated Open-Chained Active Mechanism," Mathematical Biosciences, Vol.28, 1976.

26. Orin, D. E., McGhee, R. B., Vukobratotic, M. and Hafloch, G., "Kinematic and Kinetic Analysis of Open-chain Linkages Utilizing Newton-Euler Methods," Mathematical Biosciences, Vol.43, 1/2, 1978.

27. Luh, J. Y. S., Walker, M. W., and Paul, R. P. C., "On-Line Computational Scheme for Mechanical Manipulators," ASME Journal of Dynamic Systems, Measurement, and Control, Vol.102, 1980.

28. Paul, R. P., Luh, J. Y. S., et al, "Advanced Industrial Robot Control Systems," Purdue University, RR78-25, 1978.

29. Hollerbach, J. M., "A Recursive Formulation of Lagrangian Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity," IEEE Transaction on Systems, Man, Cybernetics, Vol.10, No.11, Nov.1980.

30. Hollerbach, J. M., "Wrist-Partitioned Inverse Kinematic Accelerations and Manipulator Dynamics," The International Journal of Robotics Research, Vol.2, No.4, 1983.

31. Kanade, T., Khosla, P. K. and Tanaka, N., "Real-Time Control of CMU Direct-Drive Arm II Using Customized Inverse Dynamics," IEEE Conference On Decision and Control, Las Vegas, 1984.

32. Dillon, S. R., "Computer Assisted Equation Generation In Linkage Dynamics," Doctoral Dissertation, Ohio State University, 1973.

33. Langrana N. A. and Bartel D. L., "An Automated Method for Dynamic Analysis of Spatial Linkages for Biomechanical Applications," ASME Journal of Engineering for Industry, May 1975.

34. Luh, J. Y. S. and Lin, C. S., "Automatic Generation of Dynamic Equations for Mechanical Manipulators," Joint Automatic Control Conference, Charlottesville, CA, 1981.

35. Thomas, M. and Tesar, D., "Dynamic Modeling of Serial Manipulator Arms," ASME Journal of Dynamic Systems, Measurement, and Control, Vol.104, Sept. 1982.

36. Kane, T. R., Dynamics: Theory and Applications, McGraw-Hill Book Company, New York, 1985.

37. Radetsky, Peter, "The Man Who Mastered Motion," Science, May 1986.

38. Kane, T. R. and Levinson, D. A., "Formulation of Equations of Motion for Complex Spacecrafts," Journal of Guidance Control, Vol.2, No.3, 1980.

39. Kane, T. R. Likins, P. W., and Levinson, D. A., Spacecraft Dynamics, McGraw-Hill, New York, 1983.

40. Huston, R. L. and Kelly, F. A., "The Development of Equations of Motion of Single-arm robots," IEEE Trans. Sys., Man, Cyber. SMC-10(11):730-736, 1982.

41. Huston, R. L. and Kelly, F. A., "New Approaches in Robot Dynamics and Control," Proc. 2nd Int. Computer Eng. Conf., San Diego, Aug. 1982.

42. Kane, T. R. and Levinson, D. A., "The Use of Kane's Dynamical Equations in Robotics," The International Journal of Robotics Research, Vol.2, No.3, Fall 1983.

43. Kane, T. R. and Levinson D. A., "Multibody Dynamics," ASME Journal of Applied Mechanics, Vol.50, Dec. 1983.

44. Kane, T. R. and Faessler, H., "Dynamics of Robots and Manipulators Involving Closed Loops," 5th CISM-IFTIMM Symp. Theory and Practice of Robots and Manipulators, Udine, Italy, June, 1984.

45. Faessler, H., "Computer-Assisted Generation of Dynamical Equations for Multibody Systems," The International Journal of Robotics Research, Vol.5, No.3, Fall, 1986.

46. Paul, R. P., Robot Manipulators: Mathematics, Programming, and Control, The MIT Press, Cambridge, MA. 1981.

47. Geradin, M., Idelsohn, S., and Hogge, M., "Computational Strategies for the Solution of Large Nonlinear Problems Via Quasi-Newton Methods," Computers and Structures, Vol.13, 1981.

48. Rosenthal, D. E. and Sherman, M. A., "High Performance Multibody Simulations via Symbolic Equation Manipulation and Kane's Method," The Journal of the Astronautical Sciences, Vol.34, No.3, July-September, 1986.

49. Walker, M. W. and Orin, D. E., "Efficient Dynamic Computer Simulation of Robotic Mechanisms," the ASME Journal of Dynamic Systems, Measurement and Control, Vol. 104, 1982.

50. Silver, W. M., "On the Equivalence of Lagrangian and Newton-Euler Dynamics for Manipulators," the International Journal of Robotics Research, Vol. 1(2), 1982.

APPENDICES

APPENDIX 1. INTERMEDIATE VARIABLES

$Z_1=(u_1c_3-u_2s_3)/s_2$
$Z_2=-u_1s_3-u_2c_3$
$Z_3=u_3+(u_2s_3-u_1c_3)c_2/s_2$
$Z_4=c_3/s_2$
$Z_5=-s_3/s_2$
$Z_6=u_1c_3-u_2s_3$
$Z_7=-Z_2$
$Z_8=Z_4c_2$
$Z_9=Z_5c_2$
$Z_{10}=Z_8u_1+Z_9u_2$
$Z_{11}=u_1c_4+u_2s_4$
$Z_{12}=-u_1s_4+u_2c_4$
$Z_{13}=u_3+u_4$
$Z_{14}=c_{45}$
$Z_{15}=s_{45}$
$Z_{16}=-Z_{15}$
$Z_{17}=u_1Z_{14}+u_2Z_{15}$
$Z_{18}=-Z_{13}-u_5$
$Z_{19}=u_1Z_{16}+u_2Z_{14}$
$Z_{20}=Z_{14}c_6$
$Z_{21}=Z_{15}c_6$
$Z_{22}=Z_{14}s_6$
$Z_{23}=Z_{15}s_6$
$Z_{24}=Z_{20}u_1+Z_{21}u_2+Z_{18}s_6$
$Z_{25}=-Z_{22}u_1-Z_{23}u_2+Z_{18}c_6$
$Z_{26}=Z_{16}u_1+Z_{14}u_2+u_6$
$Z_{27}=L_{13}Z_4$
$Z_{28}=L_{13}Z_5$
$Z_{29}=Z_{27}u_1+Z_{28}u_2$
$Z_{30}=L_{22}Z_8$
$Z_{31}=L_{22}Z_9$
$Z_{32}=-L_{22}c_3$
$Z_{33}=L_{22}s_3$

$Z_{34}=Z_{30}u_1+Z_{31}u_2$

$Z_{35}=Z_{32}u_1+Z_{33}u_2$

$Z_{36}=L_{33}u_2$

$Z_{37}=L_{31}u_3-L_{33}u_1$

$Z_{38}=-L_{31}u_2$

$Z_{39}=(L_{33}+L_{43})s_4$

$Z_{40}=(L_{33}+L_{43})c_4$

$Z_{41}=L_1s_4$

$Z_{42}=L_1c_4+L_{41}$

$Z_{43}=L_{41}s_4$

$Z_{44}=L_1+L_{41}c_4$

$Z_{45}=-Z_{39}u_1+Z_{40}u_2+Z_{41}u_3$

$Z_{46}=-Z_{40}u_1-Z_{39}u_2+Z_{42}u_3+L_{41}u_4$

$Z_{47}=Z_{43}u_1-Z_{44}u_2$

$Z_{48}=L_1c_4+L_2$

$Z_{49}=L_2s_4$

$Z_{50}=L_1+L_2c_4$

$Z_{51}=L_{33}+L_{43}-L_{52}$

$Z_{52}=Z_{39}c_5+Z_{40}s_5$

$Z_{53}=Z_{40}c_5-Z_{39}s_5$

$Z_{54}=Z_{15}Z_{51}-Z_{52}$

$Z_{55}=Z_{53}-Z_{14}Z_{51}$

$Z_{56}=Z_{41}c_5+Z_{48}s_5$

$Z_{57}=L_2s_5$

$Z_{58}=Z_{48}c_5-Z_{41}s_5$

$Z_{59}=L_2c_5$

$Z_{60}=Z_{54}u_1+Z_{55}u_2+Z_{56}u_3+Z_{57}u_4$

$Z_{61}=-Z_{49}u_1+Z_{50}u_2$

$Z_{62}=-Z_{55}u_1+Z_{54}u_2+Z_{58}u_3+Z_{59}u_4$

$Z_{63}=L_{33}+L_{43}$

$Z_{64}=Z_{15}Z_{63}-Z_{52}$

$Z_{65}=Z_{53}-Z_{14}Z_{63}$

$Z_{66}=Z_{56}-L_{63}$

$Z_{67}=Z_{57}-L_{63}$

$Z_{68}=-Z_{49}-L_{63}Z_{14}$

$Z_{69}=Z_{50}-L_{63}Z_{15}$

$Z_{70}=Z_{64}u_1+Z_{65}u_2+Z_{66}u_3+Z_{67}u_4-L_{63}u_5$
$Z_{71}=Z_{68}u_1+Z_{69}u_2$
$Z_{72}=-Z_{65}u_1+Z_{64}u_2+Z_{58}u_3+Z_{59}u_4$
$Z_{73}=-(s_2s_3Z_3+c_2c_3Z_2)/s_2^2$
$Z_{74}=(c_2s_2Z_2-s_2c_3Z_3)/s_2^2$
$Z_{75}=Z_{73}u_1+Z_{74}u_2$
$Z_{76}=Z_{73}c_2-Z_2Z_4s_2$
$Z_{77}=Z_{74}c_2-Z_2Z_5s_2$
$Z_{78}=-(u_1s_3+u_2c_3)Z_3$
$Z_{79}=(u_1c_3-u_2s_3)Z_3$
$Z_{80}=Z_{76}u_1+Z_{77}u_2$
$Z_{81}=(c_4u_2-s_4u_1)u_4$
$Z_{82}=-(c_4u_1+s_4u_2)u_4$
$Z_{83}=-s_{45}(u_4+u_5)$
$Z_{84}=c_{45}(u_4+u_5)$
$Z_{85}=Z_{83}u_1+Z_{84}u_2$
$Z_{86}=-Z_{84}u_1+Z_{83}u_2$
$Z_{87}=Z_{83}c_6-Z_{14}s_6u_6$
$Z_{88}=Z_{84}c_6-Z_{15}s_6u_6$
$Z_{89}=Z_{83}s_6+Z_{14}c_6u_6$
$Z_{90}=Z_{84}s_6+Z_{15}c_6u_6$
$Z_{91}=Z_{87}u_1+Z_{88}u_2+Z_{18}c_6u_6$
$Z_{92}=Z_{89}u_1+Z_{90}u_2+Z_{18}s_6u_6$
$Z_{93}=-Z_{84}u_1+Z_{83}u_2$
$Z_{94}=L_{13}(Z_{73}u_1+Z_{74}u_2)$
$Z_{95}=-Z_1Z_{29}$
$Z_{96}=L_{22}Z_{76}$
$Z_{97}=L_{22}Z_{77}$
$Z_{98}=L_{22}s_3Z_3$
$Z_{99}=L_{22}c_3Z_3$
$Z_{100}=Z_{96}u_1+Z_{97}u_2+Z_7Z_{35}$
$Z_{101}=Z_{10}Z_{34}-Z_6Z_{35}$
$Z_{102}=Z_{98}u_1+Z_{99}u_2-Z_7Z_{34}$
$Z_{103}=Z_{38}u_2-Z_{37}u_3$
$Z_{104}=Z_{36}u_3-Z_{38}u_1$
$Z_{105}=Z_{37}u_1-Z_{36}u_2$

$Z_{106}=-(L_{33}+L_{43})c_4u_4$

$Z_{107}=-(L_{33}+L_{43})s_4u_4$

$Z_{108}=L_1c_4u_4$

$Z_{109}=-L_1s_4u_4$

$Z_{110}=L_{41}c_4u_4$

$Z_{111}=-L_{41}s_4u_4$

$Z_{112}=Z_{106}u_1+Z_{107}u_2+Z_{108}u_3+(Z_{12}Z_{47}-Z_{13}Z_{46})$

$Z_{113}=-Z_{107}u_1+Z_{106}u_2+Z_{109}u_3+(Z_{13}Z_{45}-Z_{11}Z_{47})$

$Z_{114}=Z_{110}u_1-Z_{111}u_2+(Z_{11}Z_{46}-Z_{12}Z_{45})$

$Z_{115}=Z_{84}Z_{51}-(L_{33}+L_{43})(u_4+u_5)c_{45}$

$Z_{116}=-Z_{83}Z_{51}+(Z_{107}-Z_{39}u_5)c_5-(-Z_{106}+Z_{40}u_5)s_5$

$Z_{117}=L_1c_{45}(u_4+u_5)+L_2c_5u_5$

$Z_{118}=L_2c_5u_5$

$Z_{119}=L_2c_4u_4$

$Z_{120}=-L_2s_4u_4$

$Z_{121}=-L_1s_4c_5u_4-Z_{48}s_5u_5-Z_{108}s_5-Z_{41}c_5u_5$

$Z_{122}=-L_2s_5u_5$

$Z_{125}=Z_{115}u_1+Z_{116}u_2+Z_{117}u_3+Z_{118}u_4+Z_{18}Z_{62}-Z_{19}Z_{61}$

$Z_{126}=-Z_{119}u_1+Z_{120}u_2+Z_{19}Z_{60}-Z_{17}Z_{62}$

$Z_{127}=-Z_{116}u_1+Z_{115}u_2+Z_{121}u_3+Z_{122}u_4+Z_{17}Z_{61}-Z_{18}Z_{60}$

$Z_{128}=Z_{84}Z_{63}-(L_{33}+L_{43})(u_4+u_5)c_{45}$

$Z_{129}=-Z_{83}Z_{63}+(Z_{107}-Z_{39}u_5)c_5-(-Z_{106}+Z_{40}u_5)s_5$

$Z_{130}=-Z_{119}-L_{63}Z_{83}$

$Z_{131}=Z_{120}-L_{63}Z_{84}$

$Z_{132}=Z_{18}Z_{72}-Z_{19}Z_{71}$

$Z_{133}=Z_{19}Z_{70}-Z_{17}Z_{72}$

$Z_{134}=Z_{17}Z_{71}-Z_{18}Z_{70}$

$Z_{135}=Z_{128}u_1+Z_{129}u_2+Z_{117}u_3+Z_{118}u_4+Z_{132}$

$Z_{136}=Z_{130}u_1+Z_{131}u_2+Z_{133}$

$Z_{137}=-Z_{129}u_1+Z_{128}u_2+Z_{121}u_3+Z_{122}u_4+Z_{134}$

$Z_{152}=-A_2Z_4$

$Z_{153}=-m_AZ_{27}$

$Z_{154}=Z_{152}Z_4+Z_{153}Z_{27}$

$Z_{155}=Z_{152}Z_5+Z_{153}Z_{28}$

$Z_{156}=Z_{152}Z_{75}+Z_{153}Z_{94}$

$Z_{157}=-A_2Z_5$

$Z_{158}=-m_A Z_{28}$

$Z_{159}=Z_{157}Z_4+Z_{158}Z_{27}$

$Z_{160}=Z_{157}Z_5+Z_{158}Z_{28}$

$Z_{161}=Z_{157}Z_{75}+Z_{158}Z_{94}$

$Z_{162}=(B_3-B_2)Z_7Z_{10}$

$Z_{163}=(B_1-B_3)Z_6Z_{10}$

$Z_{164}=(B_2-B_1)Z_6Z_7$

$Z_{165}=-(B_1c_3{}^2+B_2s_3{}^2+B_3Z_8{}^2+m_BZ_{30}{}^2+m_BZ_{32}{}^2)$

$Z166=-(-B_1c_3s_3+B_2c_3s_3+B_3Z_8Z_9+m_BZ_{30}Z_{31}+m_BZ_{32}Z_{33})$

$Z167=-(Z_{162}c_3+Z_{163}s_3+Z_{164}Z_8+B_1c_3Z_{78}+B_2s_3Z_{79}+B_3Z_8Z_{80}$
$+m_BZ_{30}Z_{100}+m_BZ_{32}Z_{102})$

$Z_{168}=-(-B_1s_3c_3+B_2c_3s_3+B_3Z_8Z_9+m_BZ_{30}Z_{31}+m_BZ_{32}Z_{33})$

$Z_{169}=-(B_1s_3{}^2+B_2c_3{}^2+B_3Z_9{}^2+m_BZ_{31}{}^2+m_BZ_{33}{}^2)$

$Z_{170}=-(-B_1s_3Z_{78}-Z_{162}s_3+B_2C_3Z_{79}+Z_{163}c_3+B_3Z_9Z_{80}+Z_{164}Z_9$
$+m_BZ_{31}Z_{100}+m_BZ_{33}Z_{102})$

$Z_{171}=(C_3-C_2)u_2u_3$

$Z_{172}=(C_1-C_3)u_1u_3$

$Z_{173}=(C_2-C_1)u_1u_2$

$Z_{174}=-C_1-m_CL_{33}{}^2$

$Z_{175}=m_CL_{33}L_{31}$

$Z_{176}=-Z_{171}+m_CL_{33}Z_{104}$

$Z_{178}=-C_2-m_CL_{31}{}^2-m_CL_{33}{}^2$

$Z_{179}=-Z_{172}-m_CL_{33}Z_{103}+m_CL_{31}Z_{105}$

$Z_{180}=m_CL_{31}L_{33}$

$Z_{181}=-C_3-m_CL_{31}{}^2$

$Z_{182}=-Z_{173}-m_CL_{31}Z_{104}$

$Z_{183}=(D_3-D_2)Z_{12}Z_{13}$

$Z_{184}=(D_1-D_3)Z_{11}Z_{13}$

$Z_{185}=(D_2-D_1)Z_{11}Z_{12}$

$Z_{186}=-(D_1c_4{}^2+D_2s_4{}^2+m_D(Z_{39}{}^2+Z_{40}{}^2+Z_{43}{}^2))$

$Z_{187}=(D_2-D_1)c_4s_4+m_DZ_{43}Z_{44}$

$Z_{188}=m_DZ_{39}Z_{41}+m_DZ_{40}Z_{42}$

$Z_{189}=-c_4(D_1Z_{81}+Z_{183})+s_4(D_2Z_{82}+Z_{184})+m_D(Z_{39}Z_{112}+Z_{40}Z_{113}$
$-Z_{43}Z_{114})$

$Z_{189A}=-D_1s_4{}^2-D_2c_4{}^2-m_D(Z_{40}{}^2+Z_{39}{}^2+Z_{44}{}^2)$

$Z_{190}=-m_D(Z_{40}Z_{41}-Z_{39}Z_{42})$

$Z_{191}=-s_4(D_1Z_{81}+Z_{183})-c_4(D_2Z_{82}+Z_{184})+mD(-Z_{40}Z_{112}$
$+Z_{39}Z_{113}+Z_{44}Z_{114})$
$Z_{192}=-D_3-m_D(Z_{41}{}^2+Z_{42}{}^2)$
$Z_{193}=-Z_{185}-m_D(Z_{41}Z_{112}+Z_{42}Z_{113})$
$Z_{194}=m_DZ_{40}L_{41}$
$Z_{195}=m_DZ_{39}L_{41}$
$Z_{196}=-D_3-m_DL_{41}Z_{42}$
$Z_{197}=-D_3-m_DL_{41}{}^2$
$Z_{198}=-Z_{185}-m_DL_{41}Z_{113}$
$Z_{199}=(E_3-E_2)Z_{18}Z_{19}$
$Z_{200}=(E_1-E_3)Z_{17}Z_{19}$
$Z_{201}=(E_2-E_1)Z_{17}Z_{18}$
$Z_{202}=-E_1Z_{14}{}^2-E_3Z_{16}2-m_E(Z_{54}{}^2+Z_{49}{}^2+Z_{55}{}^2)$
$Z_{203}=-E_1Z_{14}Z_{15}-E_3Z_{14}Z_{16}-m_E(Z_{54}Z_{55}-Z_{49}Z_{50}-Z_{54}Z_{55})$
$Z_{204}=-m_E(Z_{54}Z_{56}-Z_{55}Z_{58})$
$Z_{205}=-m_E(Z_{54}Z_{57}-Z_{55}Z_{59})$
$Z_{206}=-Z_{14}(E_1Z_{85}+Z_{199})-Z_{16}(E_3Z_{86}+Z_{201})$
$-m_E(Z_{54}Z_{125}-Z_{49}Z_{126}-Z_{55}Z_{127})$
$Z_{207}=-E_1Z_{15}{}^2-E_3Z_{14}{}^2-m_E(Z_{55}{}^2+Z_{50}{}^2+Z_{54}{}^2)$
$Z_{208}=-m_E(Z_{55}Z_{56}+Z_{54}Z_{58})$
$Z_{209}=-m_E(Z_{55}Z_{57}+Z_{54}Z_{59})$
$Z_{210}=-Z_{15}(E_1Z_{85}+Z_{199})-Z_{14}(E_3Z_{86}+Z_{201})$
$-m_E(Z_{55}Z_{125}+Z_{50}Z_{126}+Z54*Z127)$
$Z_{211}=-E_2-m_E(Z_{56}{}^2+Z_{58}{}^2)$
$Z_{212}=-E_2-m_E(Z_{56}Z_{57}+Z_{58}Z_{59})$
$Z_{213}=Z_{200}-m_E(Z_{56}Z_{125}+Z_{58}Z_{127})$
$Z_{214}=-E_2-m_E(Z_{57}{}^2+Z_{59}{}^2)$
$Z_{215}=Z_{200}-m_E(Z_{57}Z_{125}+Z_{59}Z_{127})$
$Z_{216}=(F_3-F_2)Z_{25}Z_{26}$
$Z_{217}=(F_1-F_3)Z_{24}Z_{26}$
$Z_{218}=(F_2-F_1)Z_{24}Z_{25}$
$Z_{219}=-(F_1Z_{20}{}^2+F_2Z_{22}{}^2+F_3Z_{16}{}^2+m_F(Z_{64}{}^2+Z_{68}{}^2+Z_{65}{}^2))$
$Z_{220}=-(F_1Z_{20}Z_{21}+F_2Z_{22}Z_{23}+F_3Z_{14}Z_{16}+m_FZ_{68}Z_{69})$
$Z_{221}=F_1Z_{20}s_6-F_2Z_{22}c_6-m_F(Z_{64}Z_{66}-Z_{65}Z_{58})$
$Z_{222}=F_1Z_{20}s_6-F_2Z_{22}c_6-m_F(Z_{64}Z_{67}-Z_{65}Z_{59})$
$Z_{223}=F_1Z_{20}s_6-F_2Z_{22}c_6+m_FZ_{64}L_{63}$

