Modeling joint kinetics in the Tkatchev release move

Martha Nichols-Ketchum
Iowa State University

Follow this and additional works at: http://lib.dr.iastate.edu/rtd
Part of the Applied Mechanics Commons, and the Biomedical Engineering and Bioengineering Commons

Recommended Citation
http://lib.dr.iastate.edu/rtd/10499

This Dissertation is brought to you for free and open access by Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI
University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700  800/521-0600
Modeling joint kinetics in the Tkatchev release move

Nichols-Ketchum, Martha Ann, Ph.D.

Iowa State University, 1994
Modeling joint kinetics in the Tkatchev release move

by

Martha Nichols-Ketchum

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Interdepartmental Program: Biomedical Engineering
Major: Biomedical Engineering

Approved:
Signature was redacted for privacy.

In Charge of Major Work
Signature was redacted for privacy.

For the Interdepartmental Program
Signature was redacted for privacy.

For the Graduate College

Iowa State University
Ames, Iowa

1994
# TABLE OF CONTENTS

**INTRODUCTION** .......................... 1

**REVIEW OF LITERATURE** .......................... 5

- The Tkatchev .................................. 5
- Giant Swing .................................. 8
- Computer Simulation Models .................. 10
  - Human body models/dimensionality ........ 11
  - Use of forward or inverse dynamic solutions 15
  - Data acquisition .......................... 16
  - Data processing ............................ 17
- Gymnastics Injuries ........................... 19

**THEORY, MATERIALS AND METHODS** ............. 22

- Preliminary Investigations ................... 23
- Analysis ................................... 25
  - Subjects ................................ 25
  - Strength and flexibility .................. 27
  - Anthropometry ............................ 27
  - Calibration of space for Ariel system .... 29
  - Determination of bar forces from strain gage data 39
  - Videotape protocol ......................... 47
  - Data reduction from motion analysis system 50
  - Determination of rotation matrices ....... 54
Calculation of kinematics of each segment. . . . . . . . 59
Determination of joint forces and torques. . . . . . . . 62
Determination of joint rotation angles. . . . . . . . . . . 64
Output from analysis. . . . . . . . . . . . . . . . . . . . . . . . 65
Theoretical determination of bar deflections. . . . . . . . 66

Development of Equations for the
Computer Simulation Model . . . . . . . . . . . . . . . . . . . 67

Validation of Computer Simulation Model. . . . . . . . . . . . 74
Running the Computer Simulation. . . . . . . . . . . . . . . . . 75

RESULTS AND DISCUSSION. . . . . . . . . . . . . . . . . . . . . 78

Part 1—Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 78
Strength and range of motion data. . . . . . . . . . . . . . . . . 78
Range of motion of joints
during the Tkatchev performance. . . . . . . . . . . . . . . . . 80
Peak forces and torques during Tkatchev performance. . . 84
Center of gravity trajectory. . . . . . . . . . . . . . . . . . . . . 98
Comparison of theoretical bar deflections with
measured bar deflections. . . . . . . . . . . . . . . . . . . . . . . . . 99
Comparison of analysis results with
Ariel default calculations. . . . . . . . . . . . . . . . . . . . . . . . 103
Comparison of kinematics with
previously published data. . . . . . . . . . . . . . . . . . . . . . . . 105
Results of the redundant equations
for the force at the lower back
and the moment at the upper back. . . . . . . . . . . . . . . . . . 107
Modeling assumptions and errors involved. . . . . . . . . . . . 108
Experimental problems/suggestions for further study. . . . 113
Limitations of the experimental approach. . . . . . . . . . . . 115
**Part 2—Modeling and Simulation**

- Validation of model 117
- Results of experiments with simulation program 126
- Problems in simulation model and simulation program 127
- Recommendations for further study 130

**CONCLUSIONS** 133

**REFERENCES** 135

**ACKNOWLEDGEMENTS** 140

**APPENDIX A:** DERIVATION OF EQUATIONS FOR SIMULATION 142

**APPENDIX B:** INFORMED WRITTEN CONSENT FORM FOR PARTICIPANTS IN A STUDY OF RELEASE MOVES ON THE HORIZONTAL BAR 185

**APPENDIX C:** ANTHROPOMETRIC MEASUREMENTS 187

**APPENDIX D:** PROGRAM TO SYNCHRONIZE STRAIN GAGE DATA WITH THE TIME DATA FOR THE DIGITIZED MARKER LOCATIONS 191

**APPENDIX E:** ANALYSIS PROGRAMS AND SUBROUTINES 193

**APPENDIX F:** THEORETICAL VALUES FOR MARKER LOCATIONS FROM HANAVAN MODEL USING THE SEGMENT CENTER OF MASS AS THE ORIGIN POINT 238

**APPENDIX G:** COMPUTER PROGRAM FOR SIMULATION 240
INTRODUCTION

Competitive gymnastics at the college and elite levels has become progressively more difficult and dangerous over the past 15 to 20 years. Gymnasts are now given bonus points for "courage, originality and virtuosity" in routines. This has led to continual development of unprecedented skills. "Since skills that are being pioneered have no history for the coach and gymnast to call upon, the method of teaching them revolves around trial and error . . . . Mistakes in learning usually result in falls that are potentially dangerous if not properly guarded against. Further, the errors of performance may bring about unpredictable injuries due to abnormal loading of tissue structures by continued repetition of the same error over and over" (Sands 1981, p.1). Release moves on the horizontal bar have become increasingly original and potentially dangerous. Little quantitative data has been published which can provide information on the optimal performance of release moves or on factors that affect performer safety.

The present work develops a computer simulation model of the Tkatchev release move (see Figure 1). The model predicts the orientation of the body, position of the body, and resulting joint forces and torques during the skill performance. The model is based on the inverse dynamic solution using Newtonian mechanics and is three dimensional in nature.
Figure 1: The Tkatchev release move (Fink 1987)

The Tkatchev is performed on the men's horizontal bar. The skill was devised by a Soviet biomechanist before being attempted by a gymnast (Nissinen et al. 1985). The Tkatchev, also called the reverse hecht in the literature, can be described in four phases: the giant swing, the release, the flight and the recatch. During the giant swing, the gymnast may planche (let the shoulders lean forward over the high bar) at the top of the swing to improve control of the move (Talecky and Steel 1991). During the downswing the gymnast "beats" by hyperextending the hips and then strongly flexes during the upswing to attain momentum (Fink 1987, Sands 1981). At the point of release, the gymnast hyperextends the hips and hyperflexes the shoulders, then stops the hip extension.
suddenly to initiate a forward rotation during flight (Fink 1987). During flight, the gymnast completes a quarter somersault forward in a straddled, piked position and then extends the body before recatch. A good recatch will exhibit little or no "jerk" and the gymnast will move smoothly into the next skill.

There are three key points of interest in the performance of the skill related to the forces produced. The first is at the bottom of the giant swing where the gymnast "beats." The next is at the point of release, where there is forcible hyperflexion of the shoulder and extension of the hips. The third is at recatch, where the path of the center of gravity and the body orientation influence the forces produced.

The present study was done in two parts. Part 1 was a quantitative description of reverse hecht performance and joint kinetics for two different performers. The results were derived from video analysis of several performances. Part 2 was the development of the computer simulation model that predicts kinematics, resultant joint forces and resultant joint torques in the reverse hecht.

For Part 1, two college-level male gymnasts were videotaped performing the Tkatchev on an instrumented horizontal bar. Strain gages were applied to provide data on horizontal and vertical reaction forces on the bar. Joint
forces and torques were determined using an inverse dynamic algorithm.

The general pattern of movement found in these studies was used as a template of the skill for the computer simulation model. The model uses the template, along with the user-input parameters, to calculate the performance characteristics and joint forces and torques of the move when selected input parameters are altered.

The computer simulation model of the Tkatchev was intended as a tool for coaches and researchers who want to assess the effects of changes in performance parameters of the forces and torques at the gymnast’s joints. The model represented by the computer program is transparent to the user, so reprogramming is not necessary each time a change is made. This involves a menu-driven program that permits qualitative input and provides graphical output.

Parameters which can be investigated with the current model include the angular speed of the giant swing, the inertial parameters of the gymnast, the amount of friction at the hand, the range of motion at the joints, the timing of the release, and the material and size of the bar. This model could readily be extended to include other parameters as variables.
The literature reviewed consisted of studies on the Tkatchev and related skills such as the overgrip giant swing, which leads into the release. Computer simulation models used in sports biomechanics, gait analysis and automotive crash modeling were reviewed. In addition, a brief review of the frequency and extent of gymnastics injuries is included.

The Tkatchev

Researchers have investigated the kinematics associated with the Tkatchev, but to date no kinetic studies have been reported. Gervais and Tally (1993) examined the kinematics of the beat swing and the release of three release moves: the Tkatchev, the Geinger and the Jaeger. They filmed the release skills during a competition and had an international judge rate the skills for execution. The investigators correlated mechanical variables with the judge's score. Nine Tkatcheves were analyzed. Two were initiated from a one-arm takeoff, while the rest were initiated from an overgrip giant swing. The study found an average takeoff angle of 52° and an average speed at release of 3.31 m/sec. The center of mass averaged .52 m above the bar at release, .90 m above the bar at the peak of the flight, and .24 m above the bar at the recatch. At release, the average angular momentum was 24.94 kg·m²/sec. High scores from the judge were correlated with a lower
maximum height of flight, a lower release speed and a lower takeoff angle. All these characteristics contributed to increased horizontal displacement. The average hip angle at the start and the end of the beat was $-40^\circ$ and $59^\circ$, respectively. Gervais and Tally also reported that the Tkatcheves had greater shoulder motion at release than the other skills reviewed.

Prassas (1990) conducted a case study analysis of the reverse hecht (Tkatchev) on the horizontal bar. Using the Ariel Performance Analysis System, he examined film of gymnasts performing the Tkatchev. He presented kinematic data from two Tkatcheves and found that the path of the center of gravity was noticeably flatter for the unsuccessful Tkatchev than for the successful one. He compared some kinematic parameters of a successful and unsuccessful performance and found significant differences in horizontal displacement of the mass center and the trajectory angle. These results differ from the conclusions of Gervais and Tally (1993) who reported better scores for flatter trajectories (lower takeoff angle). This implies that an optimal trajectory exists and that comparative studies with small numbers of subjects may be misleading.

Jinggang et al. (1989) analyzed the Tkatchev to optimize the performance of one gymnast. For this gymnast, they found the optimal joint angles throughout the giant swing, the
optimal release point, and the optimal trajectory for the flight phase. Although velocities and joint angles were analyzed, no kinetic parameters were calculated.

Sands (1981) conducted a kinematic analysis of the Tkatchev as executed by an elite female gymnast. While no force analysis was done, Sands expressed concern that hyperlordosis in the giant swing beat and before the release may damage the gymnast’s back. He also noted that the recoils of the bar aid the gymnast in completing the move.

Fink (1987) published a theoretical analysis of the factors that affect the performance of the Tkatchev. He listed three primary factors:

1) transfer of angular momentum at the point of release, when the rotation of the legs is suddenly stopped and angular momentum is transferred to the entire body;
2) action-reaction, in which the hip extension before release causes hyperflexion at the shoulder, resulting in a downward force on the bar and hip flexion during flight causing the torso to flex toward the legs;
3) reaction force, in which the bar provides an upward force on the body equal to the downward force applied to the bar by the gymnast’s hand and provides a torque causing forward rotation about the center of gravity.
Giant Swing

The giant swing is the preparatory skill for the reverse hecht part of the Tkatchev. Velocities developed during the giant swing govern the success or failure of the flight phase. Because the giant swing is a lead-in or connecting move in many horizontal bar and uneven parallel bar skills, it has been studied extensively.

Smith (1981) estimated that, during the giant swing, a female gymnast exerts forces on the bar equal to five times the body weight of the performer. He proposed that training programs be structured to ensure that gymnasts can maintain their grip when exposed to forces of this magnitude. Yamashita et al. (1979) also documented forces of up to five times the performer’s body weight during the performance of a giant swing.

Kopp and Reid (1980) analyzed forces and torques exerted on the high bar during forward and backward giant swings. They proposed that deviations from the "standard" center of gravity curve would be useful in correction of giant swing execution errors. They reported maximum forces ranging from 3.45 to 3.70 times the body weight of the gymnast occurring in the third quadrant, just past the bottom of the swing.

Cheetham (1985) analyzed the trajectory and the angular velocity of the mass center for three giant swing variations on the horizontal bar. The three swings analyzed were the
regular giant, used to join skills; the dismount giant, which contains the release for the dismount; and the windup giant, which is used to accelerate the gymnast for the dismount. Cheetham found that the path of the center of gravity was a tilted oval shape for the dismount giant, probably due to increased piking (hip flexion) during the upswing. In addition, the angular velocity was highest during the dismount giant.

Witten (1990) conducted a kinematic and kinetic analysis of the overgrip giant swing on the uneven parallel bars using club level female gymnasts, aged 10-16, as subjects. Witten chose her experimental subjects to represent in age the population of female gymnasts participating in the sport. Witten used Dempster’s segmental data (Dempster 1955, cited by Witten 1990) for inertial properties of the body and calculated muscle moments and joint forces at the elbow and shoulder. Maximum forces at the elbow and shoulder were about twice the gymnast’s weight, and mean maximum moments were 96.8 N·m at the elbow and 134.3 N·m at the shoulder. The maximum force exerted on the bar had a mean value of 3.1 times the gymnast’s weight. In analyzing force data, Witten assumed that forces on the bar acted at the center of the wrist joint and that friction forces between the gymnast’s hands and the bar were negligible.
Many of the concepts in kinematic and kinetic analysis used by these authors were adapted for use in the current work. The data from the giant swing and the Tkatchev were used as a basis of comparison for the values calculated in Part 1. Although there have been several studies on the Tkatchev and its lead-in skill, the giant swing, there have not been any studies that are three-dimensional, determine kinetic parameters, and model the skill under various circumstances. The current study addresses these issues.

Computer Simulation Models

Vaughan (1984) defines a computer simulation model as "the use of a validated computer model to carry out 'experiments,' under carefully controlled conditions on the real-world system that has been modeled." (p.373) A computer model was defined as "the setting up of mathematical equations to describe the system of interest, the gathering of appropriate input data, and the incorporation of these equations into a computer program." (p.373) Development of the model requires determination of the human body model to use, the number of dimensions to use, and whether to use forward or inverse dynamics.
Human body models/dimensionality

The human body is composed of complex, irregular shapes of nonhomogeneous density. Its joints are a combination of sliding joints, and joints that rotate about one, two or three axes. The shoulder joint and scapula have several possible directions of motion, and in actuality, most of the joints have six degrees of freedom (three rotations, and three translations). In order to have workable, efficient equations of motion, researchers have constructed idealized models of the body. The degree of sophistication has varied considerably, depending on the purpose of the model and the dimensionality (considering the body as planar or three dimensional).

Most human body models used in computer simulations have idealized the segments as rigid bodies, with the number of segments ranging from two or three up to seventeen. In planar modeling, researchers have often used linkage models for the human body. The segments have been idealized as thin rods with moments of inertia derived from tabulated data such as those from Dempster (1955, cited by Witten 1990, Dapena 1979, Youm et al. 1973, and Yeadon 1990). Dempster used a cadaver study to measure the moments of inertia of various segments under laboratory conditions.

Dapena (1979) used a three-dimensional model of the human body to study the kinematics of airborne movements. He used a
15-segment human body model, where all segments except the trunk were treated as thin rods with zero moments of inertia about the longitudinal axis. Segment masses and centers of mass were taken from Dempster’s data (1955, cited by Dapena 1979) and other inertial parameters were taken from Whitsett (1963, cited by Dapena 1979). The model allowed the two trunk segments to twist, but not bend, relative to each other, and did not include motion at the ankle and wrist. Ramey and Yang (1981) modeled the free fall phase of athletic activities using a nine segment human body model whose inertia parameters and segmental lengths were taken from data for the fiftieth percentile of United States Air Force Personnel (Scher and Kane 1969, cited by Ramey and Yang 1981). Youm et al. (1973) simulated kicking using a four-bar linkage in which mass and inertial properties were derived from Dempster (1955, cited by Youm et al. 1973).

Others have modeled segments as regular geometric shapes and joints as ball and socket joints or hinge joints. Hanavan (1964) used ellipsoids, frustra of cones and elliptical cylinders to model segments of the body. He utilized anthropometric measurements to determine segment geometric parameters. Hanavan’s model has been used by many investigators simulating and analyzing gymnastics skills. Nissinen et al. (1985) used this model in a two-dimensional, six-segment model of horizontal bar dismount kinematics.
Miller (1970) developed a kinematic simulation model of diving that was quasi-three-dimensional and used the Hanavan human body model. Miller limited her study to a four-segment model: trunk, legs, and two arms. The arms were the only segments that could move out of the sagittal plane. This limited the dives that could be analyzed to non-twisting dives in the pike or layout position. Spaepen et al. (1983) used the Hanavan model to develop a 15-segment homogeneous mass model with three degrees of freedom per joint. They used this model to investigate the take-off and flight phases of various gymnastics maneuvers. Van Gheluwe (1981) used Hanavan’s model to perform a three-dimensional computer simulation of an airborne backward twist somersault. Duck (1978) used a three-link planar model of the human body to model gymnasts’ movements on the horizontal bar. He used the properties found by the Hanavan model and reduced it to three links.

Yeadon (1990) developed a mathematical model of the human body which uses 40 geometric solids that are specified by 95 anthropometric measurements. Inertia parameters for body models up to 20 segments could be obtained using Yeadon’s equations. He assumed that segments are rigid bodies and that no movement occurs at the neck, wrists, or ankles. All body segments other than the head were represented by a number of
stadium solids or truncated cones. He used Dempster’s density data (1955, cited by Yeadon 1990) to calculate the mass of each segment or sub-segment.

In addition to athletic activities, human body models have been used in gait simulations and automotive simulation. In automotive applications, vibrations models have been extensively utilized. Segal (1970) used an eight segment pin-jointed human body model with coulomb friction and motion-limiting stops at the joints. Vulcan and King (1970) modeled the body using rigid bodies connected by linear and torsional springs and dampers. In gait simulations, pin jointed models like those used in athletic activities and vibrations models have both been used. Amirouche et al. (1990) used a two-dimensional model with linear and nonlinear springs and dampers at the joints. Ju and Mansour (1988) used two and three dimensional models with rigid links and pin joints. Siegler et al. (1982) used a concentrated mass supported by two elastic and viscous straight legs.

In the current research, it was decided to use a three-dimensional human body model. The mode used Hanavan’s (1964) properties for inertias and derives masses from Clauser et al. (1969). It extends Hanavan’s model to three trunk segments and neglects motion at the wrist and ankle. This approach was chosen to provide enough detail to find forces at the shoulders and at several points in the back, while not
requiring extensive anthropometric measurements or calculations to find inertial properties.

**Use of forward or inverse dynamic solutions**

The two primary methods of performing kinetic analyses are the forward method (forces and torques as input, kinematic data calculated) and the inverse solution (kinematic information is known, forces and torques are calculated).


The present study used a combination of the forward and inverse solution methods in the model. Because alterations in joint kinematics are the usual method of coaching, they were the primary input for the model. While the coach can view the joint angles and tell the gymnast to increase or decrease the angle, he/she cannot observe the amount of force or torque used to create the movement. The external forces were found
in terms of the joint angles and the whole body kinematics using a vibrations model, and moments were estimated as proportional to the forces. Joint forces and torques were found using the inverse approach after the kinematics were calculated from the forward approach.

Data acquisition

The analyses and models of various skills used similar cinematographic techniques to obtain kinematic data. The primary differences in techniques were in the use of video or film, the frame rate of the video or film used, the smoothing technique, and the number of cameras used.

Of those studying the Tkatchev, Gervais and Tally (1993) used one camera, filming at 50 frames per second (fps) and a Butterworth filter for smoothing the displacement data. Prassas (1990) videotaped the skill using two cameras at 200 and at 60 fps, and used a digital filter for smoothing. Sands (1981) used a single camera in the sagittal plane at 100 fps. Other investigators who analyzed horizontal bar skills included Cheetham (1985) who studied the giant swing using film at 50 fps and Duck (1978) who filmed using one camera at 200 fps. Those analyzing women’s gymnastics skills on the uneven parallel bars included Witten (1990), who filmed at 100 fps and Bentham (1987) who filmed at 150 fps. The present study used four video cameras, filming at 30 frames per
second. While a higher frame rate would have been preferable, equipment was not available to conduct higher frame rate studies. Although previous studies generally used film rather than videotape, Angulo and Dapena (1992) found that within the volume of the control object, video and film analyses are both precise enough "for most practical purposes." They also reported that although video was inferior for points outside the control volume, the average error in distances for landmarks outside the control volume was only 1.3%, which "may be acceptable" for some purposes. Preliminary studies showed that smooth and realistic acceleration curves could be obtained from 30 fps videotapes.

Data processing

Smoothing and differentiation techniques have been studied by several researchers. There is noise in the displacement data from motion of the markers on the skin, inaccuracies in digitizing, approximations in linear regression in the direct linear transformation (DLT), and other variables associated with the data acquisition process (Wood 1982). There are four basic approaches which have been utilized in smoothing and differentiation of biomechanical data. These are digital filtering followed by finite difference techniques, polynomial approximations, splines and Fourier series approximations. These have been reviewed and
compared by various researchers. Andrews et al. (1982) compared two digital filters, cubic spline approximations and two Fourier series approximations. Using two different sets of test data, they found that different approaches worked better for each set of data. They concluded that there is no one method that is best in all situations. Pezzack et al. (1977) compared Chebychev polynomial approximations with a digital filter followed by finite differences. They found that the digital filter provided the closest match to analog acceleration curves. Woltring (1985) discussed the choice of the optimal smoothing and differentiation techniques for different types of data. He noted that most of the techniques are interrelated and reported that optimally regularized quintic splines were useful for smoothing and differentiation. He also reported that since accelerations are ill-defined by noisy position data, a wide variety of acceleration patterns can yield identical displacement. Therefore, in smoothing, the researcher looks at minimizing or maximizing other factors, such as minimizing the amount of jerk. Woltring indicated that the techniques in data collection can have a great effect on the accuracy of displacement data and its derivatives. The sampling frequency must be high enough to provide accurate time history of the displacement if accurate derivatives are to be computed. Wood (1982) reviewed least squares polynomial approximations, splines, digital filtering
and Fourier analysis approaches. In comparing these methods he concluded that all procedures provided good fits to noisy displacement data except for polynomials. The best fit to the second derivatives was found using quintic splines and an optimally regularized Fourier filter. Digital filters and cubic splines also provided adequate results. He concluded that "any procedure that can be shown to produce valid results within the context of the motion being analyzed is acceptable."

Motion analysis labs generally use either a digital filter and finite differences or cubic splines (Wood 1982) for smoothing and differentiation. These approaches, along with quintic splines and Fourier series, were available for the current research. Digital filters, cubic splines and quintic splines were compared. Cubic splines were chosen due to the ease of use and options available. The smoothing results were comparable for all methods.

**Gymnastics Injuries**

The literature consistently identifies the sport of gymnastics as dangerous, with injury rates as high as those in wrestling, football and lacrosse (Mandelbaum et al. 1989). Caine et al. (1989) reported that 56% of female gymnasts' injuries occurred due to chronic overuse and 72.2% of lower back injuries recurred during a one year study period.
Mandelbaum et al. (1989) reported on wrist pain syndrome in gymnasts and found 75% of male gymnasts and 33% of female gymnasts surveyed had wrist pain for at least four months. Gymnasts with more years of participation were more likely to have an ulnar variance. Much of the wrist pain in males was correlated to their pommel horse performances. Repetitive compressive impact on the epiphyseal structures was suspected to have a cumulative negative effect on the normal growth mechanism.

Szot et al. (1985) reported on the radiological changes occurring in elite male gymnasts and the interrelationship between symptoms of pain and coexisting organic changes of the osteoarticular system. In this study of 41 men with 6-20 years of gymnastics training, 65.8% showed radiological changes in the spinal column, 59.8% in the shoulders, 73.2% in the elbows and 58.5% in the wrist joints. In most subjects the radiological changes were "of a malformative degenerative type." They found more damage with increased years of training. In addition, they reported that radiological malformations preceded the associated pain, since gymnasts early in their training reported less pain, although the damage was already present in their x-rays.