$Z_{224}=-Z_{20}(F_1Z_{91}+Z_{216})+Z_{22}(-F_2Z_{92}+Z_{217})-Z_{16}(F_3Z_{93}+Z_{218})$
$\quad -m_F(Z_{64}Z_{135}+Z_{68}Z_{136}-Z_{65}Z_{137})$

$Z_{225}=-(F_1Z_{21}{}^2+F_2Z_{23}{}^2+F_3Z_{14}{}^2+m_F(Z_{65}{}^2+Z_{69}{}^2+Z_{64}{}^2))$

$Z_{226}=F_1Z_{21}s_6-F_2Z_{23}c_6-m_F(Z_{65}Z_{66}+Z_{64}Z_{58})$

$Z_{227}=F_1Z_{21}s_6-F_2Z_{23}c_6-m_F(Z_{65}Z_{67}+Z_{59}Z_{64})$

$Z_{228}=F_1Z_{21}s_6-F_2Z_{23}c_6+m_FL_{63}Z_{65}$

$Z_{229}=-Z_{21}(F_1Z_{91}+Z_{216})+Z_{23}(Z_{217}-F_2Z_{92})-Z_{14}(F_3Z_{93}+Z_{218})$
$\quad -m_F(Z_{65}Z_{135}+Z_{69}Z_{136}+Z_{64}Z_{137})$

$Z_{230}=-(F_1s_6{}^2+F_2c_6{}^2+m_F(Z_{66}{}^2+Z_{58}{}^2))$

$Z_{231}=-(F_1s_6{}^2+F_2c_6{}^2+m_F(Z_{66}Z_{67}+Z_{58}Z_{59}))$

$Z_{232}=-(F_1s_6{}^2+F_2c_6{}^2-m_FL_{63}Z_{66})$

$Z_{234}=s_6(F_1Z_{91}+Z_{216})+c_6(Z_{217}-F_2Z_{92})-m_F(Z_{66}Z_{135}+Z_{58}Z_{137})$

$Z_{235}=-(F_1s_6{}^2+F_2c_6{}^2+m_F(Z_{67}{}^2+Z_{59}{}^2))$

$Z_{236}=-(F_1s_6{}^2+F_2c_6{}^2-m_FL_{63}Z_{67})$

$Z_{237}=s_6(F_1Z_{91}+Z_{216})-c_6(F_2Z_{92}-Z_{217})-m_F(Z_{67}Z_{135}+Z_{59}Z_{137})$