Silvij and Nocini (1982) and Nocini and Silvij (1982) found evidence that overloading during gymnastics causes chronic pain and structural alterations due to microtrauma at
the shoulder and elbow. They diagnosed conditions such as "gymnast’s elbow" and "gymnast’s shoulder."

These studies demonstrate that many gymnastics activities can cause chronic, permanent damage to the gymnast. If gymnasts are to retire and not live in pain for the rest of their lives, research into minimizing the causes of high stress levels on the joints is necessary. The current work provides a tool for determining the demands of some gymnastics skills on the body under various situations.
THEORY, MATERIALS AND METHODS

In this chapter, the mathematics and mechanics theory used to develop the simulation model, the materials used for the experiments and the methods followed in experimental and programming sections are discussed.

A flowchart of the process followed in developing the model is shown below in Figure 2. Each major section of this chapter includes a flowchart summarizing the steps taken in that portion of this research.

Figure 2: Flowchart of simulation process
Preliminary Investigations

Eight Tkatchevs were videotaped during the 1992 Big 8 Mens' Gymnastics Meet to gain experience in the use of the Ariel Performance motion analysis system and to obtain preliminary data on the forces and torques in the Tkatchev release move. These Tkatchevs were digitized by estimating joint centers and using Ariel data (based on Dempster 1955, cited by APAS Manual 1989) for segment moments of inertia and center of gravity location. The segments used were: hands, forearms, upper arms, three trunk segments, legs and head. The Ariel kinetics module was used to estimate the forces and torques in these Tkatchevs. The results were used to provide a basis of comparison for later experimental results as well as to determine the minimum number of segments required to model the human body in the Tkatchev.

In these preliminary investigations, most joints showed force peaks at three points in the move: the bottom of the giant swing, the release point, and the recatch. The largest force and torque peaks occurred during the giant swings. The highest force peaks were between one and six times the gymnasts' body weight, and peak torques about the z-axis (sagittal plane) were between 200 and 600 N·m. In general, the maximum torques in the giant swing were in the upper back, at the release were in the lower back, and at recatch were in
the shoulders and upper back. There were some indications of a trade-off between lower and upper body motion: the subjects with the highest lower body forces and torques experienced the lowest upper body forces and torques. Figure 3 shows example plots of peak forces and torques calculated using the Ariel system software.

![Diagram](image)

(a) Shoulder torques

![Diagram](image)

(b) Shoulder forces

Figure 3: Example plots from preliminary studies
Analysis

In the analysis portion of the research, gymnasts were videotaped performing the Tkatchev, strain gages were used to collect data from the bar, and computer programs were used to digitize the videotaped frames and calculate the kinetics and kinematics of the body during the performance. Figure 4 is a flowchart of the analysis process.

Subjects

The use of human subjects was approved by the Iowa State University Human Subjects Committee. Two members of the University of Iowa men's gymnastics team volunteered as subjects. The subjects were aged 19 and 21; their average height was 173.5 cm and average mass was 71.4 kg. Each gymnast provided information on previous injuries sustained, so correlations between forces experienced and any residual effects of the injuries could be investigated. Subject 1 had some stiffness and limited range of motion in his right elbow, due to a tendon injury experienced two years previously. Subject 2 had spondylolisthesis in his back three years previously and had undergone therapy for correction. He reported no recurrent pain from the injury. Both subjects had performed the Tkatchev during competition in the previous season. Subject 1 was considered by his coach to be somewhat more proficient than Subject 2.
Set up experiments

Use computer program to process video data

Calculate the kinetics of each segment

Determine the rotation matrices between global and segment coordinates

Calculate the kinetics at each joint

Store data for use in simulation model

Figure 4: Flowchart of analysis process.
The gymnasts' coach signed a letter of permission to allow the gymnasts to participate and was present at the videotape session. The gymnasts signed informed consent forms describing their participation in the study and all possible risks (see Appendix B).

**Strength and flexibility**

A Biodex dynamometer, a computerized testing and rehabilitation system, was used to measure the maximum muscular torques and range of motion at the shoulders, elbows and lower back. The Biodex system could be programmed for the number of trials and output statistical data from the strength testing.

The subjects were tested for maximum torques at speeds of 180 and 300 degrees per second. Five repetitions were done at each speed.

**Anthropometry**

A modified Hanavan model of the human body was used (Miller and Morrison, 1975) to estimate segment centers of mass, moments of inertia, and masses. The Hanavan model represents the human body by 15 simple geometric solids of uniform density. The head was depicted as an ellipsoid of revolution, the upper and lower torso as right elliptical cylinders and the hands as solid spheres. All other segments
(upper and lower arms, thighs, shanks, and feet) were portrayed as frustra of right circular cones. The joints were modeled as follows: neck: hinge joint; shoulder: ball and socket joint; elbow: hinge joint; spine: ball and socket joints; hip: ball and socket joint; knee: hinge joint.

The trunk was divided into three parts, based on the points where the back appeared to flex in the preliminary studies. The upper torso was defined to be from the top of the shoulders to the bottom of the sternum, the center segment from the bottom of the sternum to the waist, and the lower torso from the waist to the buttocks. All spinal movements were assumed to occur at the "joints" between these defined segments (i.e., the motion at the individual vertebrae was disregarded).

The foot and shank were treated as a single segment, since (because of the small mass of the foot) the movements at the ankle could not significantly contribute to the performance of the Tkatchev or to the forces at the other joints. The wrist was also considered immobile, although some important movement occurs at the wrist joint. Because of the size of the frame of reference of measurements, motion at the wrist could not be accurately measured (markers would be too close together to be distinguishable). Preliminary investigations showed that elbow and wrist torques were approximately equal throughout most of the move. Therefore,
the movement at the wrist probably did not contribute significantly to the torques of interest at the shoulders and back.

Methods for obtaining anthropometric measurements are found in Hanavan (1964) and Clauser et al. (1969). Equipment needed to perform these anthropometric measurements included: skin calipers, a scale, and an anthropometer. The definition of each measurement taken is provided in Appendix C. The anthropometric data for the subjects are shown in Table 1.

From the anthropometric data, segmental masses were calculated using Clauser’s regression equations as reported by Miller and Morrison (1975). These are given in Table 2. Any discrepancy between calculated total weight and the actual total weight of the gymnast was distributed proportionally among the segments (Miller and Morrison 1975).

Principal mass moments of inertia (Ixx, Iyy, Izz) were calculated using segment weights and anthropometric measurements defining lengths and circumferences. The formulae used for mass moments of inertia are given in Table 3.

**Calibration of space for Ariel system**

The direct linear transformation (DLT) is the method by which two-dimensional data from two or more video cameras is converted into three-dimensional positions of markers. The
<table>
<thead>
<tr>
<th>Measurement (cm)</th>
<th>Subject 1</th>
<th>Subject 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head circumference</td>
<td>56.5 ± 0.1</td>
<td>58.3 ± 0.2</td>
</tr>
<tr>
<td>Chest circumference</td>
<td>97.3 ± 0.3</td>
<td>97.7 ± 0.7</td>
</tr>
<tr>
<td>Axillary circumference</td>
<td>33.1 ± 0.4</td>
<td>31.7 ± 0.4</td>
</tr>
<tr>
<td>Elbow circumference</td>
<td>28.6 ± 0.4</td>
<td>27.0 ± 0.0</td>
</tr>
<tr>
<td>Forearm circumference</td>
<td>29.5 ± 0.0</td>
<td>27.8 ± 0.3</td>
</tr>
<tr>
<td>Wrist circumference</td>
<td>18.9 ± 0.1</td>
<td>18.3 ± 0.2</td>
</tr>
<tr>
<td>Fist circumference</td>
<td>29.3 ± 0.3</td>
<td>28.4 ± 0.2</td>
</tr>
<tr>
<td>Thigh circumference</td>
<td>49.9 ± 0.5</td>
<td>49.9 ± 0.1</td>
</tr>
<tr>
<td>Knee circumference</td>
<td>36.4 ± 0.1</td>
<td>35.1 ± 0.1</td>
</tr>
<tr>
<td>Calf circumference</td>
<td>36.5 ± 0.1</td>
<td>34.6 ± 0.1</td>
</tr>
<tr>
<td>Ankle circumference</td>
<td>25.7 ± 0.4</td>
<td>26.3 ± 0.2</td>
</tr>
<tr>
<td>Iliac fat</td>
<td>1.0 ± 0.0</td>
<td>0.6 ± 0.0</td>
</tr>
<tr>
<td>Chest breadth</td>
<td>32.0 ± 0.2</td>
<td>30.4 ± 0.4</td>
</tr>
<tr>
<td>Hip breadth</td>
<td>30.8 ± 0.3</td>
<td>30.4 ± 0.1</td>
</tr>
<tr>
<td>Chest depth</td>
<td>24.4 ± 0.1</td>
<td>26.0 ± 0.2</td>
</tr>
<tr>
<td>Waist depth</td>
<td>20.9 ± 0.1</td>
<td>20.5 ± 0.1</td>
</tr>
<tr>
<td>Buttock depth</td>
<td>25.1 ± 0.4</td>
<td>24.5 ± 0.3</td>
</tr>
<tr>
<td>Upper arm length</td>
<td>26.3 ± 0.3</td>
<td>31.1 ± 0.1</td>
</tr>
<tr>
<td>Forearm length</td>
<td>25.4 ± 0.2</td>
<td>26.4 ± 0.4</td>
</tr>
<tr>
<td>Wrist breadth</td>
<td>6.9 ± 0.1</td>
<td>6.5 ± 0.0</td>
</tr>
<tr>
<td>Hand breadth</td>
<td>9.5 ± 0.1</td>
<td>9.2 ± 0.1</td>
</tr>
<tr>
<td>Chin-neck interval</td>
<td>20.9 ± 0.1</td>
<td>22.1 ± 0.2</td>
</tr>
<tr>
<td>Shoulder height</td>
<td>133.7 ± 0.3</td>
<td>136.9 ± 0.2</td>
</tr>
<tr>
<td>Substernal height</td>
<td>122.5 ± 0.2</td>
<td>128.2 ± 0.8</td>
</tr>
<tr>
<td>Trochanterion height</td>
<td>91.5 ± 0.1</td>
<td>95.5 ± 0.0</td>
</tr>
<tr>
<td>Sitting height</td>
<td>87.5 ± 0.1</td>
<td>86.6 ± 0.4</td>
</tr>
<tr>
<td>Tibiale height</td>
<td>45.1 ± 0.2</td>
<td>50.2 ± 0.6</td>
</tr>
<tr>
<td>Sphyrion height</td>
<td>8.0 ± 0.1</td>
<td>7.6 ± 0.5</td>
</tr>
<tr>
<td>Foot length</td>
<td>26.3 ± 0.2</td>
<td>25.5 ± 0.1</td>
</tr>
<tr>
<td>Stature</td>
<td>172.4 ± 0.1</td>
<td>174.5 ± 0.3</td>
</tr>
</tbody>
</table>

* Dropped outlier, averaged 2 values.
Table 2: Segmental Mass Formulas (Miller and Morrison 1975)

<table>
<thead>
<tr>
<th>Body Segment Mass (kg)</th>
<th>Regression Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>(=0.104\text{Head Circumference} +0.015\text{Mass}-2.189)</td>
</tr>
<tr>
<td>Trunk</td>
<td>(=0.349\text{Mass}+0.423\text{Trunk Length} +0.229\text{Chest Circumference} -35.460)</td>
</tr>
<tr>
<td>Upper Arm</td>
<td>(=0.007\text{Mass}+0.092\text{Axillary Circumference}+0.050\text{Upper Arm Length}-3.101)</td>
</tr>
<tr>
<td>Forearm</td>
<td>(=0.081\text{Wrist Circumference} +0.052\text{Forearm Circumference} -1.650)</td>
</tr>
<tr>
<td>Hand</td>
<td>(=0.029\text{Wrist Circumference} +0.075\text{Wrist Breadth} + 0.031\text{Hand Breadth}-0.746)</td>
</tr>
<tr>
<td>Thigh</td>
<td>(=0.074\text{Mass}+0.123\text{Thigh Circumference}+0.027\text{Iliac Fat}-4.216)</td>
</tr>
<tr>
<td>Lower Leg</td>
<td>(=0.111\text{Calf Circumference} +0.047\text{Tibiale Height} +0.074\text{Ankle Circumference}-4.208)</td>
</tr>
<tr>
<td>Foot</td>
<td>(=0.003\text{Mass}+0.048\text{Ankle Circumference}+0.027\text{Foot Length}-0.869)</td>
</tr>
</tbody>
</table>

Fat in mm
All other dimensions in cm

Table 3: Mass moments of inertia of segments (Hanavan 1964)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Ixx</th>
<th>Iyy</th>
<th>Izz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ellipsoid of revolution</td>
<td>(m/5*(b^2+c^2))</td>
<td>(m/5*(a^2+c^2))</td>
<td>(m/5*(a^2+b^2))</td>
</tr>
<tr>
<td>Right elliptical cylinder</td>
<td>(m/12*(3*b^2+h^2))</td>
<td>(m/12*(3*a^2+h^2))</td>
<td>(m/4*(a^2+b^2))</td>
</tr>
<tr>
<td>Sphere</td>
<td>(2<em>m</em>r^2/5)</td>
<td>Ixx</td>
<td>Ixx</td>
</tr>
<tr>
<td>Frustra of right circular cone***</td>
<td>(A<em>m^2/(D</em>L))</td>
<td>Ixx</td>
<td>(3<em>m</em>(R^2-r^3))</td>
</tr>
<tr>
<td></td>
<td>+B<em>m</em>L^2</td>
<td></td>
<td>(10*(R^3-r^3))</td>
</tr>
</tbody>
</table>

\[^*] A=9\frac{\sqrt{20\pi}}{(1+(R/r)+(R/r)^2+(R/r)^3+(R/r)^4)}\]

\[^*\] B=3/80*(1+4*(R/r)+10*(R/r)^2+4*(R/r)^3+(R/r)^4)

\[^*\] D=3*m/(L*(r^2+r*R+R^2)*π)
DLT requires a calibration object whose marker positions are known in three dimensions. A linear regression algorithm is used to calculate the relationship between two-dimensional image coordinates and three-dimensional space data.

Scaled two-dimensional image coordinates are retrieved from the videotape or film during digitization of the frames. The calibration object is used to define a three-dimensional reference frame in space. Two vectors, \( \vec{h}_i \) and \( \vec{h}_j \), along with a fixed point, define a two-dimensional coordinate system for the image plane. If point I is the video image of point O and point O has coordinates \((x,y,z)\) with respect to the 3-d reference frame, and point I has digitizer coordinates \((U,V)\) with respect to the image coordinate system, then the general form of the 3-d to 2-d image coordinate transformation is:

\[
\begin{align*}
U &= \frac{(Ax+By+Cz+D)}{(Ex+Fy+Gz+D)} \\
V &= \frac{(Hx+Iy+Kz+L)}{(Ex+Fy+Gz+D)}
\end{align*}
\]

The coefficients A-L are found experimentally using the calibration object. Given six or more known locations in the three-dimensional space (defined by the calibration object) and the two-dimensional data from at least two cameras, a set of equations, using the above relationships, can be set up to solve for A-L using the method of least squares.

An object was designed to attach to the bar to calibrate the space in which the gymnast performed the Tkatchev, for use with the DLT. The reference object was made from PVC pipe in order to be lightweight and durable. The calibration points were rubber balls glued to the ends of the pipe and covered with reflective tape. There were eight calibration points, four above and four below the bar. The device is pictured in Figure 5.

The object defined an area about 200 cm square about the bar. The global coordinates of the calibration points were found by measuring the device, and are listed in Table 4. Points within this calibration space were expected to be reliable (Wood and Marshall 1986). In the videotaping, the ankle markers were outside the calibration space at some points during the move. These were expected to be less reliable than the other measured marker positions, and required more smoothing.

The calibration object was attached to the bar before the gymnasts were videotaped. The object was videotaped from all cameras and the videotape was digitized and transformed to provide the parameters for the DLT. After the first gymnast performed,
Figure 5: Calibration object
Table 4: Position of calibration markers in global coordinates

<table>
<thead>
<tr>
<th>Marker #</th>
<th>X (cm)</th>
<th>Y (cm)</th>
<th>Z(cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>3.5</td>
<td>0.0</td>
<td>99.4</td>
</tr>
<tr>
<td>3</td>
<td>248.5</td>
<td>-4.0</td>
<td>77.8</td>
</tr>
<tr>
<td>4</td>
<td>244.5</td>
<td>-4.0</td>
<td>-3.2</td>
</tr>
<tr>
<td>5</td>
<td>255.2</td>
<td>202.5</td>
<td>-13.4</td>
</tr>
<tr>
<td>6</td>
<td>256.0</td>
<td>194.5</td>
<td>99.0</td>
</tr>
<tr>
<td>7</td>
<td>-14.1</td>
<td>210.5</td>
<td>94.8</td>
</tr>
<tr>
<td>8</td>
<td>-20.9</td>
<td>208.0</td>
<td>-12.3</td>
</tr>
</tbody>
</table>

The cameras were moved to improve capture of points high above the bar. The calibration object was videotaped a second time after the experiments with the cameras in their new location, and this calibration frame was used for the data from Subject 2.

The accuracy of calibration was checked by digitizing ten frames of the calibration object in a static position and comparing the computed locations of the markers with their measured locations. This was done separately for the Subject 1 and Subject 2 locations. Results of this check are summarized in Figures 6 and 7 and in Table 5.

The errors in the z-direction (along the axis of the bar) are higher for Subject 1. This occurred because one of the two cameras used for digitizing Subject 1 data was aligned
Figure 6: Errors in calibration marker positions, Subject 1
Figure 7: Errors in calibration marker positions. Subject 2
Table 5. Marker errors in calibration test

<table>
<thead>
<tr>
<th>Marker</th>
<th>X error (cm)</th>
<th>Y error (cm)</th>
<th>Z error cm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sub 1</td>
<td>Sub 2</td>
<td>Sub 1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Avg</td>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

with the z-axis, which made it difficult to differentiate points along the axis. Since the Ariel system "projects" a trajectory for the points, some error resulted if a small difference occurred from the first to the second differentiated frame. In addition, since the PVC pipe from which the calibration object was made was not completely rigid, there may have been some movement of the markers in this "static" calibration check. Angulo and Dapena (1992) reported errors in control object points of up to 1.5% of the distance surveyed. This would allow up to 3 cm error in the calibration points. Although the errors reported here are higher than 3 cm in some cases, they are small enough to permit adequate resolution of most points. The higher errors reported here are probably due to measurement errors, movement of the markers, poor resolution of the object due to
inadequate lighting, and less than optimal location of cameras for resolution of calibration points.

Determination of bar forces from strain gage data

Strain gages were applied to the horizontal bar (a steel bar 240 cm long and 2.8 cm diameter) to determine the external forces (vertical and horizontal forces) that occurred at each hand. Two pairs of strain gages were applied at each end of the bar, one pair on the top surface and one pair on the side (Figure 8).

Figure 8: Strain gages attached to bar
Static strength of material theory was used to determine the forces at each hand from the strains measured. As shown in Figure 9, the shear diagram, $V_L$ is the shear force on the left end of the bar, $V_R$ is the shear force on the right end of the bar, $M_L$ is the moment on the left end of the bar, and $M_R$ is the moment on the right end of the bar. Strain at a given gage is denoted by $\epsilon$.

![Figure 9: Shear diagram](image)

From strength of materials it is known that at any point on the surface, $\epsilon_x = -\frac{M(x)}{EI} \cdot \frac{r}{EI}$ is constant for constant cross-section, it will be denoted as $C$ in the following equations. Then

$$\epsilon_1 = M(a) \cdot C ; \quad M(a) = \frac{\epsilon_1}{C} \quad (2)$$

and similarly,

$$\epsilon_2 = M(b) \cdot C ; \quad M(b) = \frac{\epsilon_2}{C} \quad (3)$$
From the shear-moment diagrams (Figures 9 and 10), it can be seen that

\[ M(b) - M(a) = VL \cdot (b-a) \]  (4)

so \( VL \) can be found from strain data:

\[ VL = \frac{M(b) - M(a)}{(b-a)} = \frac{1}{(b-a)} \cdot \frac{(\epsilon_2 - \epsilon_1)}{C} \]  (5)

From the shear-moment relationships,

\[ ML + VL \cdot b = M(b) \]  (6)

then \( ML \) can be found

\[ ML = \frac{\epsilon_2}{C} - VL \cdot b \]  (7)
The moment at P1 can then be found as

\[ M_1 = M_L + V_L \cdot C \]  \hspace{1cm} (8)

Similarly, for the right-hand side of the bar,

\[ V_R = \frac{1}{b-a} \cdot \frac{\epsilon_3 - \epsilon_4}{C} \]  \hspace{1cm} (9)

\[ M_R = \frac{\epsilon_3}{C} - V_R \cdot b \]  \hspace{1cm} (10)

\[ M_2 = M_R + V_R \cdot (L - d) \]  \hspace{1cm} (11)

Again, from the shear-moment relationships,

\[ M_2 - M_1 = (V_L - P_1) \cdot (d - c) \]  \hspace{1cm} (12)

and P1 can be found from

\[ P_1 = V_L - \frac{M_2 - M_1}{d - c} \]  \hspace{1cm} (13)

P2 is found using the shear force relationship

\[ P_2 = V_R + V_L - P_1 \]  \hspace{1cm} (14)

Horizontal forces can be found using an identical
derivation, with gages 5-8 replacing 1-4 in the equations. The distances \( a \) and \( b \) were measured when the strain gages were applied. The distances \( c \) and \( d \) were determined using the wrist center location in relationship to the ends of the bar. These were calculated using the three-dimensional position data. The constant \( C \) was determined using static calibration of the system and was checked using known values for the Young's Modulus and the area moment of inertia of the steel bar.

An approximation of the error involved in using static calibration for this dynamic system was done. In vibrations, the magnification ratio is the ratio of the dynamic displacement to the static displacement under the same forcing conditions. For a system with negligible damping, the magnification ratio is:

\[
\frac{\delta}{\delta_{st}} = \frac{1}{1 - \left(\frac{\omega}{\omega_n}\right)^2} \quad (15)
\]

If the bar/gymnast system is viewed as a lumped mass of 73 kg on a vibrating bar with pin connections at the ends, the natural frequency of the system is

\[
\omega_n = \sqrt{\frac{k_{eq}}{M + 0.5m}} \quad (16)
\]

where the equivalent spring constant of the system is
the lumped mass $M$ is 73.4 kg, and the mass of the bar $m$ is 9.6 kg. Then the natural frequency of the system $\omega_n$ is 13.48 rad/sec.

From preliminary investigations, it appeared that the forcing frequency at the hands was about 4.5 rad/sec. This would result in a magnification ratio of 1.12. This would mean that the method of using static calibration results in estimated forces about 12% above the actual applied forces. This overestimate would increase as the natural frequency and the forcing frequency get closer together.

The bar was instrumented with eight MicroMeasurements EA-13-120LZ-120 strain gages, having gage factors of 2.115 ± 0.5% and a resistance of 120 Ω. Although the strain gages used were designed for static data collection, they were adequate for this application because the experiment was of short duration. The primary advantage of using "dynamic" strain gages would be a longer life under cyclic stresses. Data from the strain gages were conditioned and amplified using a Measurements Group 2100 Strain Gage Conditioner and Amplifier System. The amplified data were digitized using a Bioinstrumentation A/D board and input into an AT-compatible
386 personal computer. Data acquisition and storage was controlled using the APAS analog module.

The strain gages were affixed to the side and the top of the bar at each end. The first was located at 1 inch and the second at 13 inches from the point where the bar was pinned. The strain gage data was sampled at 210 Hz and was later manually synchronized with the kinematic data resulting from the digitized videotape frames. The release point was used as the synchronizing event.