$Z_{238}=-F_1s_6{}^2-F_2c_6{}^2-m_FL_{63}{}^2$

$Z_{239}=s_6(F_1Z_{91}+Z_{216})+c_6(Z_{217}-F_2Z_{92})+m_FL_{63}Z_{135}$

$Z_{240}=-F_3Z_{16}$

$Z_{241}=-F_3Z_{14}$

$Z_{242}=-F_3Z_{93}-Z_{218}$

$Z_{243}=Z_{156}+Z_{167}+Z_{176}+Z_{189}+Z_{206}+Z_{224}$

$Z_{244}=Z_{161}+Z_{170}+Z_{179}+Z_{191}+Z_{210}+Z_{229}$

$Z_{245}=Z_{182}+Z_{193}+Z_{213}+Z_{234}$

$Z_{246}=Z_{198}+Z_{215}+Z_{237}$

$Z_{247}=Z_{200}+Z_{239}$

$Z_{248}=Z_{242}$

$Z_{251}=m_As_1$

$Z_{252}=Z_{251}Z_{27}$

$Z_{253}=Z_{251}Z_{28}$

$Z_{254}=m_Bs_1c_2$

$Z_{255}=-m_Bs_1s_2$

$Z_{256}=Z_{30}Z_{254}+Z_{32}Z_{255}$

$Z_{257}=Z_{31}Z_{254}+Z_{33}Z_{255}$

$Z_{258}=c1c_3-s_1c_2s_3$

$Z_{259}=-m_CL_{33}Z_{258}$

$Z_{260}=m_C(L_{33}(s_1c_2c_3+c_1s_3)+L_{31}s_1s_2)$

$Z_{261}=m_C L_{31} Z_{258}$

$Z_{262}=s_1 c_2 c_{34}+c_1 s_{34}$

$Z_{263}=c_1 c_{34}-s_1 c_2 s_{34}$

$Z_{264}=-s_1 s_2$

$Z_{265}=-m_D(Z_{39}Z_{262}+Z_{40}Z_{263}-Z_{43}Z_{264})$

$Z_{266}=m_D(Z_{40}Z_{262}-Z_{39}Z_{263}-Z_{44}Z_{264})$

$Z_{267}=m_D(Z_{41}Z_{262}+Z_{42}Z_{263})$

$Z_{268}=m_D L_{41} Z_{263}$

$Z_{269}=s_1 c_2 c_{345}+c_1 s_{345}$

$Z_{270}=s_1 s_2$

$Z_{271}=c_1 c_{345}-s_1 c_2 s_{345}$

$Z_{272}=m_E(Z_{54}Z_{269}-Z_{49}Z_{270}-Z_{55}Z_{271})$

$Z_{273}=m_E(Z_{55}Z_{269}+Z_{50}Z_{270}+Z_{54}Z_{271})$

$Z_{274}=m_E(Z_{56}Z_{269}+Z_{58}Z_{271})$

$Z_{275}=m_E(Z_{57}Z_{269}+Z_{59}Z_{271})$

$Z_{276}=m_F(Z_{64}Z_{269}+Z_{68}Z_{270}-Z_{65}Z_{271})$

$Z_{277}=m_F(Z_{65}Z_{269}+Z_{69}Z_{270}+Z_{64}Z_{271})$

$Z_{278}=m_F(Z_{66}Z_{269}+Z_{58}Z_{271})$

$Z_{279}=m_F(Z_{67}Z_{269}+Z_{59}Z_{271})$

$Z_{280}=-m_F L_{63} Z_{269}$

$Z_{281}=g(Z_{252}+Z_{256}+Z_{259}+Z_{265}+Z_{272}+Z_{276})$

$Z_{282}=g(Z_{253}+Z_{257}+Z_{260}+Z_{266}+Z_{273}+Z_{277})$

$Z_{283}=g(Z_{261}+Z_{267}+Z_{274}+Z_{278})$

$Z_{284}=g(Z_{268}+Z_{275}+Z_{279})$

$Z_{285}=gZ_{280}$

$X_{11}=Z_{154}+Z_{165}+Z_{174}+Z_{186}+Z_{202}+Z_{219}$

$X_{12}=Z_{155}+Z_{166}+Z_{187}+Z_{203}+Z_{220}$

$X_{13}=Z_{175}+Z_{188}+Z_{204}+Z_{221}$

$X_{14}=Z_{194}+Z_{205}+Z_{222}$

$X_{15}=Z_{223}$

$X_{16}=Z_{240}$

$X_{21}=X_{12}$

$X_{22}=Z_{160}+Z_{169}+Z_{178}+Z_{189A}+Z_{207}+Z_{225}$

$X_{23}=Z_{190}+Z_{208}+Z_{226}$

$X_{24}=Z_{195}+Z_{209}+Z_{227}$

$X_{25}=Z_{228}$

$X_{26}=Z_{241}$

$X_{31}=X_{13}$

$X_{32}=X_{23}$

$X_{33}=Z_{181}+Z_{192}+Z_{211}+Z_{230}$

$X_{34}=Z_{196}+Z_{212}+Z_{231}$

$X_{35}=-E_2+Z_{232}$

$X_{36}=0.0$

$X_{41}=X_{14}$

$X_{42}=X_{24}$

$X_{43}=X_{34}$

$X_{44}=Z_{197}+Z_{214}+Z_{235}$

$X_{45}=-E_2+Z_{236}$

$X_{46}=0.0$

$X_{51}=X_{15}$

$X_{52}=X_{25}$

$X_{53}=X_{35}$

$X_{54}=X_{45}$

$X_{55}=-E_2+Z_{238}$

$X_{56}=0.0$

$X_{61}=X_{16}$

$X_{62}=X_{26}$

$X_{63}=X_{63}$

$X_{64}=X_{46}$

$X_{65}=X_{56}$

$X_{66}=-F_3$

APPENDIX 2. PROGRAM TORQUE

```
C     ************************************************************
C     *                                                          *
C     *              PROGRAM      TORQUE                         *
C     *                                                          *
C     *           WRITTEN BY: TINGLIN NIE                        *
C     *           DATE WRITTEN: OCT. 1986                        *
C     *                                                          *
C     *   THIS PROGRAM IS TO CALCULATE THE TORQUES APPLIED TO    *
C     *   EACH OF THE SIX LINKS OF INTELLEDEX 605 ROBOT THAT     *
C     *   ARE NEEDED TO CARRY OUT A SPECIFIED MOTION.            *
C     *                                                          *
C     ************************************************************
C
C
      PROGRAM TORQUE
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,N,NPRINT
      COMMON/TANDR/TIME,TSPAN
      COMMON/COORD/Q1,Q2,Q3,Q4,Q5,Q6,QP1,QP2,QP3,QP4,QP5,QP6
      COMMON/ACCEL/QDP1,QDP2,QDP3,QDP4,QDP5,QDP6
      COMMON/GSPD/U1,U2,U3,U4,U5,U6,UP1,UP2,UP3,UP4,UP5,UP6
      COMMON/STRDT/L12,L13,L22,L31,L33,L41,L43,L52,L63,L1,L2
      COMMON/IPTDT/AM,BM,CM,DM,EM,FM,AI1,AI2,AI3,BI1,BI2,BI3,
     .        CI1,CI2,CI3,DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3
      COMMON/OUTT/T1,T2,T3,T4,T5,T6
      OPEN (5,FILE='DATA1')
      OPEN (8,FILE='OUT1')
      OPEN(9,FILE='1OUT1')
C
C   INPUT THE VALUES OF MASSES, INERTIAS, MASS CENTER COORDINATES
C
      READ (5,100)AM,BM,CM,DM,EM,FM
  100 FORMAT (6F7.4)
      READ (5,100) AI1,AI2,AI3,BI1,BI2,BI3
      READ (5,100) CI1,CI2,CI3,DI1,DI2,DI3
      READ (5,100) EI1,EI2,EI3,FI1,FI2,FI3
      READ (5,110) L12,L13,L22,L31,L33,L41,L43,L52,L63
  110 FORMAT (9F7.4)
      READ (5,120) L1,L2
  120 FORMAT (2F7.4)
      READ (5,130) TSPAN
      READ (5,130) TSTEP
  130 FORMAT (F7.4)
      READ (5,135) NPRINT
  135 FORMAT (I4)
C
C   PRINT TITLE FOR THE OUTPUT
C
      WRITE (8,140)
  140 FORMAT (////,30X,16HTABLE OF TORQUES/20X,
     .      38HNEEDED TO CARRY OUT A SPECIFIED MOTION)
      WRITE (8,150)
  150 FORMAT (///,1X,19HTHE GIVEN DATA ARE:)
      WRITE (8,160)
  160 FORMAT (//,3X,23HMASS OF EACH LINK (Kg):)
```



```
      WRITE (8,170) AM,BM,CM,DM,EM,FM
  170 FORMAT (//,8X,5HMA = ,F7.4/8X,5HMB = ,F7.4/8X,5HMC = ,F7.4/
     .     8X,5HMD = ,F7.4/8X,5HME = ,F7.4/8X,5HMF = ,F7.4)
      WRITE (8,180) AI1,AI2,AI3,BI1,BI2,BI3,CI1,CI2,CI3,
     .            DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3
  180 FORMAT (//,3X,31HINERTIAS OF EACH LINK (Kg-m^2)://
     .        8X,5HA1 = ,F7.4,8X,5HA2 = ,F7.4,8X,5HA3 = ,F7.4/
     .        8X,5HB1 = ,F7.4,8X,5HB2 = ,F7.4,8X,5HB3 = ,F7.4/
     .        8X,5HC1 = ,F7.4,8X,5HC2 = ,F7.4,8X,5HC3 = ,F7.4/
     .        8X,5HD1 = ,F7.4,8X,5HD2 = ,F7.4,8X,5HD3 = ,F7.4/
     .        8X,5HE1 = ,F7.4,8X,5HE2 = ,F7.4,8X,5HE3 = ,F7.4/
     .        8X,5HF1 = ,F7.4,8X,5HF2 = ,F7.4,8X,5HF3 = ,F7.4)
      WRITE (8,190) L12,L13,L22,L31,L33,L41,L43,L52,L63
  190 FORMAT (//,3X,40HTHE COORDINATES OF MASS CENTERS (meter)://
     .        8X,6HL12 = ,F7.4/8X,6HL13 = ,F7.4/8X,6HL22 = ,F7.4/
     .        8X,6HL31 = ,F7.4/8X,6HL33 = ,F7.4/8X,6HL41 = ,F7.4/
     .        8X,6HL43 = ,F7.4/8X,6HL52 = ,F7.4/8X,6HL63 = ,F7.4)
      WRITE (8,200) L1,L2
  200 FORMAT (8X,6HL1  = ,F7.4/8X,6HL2  = ,F7.4)
      WRITE (8,210)TSPAN,TSTEP
  210 FORMAT (//,3X,11HOTHER DATA://
     .        8X,12HTIME SPAN = ,F8.4,8X,
     .        12HTIME STEP = ,F8.4)
      WRITE (8,220)
  220 FORMAT (//,3X,38HTHE TORQUES APPLIED TO EACH LINK (Nm):)
      WRITE (8,230)
  230 FORMAT (//,1X,4HTIME,6X,2HT1,9X,2HT2,9X,2HT3,9X,2HT4,
     .          9X,2HT5,9X,2HT6//)
C
C   MAIN PROGRAM TO CALL THREE DIFFERENT SUBROUTINES
C
      TIME=0.0
      N=IDINT(TSPAN/TSTEP)
      N=N+2
      DO 250 I=1,N
C   TO CALCULATE GENERALIZED COORDINATES
      CALL GCOORD
C   TO CALCULATE GENERALIZED SPEEDS
      CALL GSPEED
C   TO CALCULATE THE ACTIVE FORCES, INERTIA FORCES AND
C   FINALLY CALCULATE THE TORQUES
      CALL CLCLTN
      WRITE (9,240) TIME,T1,T2,T3,T4,T5,T6
  240 FORMAT (F5.3,1X,6G16.8)
      IF(I.EQ.1 .OR. MOD(I,NPRINT).EQ.0) THEN
      WRITE(8,235) TIME,T1,T2,T3,T4,T5,T6
  235 FORMAT(F5.3,1X,6F11.5)
      ENDIF
      TIME=TIME+TSTEP
  250 CONTINUE
      STOP
      END
C
C   THIS SUBROUTINE IS TO CALCULATE THE GENERALIZED COOR-
C   OINATES, ANGULAR VELOCITIES AND ANGULAR ACCELERATIONS
```

```
C
      SUBROUTINE GCOORD
      IMPLICIT DOUBLE PRECISION (A-Z)
      COMMON/TANOR/TIME,TSPAN
      COMMON/COORD/Q1,Q2,Q3,Q4,Q5,Q6,QP1,QP2,QP3,QP4,QP5,QP6
      COMMON/ACCEL/QDP1,QDP2,QDP3,QDP4,QDP5,QDP6
C
      T=TIME
      TF=TSPAN
      PI=DACOS(-1.0D0)
      QQ=T-TF*DSIN(2*PI*T/TF)/(2*PI)
      Q1=QQ*PI/(3.*TF)
      Q2=PI/2.-QQ*PI/(6.*TF)
      Q3=Q1
      Q4=Q1
      Q5=Q1
      Q6=Q1
      QP1=PI*(1.-DCOS(2.*PI*T/TF))/(TF*3.)
      QP2=-QP1/2.
      QP3=QP1
      QP4=QP1
      QP5=QP1
      QP6=QP1
      QDP1=2.*PI**2/(3.*TF**2)*DSIN(2.*PI*T/TF)
      QDP2=-QDP1/2.
      QDP3=QDP1
      QDP4=QDP1
      QDP5=QDP1
      QDP6=QDP1
      RETURN
      END
C
C   THIS SUBROUTINE IS TO CALCULATE THE GENERALIZED SPEEDS
C
      SUBROUTINE GSPEED
      IMPLICIT DOUBLE PRECISION (A-Z)
      COMMON/COORD/Q1,Q2,Q3,Q4,Q5,Q6,QP1,QP2,QP3,QP4,QP5,QP6
      COMMON/ACCEL/QDP1,QDP2,QDP3,QDP4,QDP5,QDP6
      COMMON/GSPD/U1,U2,U3,U4,U5,U6,UP1,UP2,UP3,UP4,UP5,UP6
C
      S2=DSIN(Q2)
      S3=DSIN(Q3)
      C2=DCOS(Q2)
      C3=DCOS(Q3)
      U1=QP1*S2*C3-QP2*S3
      U2=-QP1*S2*S3-QP2*C3
      U3=QP1*C2+QP3
      U4=QP4
      U5=QP5
      U6=QP6
      UP1=QDP1*S2*C3+QP1*QP2*C2*C3-QP1*QP3*S2*S3-QDP2*S3-QP2*QP3*C3
      UP2=-(QDP1*S2*S3+QP1*QP2*C2*S3+QP1*QP3*S2*C3+QDP2*C3-QP2*QP3*S3)
      UP3=QDP1*C2-QP1*QP2*S2+QDP3
      UP4=QDP4
      UP5=QDP5
```



```
      UP6=QDP6
C
      RETURN
      END
C
C   THIS SUB ROUTINE IS TO CALCULATE THE ACTIVE FORCES,
C   THE INERTIA FORCES, AND FINALLY CALCULATE THE TORQUES.
C
      SUBROUTINE CLCLTN
      IMPLICIT DOUBLE PRECISION (A-Z)
      COMMON/TANOR/TIME,TSPAN
      COMMON/COORD/Q1,Q2,Q3,Q4,Q5,Q6,QP1,QP2,QP3,QP4,QP5,QP6
      COMMON/ACCEL/QDP1,QDP2,QDP3,QDP4,QDP5,QDP6
      COMMON/GSPD/U1,U2,U3,U4,U5,U6,UP1,UP2,UP3,UP4,UP5,UP6
      COMMON/STROT/L12,L13,L22,L31,L33,L41,L43,L52,L63,L1,L2
      COMMON/IPTOT/AM,BM,CM,DM,EM,FM,AI1,AI2,AI3,BI1,BI2,BI3,
     ,         CI1,CI2,CI3,DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3
      COMMON/OUTT/T1,T2,T3,T4,T5,T6
C
C   DEFINE SYMBOLS
C
      G=9.81
      C1=DCOS(Q1)
      C2=DCOS(Q2)
      C3=DCOS(Q3)
      C4=DCOS(Q4)
      C5=DCOS(Q5)
      C6=DCOS(Q6)
      S1=DSIN(Q1)
      S2=DSIN(Q2)
      S3=DSIN(Q3)
      S4=DSIN(Q4)