Calibration of strain gages consisted of computing the constant \( C = -\frac{E I}{r} \) for the strain gage set-up used. The constant was first computed with a single static load applied at different points along the bar. With a single load, the force was independent of its location, and Equation (13) reduces to

\[
P I = \frac{1}{C} \cdot \frac{(\varepsilon_2 + \varepsilon_3 - \varepsilon_1 - \varepsilon_4)}{(b-a)}
\]

(18)

The results from the calibration are shown in Figure 11. Using linear regression, \( C \) was 3.85 in·lb per \( \mu \varepsilon \), (43.497 N·cm/\( \mu \varepsilon \)). Using \( E = 30 \times 10^6 \) psi and the bar diameter of 1.102 inches, the calculated value of \( C \) was 3.924 in·lb/\( \mu \varepsilon \), which is 1.9% higher than the measured value. It was concluded that the calibration process was correct.
Figure 11: Calibration data, single load: load vs. strain

The calibration was then checked by applying two loads along the bar. Table 6 shows the data from the two-load application. It appears that with loads that are very close together, there is significant error. This is probably due to insufficient differentiation of the strain from each force. That is, the significance of the strain is less than the differences expected. This was not a serious problem in the experiments performed, since the hands were at least shoulder distance apart. This distance was large enough for adequate differentiation of force results.
### Table 6: Calibration data for strain gages

<table>
<thead>
<tr>
<th>Value</th>
<th>Trial 1</th>
<th>Trial 2</th>
<th>Trial 3</th>
<th>Trial 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1 (lb)</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>65</td>
</tr>
<tr>
<td>P2 (lb)</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>c (in)</td>
<td>28.5</td>
<td>23.25</td>
<td>65.5</td>
<td>32.75</td>
</tr>
<tr>
<td>d (in)</td>
<td>63</td>
<td>29.5</td>
<td>69</td>
<td>69</td>
</tr>
<tr>
<td>d-c(in)</td>
<td>34.50</td>
<td>6.25</td>
<td>3.50</td>
<td>36.25</td>
</tr>
<tr>
<td>VL (lb)</td>
<td>52.2</td>
<td>74.4</td>
<td>29.8</td>
<td>57.7</td>
</tr>
<tr>
<td>ML(in-lb)</td>
<td>-10.1</td>
<td>-8.0</td>
<td>-22.0</td>
<td>-26.5</td>
</tr>
<tr>
<td>VR (lb)</td>
<td>46.8</td>
<td>25.3</td>
<td>69.2</td>
<td>56.1</td>
</tr>
<tr>
<td>MR(in-lb)</td>
<td>-1.4</td>
<td>-2.8</td>
<td>10.6</td>
<td>-2.8</td>
</tr>
<tr>
<td>M1(in-lb)</td>
<td>1477.6</td>
<td>1721.8</td>
<td>1929.9</td>
<td>1863.2</td>
</tr>
<tr>
<td>M2(in-lb)</td>
<td>1543.0</td>
<td>1654.3</td>
<td>1809.8</td>
<td>1455.8</td>
</tr>
<tr>
<td>P1 (lb)</td>
<td>50.3</td>
<td>85.3</td>
<td>64.1</td>
<td>68.9</td>
</tr>
<tr>
<td>P2 (lb)</td>
<td>48.7</td>
<td>14.4</td>
<td>35.1</td>
<td>44.9</td>
</tr>
<tr>
<td>P1+P2(lb)</td>
<td>99.0</td>
<td>99.7</td>
<td>99.0</td>
<td>113.8</td>
</tr>
<tr>
<td>Error, P1</td>
<td>+0.6%</td>
<td>+70.6%</td>
<td>+28.0%</td>
<td>+6.0%</td>
</tr>
<tr>
<td>Error, P2</td>
<td>-2.6%</td>
<td>-71.2%</td>
<td>-29.0%</td>
<td>-10.2%</td>
</tr>
<tr>
<td>Error, P1+P2</td>
<td>-1.0%</td>
<td>-0.3%</td>
<td>-1.0%</td>
<td>-1.0%</td>
</tr>
</tbody>
</table>

**Videotape protocol**

Four cameras were used to videotape the gymnasts. The cameras were Panasonic Model AG450, Panasonic Model AG190, Panasonic Model AG185 and Panasonic Model AG180. Four Scotch EG+ T-120 videotapes were used.

The subjects were videotaped using a four-camera arrangement, as illustrated in Figure 12. Because of the proximity of the bar to the wall, cameras could not be set up to view the move directly from the gymnast’s left side.
The camera arrangement used provided adequate reproduction of most points; however, there were parts of the movement where the left side markers were invisible to three of the cameras. This required some estimation of the location of some markers while digitizing to have an adequate reproduction of the move.

![Figure 12: Camera positions](image-url)
no shirts to minimize errors associated with the movements of clothing relative to the body. Body segments were identified using reflective markers made by students at Iowa State University. The markers were Styrofoam balls covered with 3M™ reflective tape, with bingo chips glued to the bottom. They were attached to the gymnasts' skin using spirit gum. The reflective markers were placed at the following locations (with reference to anatomical position):

- **Head:** forehead, in front of the left ear, and in front of the right ear,
- **Forearm:** lateral side of elbow and medial side of wrist,
- **Upper arm:** lateral side of elbow, half way up the posterior side of the upper arm, lateral side of the shoulder,
- **Upper trunk:** superior edge of the sternum, substernal notch, and on the neck opposite the superior edge of the sternum,
- **Mid trunk:** left and right sides of waist, and on vertebral column opposite the substernal notch,
- **Lower trunk:** left and right sides of waist, left and right anterior superior iliac spine,
- **Upper leg:** lateral side of knee, half way up anterior thigh, half way up posterior thigh, greater trochanter,
- **Lower leg:** lateral side of knee and lateral malleolus.

The skill was videotaped at 30 frames per second. In
The skill was videotaped at 30 frames per second. In preliminary studies, this frame rate was found to provide adequate resolution for the Tkatchev. The frame rate of each camera was checked using an analog clock for 60 seconds prior to and after data acquisition.

Nine trials were recorded from Subject 1 and ten from Subject 2. Trials were chosen for analysis based on completion of the move without spotting assistance, points of release and recatch clearly visible on the tape, and all data acquisition equipment functioning simultaneously and properly. Two trials were selected and digitized for each subject.

The data from all four cameras and the strain gages were synchronized using the point of release as the synchronizing event. The camera views were synchronized using Ariel software prior to transformation of the data into three-dimensional coordinates. The release point was identified in each view while digitizing. The displacement data and the strain gage data were synchronized at this point using the software specifically written for this study (Appendix D).

Data reduction from motion analysis system

The Ariel Performance Analysis System (APAS) was used for much of the data reduction in this study. APAS is a
software and hardware system specifically designed to analyze videotape or film using biomechanical principles. It consists of modules that

1) transform video images into computer files,
2) allow the user to digitize points (joints) on the image,
3) transform digitized data from two or more cameras into three-dimensional data,
4) smooth the digitized data,
5) calculate kinetic and kinematic data,
and 6) provide output in the form of animated figures, graphs of any desired kinetic or kinematic parameter, and data or spreadsheet files for further processing by the user.

The system limitations are as follows:

1) The Ariel software operates on the assumption that each segment is defined by the joints at each end (a stick-figure representation). Therefore, it assumes that rotations along the segment’s long axis are negligible in all calculations.
2) APAS provides output only in global coordinates; kinematic and kinetic parameters cannot be calculated in segment coordinates unless segments are aligned with the global coordinates.
3) The kinetics module cannot be used with non-system
segments; the user must use the default segments provided by the software. APAS provides for only one trunk segment, so the division of the trunk into two or three segments could not be done accurately using the software provided.

4) The kinetics module cannot interface with analog external force data, so it uses assumptions to calculate forces and torques at the joints. It assumes zero moments at the external force point, and apparently uses a optimization scheme of some sort to solve the closed loop problem.

The following processes were done using the software available with APAS: The videotape frames of interest were selected and captured into digital computer files. The locations of the reflective markers on each video frame were manually digitized. The data from the four cameras were transformed into three-dimensional data using the DLT algorithm. Cubic spline smoothing was performed on all the raw displacement data to filter out noise and extraneous data. Various smoothing algorithms were available from the APAS system; cubic splines, digital filtering and quintic splines were compared on test data. The difference in resulting accelerations was minimal, with cubic splines providing adequate reproduction of the signal without cutting off the peaks. The cubic splines algorithm also
allowed "impact/relax" areas in the curves where smoothing could be increased or decreased due to the nature of the move or the data. Cubic splines were therefore used in order to take advantage of this feature. The closeness of fit parameter for each point’s displacement was chosen by viewing the acceleration curves. The closeness of fit parameter corresponded to the maximum distance from the raw data that the smoothed data could be; that is, the smaller the parameter, the less smoothing performed. The smoothing was determined to be adequate if the acceleration curve showed no sudden spikes, which would indicate random noise (APAS User’s Guide 1989). Displacement, velocity, and acceleration data calculated by the Ariel program were printed to a DOS file for access by the Fortran programs used to analyze the skill (Appendix D).

Secondly, the strain gage output was converted for use in the force/torque program. The data were output from the Ariel system into a Lotus 1-2-3 spreadsheet, sent to a file, and converted to a form that could be used more easily. The synchronizing point was found where the strains crossed the zero level (at the release). The strain gage data was interpolated to correspond with the kinematic data using a cubic splines routine (Kahaner, Moler and Nash, 1989). The interpolated data was then sent to a file for use in the calculation of forces and torques for the body.
that could not be easily used as a data file. A routine was written that converted these data into three files (position, velocity and acceleration) that could be used as data files in the Fortran program.

**Determination of rotation matrices**

The conversion of vectors from global to segment coordinates and conversion between two segments were found using the methods of Spoor and Veldpaus (1980). Assume that in the initial videotape reference frame, the displacement vector for any marker is defined by a vector \( \mathbf{a} \) and in another frame the displacement vector to the same marker is defined by vector \( \mathbf{p} \). Then the relationship between the two positions is:

\[
\mathbf{p} = [\mathbf{R}] \mathbf{a} + \mathbf{b}
\]  

(19)

where \([\mathbf{R}]\) is a rotation matrix and \( \mathbf{b} \) is a translation vector. Since only the rotation information was needed, the method for finding \([\mathbf{R}]\) was used. This method requires position vectors be defined for at least three points on each segment (i.e., there must be three markers to define a segment). If \( \mathbf{a} \) defines the marker positions when the segment is aligned with the global coordinates, then \([\mathbf{R}]\), the rotation matrix between global and segment coordinates, can be found using \( \mathbf{a} \) and \( \mathbf{p} \) at any other point in the move. Details are given in Spoor and
The rotation matrix between two segments (i.e., from the shank to the thigh segment) can be found from the rotation matrices for each segment from global coordinates. If $[R_i]$ and $[R_j]$ are the rotation matrices for the shank and thigh segments, respectively, then for any vector $\mathbf{v}$,

$$\mathbf{v}_{global} = [R_i] \mathbf{v}_{shank} \quad (20)$$

and

$$\mathbf{v}_{global} = [R_j] \mathbf{v}_{thigh} \quad (21)$$

so

$$[R_j] \mathbf{v}_{thigh} = [R_i] \mathbf{v}_{shank} \quad (22)$$

then, since the rotation matrices are orthogonal $[R_i]^T = [R_i]$ and

$$\mathbf{v}_{thigh} = [R_j]^T [R_i] \mathbf{v}_{shank} \quad (23)$$

so the rotation matrix from shank to thigh

$$[R_{12}] = [R_j]^T [R_i] \quad (24)$$

Similarly, for any two segments $i$ and $j$, the rotation matrix
Similarly, for any two segments i and j, the rotation matrix from segment i to segment j \( [R_{ij}] \) is:

\[
[R_{ij}] = [R_j]^T[R_i]
\] (25)

For the shank and forearm segments, only two markers were used since the rotations at the elbow and knee joints were assumed to be only about the z-axis (mediolateral axis) of the thigh and upper arm. An algorithm was developed to deal with these special cases.

Equation (25) was rearranged so that the rotation matrix of one segment with respect to the global coordinates was written in terms of the adjacent segment’s rotation matrix and the rotation matrix between the two segments:

\[
[R_i] = [R_j][R_{ij}]
\] (26)

Since the rotation between the two segments was assumed to be only about the z-axis, the rotation matrix \( [R_{ij}] \) could be written as:
Let the vector from marker 1 to marker 2 on the distal segment be labeled $\vec{a}$ in distal segment coordinates, $\vec{p}$ in global coordinates, and $\vec{p}'$ in proximal segment coordinates. Using $[R_{ij}]^T \vec{p} = \vec{p}'$, the vector can be determined in proximal segment coordinates. Then using $[R_{ij}] \vec{a} = \vec{p}'$, we get the relationships,

$$p'_1 = a_1 \cos \theta - a_2 \sin \theta$$
$$p'_2 = a_1 \sin \theta + a_2 \cos \theta$$

Since only the cosine and sine of $\theta$ are unknown, the two equations can be solved simultaneously to obtain the rotation angle. Then $\sin \theta$ can be found by the relationship:

$$\sin \theta = \frac{p'_2 a_1 - p'_1 a_2}{a_1^2 + a_2^2}$$
and \( \cos \theta \) can be found using:

\[
\cos \theta = \sqrt{1 - \sin^2 \theta}
\]  

(30)

The rotation matrices for each segment and between segments were computed using the position data from the APAS program. Initially, a reference location was determined for each marker in segment coordinates using their theoretical locations from the Hanavan model. At these reference points, the segment would be aligned with the global coordinate system, as shown in Figure 13.