      S5=DSIN(Q5)
      S6=DSIN(Q6)
      C34=DCOS(Q3+Q4)
      C45=DCOS(Q4+Q5)
      C345=DCOS(Q3+Q4+Q5)
      S34=DSIN(Q3+Q4)
      S45=DSIN(Q4+Q5)
      S345=DSIN(Q3+Q4+Q5)
C
C   CALCULATE INTERMEDIATE VARIABLE Z'S
C
      Z1=(U1*C3-U2*S3)/S2
      Z2=-U1*S3-U2*C3
      Z3=U3+(U2*S3-U1*C3)*C2/S2
      Z4=C3/S2
      Z5=-S3/S2
      Z6=U1*C3-U2*S3
      Z7=-Z2
      Z8=Z4*C2
      Z9=Z5*C2
      Z10=Z8*U1+Z9*U2
      Z11=U1*C4+U2*S4
      Z12=-U1*S4+U2*C4
```

```
Z13=U3+U4
Z14=C45
Z15=S45
Z16=-Z15
Z17=U1*Z14+U2*Z15
Z18=-Z13-U5
Z19=U1*Z16+U2*Z14
Z20=Z14*C6
Z21=Z15*C6
Z22=Z14*S6
Z23=Z15*S6
Z24=Z20*U1+Z21*U2+Z18*S6
Z25=-Z22*U1-Z23*U2+Z18*C6
Z26=Z16*U1+Z14*U2+U6
Z27=L13*Z4
Z28=L13*Z5
Z29=Z27*U1+Z28*U2
Z30=L22*Z8
Z31=L22*Z9
Z32=-L22*C3
Z33=L22*S3
Z34=Z30*U1+Z31*U2
Z35=Z32*U1+Z33*U2
Z36=L33*U2
Z37=L31*U3-L33*U1
Z38=-L31*U2
Z39=(L33+L43)*S4
Z40=(L33+L43)*C4
Z41=L1*S4
Z42=L1*C4+L41
Z43=L41*S4
Z44=L1+L41*C4
Z45=-Z39*U1+Z40*U2+Z41*U3
Z46=-Z40*U1-Z39*U2+Z42*U3+L41*U4
Z47=Z43*U1-Z44*U2
Z48=L1*C4+L2
Z49=L2*S4
Z50=L1+L2*C4
Z51=L33+L43-L52
Z52=Z39*C5+Z40*S5
Z53=Z40*C5-Z39*S5
Z54=Z15*Z51-Z52
Z55=Z53-Z14*Z51
Z56=Z41*C5+Z48*S5
Z57=L2*S5
Z58=Z48*C5-Z41*S5
Z59=L2*C5
Z60=Z54*U1+Z55*U2+Z56*U3+Z57*U4
Z61=-Z49*U1+Z50*U2
Z62=-Z55*U1+Z54*U2+Z58*U3+Z59*U4
Z63=L33+L43
Z64=Z15*Z63-Z52
Z65=Z53-Z14*Z63
Z66=Z56-L63
Z67=Z57-L63
```



```
Z68=-Z49-L63*Z14
Z69=Z50-L63*Z15
Z70=Z64*U1+Z65*U2+Z66*U3+Z67*U4-L63*U5
Z71=Z68*U1+Z69*U2
Z72=-Z65*U1+Z64*U2+Z58*U3+Z59*U4
Z73=-(S2*S3*Z3+C2*C3*Z2)/S2**2
Z74=(C2*S3*Z2-S2*C3*Z3)/S2**2
Z75=Z73*U1+Z74*U2
Z76=Z73*C2-Z2*Z4*S2
Z77=Z74*C2-Z2*Z5*S2
Z78=-(U1*S3+U2*C3)*Z3
Z79=(U1*C3-U2*S3)*Z3
Z80=Z76*U1+Z77*U2
Z81=(C4*U2-S4*U1)*U4
Z82=-(C4*U1+S4*U2)*U4
Z83=-S45*(U4+U5)
Z84=C45*(U4+U5)
Z85=Z83*U1+Z84*U2
Z86=-Z84*U1+Z83*U2
Z87=Z83*C6-Z14*S6*U6
Z88=Z84*C6-Z15*S6*U6
Z89=Z83*S6+Z14*C6*U6
Z90=Z84*S6+Z15*C6*U6
Z91=Z87*U1+Z88*U2+Z18*C6*U6
Z92=Z89*U1+Z90*U2+Z18*S6*U6
Z93=-Z84*U1+Z83*U2
Z94=L13*(Z73*U1+Z74*U2)
Z95=-Z1*Z29
Z96=L22*Z76
Z97=L22*Z77
Z98=L22*S3*Z3
Z99=L22*C3*Z3
Z100=Z96*U1+Z97*U2+Z7*Z35
Z101=Z10*Z34-Z6*Z35
Z102=Z98*U1+Z99*U2-Z7*Z34
Z103=Z38*U2-Z37*U3
Z104=Z36*U3-Z38*U1
Z105=Z37*U1-Z36*U2
Z106=-(L33+L43)*C4*U4
Z107=-(L33+L43)*S4*U4
Z108=L1*C4*U4
Z109=-L1*S4*U4
Z110=L41*C4*U4
Z111=-L41*S4*U4
Z112=Z106*U1+Z107*U2+Z108*U3+(Z12*Z47-Z13*Z46)
Z113=-Z107*U1+Z106*U2+Z109*U3+(Z13*Z45-Z11*Z47)
Z114=Z110*U1-Z111*U2+(Z11*Z46-Z12*Z45)
Z115=Z84*Z51-(L33+L43)*(U4+U5)*C45
Z116=-Z83*Z51+(Z107-Z39*U5)*C5-(-Z106+Z40*U5)*S5
Z117=L1*C45*(U4+U5)+L2*C5*U5
Z118=L2*C5*U5
Z119=L2*C4*U4
Z120=-L2*S4*U4
Z121=-L1*S4*C5*U4-Z48*S5*U5-Z108*S5-Z41*C5*U5
Z122=-L2*S5*U5
```

```
Z125=Z115*U1+Z116*U2+Z117*U3+Z118*U4+Z18*Z62-Z19*Z61
Z126=-Z119*U1+Z120*U2+Z19*Z60-Z17*Z62
Z127=-Z116*U1+Z115*U2+Z121*U3+Z122*U4+Z17*Z61-Z18*Z60
Z128=Z84*Z63-(L33+L43)*(U4+U5)*C45
Z129=-Z83*Z63+(Z107-Z39*U5)*C5-(-Z106+Z40*U5)*S5
Z130=-Z119-L63*Z83
Z131=Z120-L63*Z84
Z132=Z18*Z72-Z19*Z71
Z133=Z19*Z70-Z17*Z72
Z134=Z17*Z71-Z18*Z70
Z135=Z128*U1+Z129*U2+Z117*U3+Z118*U4+Z132
Z136=Z130*U1+Z131*U2+Z133
Z137=-Z129*U1+Z128*U2+Z121*U3+Z122*U4+Z134
Z152=-AI2*Z4
Z153=-AM*Z27
Z154=Z152*Z4+Z153*Z27
Z155=Z152*Z5+Z153*Z28
Z156=Z152*Z75+Z153*Z94
Z157=-AI2*Z5
Z158=-AM*Z28
Z159=Z157*Z4+Z158*Z27
Z160=Z157*Z5+Z158*Z28
Z161=Z157*Z75+Z158*Z94
Z162=(BI3-BI2)*Z7*Z10
Z163=(BI1-BI3)*Z6*Z10
Z164=(BI2-BI1)*Z6*Z7
Z165=-(BI1*C3**2+BI2*S3**2+BI3*Z8**2+BM*Z30**2+BM*Z32**2)
Z166=-(-BI1*C3*S3+BI2*C3*S3+BI3*Z8*Z9+BM*Z30*Z31+BM*Z32*Z33)
Z167=-(Z162*C3+Z163*S3+Z164*Z8+BI1*C3*Z78+BI2*S3*Z79+BI3*Z8*Z80
      +BM*Z30*Z100+BM*Z32*Z102)
Z168=-(-BI1*S3*C3+BI2*C3*S3+BI3*Z8*Z9+BM*Z30*Z31+BM*Z32*Z33)
Z169=-(BI1*S3**2+BI2*C3**2+BI3*Z9**2+BM*Z31**2+BM*Z33**2)
Z170=-(-BI1*S3*Z78-Z162*S3+BI2*C3*Z79+Z163*C3+BI3*Z9*Z80+Z164*Z9
      +BM*Z31*Z100+BM*Z33*Z102)
Z171=(CI3-CI2)*U2*U3
Z172=(CI1-CI3)*U1*U3
Z173=(CI2-CI1)*U1*U2
Z174=-CI1-CM*L33**2
Z175=CM*L33*L31
Z176=-Z171+CM*L33*Z104
Z178=-CI2-CM*L31**2-CM*L33**2
Z179=-Z172-CM*L33*Z103+CM*L31*Z105
Z180=CM*L31*L33
Z181=-CI3-CM*L31**2
Z182=-Z173-CM*L31*Z104
Z183=(DI3-DI2)*Z12*Z13
Z184=(DI1-DI3)*Z11*Z13
Z185=(DI2-DI1)*Z11*Z12
Z186=-(DI1*C4**2+DI2*S4**2+DM*(Z39**2+Z40**2+Z43**2))
Z187=(DI2-DI1)*C4*S4+DM*Z43*Z44
Z188=DM*Z39*Z41+DM*Z40*Z42
Z189=-C4*(DI1*Z81+Z183)+S4*(DI2*Z82+Z184)+DM*(Z39*Z112+Z40*Z113
      -Z43*Z114)
Z189A=-DI1*S4**2-DI2*C4**2-DM*(Z40**2+Z39**2+Z44**2)
Z190=-DM*(Z40*Z41-Z39*Z42)
```



```
Z191=-S4*(DI1*Z81+Z183)-C4*(DI2*Z82+Z184)+DM*(-Z40*Z112+Z39*Z113
      +Z44*Z114)
Z192=-DI3-DM*(Z41**2+Z42**2)
Z193=-Z185-DM*(Z41*Z112+Z42*Z113)
Z194=DM*Z40*L41
Z195=DM*Z39*L41
Z196=-DI3-DM*L41*Z42
Z197=-DI3-DM*L41**2
Z198=-Z185-DM*L41*Z113
Z199=(EI3-EI2)*Z18*Z19
Z200=(EI1-EI3)*Z17*Z19
Z201=(EI2-EI1)*Z17*Z18
Z202=-EI1*Z14**2-EI3*Z16**2-EM*(Z54**2+Z49**2+Z55**2)
Z203=-EI1*Z14*Z15-EI3*Z14*Z16-EM*(Z54*Z55-Z49*Z50-Z54*Z55)
Z204=-EM*(Z54*Z56-Z55*Z58)
Z205=-EM*(Z54*Z57-Z55*Z59)
Z206=-Z14*(EI1*Z85+Z199)-Z16*(EI3*Z86+Z201)-EM*(Z54*Z125-Z49*Z126
      -Z55*Z127)
Z207=-EI1*Z15**2-EI3*Z14**2-EM*(Z55**2+Z50**2+Z54**2)
Z208=-EM*(Z55*Z56+Z54*Z58)
Z209=-EM*(Z55*Z57+Z54*Z59)
Z210=-Z15*(EI1*Z85+Z199)-Z14*(EI3*Z86+Z201)-EM*(Z55*Z125+Z50*Z126
      +Z54*Z127)
Z211=-EI2-EM*(Z56**2+Z58**2)
Z212=-EI2-EM*(Z56*Z57+Z58*Z59)
Z213=Z200-EM*(Z56*Z125+Z58*Z127)
Z214=-EI2-EM*(Z57**2+Z59**2)
Z215=Z200-EM*(Z57*Z125+Z59*Z127)
Z216=(FI3-FI2)*Z25*Z26
Z217=(FI1-FI3)*Z24*Z26
Z218=(FI2-FI1)*Z24*Z25
Z219=-(FI1*Z20**2+FI2*Z22**2+FI3*Z16**2+FM*(Z64**2+Z68**2
      +Z65**2))
Z220=-(FI1*Z20*Z21+FI2*Z22*Z23+FI3*Z14*Z16+FM*Z68*Z69)
Z221=FI1*Z20*S6-FI2*Z22*C6-FM*(Z64*Z66-Z65*Z58)
Z222=FI1*Z20*S6-FI2*Z22*C6-FM*(Z64*Z67-Z65*Z59)
Z223=FI1*Z20*S6-FI2*Z22*C6+FM*Z64*L63
Z224=-Z20*(FI1*Z91+Z216)+Z22*(-FI2*Z92+Z217)-Z16*(FI3*Z93+Z218)
      -FM*(Z64*Z135+Z68*Z136-Z65*Z137)
Z225=-(FI1*Z21**2+FI2*Z23**2+FI3*Z14**2+FM*(Z65**2+Z69**2+Z64**2))
Z226=FI1*Z21*S6-FI2*Z23*C6-FM*(Z65*Z66+Z64*Z58)
Z227=FI1*Z21*S6-FI2*Z23*C6-FM*(Z65*Z67+Z59*Z64)
Z228=FI1*Z21*S6-FI2*Z23*C6+FM*L63*Z65
Z229=-Z21*(FI1*Z91+Z216)+Z23*(Z217-FI2*Z92)-Z14*(FI3*Z93+Z218)
      -FM*(Z65*Z135+Z69*Z136+Z64*Z137)
Z230=-(FI1*S6**2+FI2*C6**2+FM*(Z66**2+Z58**2))
Z231=-(FI1*S6**2+FI2*C6**2+FM*(Z66*Z67+Z58*Z59))
Z232=-(FI1*S6**2+FI2*C6**2-FM*L63*Z66)
Z234=S6*(FI1*Z91+Z216)+C6*(Z217-FI2*Z92)-FM*(Z66*Z135+Z58*Z137)
Z235=-(FI1*S6**2+FI2*C6**2+FM*(Z67**2+Z59**2))
Z236=-(FI1*S6**2+FI2*C6**2-FM*L63*Z67)
Z237=S6*(FI1*Z91+Z216)-C6*(FI2*Z92-Z217)-FM*(Z67*Z135+Z59*Z137)
Z238=-FI1*S6**2-FI2*C6**2-FM*L63**2
Z239=S6*(FI1*Z91+Z216)+C6*(Z217-FI2*Z92)+FM*L63*Z135
Z240=-FI3*Z16
```

```
      Z241=-FI3*Z14
      Z242=-FI3*Z93-Z218
C
      Z243=Z156+Z167+Z176+Z189+Z206+Z224
      Z244=Z161+Z170+Z179+Z191+Z210+Z229
      Z245=Z182+Z193+Z213+Z234
      Z246=Z198+Z215+Z237
      Z247=Z200+Z239
      Z248=Z242
C
C   CALCULATE THE GENERALIZED INERTIA FORCES
C
      X11=Z154+Z165+Z174+Z186+Z202+Z219
      X12=Z155+Z166+Z187+Z203+Z220
      X13=Z175+Z188+Z204+Z221
      X14=Z194+Z205+Z222
      X15=Z223
      X16=Z240
C
      X21=Z159+Z168+Z187+Z203+Z220
      X22=Z160+Z169+Z178+Z189A+Z207+Z225
      X23=Z190+Z208+Z226
      X24=Z195+Z209+Z227
      X25=Z228
      X26=Z241
C
      X31=Z180+Z188+Z204+Z221
      X32=Z190+Z208+Z226
      X33=Z181+Z192+Z211+Z230
      X34=Z196+Z212+Z231
      X35=-EI2+Z232
      X36=0.0
C
      X41=Z194+Z205+Z222
      X42=Z195+Z209+Z227
      X43=Z196+Z212+Z231
      X44=Z197+Z214+Z235
      X45=-EI2+Z236
      X46=0.0
C
      X51=Z223
      X52=Z228
      X53=-EI2+Z232
      X54=-EI2+Z236
      X55=-EI2+Z238
      X56=0.0
C
      X61=Z240
      X62=Z241
      X63=0.0
      X64=0.0
      X65=0.0
      X66=-FI3
C
      KI1=X11*UP1+X12*UP2+X13*UP3+X14*UP4+X15*UP5+X16*UP6+Z243
```



```
      KI2=X21*UP1+X22*UP2+X23*UP3+X24*UP4+X25*UP5+X26*UP6+Z244
      KI3=X31*UP1+X32*UP2+X33*UP3+X34*UP4+X35*UP5+X36*UP6+Z245
      KI4=X41*UP1+X42*UP2+X43*UP3+X44*UP4+X45*UP5+X46*UP6+Z246
      KI5=X51*UP1+X52*UP2+X53*UP3+X54*UP4+X55*UP5+X56*UP6+Z247
      KI6=X61*UP1+X62*UP2+X63*UP3+X64*UP4+X65*UP5+X66*UP6+Z248
C
C   CALCULATE THE GENERALIZED ACTIVE FORCES
C
      Z251=AM*S1
      Z252=Z251*Z27
      Z253=Z251*Z28
      Z254=BM*S1*C2
      Z255=-BM*S1*S2
      Z256=Z30*Z254+Z32*Z255
      Z257=Z31*Z254+Z33*Z255
      Z258=C1*C3-S1*C2*S3
      Z259=-CM*L33*Z258
      Z260=CM*(L33*(S1*C2*C3+C1*S3)+L31*S1*S2)
      Z261=CM*L31*Z258
      Z262=S1*C2*C34+C1*S34
      Z263=C1*C34-S1*C2*S34
      Z264=-S1*S2
      Z265=-DM*(Z39*Z262+Z40*Z263-Z43*Z264)
      Z266=DM*(Z40*Z262-Z39*Z263-Z44*Z264)
      Z267=DM*(Z41*Z262+Z42*Z263)
      Z268=DM*L41*Z263
      Z269=S1*C2*C345+C1*S345
      Z270=S1*S2
      Z271=C1*C345-S1*C2*S345
      Z272=EM*(Z54*Z269-Z49*Z270-Z55*Z271)
      Z273=EM*(Z55*Z269+Z50*Z270+Z54*Z271)
      Z274=EM*(Z56*Z269+Z58*Z271)
      Z275=EM*(Z57*Z269+Z59*Z271)
      Z276=FM*(Z64*Z269+Z68*Z270-Z65*Z271)
      Z277=FM*(Z65*Z269+Z69*Z270+Z64*Z271)
      Z278=FM*(Z66*Z269+Z58*Z271)
      Z279=FM*(Z67*Z269+Z59*Z271)
      Z280=-FM*L63*Z269
      Z281=G*(Z252+Z256+Z259+Z265+Z272+Z276)
      Z282=G*(Z253+Z257+Z260+Z266+Z273+Z277)
      Z283=G*(Z261+Z267+Z274+Z278)
      Z284=G*(Z268+Z275+Z279)
      Z285=G*Z280
C
C   CALCULATE THE TORQUES NEEDED TO CARRY OUT THE SPECIFIED MOTION
C
      T3=-(KI3+Z283)
      T4=-(KI4+Z284)
      T5=-(KI5+Z285)
      T6=-KI6
C
      V1=-KI1-Z281+T3*Z8
      V2=-KI2-Z282+T3*Z9
      DET=-1.0/S2
C
```

```
      T1=(V2*S3-V1*C3)/DET
      T2=(Z4*V2-Z5*V1)/DET
C
C    RETURN THE CALCULATED VALUES TO THE MAIN PROGRAM
C
      RETURN
      END
```

APPENDIX 3. PROGRAM ANGLE

```
C     ************************************************************
C     *                                                          *
C     *              PROGRAM       ANGLE                         *
C     *                                                          *
C     *              WRITTEN BY: TINGLIN NIE                     *
C     *              DATE WRITTEN: OCT. 1986                     *
C     *                                                          *
C     *   THIS PROGRAM IS TO FIND THE SOLUTION OF THE KANE'S DY- *
C     *   NAMICAL EQUATIONS FOR INTELLEDEX 605 ROBOT. WITH THE   *
C     *   GENERALIZED ACTIVE FORCES GIVEN, THIS PROGRAM WILL     *
C     *   SOLVE THE DYNAMICAL EQUATIONS FOR THE GENERALIZED      *
C     *   COORDINATES, THE ROTATION ANGLES Q1,Q2,Q3,Q4,Q5 AND Q6. *
C     *                                                          *
C     ************************************************************
C
C
      PROGRAM ANGLE
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,ITER,ITIME,ITMAX,J,M,MAXITR,N,NPRINT,TQTYPE
      DIMENSION A(12,12),V(12),Q0(6),QD0(6),U0(6),QOLD(12),Q(12),FN(12)
      DIMENSION TQ(12),QOLDEST(12)
      COMMON/TYPE/TQTYPE
      COMMON/TANOR/TIME,TSPAN
      COMMON/STEP/TSTEP
      COMMON/STRDT/L12,L13,L22,L31,L33,L41,L43,L52,L63,L1,L2
      COMMON/IPTDT/AM,BM,CM,DM,EM,FM,AI1,AI2,AI3,BI1,BI2,BI3,
     ,        CI1,CI2,CI3,DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3

      COMMON/EPSI/XESPI,FEPSI
      COMMON/OLD/QOLD,V,BTA,TQ,QOLDEST,ITIME
      OPEN (7,FILE='DATA2')
      OPEN (8,FILE='TOUT1')
      OPEN (9,FILE='AOUT1')
C
C   INPUT THE VALUES OF MASSES, INERTIAS, MASS CENTER COORDINATES
C
      READ (7,100)AM,BM,CM,DM,EM,FM
  100 FORMAT (6F7.4)
      READ (7,100) AI1,AI2,AI3,BI1,BI2,BI3
      READ (7,100) CI1,CI2,CI3,DI1,DI2,DI3
      READ (7,100) EI1,EI2,EI3,FI1,FI2,FI3
      READ (7,110) L12,L13,L22,L31,L33,L41,L43,L52,L63
  110 FORMAT (9F7.4)
      READ (7,120) L1,L2
  120 FORMAT (2F7.4)
      READ (7,130) TSPAN
      READ (7,130) TSTEP
  130 FORMAT (F7.4)
      READ (7,135) (Q0(I),I=1,6)
  135 FORMAT (6F12.9)
      READ (7,135) (QD0(I),I=1,6)
      READ (7,*) MAXITR,ERRMAX,XEPSI,FEPSI,NPRINT
  137 FORMAT (I5,3(F10.7,2X),I4)
C
C   PRINT TITLE FOR THE OUTPUT
```

1

```
C
      WRITE (9,140)
  140 FORMAT (////,25X,18HNUMERICAL SOLUTION/ 20X,
     ,        31H( THE GENERALIZED COORDINATES ))
      WRITE (9,150)
  150 FORMAT (///,1X,19HTHE GIVEN DATA ARE:)
      WRITE (9,160)
  160 FORMAT (//,3X,23HMASS OF EACH LINK (Kg):)
      WRITE (9,170) AM,BM,CM,DM,EM,FM
  170 FORMAT (//,8X,5HMA = ,F7.4/8X,5HMB = ,F7.4/8X,5HMC = ,F7.4/
     ,        8X,5HMD = ,F7.4/8X,5HME = ,F7.4/8X,5HMF = ,F7.4)
      WRITE (9,180) AI1,AI2,AI3,BI1,BI2,BI3,CI1,CI2,CI3,
     ,              DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3
  180 FORMAT (//,3X,31HINERTIAS OF EACH LINK (Kg-m^2)://
     ,        8X,5HA1 = ,F7.4,8X,5HA2 = ,F7.4,8X,5HA3 = ,F7.4/
C    ,        8X,5HB1 = ,F7.4,8X,5HB2 = ,F7.4,8X,5HB3 = ,F7.4/
     ,        8X,5HB1 = ,F7.4,8X,5HB2 = ,F7.4,8X,5HB3 = ,F7.4/
     ,        8X,5HC1 = ,F7.4,8X,5HC2 = ,F7.4,8X,5HC3 = ,F7.4/
     ,        8X,5HD1 = ,F7.4,8X,5HD2 = ,F7.4,8X,5HD3 = ,F7.4/
     ,        8X,5HE1 = ,F7.4,8X,5HE2 = ,F7.4,8X,5HE3 = ,F7.4/
     ,        8X,5HF1 = ,F7.4,8X,5HF2 = ,F7.4,8X,5HF3 = ,F7.4)
      WRITE (9,190) L12,L13,L22,L31,L33,L41,L43,L52,L63
  190 FORMAT (//,3X,40HTHE COORDINATES OF MASS CENTERS (meter)://
     ,        8X,6HL12 = ,F7.4/8X,6HL13 = ,F7.4/8X,6HL22 = ,F7.4/
     ,        8X,6HL31 = ,F7.4/8X,6HL33 = ,F7.4/8X,6HL41 = ,F7.4/
     ,        8X,6HL43 = ,F7.4/8X,6HL52 = ,F7.4/8X,6HL63 = ,F7.4)
      WRITE (9,200) L1,L2
  200 FORMAT (8X,6HL1  = ,F7.4/8X,6HL2  = ,F7.4)
      WRITE (9,210)TSPAN,TSTEP
  210 FORMAT (//,3X,11HOTHER DATA://
     ,        8X,12HTIME SPAN = ,F8.4,8X,
     ,        12HTIME STEP = ,F8.4)
      WRITE (9,213)
  213 FORMAT (//,3X,35HTHE GIVEN INITIAL VALUES (RAD) ARE:)
      WRITE (9,215) (I,Q0(I),I,QD0(I),I=1,6)
  215 FORMAT (/,6(8X,3HQ0(,I1,4H) = ,F9.6,8X,4HQD0(,I1,4H) = ,F9.6/))
      WRITE (9,220)
  220 FORMAT (//,3X,40HTHE GENERALIZED COORDINATES Q1-Q6 (RAD):)
      WRITE (9,230)
  230 FORMAT (//,1X,4HTIME,6X,2HQ1,9X,2HQ2,9X,2HQ3,9X,2HQ4,
     ,        9X,2HQ5,9X,2HQ6//)
      WRITE (*,*) 'PLEASE DECIDE THE METHOD:   '
      WRITE (*,*) 'EXPLICIT:  0      '
      WRITE (*,*) 'IMPLICIT:  1      '
      WRITE (*,*) 'CRANK-NELSON:   0---1 '
      READ (*,*) BTA
C
C   MAIN PROGRAM TO CALL THREE DIFFERENT SUBROUTINES
C
      TIME=0.0
      ITMAX=DINT(TSPAN/TSTEP+1)
      N=12
C
C CALL GUESS
C
```