Figure 13: Coordinate systems
The marker locations for the reference position were put in a reference file for the subject. These reference positions were compared manually with relative position vectors in several digitized frames where at least two points in a segment were aligned along a global axis. For example, the upper arm segment is aligned with the global coordinate system when the elbow and shoulder have the same x-coordinate. The averaged position vectors from the digitized frames were used as corrected reference frame data. If there was a large discrepancy between the theoretical and the averaged reference data, the marker was either disregarded or the theoretical value used (assumed that the digitization was less accurate, since there seemed to be many errors in the digitization process). The theoretical location of markers, from the Hanavan model, are tabulated in Appendix F. The reference data was then used as the $\mathbf{a}$ vectors in Equation (16), and used to calculate rotation matrices for each frame.

**Calculation of kinematics of each segment**

The angular velocity and angular acceleration of each segment were determined from the methods of Verstraete and Soutas-Little (1990). As with the rotation matrix, a minimum of three markers were required for this determination. The authors indicated that four markers
increased the accuracy markedly. For each segment's markers, 1..n, the global position, velocity and acceleration vectors were known from digitizing and differentiating the Ariel data. From these, the relative position, velocity and acceleration vectors between any two markers were determined using subtraction. For angular velocity, the relationship

\[ \vec{v}_{i/j} = \vec{\omega} \times \vec{x}_{i/j} \]  

was used, where \( \omega \) is the angular velocity of the object with respect to the global reference frame. A system of algebraic equations was then set up and solved for \( \omega \) using least squares techniques. See Verstraete and Soutas-Little (1990) for details. Similarly, for angular acceleration, if \( \alpha \) is the angular acceleration of the object with respect to the fixed reference frame, then the relationship:

\[ \vec{a}_{i/j} = \vec{\omega} \times (\vec{\omega} \times \vec{x}_{i/j}) + \vec{\alpha} \times \vec{x}_{i/j} \]  

is used and solved for \( \alpha \) using least squares techniques. The vectors \( \omega \) and \( \alpha \) are then rotated into segment coordinates using \([R_i]\) for the segment of interest.

For the forearm and shank segments, an algorithm was
developed to deal with the assumption that there was only one rotation at the knee and elbow joints. If the proximal segment is again segment 2 and the distal segment is segment 1, then (Kane and Levinson 1985) the angular velocity of the distal segment with respect to the global coordinate system can be written as:

$$\omega^{A1} = \omega^{A2} + \omega^{21}$$  \hspace{1cm} (33)

The angular velocity of the distal segment can be rewritten as:

$$\omega^{A1} = \omega^{A2}_{21} + \omega^{A2}_{22} + (\omega^{A2} + \omega^{21}) \bar{n}_{23}$$  \hspace{1cm} (34)

where $\omega^{21}$ is the first derivative of the joint rotation at the joint of interest.

The angular acceleration for the two-marker segments was computed as follows:

$$\ddot{\omega}^{A1} = \ddot{\omega}^{A2} + \frac{d}{dt} (\omega^{21}) \bar{n}_{23} + \omega^{21} \frac{d}{dt} (\bar{n}_{23})$$

$$= \ddot{\omega}^{A2} + \omega^{21} \bar{n}_{23} + \omega^{21} \chi (\ddot{\omega}^{A2} \bar{n}_{23})$$  \hspace{1cm} (35)

$$= (\ddot{\omega}^{A2} + \omega^{21} \omega^{A2}) \bar{n}_{21} + (\ddot{\omega}^{A2} - \omega^{21} \omega^{A2}) \bar{n}_{22} + (\ddot{\omega}^{A2} + \omega^{21}) \bar{n}_{23}$$
All the variables in the first two terms of Equation (35) are known from previous calculations; the only component we need to calculate is in the $n_{12}$ direction, which has the magnitude of $\alpha^{21}+\alpha^{32}$, which can be found by setting $\alpha^{21}$ equal to the second derivative of the joint rotation at the joint of interest. The first and second derivatives of the rotations were found using a cubic splines program.

The linear acceleration of the center of gravity of the segment was determined using the relative acceleration relationship:

$$\ddot{a}_{cg} = \ddot{a}_{marker} + \dddot{X} \times r + \dddot{\omega} X (\dddot{r})$$  \hspace{1cm} (36)

where $r$ is the position vector from the segment’s center of gravity to the marker whose acceleration is defined. The marker acceleration was determined in segment coordinates using the rotation matrix, and then the center of gravity acceleration was calculated using Equation (36).

**Determination of joint forces and torques**

Forces and torques at joints were determined using the Newtonian relationships:

$$\Sigma \vec{F} = m\ddot{a}_g$$  \hspace{1cm} (37)
and

\[ \Sigma \overrightarrow{M} = \frac{d}{dt} \overrightarrow{H}_g \]  

(38)

where the change in momentum, \( \overrightarrow{H}_s \) was determined using the Newton-Euler equations:

\[
\begin{align*}
\dot{H}_x &= I_{xx} \alpha_x - (I_{yy} - I_{zz}) \omega_y \omega_z \\
\dot{H}_y &= I_{yy} \alpha_y - (I_{zz} - I_{xx}) \omega_z \omega_x \\
\dot{H}_z &= I_{zz} \alpha_z - (I_{xx} - I_{yy}) \omega_x \omega_y
\end{align*}
\]

(39)

All vectors were determined in segment coordinates. The forces were determined by first using hand forces found using the strain gage methods described previously. The joint forces were then computed segment by segment as follows: forearms (elbow forces), upper arms (shoulder forces), head (neck forces), upper trunk (upper spine forces), mid spine (lower back forces), shanks (knee forces), thighs (hip forces). One extraneous equation for the pelvis segment was used as a check on the accuracy. It was not expected to be in complete agreement because of errors associated with the relative position vectors, and kinematic calculations.
The moments at the hands were assumed to be primarily frictional and therefore proportional to the forces at the hands and only in the direction opposing the rotation (that is, the global z-direction). The hand moments were calculated using $M = k |F|$, where $k = 0.03$ (based on literature values from Duck 1978 and Witten 1990). $M$ is in Newton-meters and $F$ in Newtons. The moments were determined segment by segment as follows: shanks (knee moments), thighs (hip moments), pelvis (lower back), mid-trunk (upper back), head (neck), then the forearm, upper arm and upper trunk segments. As with the forces, there was an extra equation used as a check on the accuracy.

**Determination of joint rotation angles**

The joint rotation angles about the distal segment's z-x-y axes were calculated and sent to data files for use as input for the simulation program.

The joint angles were determined using the rotation matrices for each segment calculated as described above. If the rotation matrix between two adjacent segments is $[R_i]$, then the joint angles can be defined as three independent rotations, about the distal segment’s z-axis, then the x axis, then the y axis. If these three rotation angles are $\theta_1$, $\theta_2$, and $\theta_3$, then the rotation matrix between the segments
\[ [R] = \begin{bmatrix}
  c\theta_1 c\theta_3 - s\theta_1 s\theta_2 s\theta_3 & -s\theta_1 c\theta_2 & c\theta_1 s\theta_3 + s\theta_1 s\theta_2 c\theta_3 \\
  s\theta_1 c\theta_3 + c\theta_1 s\theta_2 s\theta_3 & c\theta_1 c\theta_2 & s\theta_1 s\theta_3 - c\theta_1 s\theta_2 c\theta_3 \\
  -s\theta_2 c\theta_3 & s\theta_2 & c\theta_2 c\theta_3
\end{bmatrix} \] (40)

where \( c \) denotes cosine and \( s \) denotes sine.

Using trigonometric relationships, the rotation angles were computed from the terms of the rotation matrix as follows:

\[ \theta_1 = \tan^{-1}\left(\frac{-R_{12}}{R_{22}}\right) \]
\[ \theta_2 = \tan^{-1}\left(\frac{R_{32}}{\sqrt{R_{12}^2 + R_{22}^2}}\right) \] (41)
\[ \theta_3 = \tan^{-1}\left(\frac{-R_{21}}{R_{33}}\right) \]

Output from analysis

Forces, torques, kinematics, joint angles, and center of gravity trajectory were output to data files, converted to form usable with Quattro Pro for Windows, and imported to spreadsheet files for plotting and analysis.
Theoretical determination of bar deflections

The bar deflections measured at the hands were compared with those expected from strength of materials theory. The bar was modeled as being pinned at both ends, although there were minimal end moments calculated during the calibration. For a bar pinned at both ends, the deflection, $y$, at a point a distance $x$ from the left end of the bar when a single load $P$ is applied at a distance $a$ from the left end of the bar, and a distance $b$ from the right end of the bar:

$$x < a \Rightarrow y = \frac{Pb}{6EL} (x^3 - (L^2 - b^2) x)$$  \hspace{1cm} (42)$$

$$x = a \Rightarrow y = \frac{-Pb^2 a^2}{3EL}$$  \hspace{1cm} (43)$$

$$x > a \Rightarrow y = \frac{Pa}{6EL} [(L-a)^3 - (L^2 - a^2) (L-x)]$$  \hspace{1cm} (44)$$

Using the principle of superposition, if there is a load $P_1$ at a distance $c$ from the left end and a load $P_2$ at a distance $d$ from the left end of a bar of length $L$, then the deflection at $c$ from $P_1$ is

$$y_{11} = \frac{-P_1 (L-c)^2 c^2}{3EL}$$  \hspace{1cm} (45)$$
and the deflection at \( c \) from \( P_2 \) is

\[
y_{12} = \frac{P_2 (L-d)}{6EI} \left[ c^3 - (L^2 - (L-d)^2) c \right]
\]  

(46)

Then the total deflection at \( c \) is

\[
y_1 = y_{11} + y_{12}
\]  

(47)

Similarly, at \( d \), the deflection from \( P_1 \) is

\[
y_{21} = \frac{P_1 c}{6EI} \left[ (L-d)^3 - (L^2 - c^2) (L-d) \right]
\]  

(48)

and the deflection at \( d \) from \( P_2 \) is

\[
y_{22} = \frac{-P_2 (L-d)^2 d^2}{3EI}
\]  

(49)

And by the principle of superposition,

\[
y_2 = y_{21} + y_{22}
\]  

(50)

**Development of Equations for the Computer Simulation Model**

The equations for the computer simulation model were developed using joint angular kinematics, initial body orientation, and the center of gravity trajectory about the
bar as input. The primary results of the program were the joint kinetics and the orientation of the body in space. Figure 14 shows a flowchart of the mathematics used in developing the simulation.

The computer simulation was developed entirely in Microsoft Fortran on a 486 IBM PC compatible computer. The 12-segment computer simulation model was developed as follows: The Tkatchev was divided into three phases, the giant swing, the flight and the recatch. The giant swing and the recatch were the hand contact phases, while there was no contact with the bar during the flight.

For all phases, the mid-trunk segment was used as the reference segment. The orientation and kinematics of the other segments were calculated with respect to the reference segment.

During the giant swing phase, the input parameters were the joint angles versus time and the angular position of the center of gravity with respect to the bar versus time. The vector $\mathbf{r}$ from the hands to the center of gravity was calculated, using the average distance from the hands to the center of gravity, essentially assuming that the hands were fixed at the bar, rather than deflecting with the bar. An attempt was made to use the bar deflections in an iterative process, but it was unsuccessful because the noise in the
Determine parameters to be used as input/output

Use bar properties, CG angle in global coordinate system to determine forces at hands and CG path

Determine reference segment orientation

Use reference segment kinematics with joint rotations to determine the kinematics of all segments

Determine joint forces and torques using Newtonian equations

Figure 14. Flowchart of simulation mathematics.
data resulted in the iterations rapidly becoming unstable. The $r$ and $\theta$ data were differentiated using cubic splines. The data was passed through a 7-point moving filter to eliminate high frequency noise associated with the differentiation process. Then the center of gravity acceleration was calculated using the $r$-$\theta$ planar kinematic relationships:

\[
\begin{align*}
    a_r &= r - r \theta^2 \\
    a_\theta &= r \theta + 2r \theta \\
    a_x &= a_r \cos \theta - a_\theta \sin \theta \\
    a_y &= a_r \sin \theta + a_\theta \cos \theta
\end{align*}
\]  \tag{51}

The sum of the forces at the hands were calculated using:

\[
\Sigma \vec{F} = m \vec{a}_{cg}
\]  \tag{52}

then

\[
\begin{align*}
\Sigma F_{x_{\text{hand}}} &= m a_{cg} \\
\Sigma F_{y_{\text{hand}}} &= m a_{cg} + mg
\end{align*}
\]  \tag{53}

The total force at the hand was then divided between the two hands based on parameters input by the user.

Using the positions of the hands and the center of gravity in global and in reference segment coordinates, the
Once the position and orientation of the reference segment were known at each frame during the giant swing, the kinematics of the reference segment were determined using:

\[
\begin{align*}
\omega_1^A &= \dot{\beta} \gamma - \dot{\alpha} \beta \gamma \\
\omega_2^A &= \dot{\alpha} \beta + \dot{\gamma} \\
\omega_3^A &= \dot{\alpha} \beta \gamma + \dot{\beta} \gamma
\end{align*}
\]

and

\[
\begin{align*}
\alpha_1^A &= \beta \gamma - \dot{\beta} \gamma + \dot{\gamma} - \dot{\alpha} \beta \gamma + \dot{\alpha} \beta \gamma + \dot{\gamma} - \dot{\alpha} \beta \gamma + \dot{\gamma} - \dot{\alpha} \beta \gamma + \dot{\gamma} - \dot{\alpha} \beta \gamma + \dot{\gamma} - \dot{\alpha} \beta \gamma + \dot{\gamma}
\end{align*}
\]

where \( \alpha, \beta, \) and \( \gamma \) are the rotation angles about the 3-1-2 axes of the reference segment.

The kinematics of each of the segments were then calculated from the kinematics of the reference segment and the relative kinematic relationships resulting from the geometry and the joint angles. These relationships were defined in Huston and Passerello (1971):

\[
\vec{\omega}^A = \vec{\omega}^{A1} + \vec{\omega}^{1i}
\]

\[
\alpha^A = \omega_{i1} \vec{b}_1 + \omega_{i2} \frac{d}{dt} \vec{b}_1 + \omega_{i2} \frac{d}{dt} \vec{b}_2 + \omega_{i3} \frac{d}{dt} \vec{b}_3 + \omega_{i3} \frac{d}{dt} \vec{b}_3
\]

and

\[
\vec{a}_{ji} = \vec{a}_{ci} + \dot{q}_j + \vec{p}_{ci}
\]

where \( \vec{b}_1, \vec{b}_2, \) and \( \vec{b}_3 \) are the unit vectors aligned with the
\[ \vec{a}_{\text{j}} = \vec{a}_{\text{CO}} + \vec{a}_{\text{j}} + \vec{P}_{\text{CO}} \] (58)

where \( \vec{b}_1, \vec{b}_2, \) and \( \vec{b}_3 \) are the unit vectors aligned with the principle axes of the reference segment, \( \vec{q}_j \) is the position vector of the center of gravity of segment \( j \) with respect to the reference segment center of gravity, and \( \vec{P}_{\text{CO}} \) is the position vector of the center of gravity of the whole body with respect to the center of gravity of the reference segment. The details of the derivations for each segment are provided in Appendix A.

After the segment kinematics were found, the joint kinetics could be calculated using the same Newtonian relationships defined in the analysis portion of this research.

In the flight phase of the Tkatchev, the principles of conservation of momentum were used to determine the body position and orientation at each time point within the flight. The velocity at release and the angular momentum of the segments about the whole body center of gravity were input by the user during the simulation. The position of the center of gravity during the flight was calculated using:
\[ x = x_{\text{release}} + v_x t \]
\[ y = y_{\text{release}} + v_y t - \frac{1}{2} gt^2 \]
\[ z = z_{\text{release}} \]  

\[ (59) \]

The orientation of the reference segment was found using the methods of Dapena (1979). The angular momentum around the center of gravity was defined as:

\[ \vec{H}_{cg} = \sum_{i=1}^{\theta} \sum_{j=1}^{n} \vec{r}_i \omega_i + \sum_{i=1}^{n} m_i (\vec{r}_i \times \frac{d\vec{r}_i}{dt}) \]  

\[ (60) \]

All the terms in the above equation can be written in terms of the angular velocity of the reference segment (unknown) and the joint angles versus time and their derivatives (known). The vector equation above is then rewritten into 3 scalar equations and solved for the components of \( \vec{\omega}_i \) at each frame during the flight. The derivation of the equations used is given in Dapena (1979). The angular position of the reference segment in the global reference frame was then calculated using numerical integration of the angular velocity. A Runge-Kutta method was used to perform the integration.
Once the angular position and the center of gravity position are known, the segment kinematics and joint kinetics are calculated as discussed above for the giant swing phase.

During the recatch phase, the program first checked to see if the hands were close enough to the bar to recatch. A 10 cm cushion was provided to account for errors in digitization and errors resulting from the integration. If the hands did not recatch, the flight continues. If they do, then the program calculates the angular position of the center of gravity at the time of recatch. The angular speed of the center of gravity about the bar was used as input, and the angular position versus time was calculated. The recatch phase calculations were identical to those in the giant swing phase thereafter.

Validation of the Computer Simulation Model

Two variations of the Tkatchev performed by Subject 2 in Trial 9 were used to validate the model, since the results from that trial were the most accurate of the analysis results. The validation was done using the assumption of symmetry about the centerline in order to simplify the determination of angular momentum during flight and to account for any erroneous asymmetry in the analysis results. In one case, the values for the right side of the
body were used, and mirrored on the left. In the second symmetric Tkatchev, the left hand values were used and mirrored on the right. All long axis and anterior-posterior axis rotation was disregarded. When the pattern and magnitude of forces and torques were essentially the same for both the simulation and the analysis, the model was considered to be validated. Because there were various sources of error, a generous margin of error was allowed between the two methods.

The output of the simulation model was provided in two graphical forms: plots of forces and torques versus time and animated output of the gymnast on the screen. The kinematic and kinetic data were also output to files so that the user could do further analyses if needed.

The animation routines were developed using the CYLBOD subroutine and MAINJMP program developed by Jesus Dapena (Department of Kinesiology, Indiana University, Bloomington, IN 47405). Although these routines did not provide for the joints in the trunk, it was determined that they were adequate to show the body trajectory and motion during the move.

Running the Computer Simulation

In order to illustrate the use of the computer simulation program, several example simulations were run.
These included prohibiting the rotation at various joints and determining the effects on the angular velocity of the trunk, and on the forces at the right shoulder and the lower back. Other experiments included investigating the effects of increasing the leg length and body mass, the effects of increasing the giant swing speed, and the effects of different angular momentum values during flight. Table 7 summarized the experiments done.

<table>
<thead>
<tr>
<th>Simulation name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOKNEES</td>
<td>Knees are not allowed to bend</td>
</tr>
<tr>
<td>NOLOBACK</td>
<td>Lower back is not allowed to bend</td>
</tr>
<tr>
<td>NOHIPS</td>
<td>Hips are not allowed to bend</td>
</tr>
<tr>
<td>NOUPBACK</td>
<td>Upper back is not allowed to bend</td>
</tr>
<tr>
<td>FASTSWNG</td>
<td>Angular speed increased by 25%</td>
</tr>
<tr>
<td>FASTWHIP</td>
<td>Angular speed increased by 25%, hip flexion/extension range increase by 10%</td>
</tr>
<tr>
<td>LONGLEGS</td>
<td>Leg length increase by 5 cm, body mass increased by 2 kg</td>
</tr>
</tbody>
</table>

When the computer simulation program is run, the user is led through several steps to input the parameters of the program, then the simulation is run through the giant swing phase. The gymnast’s position is shown on the computer screen as each frame of data is computed. At the point of release, the user is asked for an angular momentum value to use in the flight. The user may also alter the linear
velocity at release to correct for errors in the giant swing velocities. The program stops if recatch is expected and informs the user of where the hands are located in space at that frame. It will continue until recatch and then inform the user of the recatch frame. In the postprocessing phase, the user is given the opportunity to view the animation again and to view plots of the forces and torques. The data is then written to files and can be read by the user or imported into a spreadsheet for further analysis. An example run of the simulation is provided in Appendix I.
RESULTS AND DISCUSSION

Part 1: Analysis

It is difficult to make any sweeping statements about "typical" Tkatchev performance based on measurements from two gymnasts. This section will, however, discuss the differences and similarities between and within gymnasts, as well as the joint location, timing, and magnitude of force and torque peaks, and the assumptions made in the analysis model. It will compare the results from this analysis with previous studies of the Tkatchev, and will use video observation to attempt to correlate any anomalous results with physical events.

Strength and range of motion data

As described in the Methods section, the gymnasts' strength and range of motion were tested for selected joints using the Biodex dynamometer. Table 8 summarizes the results of the strength testing for the elbows, shoulders and back. Subject 2 had more elbow and back strength than Subject 1 while both gymnasts exhibited similar strength characteristics for the shoulder joints. The range of motion of the joints in single plane motion was also measured using the Biodex dynamometer. This data is summarized in Table 9.
### Table 8: Strength data

<table>
<thead>
<tr>
<th>Joint (motion)</th>
<th>Subject 1</th>
<th>Subject 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left side</td>
<td>Right side</td>
</tr>
<tr>
<td>Elbow (flex/ext)</td>
<td>flex 49.4</td>
<td>ext 50.7</td>
</tr>
<tr>
<td>Back (flex/ext)</td>
<td>flex 148.7</td>
<td>ext 265.8</td>
</tr>
<tr>
<td>Shoulder (flex/ext)</td>
<td>flex 106.8</td>
<td>ext 89.1</td>
</tr>
<tr>
<td>Shoulder (ab/adduct)</td>
<td>ab 78.1</td>
<td>ad 90.0</td>
</tr>
</tbody>
</table>

Note that Subject 1 had a reduced range of motion in his right elbow. Subject 2 exhibited larger range of motion than Subject 1 in the shoulders. In both strength and range of motion, both subjects displayed similar characteristics.

### Table 9: Anatomical range of motion data

<table>
<thead>
<tr>
<th>Joint (motion)</th>
<th>Subject 1</th>
<th>Subject 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left side</td>
<td>Right side</td>
</tr>
<tr>
<td>Elbow (flex/ext)</td>
<td>from: -8.0 to: 145.0</td>
<td>from: -6.0 to: 127.0</td>
</tr>
<tr>
<td>Back (flex/ext)</td>
<td>from: -57.0 to: 29.0</td>
<td>N/A N/A</td>
</tr>
<tr>
<td>Shoulder (flex/ext)</td>
<td>from: -12.0 to: 168.0</td>
<td>from: -33.0 to: 180.0</td>
</tr>
<tr>
<td>Shoulder (ab/adduct)</td>
<td>from: -1.0 to: 156.0</td>
<td>from: 0.0 to: 158.0</td>
</tr>
</tbody>
</table>
is reflected in similar characteristics in the Tkatchev forces and torques.