2

```
      WRITE (*,*) 'PLEASE ENTER THE INITIAL GUESS FOR A MATRIX:'
      WRITE (*,*) 'AND VECTOR V:     '
      READ (*,*) AA,BB
      CALL GUESS(A,V,AA,BB)
C
C CALL UINTL
C
      CALL UINTL(Q0,QD0,U0)
C
      DO 240 I=1,6
      Q(I)=Q0(I)
      Q(I+6)=U0(I)
  240 CONTINUE
      WRITE (9,260) TIME,(Q(I),I=1,6)
      DO 244 I=1,N
      QOLDEST(I)=Q0(I)
  244 CONTINUE
C
      WRITE (*,*) 'PLEASE INDICATE THE DATA TYPE OF THE TORQUES'
      WRITE (*,*) 'YOU ARE GOING TO USE!'
      WRITE (*,*) 'IF USE FUNCTION,TYPE 1'
      WRITE (*,*) 'IF USE NUMERICAL VALUES, TYPE 2:'
      READ (*,*) TQTYPE
C
      DO 280 ITIME=1,ITMAX
      IF (TQTYPE .EQ. 1) GOTO 245
      READ (8,268) TIMET,(TQ(I),I=1,6)
  268 FORMAT (F5.3,1X,6G16.8)
  245 DO 250 I=1,N
      IF (ITIME .EQ. 1) THEN
      QOLD(I)=Q(I)
      Q(I)=Q(I)+V(I)*TSTEP
      ELSE
      QOLDEST(I)=QOLD(I)
      QOLD(I)=Q(I)
      Q(I)=2.0*TSTEP*V(I)+QOLDEST(I)
      ENDIF
  250 CONTINUE
C
C CALL SUBROUTINE QSOLVE TO FIND THE SOLUTION
C
      CALL QSOLVE (A,Q,FN,MAXITR,ERRMAX,ERRX,ERRF,ITER)
C
      ITER=MIN(ITER,MAXITR)
      TIME=TIME+TSTEP
C
C PRINT THE SOLUTION OR MESSAGE
C
      IF (MOD(ITIME,NPRINT) .EQ. 0) THEN
      WRITE (9,260) TIME,(Q(I),I=1,6)
  260 FORMAT (F5.3,1X,6(2X,F9.6))
      ENDIF
  280 CONTINUE
      STOP
      END
```



```
C
C
C THIS SUBROUTINE GUESS THE INITIAL VALUE OF MATRIX A AND VECTOR V
C
      SUBROUTINE GUESS (A,V,AA,BB)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,J,N
      DIMENSION A(12,12),V(12)
C
C INITIALIZATION
C
      N=12
      DO 20 I=1,N
      DO 10 J=1,N
      A(I,J)=0.0
   10 CONTINUE
      A(I,I)=AA
      V(I)=BB
   20 CONTINUE
      RETURN
      END
C
C
C THIS SUBROUTINE COMPUTE THE INITIAL VALUES OF THE GENERALIZED
C SPEEDS U0(1) THROUGH U0(6).
C
      SUBROUTINE UINTL (Q0,QD0,U0)
      IMPLICIT DOUBLE PRECISION (A-Z)
      DIMENSION Q0(6),QD0(6),U0(6)
C
C
      C2=DCOS(Q0(2))
      S2=DSIN(Q0(2))
      C3=DCOS(Q0(3))
      S3=DSIN(Q0(3))
C
C
      U0(1)=QD0(1)*S2*C3-QD0(2)*S3
      U0(2)=-QD0(1)*S2*S3-QD0(2)*C3
      U0(3)=QD0(1)*C2+QD0(3)
      U0(4)=QD0(4)
      U0(5)=QD0(5)
      U0(6)=QD0(6)
C
C
      RETURN
      END
C
C
C SUBROUTINE QSOLVE TO SOLVE THE 12 SIMULTANEOUS NONLINEAR
C DIFFERENTIAL EQUATIONS BY USING QUASI-NEWTON ALGORITHM.
C
      SUBROUTINE QSOLVE (A,X,F,MAXITR,ERRMAX,ERRX,ERRF,ITER)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,ITER,J,M,MAXITR,N
```

```
      COMMON/EPSI/XEPSI,FEPSI
      OIMENSION A(12,12),X(12),F(12),OF(12),Q(12,12),R(12,12)
C CALL SUBROUTINE CLCLTN TO FINO THE PREVIOUS VALUES OF FUNCTION
C
      CALL CLCLTN (X,F)
   15 FORMAT (' X/F: ',6E12.4)
C
C FINO THE INVERSE JACOBIAN ANO THE SOLUTION BY ITERATION
C
      M=1
      N=12
      OO 10 ITER=1,MAXITR
      ERRX=0.0
      ERRF=0.0
      ERRDF=0.0
C
      DO 20 I=1,N
      OX=0.0
      DO 30 J=1,N
      OX=OX-A(I,J)*F(J)
   30 CONTINUE
C
      X(I)=X(I)+OX
      IF (OABS(X(I)) .GT. XEPSI) THEN
      RELERR=OABS(DX/X(I))
      IF (RELERR .GT. ERRX) ERRX=RELERR
      ENOIF
   20 CONTINUE
C
      OO 40 I=1,N
      OF(I)=F(I)
   40 CONTINUE
C
C CALL CLCLTN TO FINO THE NEW VALUE OF THE FUNCTIONS
C
      CALL CLCLTN (X,F)
      OO 50 I=1,N
      DF(I)=F(I)-OF(I)
      IF (OABS(OF(I)).GT.ERROF) ERROF=DABS(DF(I))
      IF (DABS(F(I)).GT.ERRF) ERRF=OABS(F(I))
   50 CONTINUE
C
C COMPUTE THE ORHTOGONAL VECTORS Z AND R
C
      IF (ERRF .LT. ERRMAX) RETURN
      CALL ORTHO (OF,M,Q,R)
C
C
      IF (DABS(R(M,M)) .GT. ERROF*FEPSI) THEN
      OO 60 I=1,N
      OF(I)=0.0
      OO 70 J=1,N
      OF(I)=OF(I)+A(I,J)*F(J)
   70 CONTINUE
```



```
   60 CONTINUE
C
C UPDATE THE A MATRIX
C
      OO 80 I=1,N
      OO 80 J=1,N
      A(I,J)=A(I,J)-DF(I)*Q(J,M)/R(M,M)
   80 CONTINUE
      M=M+1
      ENDIF
   10 CONTINUE
C
      RETURN
      ENO
C
C
C THIS SUBROUTINE COMPUTES THE ORTHOGONAL BASIS FOR SPACE SPANNED
C BY THE INPUT VECTOR AND M-1 PREVIOUS VECTORS. THE QR OECOMPOSI-
C TION IS RETURNEO.
C
      SUBROUTINE ORTHO (B,M,Q,R)
      IMPLICIT OOUBLE PRECISION (A-Z)
      INTEGER N,M,I,J,K
      DIMENSION B(12),Q(12,12),R(12,12)
C
C
      N=12
      IF (M .LE. N) THEN
      IF (M .EQ. 1) THEN
C
C INITIALIZATION
C
      DO 10 I=1,N
      DO 20 J=1,N
      Q(I,J)=0.0
   20 CONTINUE
      Q(I,I)=1.0
   10 CONTINUE
      ENOIF
C
C TRANSFORM B VECTOR
C
      OO 30 I=1,N
      SUM=0.0
      DO 40 J=1,N
      SUM=SUM+Q(J,I)*B(J)
   40 CONTINUE
      R(I,M)=SUM
   30 CONTINUE
      IF (M .LT. N) THEN
C
C HOUSEHOLOER TRANSFORMATION
C
      RO=0.0
      DO 50 I=M,N
```

```
      RO=RO+R(I,M)*R(I,M)
   50 CONTINUE
      RO=DSQRT(RO)
      IF (R(M,M).LT.0.0) RO=-RO
      B(M)=R(M,M)+RO
      C=DSQRT(RO*B(M))
      IF (C.GT.0.0) THEN
      R(M,M)=-RO
      B(M)=B(M)/C
      DO 60 I=M+1,N
      B(I)=R(I,M)/C
      R(I,M)=0.0
   60 CONTINUE
C
C ACCUMULATE ORHTOGONAL TRANSFORMATIONS
C
      DO 70 I=1,N
      SUM=0.0
      DO 80 K=M,N
      SUM=SUM+Q(I,K)*B(K)
   80 CONTINUE
      DO 90 J=M,N
      Q(I,J)=Q(I,J)-SUM*B(J)
   90 CONTINUE
   70 CONTINUE
      ENDIF
      ENDIF
      ELSE
      M=N
      DO 100 I=1,N
      DO 110 J=2,N
      R(I,J-1)=R(I,J)
  110 CONTINUE
  100 CONTINUE
      DO 120 I=1,N
      SUM=0.0
      DO 130 J=1,N
      SUM=SUM+Q(J,I)*B(J)
  130 CONTINUE
      R(I,N)=SUM
  120 CONTINUE
      DO 140 K=1,N-1
      RO=DSQRT(R(K,K)*R(K,K)+R(K+1,K)*R(K+1,K))
      IF (R(K,K).LT.0.0) RO=-RO
      BK=R(K,K)+RO
      BKP=R(K+1,K)
      C=RO*BK
      IF (C.NE.0.0) THEN
      R(K,K)=-RO
      R(K+1,K)=0.0
      DO 150 J=K+1,N
      RO=BK*R(K,J)+BKP*R(K+1,J)
      R(K,J)=R(K,J)-RO*BK/C
      R(K+1,J)=R(K+1,J)-RO*BKP/C
  150 CONTINUE
```



```
      DO 160 I=1,N
      RO=Q(I,K)*BK+Q(I,K+1)*BKP
      Q(I,K)=Q(I,K)-RO*BK/C
      Q(I,K+1)=Q(I,K+1)-RO*BKP/C
  160 CONTINUE
      ENDIF
  140 CONTINUE
      ENDIF
      RETURN
      END
C
C
C THIS SUBROUTINE IS TO CALCULATE THE VALUES OF THE 12
C FUNCTIONS.
C
      SUBROUTINE CLCLTN (Q,FN)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,N,TQTYPE,ITIME
      DIMENSION Q(12),FN(12),QOLD(12),V(12),QM(12),TQ(12),KI(6),KA(6)
      DIMENSION QOLDEST(12)
      COMMON/TANOR/TIME,TSPAN
      COMMON/TYPE/TQTYPE
      COMMON/STEP/TSTEP
      COMMON/STROT/L12,L13,L22,L31,L33,L41,L43,L52,L63,L1,L2
      COMMON/IPTOT/AM,BM,CM,DM,EM,FM,AI1,AI2,AI3,BI1,BI2,BI3,
     .          CI1,CI2,CI3,DI1,DI2,DI3,EI1,EI2,EI3,FI1,FI2,FI3
      COMMON/OLD/QOLD,V,BTA,TQ,QOLDEST,ITIME
C
C ASSIGN Q(I) TO Q1,Q2,....,Q6 AND U1,U2,....,U6
C
      N=12
      DO 15 I=1,N
      QM(I)=(1-BTA)*QOLD(I)+Q(I)*BTA
   15 CONTINUE
      Q1=QM(1)
      Q2=QM(2)
      Q3=QM(3)
      Q4=QM(4)
      Q5=QM(5)
      Q6=QM(6)
      U1=QM(7)
      U2=QM(8)
      U3=QM(9)
      U4=QM(10)
      U5=QM(11)
      U6=QM(12)
C
C   DEFINE SYMBOLS
C
      G=9.81
      C1=DCOS(Q1)
      C2=DCOS(Q2)
      C3=DCOS(Q3)
      C4=DCOS(Q4)
      C5=DCOS(Q5)
```

```
      C6=DCOS(Q6)
      S1=DSIN(Q1)
      S2=DSIN(Q2)
      S3=DSIN(Q3)
      S4=DSIN(Q4)
      S5=DSIN(Q5)
      S6=DSIN(Q6)
      C34=DCOS(Q3+Q4)
      C45=DCOS(Q4+Q5)
      C345=DCOS(Q3+Q4+Q5)
      S34=DSIN(Q3+Q4)
      S45=DSIN(Q4+Q5)
      S345=DSIN(Q3+Q4+Q5)
C
C COMPUTE NEW VECTOR V(12)
C
      DO 25 I=1,N
      IF (ITIME .EQ. 1) THEN
      V(I)=(Q(I)-QOLD(I))/TSTEP
      ELSE
      V(I)=(3.*Q(I)-4.*QOLD(I)+QOLDEST(I))/(2.*TSTEP)
      ENDIF
   25 CONTINUE
C
C COMPUTE THE VALUES OF FUNCTIONS FN(1), FN(2),...,FN(6).
C
      FN(1)=V(1)-(U1*C3-U2*S3)/S2
      FN(2)=V(2)+U1*S3+U2*C3
      FN(3)=V(3)-U3-(U2*S3-U1*C3)*C2/S2
      FN(4)=V(4)-U4
      FN(5)=V(5)-U5
      FN(6)=V(6)-U6
C
C   CALCULATE INTERMEDIATE VARIABLE Z'S
C
      Z1=(U1*C3-U2*S3)/S2
      Z2=-U1*S3-U2*C3
      Z3=U3+(U2*S3-U1*C3)*C2/S2
      Z4=C3/S2
      Z5=-S3/S2
      Z6=U1*C3-U2*S3
      Z7=-Z2
      Z8=Z4*C2
      Z9=Z5*C2
      Z10=Z8*U1+Z9*U2
      Z11=U1*C4+U2*S4
      Z12=-U1*S4+U2*C4
      Z13=U3+U4
      Z14=C45
      Z15=S45
      Z16=-Z15
      Z17=U1*Z14+U2*Z15
      Z18=-Z13-U5
      Z19=U1*Z16+U2*Z14
      Z20=Z14*C6
```



```
      Z21=Z15*C6
      Z22=Z14*S6
      Z23=Z15*S6
      Z24=Z20*U1+Z21*U2+Z18*S6
      Z25=-Z22*U1-Z23*U2+Z18*C6
      Z26=Z16*U1+Z14*U2+U6
      Z27=L13*Z4
      Z28=L13*Z5
      Z29=Z27*U1+Z28*U2
      Z30=L22*Z8
      Z31=L22*Z9
      Z32=-L22*C3
      Z33=L22*S3
      Z34=Z30*U1+Z31*U2
      Z35=Z32*U1+Z33*U2
      Z36=L33*U2
      Z37=L31*U3-L33*U1
      Z38=-L31*U2
      Z39=(L33+L43)*S4
      Z40=(L33+L43)*C4
      Z41=L1*S4
      Z42=L1*C4+L41
      Z43=L41*S4
      Z44=L1+L41*C4
      Z45=-Z39*U1+Z40*U2+Z41*U3
      Z46=-Z40*U1-Z39*U2+Z42*U3+L41*U4
      Z47=Z43*U1-Z44*U2
      Z48=L1*C4+L2
      Z49=L2*S4
      Z50=L1+L2*C4
      Z51=L33+L43-L52
      Z52=Z39*C5+Z40*S5
      Z53=Z40*C5-Z39*S5
      Z54=Z15*Z51-Z52
      Z55=Z53-Z14*Z51
      Z56=Z41*C5+Z48*S5
      Z57=L2*S5
      Z58=Z48*C5-Z41*S5
      Z59=L2*C5
      Z60=Z54*U1+Z55*U2+Z56*U3+Z57*U4
      Z61=-Z49*U1+Z50*U2
      Z62=-Z55*U1+Z54*U2+Z58*U3+Z59*U4
      Z63=L33+L43
      Z64=Z15*Z63-Z52
      Z65=Z53-Z14*Z63
      Z66=Z56-L63
      Z67=Z57-L63
      Z68=-Z49-L63*Z14
      Z69=Z50-L63*Z15
      Z70=Z64*U1+Z65*U2+Z66*U3+Z67*U4-L63*U5
      Z71=Z68*U1+Z69*U2
      Z72=-Z65*U1+Z64*U2+Z58*U3+Z59*U4
      Z73=-(S2*S3*Z3+C2*C3*Z2)/S2**2
      Z74=(C2*S3*Z2-S2*C3*Z3)/S2**2
      Z75=Z73*U1+Z74*U2
```

```
Z76=Z73*C2-Z2*Z4*S2
Z77=Z74*C2-Z2*Z5*S2
Z78=-(U1*S3+U2*C3)*Z3
Z79=(U1*C3-U2*S3)*Z3
Z80=Z76*U1+Z77*U2
Z81=(C4*U2-S4*U1)*U4
Z82=-(C4*U1+S4*U2)*U4
Z83=-S45*(U4+U5)
Z84=C45*(U4+U5)
Z85=Z83*U1+Z84*U2
Z86=-Z84*U1+Z83*U2
Z87=Z83*C6-Z14*S6*U6
Z88=Z84*C6-Z15*S6*U6
Z89=Z83*S6+Z14*C6*U6
Z90=Z84*S6+Z15*C6*U6
Z91=Z87*U1+Z88*U2+Z18*C6*U6
Z92=Z89*U1+Z90*U2+Z18*S6*U6
Z93=-Z84*U1+Z83*U2
Z94=L13*(Z73*U1+Z74*U2)
Z95=-Z1*Z29
Z96=L22*Z76
Z97=L22*Z77
Z98=L22*S3*Z3
Z99=L22*C3*Z3
Z100=Z96*U1+Z97*U2+Z7*Z35
Z101=Z10*Z34-Z6*Z35
Z102=Z98*U1+Z99*U2-Z7*Z34
Z103=Z38*U2-Z37*U3
Z104=Z36*U3-Z38*U1
Z105=Z37*U1-Z36*U2
Z106=-(L33+L43)*C4*U4
Z107=-(L33+L43)*S4*U4
Z108=L1*C4*U4
Z109=-L1*S4*U4
Z110=L41*C4*U4
Z111=-L41*S4*U4
Z112=Z106*U1+Z107*U2+Z108*U3+(Z12*Z47-Z13*Z46)
Z113=-Z107*U1+Z106*U2+Z109*U3+(Z13*Z45-Z11*Z47)
Z114=Z110*U1-Z111*U2+(Z11*Z46-Z12*Z45)
Z115=Z84*Z51-(L33+L43)*(U4+U5)*C45
Z116=-Z83*Z51+(Z107-Z39*U5)*C5-(-Z106+Z40*U5)*S5
Z117=L1*C45*(U4+U5)+L2*C5*U5
Z118=L2*C5*U5
Z119=L2*C4*U4
Z120=-L2*S4*U4
Z121=-L1*S4*C5*U4-Z48*S5*U5-Z108*S5-Z41*C5*U5
Z122=-L2*S5*U5
Z125=Z115*U1+Z116*U2+Z117*U3+Z118*U4+Z18*Z62-Z19*Z61
Z126=-Z119*U1+Z120*U2+Z19*Z60-Z17*Z62
Z127=-Z116*U1+Z115*U2+Z121*U3+Z122*U4+Z17*Z61-Z18*Z60
Z128=Z84*Z63-(L33+L43)*(U4+U5)*C45
Z129=-Z83*Z63+(Z107-Z39*U5)*C5-(-Z106+Z40*U5)*S5
Z130=-Z119-L63*Z83
Z131=Z120-L63*Z84
Z132=Z18*Z72-Z19*Z71
```



```
Z133=Z19*Z70-Z17*Z72
Z134=Z17*Z71-Z18*Z70
Z135=Z128*U1+Z129*U2+Z117*U3+Z118*U4+Z132
Z136=Z130*U1+Z131*U2+Z133
Z137=-Z129*U1+Z128*U2+Z121*U3+Z122*U4+Z134
Z152=-AI2*Z4
Z153=-AM*Z27
Z154=Z152*Z4+Z153*Z27
Z155=Z152*Z5+Z153*Z28
Z156=Z152*Z75+Z153*Z94
Z157=-AI2*Z5
Z158=-AM*Z28
Z159=Z157*Z4+Z158*Z27
Z160=Z157*Z5+Z158*Z28
Z161=Z157*Z75+Z158*Z94
Z162=(BI3-BI2)*Z7*Z10
Z163=(BI1-BI3)*Z6*Z10
Z164=(BI2-BI1)*Z6*Z7
Z165=-(BI1*C3**2+BI2*S3**2+BI3*Z8**2+BM*Z30**2+BM*Z32**2)
Z166=-(-BI1*C3*S3+BI2*C3*S3+BI3*Z8*Z9+BM*Z30*Z31+BM*Z32*Z33)
Z167=-(Z162*C3+Z163*S3+Z164*Z8+BI1*C3*Z78+BI2*S3*Z79+BI3*Z8*Z80
,      +BM*Z30*Z100+BM*Z32*Z102)
Z168=-(-BI1*S3*C3+BI2*C3*S3+BI3*Z8*Z9+BM*Z30*Z31+BM*Z32*Z33)
Z169=-(BI1*S3**2+BI2*C3**2+BI3*Z9**2+BM*Z31**2+BM*Z33**2)
Z170=-(-BI1*S3*Z78-Z162*S3+BI2*C3*Z79+Z163*C3+BI3*Z9*Z80+Z164*Z9
,      +BM*Z31*Z100+BM*Z33*Z102)
Z171=(CI3-CI2)*U2*U3
Z172=(CI1-CI3)*U1*U3
Z173=(CI2-CI1)*U1*U2
Z174=-CI1-CM*L33**2
Z175=CM*L33*L31
Z176=-Z171+CM*L33*Z104
Z178=-CI2-CM*L31**2-CM*L33**2
Z179=-Z172-CM*L33*Z103+CM*L31*Z105
Z180=CM*L31*L33
Z181=-CI3-CM*L31**2
Z182=-Z173-CM*L31*Z104
Z183=(DI3-DI2)*Z12*Z13
Z184=(DI1-DI3)*Z11*Z13
Z185=(DI2-DI1)*Z11*Z12
Z186=-(DI1*C4**2+DI2*S4**2+DM*(Z39**2+Z40**2+Z43**2))
Z187=(DI2-DI1)*C4*S4+DM*Z43*Z44
Z188=DM*Z39*Z41+DM*Z40*Z42
Z189=-C4*(DI1*Z81+Z183)+S4*(DI2*Z82+Z184)+DM*(Z39*Z112+Z40*Z113
,      -Z43*Z114)
Z189A=-DI1*S4**2-DI2*C4**2-DM*(Z40**2+Z39**2+Z44**2)
Z190=-DM*(Z40*Z41-Z39*Z42)
Z191=-S4*(DI1*Z81+Z183)-C4*(DI2*Z82+Z184)+DM*(-Z40*Z112+Z39*Z113
,      +Z44*Z114)
Z192=-DI3-DM*(Z41**2+Z42**2)
Z193=-Z185-DM*(Z41*Z112+Z42*Z113)
Z194=DM*Z40*L41
Z195=DM*Z39*L41
Z196=-DI3-DM*L41*Z42
Z197=-DI3-DM*L41**2
```

```
      Z198=-Z185-DM*L41*Z113
      Z199=(EI3-EI2)*Z18*Z19
      Z200=(EI1-EI3)*Z17*Z19
      Z201=(EI2-EI1)*Z17*Z18
      Z202=-EI1*Z14**2-EI3*Z16**2-EM*(Z54**2+Z49**2+Z55**2)
      Z203=-EI1*Z14*Z15-EI3*Z14*Z16-EM*(Z54*Z55-Z49*Z50-Z54*Z55)
      Z204=-EM*(Z54*Z56-Z55*Z58)
      Z205=-EM*(Z54*Z57-Z55*Z59)
      Z206=-Z14*(EI1*Z85+Z199)-Z16*(EI3*Z86+Z201)-EM*(Z54*Z125-Z49*Z126
     ,     -Z55*Z127)
      Z207=-EI1*Z15**2-EI3*Z14**2-EM*(Z55**2+Z50**2+Z54**2)
      Z208=-EM*(Z55*Z56+Z54*Z58)
      Z209=-EM*(Z55*Z57+Z54*Z59)
      Z210=-Z15*(EI1*Z85+Z199)-Z14*(EI3*Z86+Z201)-EM*(Z55*Z125+Z50*Z126
     ,     +Z54*Z127)
      Z211=-EI2-EM*(Z56**2+Z58**2)
      Z212=-EI2-EM*(Z56*Z57+Z58*Z59)
      Z213=Z200-EM*(Z56*Z125+Z58*Z127)
      Z214=-EI2-EM*(Z57**2+Z59**2)
      Z215=Z200-EM*(Z57*Z125+Z59*Z127)
      Z216=(FI3-FI2)*Z25*Z26
      Z217=(FI1-FI3)*Z24*Z26
      Z218=(FI2-FI1)*Z24*Z25
      Z219=-(FI1*Z20**2+FI2*Z22**2+FI3*Z16**2+FM*(Z64**2+Z68**2
     ,      +Z65**2))
      Z220=-(FI1*Z20*Z21+FI2*Z22*Z23+FI3*Z14*Z16+FM*Z68*Z69)
      Z221=FI1*Z20*S6-FI2*Z22*C6-FM*(Z64*Z66-Z65*Z58)
      Z222=FI1*Z20*S6-FI2*Z22*C6-FM*(Z64*Z67-Z65*Z59)
      Z223=FI1*Z20*S6-FI2*Z22*C6+FM*Z64*L63
      Z224=-Z20*(FI1*Z91+Z216)+Z22*(-FI2*Z92+Z217)-Z16*(FI3*Z93+Z218)
     ,     -FM*(Z64*Z135+Z68*Z136-Z65*Z137)
      Z225=-(FI1*Z21**2+FI2*Z23**2+FI3*Z14**2+FM*(Z65**2+Z69**2+Z64**2))
      Z226=FI1*Z21*S6-FI2*Z23*C6-FM*(Z65*Z66+Z64*Z58)
      Z227=FI1*Z21*S6-FI2*Z23*C6-FM*(Z65*Z67+Z59*Z64)
      Z228=FI1*Z21*S6-FI2*Z23*C6+FM*L63*Z65
      Z229=-Z21*(FI1*Z91+Z216)+Z23*(Z217-FI2*Z92)-Z14*(FI3*Z93+Z218)
     ,      -FM*(Z65*Z135+Z69*Z136+Z64*Z137)
      Z230=-(FI1*S6**2+FI2*C6**2+FM*(Z66**2+Z58**2))
      Z231=-(FI1*S6**2+FI2*C6**2+FM*(Z66*Z67+Z58*Z59))
      Z232=-(FI1*S6**2+FI2*C6**2-FM*L63*Z66)
      Z234=S6*(FI1*Z91+Z216)+C6*(Z217-FI2*Z92)-FM*(Z66*Z135+Z58*Z137)
      Z235=-(FI1*S6**2+FI2*C6**2+FM*(Z67**2+Z59**2))
      Z236=-(FI1*S6**2+FI2*C6**2-FM*L63*Z67)
      Z237=S6*(FI1*Z91+Z216)-C6*(FI2*Z92-Z217)-FM*(Z67*Z135+Z59*Z137)
      Z238=-FI1*S6**2-FI2*C6**2-FM*L63**2
      Z239=S6*(FI1*Z91+Z216)+C6*(Z217-FI2*Z92)+FM*L63*Z135
      Z240=-FI3*Z16
      Z241=-FI3*Z14
      Z242=-FI3*Z93-Z218
C
      Z243=Z156+Z167+Z176+Z189+Z206+Z224
      Z244=Z161+Z170+Z179+Z191+Z210+Z229
      Z245=Z182+Z193+Z213+Z234
      Z246=Z198+Z215+Z237
      Z247=Z200+Z239
```



```
      Z248=Z242
C
C   CALCULATE THE GENERALIZED INERTIA FORCES
C
      X11=Z154+Z165+Z174+Z186+Z202+Z219
      X12=Z155+Z166+Z187+Z203+Z220
      X13=Z175+Z188+Z204+Z221
      X14=Z194+Z205+Z222
      X15=Z223
      X16=Z240
C
      X21=Z159+Z168+Z187+Z203+Z220
      X22=Z160+Z169+Z178+Z189A+Z207+Z225
      X23=Z190+Z208+Z226
      X24=Z195+Z209+Z227
      X25=Z228
      X26=Z241
C
      X31=Z180+Z188+Z204+Z221
      X32=Z190+Z208+Z226
      X33=Z181+Z192+Z211+Z230
      X34=Z196+Z212+Z231
      X35=-EI2+Z232
      X36=0.0
C
      X41=Z194+Z205+Z222
      X42=Z195+Z209+Z227
      X43=Z196+Z212+Z231
      X44=Z197+Z214+Z235
      X45=-EI2+Z236
      X46=0.0
C
      X51=Z223
      X52=Z228
      X53=-EI2+Z232
      X54=-EI2+Z236
      X55=-EI2+Z238
      X56=0.0
C
      X61=Z240
      X62=Z241
      X63=0.0
      X64=0.0
      X65=0.0
      X66=-FI3
C
C CALCULATE THE GENERALIZED INERTIA FORCES
C
      KI(1)=X11*V(7)+X12*V(8)+X13*V(9)+X14*V(10)+X15*V(11)+X16*V(12)
     ,      +Z243
      KI(2)=X21*V(7)+X22*V(8)+X23*V(9)+X24*V(10)+X25*V(11)+X26*V(12)
     ,      +Z244
      KI(3)=X31*V(7)+X32*V(8)+X33*V(9)+X34*V(10)+X35*V(11)+X36*V(12)
     ,      +Z245
      KI(4)=X41*V(7)+X42*V(8)+X43*V(9)+X44*V(10)+X45*V(11)+X46*V(12)
```

```
     .      +Z246
      KI(5)=X51*V(7)+X52*V(8)+X53*V(9)+X54*V(10)+X55*V(11)+X56*V(12)
     .      +Z247
      KI(6)=X61*V(7)+X62*V(8)+X63*V(9)+X64*V(10)+X65*V(11)+X66*V(12)
     .      +Z248
C
C   CALCULATE THE GENERALIZED ACTIVE FORCES
C
      IF (TQTYPE .EQ. 2) GOTO 35
      CALL TORQUE (TQ,Q,V)
C
   35 Z251=AM*S1
      Z252=Z251*Z27
      Z253=Z251*Z28
      Z254=BM*S1*C2
      Z255=-BM*S1*S2
      Z256=Z30*Z254+Z32*Z255
      Z257=Z31*Z254+Z33*Z255
      Z258=C1*C3-S1*C2*S3
      Z259=-CM*L33*Z258
      Z260=CM*(L33*(S1*C2*C3+C1*S3)+L31*S1*S2)
      Z261=CM*L31*Z258
      Z262=S1*C2*C34+C1*S34
      Z263=C1*C34-S1*C2*S34
      Z264=-S1*S2
      Z265=-DM*(Z39*Z262+Z40*Z263-Z43*Z264)
      Z266=DM*(Z40*Z262-Z39*Z263-Z44*Z264)
      Z267=DM*(Z41*Z262+Z42*Z263)
      Z268=DM*L41*Z263
      Z269=S1*C2*C345+C1*S345
      Z270=S1*S2
      Z271=C1*C345-S1*C2*S345
      Z272=EM*(Z54*Z269-Z49*Z270+Z55*Z271)
      Z273=EM*(Z55*Z269+Z50*Z270+Z54*Z271)
      Z274=EM*(Z56*Z269+Z58*Z271)
      Z275=EM*(Z57*Z269+Z59*Z271)
      Z276=FM*(Z64*Z269+Z68*Z270-Z65*Z271)
      Z277=FM*(Z65*Z269+Z69*Z270+Z64*Z271)
      Z278=FM*(Z66*Z269+Z58*Z271)
      Z279=FM*(Z67*Z269+Z59*Z271)
      Z280=-FM*L63*Z269
      Z281=G*(Z252+Z256+Z259+Z265+Z272+Z276)
      Z282=G*(Z253+Z257+Z260+Z266+Z273+Z277)
      Z283=G*(Z261+Z267+Z274+Z278)
      Z284=G*(Z268+Z275+Z279)
      Z285=G*Z280
C
      KA(1)=TQ(1)*Z4-TQ(2)*S3-TQ(3)*Z8+Z281
      KA(2)=TQ(1)*Z5-TQ(2)*C3-TQ(3)*Z9+Z282
      KA(3)=TQ(3)+Z283
      KA(4)=TQ(4)+Z284
      KA(5)=TQ(5)+Z285
      KA(6)=TQ(6)
C
C COMPUTE THE VALUES OF THE FUNCTIONS FN(7), FN(8),...,FN(12)
```



```
C
      DO 33 I=1,6
      FN(I+6)=KI(I)+KA(I)
   33 CONTINUE
C
C   RETURN THE CALCULATED VALUES TO THE SUBROUTINE QSOLVE.
C
      RETURN
      END
C
C
C THIS SUBROUTINE COMPUTE THE VALUES OF TORQUE FUNCTIONS.
C
      SUBROUTINE TORQUE (TQ,Q,V)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I
      DIMENSION TQ(12),Q(12),V(12)
C
C
      DO 45 I=1,12
      TQ(I)=0.0
   45 CONTINUE
C
C
      RETURN
      END
```

APPENDIX 4. PROGRAM AUTOTF

```
C     ************************************************************
C     *                                                          *
C     *                  PROGRAM      AUTOTF                     *
C     *                                                          *
C     *                  WRITTEN BY: TINGLIN NIE                 *
C     *                  DATE WRITTEN: MAY 1987                  *
C     *                                                          *
C     *   THIS PROGRAM IS TO CALCULATE THE TORQUES /OR FORCES    *
C     *   APPLIED TO THE LINKS OF THE ROBOT UNDER CONSIDERATION  *
C     *   TO CARRY OUT A SPECIFIED MOTION. IN OTHER WORDS, THE   *
C     *   MOTION SPECIFIED BY FUNCTIONS OR NUMERICAL DATA IS     *
C     *   INPUT, THE TORQUES /OR FORCES ARE OUTPUTS.             *
C     *                                                          *
C     ************************************************************
C
C
      PROGRAM AUTOTF
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,J,J1,J2,J3,K,L,M,N,NN,DELTA
      PARAMETER(NN=6)
      DIMENSION DELTA(NN,2)
      DIMENSION Q(NN),U(NN),UD(NN),TQ(NN)
      DIMENSION IN(NN,3),MS(NN),DL(NN,3),LL(NN,3)
      COMMON/BLK1/Q,U,UD
      COMMON/BLK2/LL,DL,MS,IN,TQ
      COMMON/BLK3/TIME,TSPAN
      COMMON/BLK4/N,L,DELTA
      OPEN (7,FILE='ATDATA')
      OPEN (8,FILE='ATOUT')
C
C
C  SPECIFY THE JOINT STATUS AND THE DIRECTION OF GRAVITY FORCES
C
      WRITE(*,*) ' PLEASE ENTER THE NUMBER OF LINKS: '
      READ(*,*) N
      WRITE(*,*) ' PLEASE INDICATE THE JOINT STATUS (IF THE JOINT IS'
      WRITE(*,*) ' REVOLUTE, TYPE "1,0"; IF THE JOINT IS PRISMATIC,'
      WRITE(*,*) ' TYPE "0,1".'
      DO 100 K=1,N
      WRITE(*,110) ' JOINT ',K,' ?'
      READ(*,*) DELTA(K,1),DELTA(K,2)
  100 CONTINUE
  110 FORMAT (A,I1,A)
      WRITE(*,*) ' PLEASE INDICATE THE DIRECTION OF THE GRAVITATIONAL'
      WRITE(*,*) ' FORCES (i.e. 1,2,3).'
      READ(*,*) L
C
C  INPUT THE MASS PROPERTY DATA AND STRUCTURAL DATA
C
      READ(7,120) (MS(K),K=1,N)
  120 FORMAT(6F7.4)
      DO 130 K=1,N
      READ(7,140) (IN(K,J),J=1,3)
  130 CONTINUE
  140 FORMAT(3F7.4)
```



```
      DO 150 K=1,N
      READ(7,140) (DL(K,J),J=1,3)
  150 CONTINUE
      DO 160 K=1,N
      READ(7,140) (LL(K,J),J=1,3)
  160 CONTINUE
      READ(7,170) TSPAN
      READ(7,170) TSTEP
  170 FORMAT(F9.5)
C
C   PRINT TITLE FOR THE OUTPUT
C
      WRITE (8,180)
  180 FORMAT (////,30X,15HTABLE OF OUTPUT/5X,
     , 61H(TORQUES /OR FORCES NEEDED TO CARRY OUT THE SPECIFIED MOTION))
      WRITE (8,190)
  190 FORMAT (///,1X,19HTHE GIVEN DATA ARE:)
      WRITE (8,200)
  200 FORMAT (//,3X,23HMASS OF EACH LINK (Kg):/)
      DO 210 K=1,N
      WRITE(8,220) K,MS(K)
  210 CONTINUE
  220 FORMAT(8X,5HMASS(,I1,4H) = ,F7.4)
      WRITE (8,230)
  230 FORMAT(//,3X,31HINERTIAS OF EACH LINK (Kg-m^2):/)
      J1=1
      J2=2
      J3=3
      DO 235 K=1,N
      WRITE(8,240) K,J1,IN(K,1),K,J2,IN(K,2),K,J3,IN(K,3)
  235 CONTINUE
  240 FORMAT(3X,3(5X,4HINR(,I1,1H,,I1,4H) = ,F7.4))
      WRITE(8,250)
  250 FORMAT (//,3X,40HTHE COORDINATES OF MASS CENTERS (meter):/)
      DO 255 K=1,N
      WRITE(8,265) K,J1,DL(K,1),K,J2,DL(K,2),K,J3,DL(K,3)
  255 CONTINUE
      WRITE(8,*) ' '
      DO 260 K=1,N
      WRITE(8,268) K,J1,LL(K,1),K,J2,LL(K,2),K,J3,LL(K,3)
  260 CONTINUE
  265 FORMAT(2X,3(6X,3HDL(,I1,1H,,I1,4H) = ,F7.4))
  268 FORMAT(2X,3(6X,3HLL(,I1,1H,,I1,4H) = ,F7.4))
      WRITE (8,270)TSPAN,TSTEP
  270 FORMAT (//,3X,11HOTHER DATA://
     ,       8X,12HTIME SPAN = ,F9.5,8X,
     ,       12HTIME STEP = ,F9.5)
      WRITE (8,280)
  280 FORMAT (//,3X,38HTHE TORQUES APPLIED TO EACH LINK (Nm):)
      WRITE (8,290)
  290 FORMAT (//,4HTIME,9X,2HT1,9X,2HT2,9X,2HT3,9X,2HT4,
     ,        9X,2HT5,9X,2HT6//)
C
C   MAIN PROGRAM
C
```

```
      TIME=0.0
      M=IDINT(TSPAN/TSTEP)
      M=M+2
      DO 300 I=1,M
C
C  TO CALL SUBROUTINE GCRO
C
      CALL GCRO
C
C  TO CALL SUBROUTINE AUTOSUB
C
      CALL AUTOSUB
      WRITE (8,295) TIME,(TQ(K),K=1,N)
  295 FORMAT (F5.3,1X,6F11.5)
      TIME=TIME+TSTEP
  300 CONTINUE
      STOP
      END
C
C   THIS SUBROUTINE IS TO CALCULATE THE GENERALIZED COOR-
C   DINATES, GENERALIZED SPEEDS AND THEIR DERIVATIVES.
C
      SUBROUTINE GCRO
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER NN
      PARAMETER(NN=6)
      COMMON/BLK1/Q,U,UD
      COMMON/BLK3/TIME,TSPAN
      COMMON/BLK4/N,L,DELTA
      DIMENSION Q(NN),U(NN),UD(NN)
C
      T=TIME
      TF=TSPAN
      PI=DACOS(-1.0D0)
      QQ=T-TF*DSIN(2*PI*T/TF)/(2*PI)
      Q(1)=QQ*PI/(3.*TF)
      Q(2)=PI/2.-QQ*PI/(6.*TF)
      Q(3)=Q(1)
      Q(4)=Q(1)
      Q(5)=Q(1)
      Q(6)=Q(1)
      U(1)=PI*(1.-DCOS(2.*PI*T/TF))/(TF*3.)
      U(2)=-PI*(1.-DCOS(2.*PI*T/TF))/(6.*TF)
      U(3)=U(1)
      U(4)=U(1)
      U(5)=U(1)
      U(6)=U(1)
      UD(1)=2.*PI**2/(3.*TF**2)*DSIN(2.*PI*T/TF)
      UD(2)=-PI**2/(3.*TF**2)*DSIN(2.*PI*T/TF)
      UD(3)=UD(1)
      UD(4)=UD(1)
      UD(5)=UD(1)
      UD(6)=UD(1)
C
      RETURN
```



```
      END
C
C   THIS SUB ROUTINE IS TO AUTOMATICALLY GENERATE
C   THE GENERALIZED ACTIVE FORCES, THE GENERALIZED
C   INERTIA FORCES, AND FINALLY CALCULATE THE TORQUES
C   /OR FORCES.
C
      SUBROUTINE AUTOSUB
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,J,K,L,M,NN,N,R,DELTA
      PARAMETER(NN=6)
      DIMENSION A(NN,3,3),AD(NN,3,3)
      DIMENSION W(NN,3),V(NN,3),S(NN+1,3),PW(NN,3,NN),
     *          PV(NN,3,NN),PS(NN+1,3,NN)
      DIMENSION PWD(NN,3,NN),PVD(NN,3,NN),PSD(NN+1,3,NN)
      DIMENSION DELTA(NN,2)
      DIMENSION MI(NN,3,NN),MID(NN,3,NN),B(NN,3),D(NN,3),E(NN,3),H(NN,3)
      DIMENSION Z(NN),Y(NN,3),P(NN,3,NN),PD(NN,3,NN)
      DIMENSION IN(NN,3),MS(NN),DL(NN,3),LL(NN,3)
      DIMENSION F(NN,NN),KSTAR(NN),KP2(NN),TQ(NN)
      DIMENSION Q(NN),U(NN),UD(NN)
      COMMON/BLK1/Q,U,UD
      COMMON/BLK2/LL,DL,MS,IN,TQ
      COMMON/BLK4/N,L,DELTA
C
C  DEFINE TRANSFORMATION MATRICES AND THEIR DERIVATIVES
C
      G=9.81
      DO 110 K=1,N
      DO 110 I=1,3
      DO 110 J=1,3
      A(K,I,J)=0.0
      AD(K,I,J)=0.0
  110 CONTINUE
      A(1,1,1)=DCOS(Q(1))
      A(1,1,3)=DSIN(Q(1))
      A(1,2,1)=DSIN(Q(1))
      A(1,2,3)=-DCOS(Q(1))
      A(1,3,2)=1.0
      A(2,1,1)=DCOS(Q(2))
      A(2,1,3)=-DSIN(Q(2))
      A(2,2,1)=DSIN(Q(2))
      A(2,2,3)=DCOS(Q(2))
      A(2,3,2)=-1.0
      A(3,1,1)=DCOS(Q(3))
      A(3,1,2)=-DSIN(Q(3))
      A(3,2,1)=DSIN(Q(3))
      A(3,2,2)=DCOS(Q(3))
      A(3,3,3)=1.0
      A(4,1,1)=DCOS(Q(4))
      A(4,1,2)=-DSIN(Q(4))
      A(4,2,1)=DSIN(Q(4))
      A(4,2,2)=DCOS(Q(4))
      A(4,3,3)=1.0
      A(5,1,1)=DCOS(Q(5))
```

```
      A(5,1,3)=-DSIN(Q(5))
      A(5,2,1)=DSIN(Q(5))
      A(5,2,3)=DCOS(Q(5))
      A(5,3,2)=-1.0
      A(6,1,1)=DCOS(Q(6))
      A(6,1,2)=-DSIN(Q(6))
      A(6,2,1)=DSIN(Q(6))
      A(6,2,2)=DCOS(Q(6))
      A(6,3,3)=1.0
C
C
      AD(1,1,1)=-DSIN(Q(1))*U(1)
      AD(1,1,3)=DCOS(Q(1))*U(1)
      AD(1,2,1)=DCOS(Q(1))*U(1)
      AD(1,2,3)=DSIN(Q(1))*U(1)
      AD(2,1,1)=-DSIN(Q(2))*U(2)
      AD(2,1,3)=-DCOS(Q(2))*U(2)
      AD(2,2,1)=DCOS(Q(2))*U(2)
      AD(2,2,3)=-DSIN(Q(2))*U(2)
      AD(3,1,1)=-U(3)*DSIN(Q(3))
      AD(3,1,2)=-U(3)*DCOS(Q(3))
      AD(3,2,1)=DCOS(Q(3))*U(3)
      AD(3,2,2)=-DSIN(Q(3))*U(3)
      AD(4,1,1)=-U(4)*DSIN(Q(4))
      AD(4,1,2)=-U(4)*DCOS(Q(4))
      AD(4,2,1)=DCOS(Q(4))*U(4)
      AD(4,2,2)=-DSIN(Q(4))*U(4)
      AD(5,1,1)=-DSIN(Q(5))*U(5)
      AD(5,1,3)=-DCOS(Q(5))*U(5)
      AD(5,2,1)=DCOS(Q(5))*U(5)
      AD(5,2,3)=-DSIN(Q(5))*U(5)
      AD(6,1,1)=-DSIN(Q(6))*U(6)
      AD(6,1,2)=-DCOS(Q(6))*U(6)
      AD(6,2,1)=DCOS(Q(6))*U(6)
      AD(6,2,2)=-DSIN(Q(6))*U(6)
C
C  INITIALIZATIONS
C
      DO 115 K=1,N
      Z(K)=0.0
      KSTAR(K)=0.0
      KP2(K)=0.0
      DO 115 J=1,3
      W(K,J)=0.0
      V(K,J)=0.0
      S(K,J)=0.0
      B(K,J)=0.0
      D(K,J)=0.0
      E(K,J)=0.0
      Y(K,J)=0.0
      DO 115 R=1,N
      PW(K,J,R)=0.0
      PWD(K,J,R)=0.0
      PV(K,J,R)=0.0
      PVD(K,J,R)=0.0
```



```
      PS(K,J,R)=0.0
      PSD(K,J,R)=0.0
      P(K,J,R)=0.0
      PD(K,J,R)=0.0
      MI(K,J,R)=0.0
      MID(K,J,R)=0.0
      F(K,R)=0.0
  115 CONTINUE
C
C  PARTIAL ANGULAR VELOCITIES
C
      IF (DELTA(1,1).EQ.1) THEN
      DO 120 J=1,3
      PW(1,J,1)=A(1,3,J)
      PWD(1,J,1)=AD(1,3,J)
  120 CONTINUE
      ENDIF
      DO 150 K=2,N
      DO 150 J=1,3
      DO 130 R=1,K-1
      DO 130 I=1,3
      PW(K,J,R)=PW(K,J,R)+PW(K-1,I,R)*A(K,I,J)
      PWD(K,J,R)=PWD(K,J,R)+PWD(K-1,I,R)*A(K,I,J)+PW(K-1,I,R)*AD(K,I,J)
  130 CONTINUE
      IF (DELTA(K,1).EQ.1) THEN
      PW(K,J,K)=A(K,3,J)
      PWD(K,J,K)=AD(K,3,J)
      ELSE
      PW(K,J,K)=0.0
      PWD(K,J,K)=0.0
      ENDIF
  150 CONTINUE
C
C  ANGULAR VELOCITIES
C
      DO 160 K=1,N
      DO 160 J=1,3
      DO 160 R=1,K
      W(K,J)=W(K,J)+PW(K,J,R)*U(R)
  160 CONTINUE
C
C  PARTIAL VELOCITIES
C
      IF (DELTA(1,1).EQ.1) THEN
      PV(1,1,1)=PW(1,2,1)*DL(1,3)-PW(1,3,1)*DL(1,2)
      PV(1,2,1)=PW(1,3,1)*DL(1,1)-PW(1,1,1)*DL(1,3)
      PV(1,3,1)=PW(1,1,1)*DL(1,2)-PW(1,2,1)*DL(1,1)
      PS(2,1,1)=PW(1,2,1)*LL(1,3)-PW(1,3,1)*LL(1,2)
      PS(2,2,1)=PW(1,3,1)*LL(1,1)-PW(1,1,1)*LL(1,3)
      PS(2,3,1)=PW(1,1,1)*LL(1,2)-PW(1,2,1)*LL(1,1)
      ELSE
      PV(1,3,1)=1.0
      PS(2,3,1)=1.0
      ENDIF
      DO 180 K=2,N
```

```
      IF (DELTA(K,1).EQ.0) THEN
      DO 165 R=1,K-1
      DO 163 J=1,3
      DO 163 I=1,3
      MI(K,J,R)=MI(K,J,R)+PS(K,I,R)*A(K,I,J)
      MID(K,J,R)=MID(K,J,R)+PSD(K,I,R)*A(K,I,J)+PS(K,I,R)*AD(K,I,J)
163   CONTINUE
      PV(K,1,R)=MI(K,1,R)+(DL(K,3)+Q(K))*PW(K,2,R)-DL(K,2)*PW(K,3,R)
      PV(K,2,R)=MI(K,2,R)+DL(K,1)*PW(K,3,R)-(DL(K,3)+Q(K))*PW(K,1,R)
      PV(K,3,R)=MI(K,3,R)+DL(K,2)*PW(K,1,R)-DL(K,1)*PW(K,2,R)
      PS(K+1,1,R)=MI(K,1,R)+(LL(K,3)+Q(K))*PW(K,2,R)-DL(K,2)*PW(K,3,R)
      PS(K+1,2,R)=MI(K,2,R)+LL(K,1)*PW(K,3,R)-(LL(K,3)+Q(K))*PW(K,1,R)
      PS(K+1,3,R)=MI(K,2,R)+LL(K,2)*PW(K,1,R)-LL(K,1)*PW(K,2,R)
      PVD(K,1,R)=MID(K,1,R)+(DL(K,3)+Q(K))*PWD(K,2,R)+U(K)*PW(K,2,R)
     .            -DL(K,2)*PWD(K,3,R)
      PVD(K,2,R)=MID(K,2,R)+DL(K,1)*PWD(K,3,R)-U(K)*PW(K,1,R)
     .            -(DL(K,3)+Q(K))*PWD(K,1,R)
      PVD(K,3,R)=MID(K,3,R)+DL(K,2)*PWD(K,1,R)-DL(K,1)*PWD(K,2,R)
      PSD(K+1,1,R)=MID(K,1,R)+(LL(K,3)+Q(K))*PWD(K,2,R)+U(K)*PW(K,2,R)
     .            -LL(K,2)*PWD(K,3,R)
      PSD(K+1,2,R)=MID(K,2,R)+LL(K,1)*PWD(K,3,R)-(LL(K,3)+Q(K))*
     .              PWD(K,1,R)-U(K)*PW(K,1,R)
      PSD(K+1,3,R)=MID(K,3,R)+LL(K,2)*PWD(K,1,R)-LL(K,1)*PWD(K,2,R)
165   CONTINUE
      PV(K,3,K)=1.0
      PS(K+1,3,K)=1.0
      ELSE
      DO 175 R=1,K
      DO 170 J=1,3
      DO 170 I=1,3
      P(K,J,R)=P(K,J,R)+PS(K,I,R)*A(K,I,J)
      PD(K,J,R)=PD(K,J,R)+PSD(K,I,R)*A(K,I,J)+PS(K,I,R)*AD(K,I,J)
170   CONTINUE
      PV(K,1,R)=PV(K,1,R)+P(K,1,R)+PW(K,2,R)*DL(K,3)-PW(K,3,R)*DL(K,2)
      PV(K,2,R)=PV(K,2,R)+P(K,2,R)+PW(K,3,R)*DL(K,1)-PW(K,1,R)*DL(K,3)
      PV(K,3,R)=PV(K,3,R)+P(K,3,R)+PW(K,1,R)*DL(K,2)-PW(K,2,R)*DL(K,1)
      PS(K+1,1,R)=PS(K+1,1,R)+P(K,1,R)+PW(K,2,R)*LL(K,3)-PW(K,3,R)
     .            *LL(K,2)
      PS(K+1,2,R)=PS(K+1,2,R)+P(K,2,R)+PW(K,3,R)*LL(K,1)-PW(K,1,R)
     .            *LL(K,3)
      PS(K+1,3,R)=PS(K+1,3,R)+P(K,3,R)+PW(K,1,R)*LL(K,2)-PW(K,2,R)
     .            *LL(K,1)
      PVD(K,1,R)=PVD(K,1,R)+PD(K,1,R)+PWD(K,2,R)*DL(K,3)-PWD(K,3,R)
     .            *DL(K,2)
      PVD(K,2,R)=PVD(K,2,R)+PD(K,2,R)+PWD(K,3,R)*DL(K,1)-PWD(K,1,R)
     .            *DL(K,3)
      PVD(K,3,R)=PVD(K,3,R)+PD(K,3,R)+PWD(K,1,R)*DL(K,2)-PWD(K,2,R)
     .            *DL(K,1)
      PSD(K+1,1,R)=PSD(K+1,1,R)+PD(K,1,R)+PWD(K,2,R)*LL(K,3)-PWD(K,3,R)
     .            *LL(K,2)
      PSD(K+1,2,R)=PSD(K+1,2,R)+PD(K,2,R)+PWD(K,3,R)*LL(K,1)-PWD(K,1,R)
     .            *LL(K,3)
      PSD(K+1,3,R)=PSD(K+1,3,R)+PD(K,3,R)+PWD(K,1,R)*LL(K,2)-PWD(K,2,R)
     .            *LL(K,1)
175   CONTINUE
```



```
      ENDIF
180   CONTINUE
C
C  VELOCITIES
C
      DO 200 K=1,N
      DO 200 J=1,3
      DO 200 R=1,K
      V(K,J)=V(K,J)+PV(K,J,R)*U(R)
      S(K,J)=S(K,J)+PS(K,J,R)*U(R)
200   CONTINUE
C
C  CALCULATE INTERMEDIATE VARIABLES
C
      DO 220 K=1,N
      B(K,1)=W(K,2)*V(K,3)-W(K,3)*V(K,2)
      B(K,2)=W(K,3)*V(K,1)-W(K,1)*V(K,3)
      B(K,3)=W(K,1)*V(K,2)-W(K,2)*V(K,1)
      DO 210 J=1,3
      DO 210 R=1,K
      D(K,J)=D(K,J)+PVD(K,J,R)*U(R)
      E(K,J)=E(K,J)+PWD(K,J,R)*U(R)
210   CONTINUE
      H(K,1)=IN(K,1)*E(K,1)+W(K,2)*W(K,3)*(IN(K,3)-IN(K,2))
      H(K,2)=IN(K,2)*E(K,2)+W(K,3)*W(K,1)*(IN(K,1)-IN(K,3))
      H(K,3)=IN(K,3)*E(K,3)+W(K,1)*W(K,2)*(IN(K,2)-IN(K,1))
220   CONTINUE
      DO 230 K=1,N
      DO 230 J=1,3
      D(K,J)=D(K,J)+B(K,J)
230   CONTINUE
C
C  CALCULATE INTERMEDIATE VARIABL Z'S
C
      DO 240 R=1,N
      DO 240 K=1,N
      DO 240 J=1,3
      Z(R)=Z(R)+PW(K,J,R)*H(K,J)+MS(K)*PV(K,J,R)*D(K,J)
240   CONTINUE
C
C  CALCULATE INERTIA COEFFICIENTS
C
      DO 250 R=1,N
      DO 250 M=1,R
      DO 245 J=1,3
      DO 245 K=1,N
      F(R,M)=F(R,M)+IN(K,J)*PW(K,J,M)*PW(K,J,R)+MS(K)*PV(K,J,M)
     .         *PV(K,J,R)
      F(M,R)=F(R,M)
245   CONTINUE
250   CONTINUE
C
C  GENERALIZED INERTIA FORCES
C
      DO 260 R=1,N
```

```
      DO 255 M=1,N
      KSTAR(R)=KSTAR(R)-F(R,M)*UO(M)
  255 CONTINUE
      KSTAR(R)=KSTAR(R)-Z(R)
  260 CONTINUE
C
C  INGREDIENTS OF GENERALIZED ACTIVE FORCES
C
      DO 270 J=1,3
      Y(1,J)=A(1,L,J)
  270 CONTINUE
      DO 280 K=2,N
      DO 280 J=1,3
      DO 280 I=1,3
      Y(K,J)=Y(K,J)+Y(K-1,I)*A(K,I,J)
  280 CONTINUE
      DO 290 R=1,N
      DO 290 K=1,N
      DO 290 J=1,3
      KP2(R)=KP2(R)+MS(K)*G*Y(K,J)*PV(K,J,R)
  290 CONTINUE
C
C  CALCULATE ACTIVE FORCES /OR TORQUES
C
      DO 300 R=1,N
      TQ(R)=-KSTAR(R)-KP2(R)
  300 CONTINUE
C
C   RETURN THE CALCULATED VALUES TO THE MAIN PROGRAM
C
      RETURN
      END
```

APPENDIX 5. PROGRAM AUTOQ

```
C      ***************************************************************
C      *                                                             *
C      *                PROGRAM      AUTOQ                           *
C      *                                                             *
C      *                WRITTEN BY: TINGLIN NIE                      *
C      *                DATE WRITTEN: JUNE 1987                      *
C      *                                                             *
C      *   THIS PROGRAM IS TO SIMULTANEOUSLY GENERATE AND SOLVE      *
C      *   THE KANE'S DYNAMICAL EQUATIONS FOR MECHANICAL             *
C      *   MANIPULATORS. WITH THE STRUCTURAL DATA AND THE            *
C      *   GENERALIZED ACTIVE FORCES GIVEN, THIS PROGRAM WILL        *
C      *   AUTOMATICALLY GENERATE THE EQUATIONS OF MOTION AND        *
C      *   SOLVE THEM FOR THE GENERALIZED COORDINATES.               *
C      *                                                             *
C      ***************************************************************
C
C
      PROGRAM AUTOQ
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,J,J1,J2,J3,K,L,M,N,NN,N2,DELTA
      INTEGER ITER,ITIME,ITMAX,MAXITR,TQTYPE,NPRINT
      PARAMETER(NN=6)
      PARAMETER(N2=12)
      DIMENSION AM(N2,N2),Q0(NN),U0(NN),QU(N2),FN(N2)
      DIMENSION DELTA(NN,2),TF(NN)
      DIMENSION QOLD(N2),QOLDEST(N2),VECTOR(N2)
      DIMENSION IN(NN,3),MS(NN),DL(NN,3),LL(NN,3)
      COMMON/BLK1/QOLD,VECTOR,BTA,TF,QOLDEST,TSTEP,ITIME
      COMMON/BLK2/LL,DL,MS,IN
      COMMON/BLK3/N,L,DELTA
      COMMON/BLK4/XEPSI,FEPSI
      OPEN (7,FILE='ATTQ')
      OPEN (8,FILE='ATAGL')
      OPEN (9,FILE='TOUT1')
C
C
C  SPECIFY THE JOINT STATUS AND THE DIRECTION OF GRAVITY FORCES
C
      WRITE(*,*) ' PLEASE ENTER THE NUMBER OF LINKS: '
      READ(*,*) N
      WRITE(*,*) ' PLEASE INDICATE THE JOINT STATUS (IF THE JOINT IS'
      WRITE(*,*) ' REVOLUTE, TYPE "1,0"; IF THE JOINT IS PRISMATIC,'
      WRITE(*,*) ' TYPE "0,1".'
      DO 100 K=1,N
      WRITE(*,110) ' JOINT ',K,' ?'
      READ(*,*) DELTA(K,1),DELTA(K,2)
  100 CONTINUE
  110 FORMAT (A,I1,A)
      WRITE(*,*) ' PLEASE INDICATE THE DIRECTION OF THE GRAVITATIONAL'
      WRITE(*,*) ' FORCES (i.e. 1,2,3).'
      READ(*,*) L
C
C  INPUT THE MASS PROPERTY DATA AND STRUCTURAL DATA
```



```
C
      READ(7,120) (MS(K),K=1,N)
  120 FORMAT(6F7.4)
      DO 130 K=1,N
      READ(7,140) (IN(K,J),J=1,3)
  130 CONTINUE
  140 FORMAT(3F7.4)
      DO 150 K=1,N
      READ(7,140) (DL(K,J),J=1,3)
  150 CONTINUE
      DO 160 K=1,N
      READ(7,140) (LL(K,J),J=1,3)
  160 CONTINUE
      READ(7,165) (Q0(I),I=1,N)
      READ(7,165) (U0(I),I=1,N)
  165 FORMAT(6F12.9)
      READ(7,*) MAXITR,ERRMAX,XEPSI,FEPSI,NPRINT
      READ(7,170) TSPAN
      READ(7,170) TSTEP
  170 FORMAT(F9.5)
C
C   PRINT TITLE FOR THE OUTPUT
C
      WRITE (8,180)
  180 FORMAT (////,19X,'NUMERICAL SOLUTION OF THE EQUATIONS OF MOTION'
     ,           /26X,'(BY AUTOMATIC METHOD)')
      WRITE (8,190)
  190 FORMAT (///,1X,19HTHE GIVEN DATA ARE:)
      WRITE (8,200)
  200 FORMAT (//,3X,23HMASS OF EACH LINK (Kg):/)
      DO 210 K=1,N
      WRITE(8,220) K,MS(K)
  210 CONTINUE
  220 FORMAT(8X,5HMASS(,I1,4H) = ,F7.4)
      WRITE (8,230)
  230 FORMAT(//,3X,31HINERTIAS OF EACH LINK (Kg-m^2):/)
      J1=1
      J2=2
      J3=3
      DO 235 K=1,N
      WRITE(8,240) K,J1,IN(K,1),K,J2,IN(K,2),K,J3,IN(K,3)
  235 CONTINUE
  240 FORMAT(3X,3(5X,4HINR(,I1,1H,,I1,4H) = ,F7.4))
      WRITE(8,250)
  250 FORMAT (//,3X,40HTHE COORDINATES OF MASS CENTERS (meter):/)
      DO 255 K=1,N
      WRITE(8,265) K,J1,DL(K,1),K,J2,DL(K,2),K,J3,DL(K,3)
  255 CONTINUE
      WRITE(8,*) ' '
      DO 260 K=1,N
      WRITE(8,270) K,J1,LL(K,1),K,J2,LL(K,2),K,J3,LL(K,3)
  260 CONTINUE
  265 FORMAT(2X,3(6X,3HDL(,I1,1H,,I1,4H) = ,F7.4))
  270 FORMAT(2X,3(6X,3HLL(,I1,1H,,I1,4H) = ,F7.4))
      WRITE(8,280)
```

```
  280 FORMAT(//,3X,35HTHE GIVEN INITIAL VALUES (RAD) ARE:/)
      WRITE (8,290) (I,Q0(I),I,U0(I),I=1,6)
  290 FORMAT (/,6(8X,3HQ0(,I1,4H) = ,F9.6,8X,3HU0(,I1,4H) = ,F9.6/))
      WRITE(8,295) TSPAN,TSTEP
  295 FORMAT(//,3X,11HOTHER DATA://8X,12HTIME SPAN = ,
     ,        F8.4,8X,12HTIME STEP = ,F8.4)
      WRITE(8,300)
  300 FORMAT (//,3X,40HTHE GENERALIZED COORDINATES Q1-QN (RAD):)
      WRITE (8,310)
  310 FORMAT (//,4HTIME,9X,2HQ1,9X,2HQ2,9X,2HQ3,9X,2HQ4,
     ,          9X,2HQ5,9X,2HQ6//)
      WRITE (*,*) ' PLEASE INDICATE THE METHOD TOU WISH TO USE.'
      WRITE (*,*) ' EXPLICIT:  0      '
      WRITE (*,*) ' IMPLICIT:  1      '
      WRITE (*,*) ' CRANK-NELSON:   0.0-1.0 '
      READ (*,*) BTA
      WRITE (*,*) ' PLEASE ENTER THE INITIAL GUESS FOR AM MATRIX'
      WRITE (*,*) ' AND VECTOR (VALUE OF TIME STEP IS SUGGESTED): '
      READ(*,*) AA,BB
C
C   MAIN PROGRAM TO CALL THREE DIFFERENT SUBROUTINES
C
      TIME=0.0
      ITMAX=DINT(TSPAN/TSTEP+1)
      CALL GUESS(AM,VECTOR,AA,BB)
      DO 320 I=1,N
      QU(I)=Q0(I)
      QU(I+N)=U0(I)
  320 CONTINUE
      WRITE (8,380) TIME,(QU(I),I=1,N)
      DO 330 I=1,N2
      QOLDEST(I)=Q0(I)
  330 CONTINUE
      WRITE (*,*) 'PLEASE INDICATE THE DATA TYPE OF THE TORQUES'
      WRITE (*,*) 'YOU ARE GOING TO USE!'
      WRITE (*,*) 'IF USE FUNCTION,TYPE 1'
      WRITE (*,*) 'IF USE NUMERICAL VALUES, TYPE 2:'
      READ (*,*) TQTYPE
C
      DO 340 ITIME=1,ITMAX
      IF (TQTYPE .EQ. 1) GOTO 360
      READ (9,350) TIMET,(TF(I),I=1,N)
  350 FORMAT (F5.3,1X,6G16.8)
  360 DO 370 I=1,N2
      IF (ITIME .EQ. 1) THEN
      QOLD(I)=QU(I)
      QU(I)=QU(I)+VECTOR(I)*TSTEP
      ELSE
      QOLDEST(I)=QOLD(I)
      QOLD(I)=QU(I)
      QU(I)=2.0*TSTEP*VECTOR(I)+QOLDEST(I)
      ENDIF
  370 CONTINUE
C
C CALL SUBROUTINE QSOLVE TO FIND SOLUTION
```



```
C
      CALL QSOLVE (AM,QU,FN,MAXITR,ERRMAX,ERRX,ERRF,ITER)
      ITER=MIN(ITER,MAXITR)
      TIME=TIME+TSTEP
      IF (MOD(ITIME,NPRINT) .EQ. 0) THEN
      WRITE (8,380) TIME,(QU(I),I=1,N)
  380 FORMAT (F5.3,1X,6(2X,F9.6))
      ENDIF
      WRITE(*,*) ' ITERATION NUMBER: ',ITIME
  340 CONTINUE
      STOP
      END
C
C
C THIS SUBROUTINE ASSIGN INITIAL VALUE TO MATRIX A AND VECTOR V
C
      SUBROUTINE GUESS (A,V,AA,BB)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,J,N
      DIMENSION A(12,12),V(12)
      N=12
      DO 20 I=1,N
      DO 10 J=1,N
      A(I,J)=0.0
   10 CONTINUE
      A(I,I)=AA
      V(I)=BB
   20 CONTINUE
      V(2)=-BB
      RETURN
      END
C
C SUBROUTINE QSOLVE TO SOLVE THE 12 SIMULTANEOUS NONLINEAR
C DIFFERENTIAL EQUATIONS BY USING QUASI-NEWTON ALGORITHM.
C
      SUBROUTINE QSOLVE (A,X,F,MAXITR,ERRMAX,ERRX,ERRF,ITER)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,ITER,J,M,MAXITR,N,N2
      PARAMETER(N2=12)
      COMMON/BLK4/XEPSI,FEPSI
      DIMENSION A(N2,N2),X(N2),F(N2),DF(N2),Q(N2,N2),R(N2,N2)
C
      CALL AUTOSUB(X,F)
C
C FIND THE INVERSE JACOBIAN AND THE SOLUTION BY ITERATION
C
      M=1
      N=12
      DO 10 ITER=1,MAXITR
      ERRX=0.0
      ERRF=0.0
      ERRDF=0.0
C
      DO 20 I=1,N
      DX=0.0
```

```
      DO 30 J=1,N
      DX=DX-A(I,J)*F(J)
   30 CONTINUE
C
      X(I)=X(I)+DX
      IF (DABS(X(I)) .GT. XEPSI) THEN
      RELERR=DABS(DX/X(I))
      IF (RELERR .GT. ERRX) ERRX=RELERR
      ENDIF
   20 CONTINUE
C
      DO 40 I=1,N
      DF(I)=F(I)
   40 CONTINUE
C
C CALL AUTOSUB TO FIND THE NEW VALUE OF THE FUNCTIONS
C
      CALL AUTOSUB(X,F)
      DO 50 I=1,N
      DF(I)=F(I)-DF(I)
      IF (DABS(DF(I)).GT.ERRDF) ERRDF=DABS(DF(I))
      IF (DABS(F(I)).GT.ERRF) ERRF=DABS(F(I))
   50 CONTINUE
C
C COMPUTE THE ORHTOGONAL VECTORS Z AND R
C
      IF (ERRF .LT. ERRMAX) RETURN
      CALL ORTHO (DF,M,Q,R)
C
C
      IF (DABS(R(M,M)) .GT. ERRDF*FEPSI) THEN
      DO 60 I=1,N
      DF(I)=0.0
      DO 70 J=1,N
      DF(I)=DF(I)+A(I,J)*F(J)
   70 CONTINUE
   60 CONTINUE
C
C UPDATE THE A MATRIX
C
      DO 80 I=1,N
      DO 80 J=1,N
      A(I,J)=A(I,J)-DF(I)*Q(J,M)/R(M,M)
   80 CONTINUE
      M=M+1
      ENDIF
   10 CONTINUE
C
      RETURN
      END
C
C
C THIS SUBROUTINE COMPUTES THE ORTHOGONAL BASIS FOR SPACE SPANNED
C BY THE INPUT VECTOR AND M-1 PREVIOUS VECTORS. THE QR DECOMPOSI-
C TION IS RETURNED.
```



```
C
      SUBROUTINE ORTHO (B,M,Q,R)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER N,M,I,J,K
      DIMENSION B(12),Q(12,12),R(12,12)
C
C
      N=12
      IF (M .LE. N) THEN
      IF (M .EQ. 1) THEN
C
C INITIALIZATION
C
      DO 10 I=1,N
      DO 20 J=1,N
      Q(I,J)=0.0
   20 CONTINUE
      Q(I,I)=1.0
   10 CONTINUE
      ENDIF
C
C TRANSFORM B VECTOR
C
      DO 30 I=1,N
      SUM=0.0
      DO 40 J=1,N
      SUM=SUM+Q(J,I)*B(J)
   40 CONTINUE
      R(I,M)=SUM
   30 CONTINUE
      IF (M .LT. N) THEN
C
C HOUSEHOLDER TRANSFORMATION
C
      RO=0.0
      DO 50 I=M,N
      RO=RO+R(I,M)*R(I,M)
   50 CONTINUE
      RO=DSQRT(RO)
      IF (R(M,M).LT.0.0) RO=-RO
      B(M)=R(M,M)+RO
      C=DSQRT(RO*B(M))
      IF (C.GT.0.0) THEN
      R(M,M)=-RO
      B(M)=B(M)/C
      DO 60 I=M+1,N
      B(I)=R(I,M)/C
      R(I,M)=0.0
   60 CONTINUE
C
C ACCUMULATE ORHTOGONAL TRANSFORMATIONS
C
      DO 70 I=1,N
      SUM=0.0
      DO 80 K=M,N
```

```
      SUM=SUM+Q(I,K)*B(K)
   80 CONTINUE
      DO 90 J=M,N
      Q(I,J)=Q(I,J)-SUM*B(J)
   90 CONTINUE
   70 CONTINUE
      ENDIF
      ENDIF
      ELSE
      M=N
      DO 100 I=1,N
      DO 110 J=2,N
      R(I,J-1)=R(I,J)
  110 CONTINUE
  100 CONTINUE
      DO 120 I=1,N
      SUM=0.0
      DO 130 J=1,N
      SUM=SUM+Q(J,I)*B(J)
  130 CONTINUE
      R(I,N)=SUM
  120 CONTINUE
      DO 140 K=1,N-1
      RO=DSQRT(R(K,K)*R(K,K)+R(K+1,K)*R(K+1,K))
      IF (R(K,K).LT.0.0) RO=-RO
      BK=R(K,K)+RO
      BKP=R(K+1,K)
      C=RO*BK
      IF (C.NE.0.0) THEN
      R(K,K)=-RO
      R(K+1,K)=0.0
      DO 150 J=K+1,N
      RO=BK*R(K,J)+BKP*R(K+1,J)
      R(K,J)=R(K,J)-RO*BK/C
      R(K+1,J)=R(K+1,J)-RO*BKP/C
  150 CONTINUE
      DO 160 I=1,N
      RO=Q(I,K)*BK+Q(I,K+1)*BKP
      Q(I,K)=Q(I,K)-RO*BK/C
      Q(I,K+1)=Q(I,K+1)-RO*BKP/C
  160 CONTINUE
      ENDIF
  140 CONTINUE
      ENDIF
      RETURN
      END
C
C   THIS SUBROUTINE IS TO CALCULATE THE GENERALIZED COOR-
C   DINATES, GENERALIZED SPEEDS AND THEIR DERIVATIVES.
C
C
C   THIS SUB ROUTINE IS TO AUTOMATICALLY GENERATE
C   THE GENERALIZED ACTIVE FORCES, THE GENERALIZED
C   INERTIA FORCES, AND FINALLY CALCULATE THE TORQUES
C   /OR FORCES.
```



```
C
      SUBROUTINE AUTOSUB(QU,FN)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I,ITIME,J,K,L,M,NN,N,N2,R,DELTA
      PARAMETER(NN=6)
      PARAMETER(N2=12)
      DIMENSION VECTOR(N2),FN(N2),QU(N2),QOLD(N2),KA(NN),QOLDEST(N2)
      DIMENSION A(NN,3,3),AD(NN,3,3)
      DIMENSION W(NN,3),V(NN,3),S(NN+1,3),PW(NN,3,NN),
     .          PV(NN,3,NN),PS(NN+1,3,NN)
      DIMENSION PWD(NN,3,NN),PVD(NN,3,NN),PSD(NN+1,3,NN)
      DIMENSION DELTA(NN,2)
      DIMENSION MI(NN,3,NN),MID(NN,3,NN),B(NN,3),D(NN,3),E(NN,3),H(NN,3)
      DIMENSION Z(NN),Y(NN,3),P(NN,3,NN),PD(NN,3,NN)
      DIMENSION IN(NN,3),MS(NN),DL(NN,3),LL(NN,3)
      DIMENSION F(NN,NN),KSTAR(NN),KP2(NN),TF(NN)
      DIMENSION Q(NN),U(NN),UD(NN)
      COMMON/BLK1/QOLD,VECTOR,BTA,TF,QOLDEST,TSTEP,ITIME
      COMMON/BLK2/LL,DL,MS,IN
      COMMON/BLK3/N,L,DELTA
C
C  ASSIGN NEW VALUES TO Q(I) AND U(I)
C
      DO 70 I=1,N
      Q(I)=(1.-BTA)*QOLD(I)+QU(I)*BTA
      U(I)=(1.-BTA)*QOLD(I+N)+QU(I+N)*BTA
   70 CONTINUE
C
C  COMPUTE NEW VECTOR
C
      DO 90 I=1,N2
      IF(ITIME.EQ.1) THEN
      VECTOR(I)=(QU(I)-QOLD(I))/TSTEP
      ELSE
      VECTOR(I)=(3.*QU(I)-4.*QOLD(I)+QOLDEST(I))/(2.*TSTEP)
      ENDIF
   90 CONTINUE
      DO 100 I=1,N
      FN(I)=VECTOR(I)-U(I)
  100 CONTINUE
C
C     FN(1)=VECTOR(1)-(U(1)*DCOS(Q(3))-U(2)*DSIN(Q(3)))/DSIN(Q(2))
C     FN(2)=VECTOR(2)+U(1)*DSIN(Q(3))+U(2)*DCOS(Q(3))
C     FN(3)=VECTOR(3)-U(3)-(U(2)*DSIN(Q(3))-U(1)*DCOS(Q(3)))*DCOS(Q(2))
C    .      /DSIN(Q(2))
C     FN(4)=VECTOR(4)-U(4)
C     FN(5)=VECTOR(5)-U(5)
C     FN(6)=VECTOR(6)-U(6)
C  DEFINE TRANSFORMATION MATRICES AND THEIR DERIVATIVES
C
      G=9.81
      DO 110 K=1,N
      DO 110 I=1,3
      DO 110 J=1,3
      A(K,I,J)=0.0
```

```
      AD(K,I,J)=0.0
  110 CONTINUE
      A(1,1,1)=DCOS(Q(1))
      A(1,1,3)=DSIN(Q(1))
      A(1,2,1)=DSIN(Q(1))
      A(1,2,3)=-DCOS(Q(1))
      A(1,3,2)=1.0
      A(2,1,1)=DCOS(Q(2))
      A(2,1,3)=-DSIN(Q(2))
      A(2,2,1)=DSIN(Q(2))
      A(2,2,3)=DCOS(Q(2))
      A(2,3,2)=-1.0
      A(3,1,1)=DCOS(Q(3))
      A(3,1,2)=-DSIN(Q(3))
      A(3,2,1)=DSIN(Q(3))
      A(3,2,2)=DCOS(Q(3))
      A(3,3,3)=1.0
      A(4,1,1)=DCOS(Q(4))
      A(4,1,2)=-DSIN(Q(4))
      A(4,2,1)=DSIN(Q(4))
      A(4,2,2)=DCOS(Q(4))
      A(4,3,3)=1.0
      A(5,1,1)=DCOS(Q(5))
      A(5,1,3)=-DSIN(Q(5))
      A(5,2,1)=DSIN(Q(5))
      A(5,2,3)=DCOS(Q(5))
      A(5,3,2)=-1.0
      A(6,1,1)=DCOS(Q(6))
      A(6,1,2)=-DSIN(Q(6))
      A(6,2,1)=DSIN(Q(6))
      A(6,2,2)=DCOS(Q(6))
      A(6,3,3)=1.0
C
C
      AD(1,1,1)=-DSIN(Q(1))*U(1)
      AD(1,1,3)=DCOS(Q(1))*U(1)
      AD(1,2,1)=DCOS(Q(1))*U(1)
      AD(1,2,3)=DSIN(Q(1))*U(1)
      AD(2,1,1)=-DSIN(Q(2))*U(2)
      AD(2,1,3)=-DCOS(Q(2))*U(2)
      AD(2,2,1)=DCOS(Q(2))*U(2)
      AD(2,2,3)=-DSIN(Q(2))*U(2)
      AD(3,1,1)=-U(3)*DSIN(Q(3))
      AD(3,1,2)=-U(3)*DCOS(Q(3))
      AD(3,2,1)=DCOS(Q(3))*U(3)
      AD(3,2,2)=-DSIN(Q(3))*U(3)
      AD(4,1,1)=-U(4)*DSIN(Q(4))
      AD(4,1,2)=-U(4)*DCOS(Q(4))
      AD(4,2,1)=DCOS(Q(4))*U(4)
      AD(4,2,2)=-DSIN(Q(4))*U(4)
      AD(5,1,1)=-DSIN(Q(5))*U(5)
      AD(5,1,3)=-DCOS(Q(5))*U(5)
      AD(5,2,1)=DCOS(Q(5))*U(5)
      AD(5,2,3)=-DSIN(Q(5))*U(5)
      AD(6,1,1)=-DSIN(Q(6))*U(6)
```



```
      AD(6,1,2)=-DCOS(Q(6))*U(6)
      AD(6,2,1)=DCOS(Q(6))*U(6)
      AD(6,2,2)=-DSIN(Q(6))*U(6)
C
C  INITIALIZATIONS
C
      DO 115 K=1,N
      Z(K)=0.0
      KSTAR(K)=0.0
      KP2(K)=0.0
      DO 115 J=1,3
      W(K,J)=0.0
      U(K,J)=0.0
      S(K,J)=0.0
      B(K,J)=0.0
      D(K,J)=0.0
      E(K,J)=0.0
      Y(K,J)=0.0
      DO 115 R=1,N
      PW(K,J,R)=0.0
      PWD(K,J,R)=0.0
      PV(K,J,R)=0.0
      PVD(K,J,R)=0.0
      PS(K,J,R)=0.0
      PSD(K,J,R)=0.0
      P(K,J,R)=0.0
      PD(K,J,R)=0.0
      MI(K,J,R)=0.0
      MID(K,J,R)=0.0
      F(K,R)=0.0
  115 CONTINUE
C
C  PARTIAL ANGULAR VELOCITIES
C
      IF (DELTA(1,1).EQ.1) THEN
      DO 120 J=1,3
      PW(1,J,1)=A(1,3,J)
      PWD(1,J,1)=AD(1,3,J)
  120 CONTINUE
      ENDIF
      DO 150 K=2,N
      DO 150 J=1,3
      DO 130 R=1,K-1
      DO 130 I=1,3
      PW(K,J,R)=PW(K,J,R)+PW(K-1,I,R)*A(K,I,J)
      PWD(K,J,R)=PWD(K,J,R)+PWD(K-1,I,R)*A(K,I,J)+PW(K-1,I,R)*AD(K,I,J)
  130 CONTINUE
      IF (DELTA(K,1).EQ.1) THEN
      PW(K,J,K)=A(K,3,J)
      PWD(K,J,K)=AD(K,3,J)
      ELSE
      PW(K,J,K)=0.0
      PWD(K,J,K)=0.0
      ENDIF
  150 CONTINUE
```

```
C
C  ANGULAR VELOCITIES
C
      DO 160 K=1,N
      DO 160 J=1,3
      DO 160 R=1,K
      W(K,J)=W(K,J)+PW(K,J,R)*U(R)
  160 CONTINUE
C
C  PARTIAL VELOCITIES
C
      IF (DELTA(1,1).EQ.1) THEN
      PV(1,1,1)=PW(1,2,1)*DL(1,3)-PW(1,3,1)*DL(1,2)
      PV(1,2,1)=PW(1,3,1)*DL(1,1)-PW(1,1,1)*DL(1,3)
      PV(1,3,1)=PW(1,1,1)*DL(1,2)-PW(1,2,1)*DL(1,1)
      PS(2,1,1)=PW(1,2,1)*LL(1,3)-PW(1,3,1)*LL(1,2)
      PS(2,2,1)=PW(1,3,1)*LL(1,1)-PW(1,1,1)*LL(1,3)
      PS(2,3,1)=PW(1,1,1)*LL(1,2)-PW(1,2,1)*LL(1,1)
      ELSE
      PV(1,3,1)=1.0
      PS(2,3,1)=1.0
      ENDIF
      DO 180 K=2,N
      IF (DELTA(K,1).EQ.0) THEN
      DO 165 R=1,K-1
      DO 163 J=1,3
      DO 163 I=1,3
      MI(K,J,R)=MI(K,J,R)+PS(K,I,R)*A(K,I,J)
      MID(K,J,R)=MID(K,J,R)+PSD(K,I,R)*A(K,I,J)+PS(K,I,R)*AD(K,I,J)
  163 CONTINUE
      PV(K,1,R)=MI(K,1,R)+(DL(K,3)+Q(K))*PW(K,2,R)-DL(K,2)*PW(K,3,R)
      PV(K,2,R)=MI(K,2,R)+DL(K,1)*PW(K,3,R)-(DL(K,3)+Q(K))*PW(K,1,R)
      PV(K,3,R)=MI(K,3,R)+DL(K,2)*PW(K,1,R)-DL(K,1)*PW(K,2,R)
      PS(K+1,1,R)=MI(K,1,R)+(LL(K,3)+Q(K))*PW(K,2,R)-DL(K,2)*PW(K,3,R)
      PS(K+1,2,R)=MI(K,2,R)+LL(K,1)*PW(K,3,R)-(LL(K,3)+Q(K))*PW(K,1,R)
      PS(K+1,3,R)=MI(K,2,R)+LL(K,2)*PW(K,1,R)-LL(K,1)*PW(K,2,R)
      PVD(K,1,R)=MID(K,1,R)+(DL(K,3)+Q(K))*PWD(K,2,R)+U(K)*PW(K,2,R)
     ,          -DL(K,2)*PWD(K,3,R)
      PVD(K,2,R)=MID(K,2,R)+DL(K,1)*PWD(K,3,R)-U(K)*PW(K,1,R)
     ,          -(DL(K,3)+Q(K))*PWD(K,1,R)
      PVD(K,3,R)=MID(K,3,R)+DL(K,2)*PWD(K,1,R)-DL(K,1)*PWD(K,2,R)
      PSD(K+1,1,R)=MID(K,1,R)+(LL(K,3)+Q(K))*PWD(K,2,R)+U(K)*PW(K,2,R)
     ,            -LL(K,2)*PWD(K,3,R)
      PSD(K+1,2,R)=MID(K,2,R)+LL(K,1)*PWD(K,3,R)-(LL(K,3)+Q(K))*
     ,            PWD(K,1,R)-U(K)*PW(K,1,R)
      PSD(K+1,3,R)=MID(K,3,R)+LL(K,2)*PWD(K,1,R)-LL(K,1)*PWD(K,2,R)
  165 CONTINUE
      PV(K,3,K)=1.0
      PS(K+1,3,K)=1.0
      ELSE
      DO 175 R=1,K
      DO 170 J=1,3
      DO 170 I=1,3
      P(K,J,R)=P(K,J,R)+PS(K,I,R)*A(K,I,J)
      PD(K,J,R)=PD(K,J,R)+PSD(K,I,R)*A(K,I,J)+PS(K,I,R)*AD(K,I,J)
```



```
  170 CONTINUE
      PV(K,1,R)=PV(K,1,R)+P(K,1,R)+PW(K,2,R)*DL(K,3)-PW(K,3,R)*DL(K,2)
      PV(K,2,R)=PV(K,2,R)+P(K,2,R)+PW(K,3,R)*DL(K,1)-PW(K,1,R)*DL(K,3)
      PV(K,3,R)=PV(K,3,R)+P(K,3,R)+PW(K,1,R)*DL(K,2)-PW(K,2,R)*DL(K,1)
      PS(K+1,1,R)=PS(K+1,1,R)+P(K,1,R)+PW(K,2,R)*LL(K,3)-PW(K,3,R)
     ,           *LL(K,2)
      PS(K+1,2,R)=PS(K+1,2,R)+P(K,2,R)+PW(K,3,R)*LL(K,1)-PW(K,1,R)
     ,           *LL(K,3)
      PS(K+1,3,R)=PS(K+1,3,R)+P(K,3,R)+PW(K,1,R)*LL(K,2)-PW(K,2,R)
     ,           *LL(K,1)
      PVD(K,1,R)=PVD(K,1,R)+PD(K,1,R)+PWD(K,2,R)*DL(K,3)-PWD(K,3,R)
     ,           *DL(K,2)
      PVD(K,2,R)=PVD(K,2,R)+PD(K,2,R)+PWD(K,3,R)*DL(K,1)-PWD(K,1,R)
     ,           *DL(K,3)
      PVD(K,3,R)=PVD(K,3,R)+PD(K,3,R)+PWD(K,1,R)*DL(K,2)-PWD(K,2,R)
     ,           *DL(K,1)
      PSD(K+1,1,R)=PSD(K+1,1,R)+PD(K,1,R)+PWD(K,2,R)*LL(K,3)-PWD(K,3,R)
     ,           *LL(K,2)
      PSD(K+1,2,R)=PSD(K+1,2,R)+PD(K,2,R)+PWD(K,3,R)*LL(K,1)-PWD(K,1,R)
     ,           *LL(K,3)
      PSD(K+1,3,R)=PSD(K+1,3,R)+PD(K,3,R)+PWD(K,1,R)*LL(K,2)-PWD(K,2,R)
     ,           *LL(K,1)
  175 CONTINUE
      ENDIF
  180 CONTINUE
C
C  VELOCITIES
C
      DO 200 K=1,N
      DO 200 J=1,3
      DO 200 R=1,K
      V(K,J)=V(K,J)+PV(K,J,R)*U(R)
      S(K,J)=S(K,J)+PS(K,J,R)*U(R)
  200 CONTINUE
C
C  CALCULATE INTERMEDIATE VARIABLES
C
      DO 220 K=1,N
      B(K,1)=W(K,2)*V(K,3)-W(K,3)*V(K,2)
      B(K,2)=W(K,3)*V(K,1)-W(K,1)*V(K,3)
      B(K,3)=W(K,1)*V(K,2)-W(K,2)*V(K,1)
      DO 210 J=1,3
      DO 210 R=1,K
      D(K,J)=D(K,J)+PVD(K,J,R)*U(R)
      E(K,J)=E(K,J)+PWD(K,J,R)*U(R)
  210 CONTINUE
      H(K,1)=IN(K,1)*E(K,1)+W(K,2)*W(K,3)*(IN(K,3)-IN(K,2))
      H(K,2)=IN(K,2)*E(K,2)+W(K,3)*W(K,1)*(IN(K,1)-IN(K,3))
      H(K,3)=IN(K,3)*E(K,3)+W(K,1)*W(K,2)*(IN(K,2)-IN(K,1))
  220 CONTINUE
      DO 230 K=1,N
      DO 230 J=1,3
      D(K,J)=D(K,J)+B(K,J)
  230 CONTINUE
C
```

```
C  CALCULATE INTERMEDIATE VARIABL Z'S
C
      DO 240 R=1,N
      DO 240 K=1,N
      DO 240 J=1,3
      Z(R)=Z(R)+PW(K,J,R)*H(K,J)+MS(K)*PV(K,J,R)*D(K,J)
  240 CONTINUE
C
C  CALCULATE INERTIA COEFFICIENTS
C
      DO 250 R=1,N
      DO 250 M=1,R
      DO 245 J=1,3
      DO 245 K=1,N
      F(R,M)=F(R,M)+IN(K,J)*PW(K,J,M)*PW(K,J,R)+MS(K)*PV(K,J,M)
     .        *PV(K,J,R)
      F(M,R)=F(R,M)
  245 CONTINUE
  250 CONTINUE
C
C  GENERALIZED INERTIA FORCES
C
      DO 260 R=1,N
      DO 255 M=1,N
      KSTAR(R)=KSTAR(R)-F(R,M)*UD(M)
  255 CONTINUE
      KSTAR(R)=KSTAR(R)-Z(R)
  260 CONTINUE
C
C  INGREDIENTS OF GENERALIZED ACTIVE FORCES
C
      DO 270 J=1,3
      Y(1,J)=A(1,L,J)
  270 CONTINUE
      DO 280 K=2,N
      DO 280 J=1,3
      DO 280 I=1,3
      Y(K,J)=Y(K,J)+Y(K-1,I)*A(K,I,J)
  280 CONTINUE
      DO 290 R=1,N
      DO 290 K=1,N
      DO 290 J=1,3
      KP2(R)=KP2(R)+MS(K)*G*Y(K,J)*PV(K,J,R)
  290 CONTINUE
C
C  GENERALIZED ACTIVE FORCES
C
      DO 300 R=1,N
      KA(R)=IF(R)+KP2(R)
  300 CONTINUE
      DO 310 R=1,N
      FN(R+N)=KA(R)+KSTAR(R)
  310 CONTINUE
      WRITE(*,400) ' FN: ',(FN(J),J=1,12)
  400 FORMAT(A,6E11.5/5X,6E11.5)
```



```
      RETURN
      END
C
C THIS SUBROUTINE COMPUTE THE VALUES OF TORQUE FUNCTIONS.
C
      SUBROUTINE TORQUE (TF,Q,V)
      IMPLICIT DOUBLE PRECISION (A-Z)
      INTEGER I
      DIMENSION TQ(12),Q(12),V(12)
C
C  DEFINE YOUR FUNCTION EXPRESSIONS OF GENERALIZED ACTIVE
C  FORCES OR/TORQUES IN THIS PART IF YOU INPUT IS FUNCTIONS.
C  SUCH AS:
      DO 45 I=1,6
      TQ(I)=0.0
   45 CONTINUE
      RETURN
      END
```