**Range of motion of joints during the Tkatchev performance**

The rotation angles of the distal segments were used as a measure of the range of motion of the joints during the Tkatchev performance. Since motions during the move are three-dimensional rather than planar, these angles do not correspond directly with the range of motion capabilities measured in the previous section. Table 10 summarizes the range of joint rotations for selected joints. Figures 15-17 are example plots of the shoulder joint angles versus time. These are for Subject 2, trial 9. Note that in all plots in this chapter, point B indicates the release point, and point C indicates the recatch point. Note that the flight phase shows some step changes in the joint angles. While there are definitely large shoulder motions during the flight, some of the jumps in the data are probably attributable to digitization errors, since the large range of motion and rapid arm motion during flight made digitization difficult due both to blur and the slow frame rate of the videotape. Subject 1 displayed somewhat less motion at the elbow than
Table 10: Average rotation angle ranges during Tkatchev performance

<table>
<thead>
<tr>
<th>Joint</th>
<th>Subject 1 average range of motion (degrees)</th>
<th>Subject 2 average range of motion (degrees)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A/P axis</td>
<td>l. axis</td>
</tr>
<tr>
<td>R.elbow</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>L.elbow</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>R.shldr</td>
<td>92</td>
<td>360</td>
</tr>
<tr>
<td>L.shldr</td>
<td>87</td>
<td>360</td>
</tr>
<tr>
<td>upper back</td>
<td>27</td>
<td>45</td>
</tr>
<tr>
<td>lower back</td>
<td>11</td>
<td>13</td>
</tr>
<tr>
<td>R. hip</td>
<td>60</td>
<td>41</td>
</tr>
<tr>
<td>L. hip</td>
<td>55</td>
<td>42</td>
</tr>
<tr>
<td>R. knee</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>L. knee</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Neck</td>
<td>42</td>
<td>55</td>
</tr>
</tbody>
</table>

Subject 2, possibly because of the range of motion deficit in the right elbow due to the previous injury. Subject 1 also had a smaller range of motion in the shoulder and neck. This may indicate that Subject 1 used his body more efficiently, since the neck contributes little to the move. The larger range of motion of the shoulder for Subject 2 coincides with the flight phase of the move. Subject 1 may reduce the motion of the arms during flight to limit extraneous movement and get to a ready position for recatch more quickly.
Figure 15: Anterior-posterior rotations of the right shoulder for Subject 2, Trial 9.

Figure 16: Longitudinal axis rotations for right shoulder of Subject 2, Trial 9.
Subject 1 had more frontal plane motion (abduction and adduction) in the shoulder than Subject 2. This indicates that Subject 1's arms came out to the side during the flight phase, while Subject 2 circled the arms overhead, then dropped the hands in front of the body. Dropping the arms to the side may provide a smoother, more direct motion for the arms, but there may be a trade-off with more lateral torque on the shoulders at the recatch.

During the performance of the Tkatchev, the upper and lower back "joints" had 70-80 degrees range of motion. These ranges are actually an aggregate of the smaller range of motions at the individual vertebrae. This indicates that motion at the back should not be neglected, as occurs in

Figure 17: Mediolateral rotation of right shoulder for Subject 2, Trial 9.
many human body models. When digitizing using the Ariel default model, it was obvious that much motion was neglected by ignoring the back movement. The line connecting the shoulder and hip was not aligned with the torso at all. It instead resembled the string of a bow, with the body being curved and only the ends meeting.

There appears to be significant twisting motion at the upper trunk. This is probably due to the asymmetry of hand and arm forces that was observed. There were also significant axial rotations at the shoulder and the hip joints. This indicates that the inclusion of axial rotations in the human body model is worthwhile.

**Peak forces and torques during Tkatchev performance**

The peak magnitudes of forces and torques from each gymnast for each joint are summarized in Tables 11 and 12. Since these values are approximations, based on a modeling process, the magnitudes have been rounded to two significant figures. This provides a feel for the data without presenting the impression of highly accurate results.

In these tables, the x-direction is the along the anterior-posterior axis of the proximal segment, the y-direction is along the long axis of the proximal segment, and the z-direction is along the mediolateral axis of the proximal segment. The reference segment orientation was
Table 11: Force peaks at each joint by gymnast

Note: Forces are in Newtons, followed by the time, in seconds, when the peak occurred (with the point of release).

<table>
<thead>
<tr>
<th>Subject 1, Trial 4</th>
<th>Joint</th>
<th>X-direction</th>
<th>Y-direction</th>
<th>Z-direction</th>
<th>Magnitude</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mag(N)</td>
<td>time</td>
<td>Mag(N)</td>
<td>time</td>
</tr>
<tr>
<td>Right hand</td>
<td>-930.</td>
<td>-0.44</td>
<td>2000.</td>
<td>0.92</td>
<td>1300.</td>
</tr>
<tr>
<td>Left hand</td>
<td>-970.</td>
<td>-0.16</td>
<td>1300.</td>
<td>0.18</td>
<td>960.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>-1400.</td>
<td>-0.44</td>
<td>-1700.</td>
<td>0.92</td>
<td>-1400.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>-1200.</td>
<td>-0.16</td>
<td>-1100.</td>
<td>0.84</td>
<td>970.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>-1100.</td>
<td>-0.84</td>
<td>-1800.</td>
<td>0.92</td>
<td>1500.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>-700.</td>
<td>-0.56</td>
<td>-1200.</td>
<td>0.48</td>
<td>1100.</td>
</tr>
<tr>
<td>Neck</td>
<td>-96.</td>
<td>0.46</td>
<td>-110.</td>
<td>0.14</td>
<td>89.</td>
</tr>
<tr>
<td>Upper back</td>
<td>1500.</td>
<td>-0.82</td>
<td>-3500.</td>
<td>-0.44</td>
<td>-590.</td>
</tr>
<tr>
<td>Lower back</td>
<td>1000.</td>
<td>0.20</td>
<td>2200.</td>
<td>0.52</td>
<td>210.</td>
</tr>
<tr>
<td>Right hip</td>
<td>500.</td>
<td>0.42</td>
<td>1000.</td>
<td>0.50</td>
<td>130.</td>
</tr>
<tr>
<td>Left hip</td>
<td>530.</td>
<td>0.18</td>
<td>970.</td>
<td>0.52</td>
<td>160.</td>
</tr>
<tr>
<td>Right knee</td>
<td>-120.</td>
<td>0.42</td>
<td>300.</td>
<td>0.52</td>
<td>-69.</td>
</tr>
<tr>
<td>Left knee</td>
<td>-110.</td>
<td>0.42</td>
<td>255.</td>
<td>0.54</td>
<td>75.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subject 1, Trial 8</th>
<th>Joint</th>
<th>X-direction</th>
<th>Y-direction</th>
<th>Z-direction</th>
<th>Magnitude</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mag(N)</td>
<td>time</td>
<td>Mag(N)</td>
<td>time</td>
</tr>
<tr>
<td>Right hand</td>
<td>-1200.</td>
<td>0.38</td>
<td>-2100.</td>
<td>0.94</td>
<td>-890.</td>
</tr>
<tr>
<td>Left hand</td>
<td>-870.</td>
<td>0.10</td>
<td>-1300.</td>
<td>0.40</td>
<td>780.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>-1800.</td>
<td>0.50</td>
<td>-1900.</td>
<td>0.94</td>
<td>-910.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>-1100.</td>
<td>0.12</td>
<td>-1100.</td>
<td>0.42</td>
<td>790.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>970.</td>
<td>0.36</td>
<td>-1700.</td>
<td>0.94</td>
<td>1600.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>430.</td>
<td>0.92</td>
<td>-1200.</td>
<td>0.42</td>
<td>1100.</td>
</tr>
<tr>
<td>Neck</td>
<td>-95.</td>
<td>0.52</td>
<td>-104.</td>
<td>0.14</td>
<td>140.</td>
</tr>
<tr>
<td>Upper back</td>
<td>2000.</td>
<td>0.36</td>
<td>-3300.</td>
<td>0.46</td>
<td>-550.</td>
</tr>
<tr>
<td>Lower back</td>
<td>950.</td>
<td>0.22</td>
<td>2000.</td>
<td>0.50</td>
<td>290.</td>
</tr>
<tr>
<td>Right hip</td>
<td>510.</td>
<td>0.22</td>
<td>890.</td>
<td>0.50</td>
<td>-89.</td>
</tr>
<tr>
<td>Left hip</td>
<td>480.</td>
<td>0.20</td>
<td>970.</td>
<td>0.50</td>
<td>140.</td>
</tr>
<tr>
<td>Right knee</td>
<td>-120.</td>
<td>0.72</td>
<td>280.</td>
<td>0.54</td>
<td>-65.</td>
</tr>
<tr>
<td>Left knee</td>
<td>-110.</td>
<td>0.78</td>
<td>300.</td>
<td>0.54</td>
<td>81.</td>
</tr>
</tbody>
</table>
Table 11 (continued)

**Subject 2, Trial 4**

<table>
<thead>
<tr>
<th>Joint</th>
<th>X-direction (N)</th>
<th>Y-direction (N)</th>
<th>Z-direction (N)</th>
<th>Magnitude (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mag time</td>
<td>Mag Time</td>
<td>Mag Time</td>
<td>Mag time</td>
</tr>
<tr>
<td>Right hand</td>
<td>-310. .50 -1400. .70 -770. .50 1500. .68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left hand</td>
<td>-930. -.54 -3300. -.54 1200. -.60 3600. -.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right elbow</td>
<td>-450. .70 -1300. .70 1200. -.60 1500. .68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left elbow</td>
<td>-1500. -.54 -3100. -.54 1200. -.60 3600. -.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R shoulder</td>
<td>620. -.54 -1400. .68 -180. -.94 1500. .68</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L shoulder</td>
<td>1700. -.52 -3200. -.58 180. -.52 3600. -.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neck</td>
<td>97. .02 -140. .20 43. .08 150. .14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Upper back</td>
<td>2500. -.54 -3000. -.58 620. -.22 3700. -.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lower back</td>
<td>-1200. .68 2000. -.60 -450. -.14 2000. -.60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right hip</td>
<td>500. -.20 870. -.62 -100. .60 940. -.60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left hip</td>
<td>490. -.20 790. -.60 -130. -.12 910. -.60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right knee</td>
<td>160. -.12 250. -.60 89. -.12 260. -.58</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left knee</td>
<td>210. -.08 260. -.60 76. .46 270. -.60</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Subject 2, Trial 9**

<table>
<thead>
<tr>
<th>Joint</th>
<th>X-direction (N)</th>
<th>Y-direction (N)</th>
<th>Z-direction (N)</th>
<th>Magnitude (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mag time</td>
<td>Mag Time</td>
<td>Mag Time</td>
<td>Mag time</td>
</tr>
<tr>
<td>Right hand</td>
<td>-380. -.46 -2000. .84 -980. .78 2100. .82</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left hand</td>
<td>-470. -.44 -2200. -.46 620. .88 2300. -.46</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right elbow</td>
<td>-830. -.46 -1900. .84 -980. .78 2100. -.80</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left elbow</td>
<td>-1000. -.46 -2100. -.48 630. .88 2300. -.48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R shoulder</td>
<td>1100. -.44 -1900. .82 -230. .80 2100. .80</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L shoulder</td>
<td>1200. -.16 -2300. -.48 220. -.16 2300. -.48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neck</td>
<td>98. .12 -130. .28 -19. -.18 160. .18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Upper back</td>
<td>2000. -.44 -3100. -.46 -610. .82 3700. -.48</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lower back</td>
<td>-1100. -.72 1900. -.52 -260. -.08 2000. -.52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right hip</td>
<td>510. -.18 870. -.52 -84. -.12 940. -.52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left hip</td>
<td>550. -.58 910. -.52 110. -.70 970. -.52</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right knee</td>
<td>170. -.06 270. -.16 -110. .52 280. -.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Left knee</td>
<td>190. -.02 320. -.52 -67. -.02 320. -.52</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 12: Moment peaks at each joint by gymnast
Note: Moments are in Newton-meters, followed in parentheses by the time with respect to release when the peak occurred.

<table>
<thead>
<tr>
<th>Subject 1, Trial 4</th>
<th>Joint</th>
<th>X-direction (N-m)</th>
<th>Y-direction (N-m)</th>
<th>Z-direction (N-m)</th>
<th>Magnitude (N-m)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mag</td>
<td>time</td>
<td>Mag</td>
<td>time</td>
<td>Mag</td>
</tr>
<tr>
<td>Right hand</td>
<td>-69.</td>
<td>.92</td>
<td>33.</td>
<td>-.82</td>
<td>-23.</td>
</tr>
<tr>
<td>Left hand</td>
<td>38.</td>
<td>-.46</td>
<td>-27.</td>
<td>-.46</td>
<td>-25.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>280.</td>
<td>.82</td>
<td>-100.</td>
<td>.90</td>
<td>-180.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>-210.</td>
<td>-.54</td>
<td>-26.</td>
<td>-.16</td>
<td>-210.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>-360.</td>
<td>-.82</td>
<td>-250.</td>
<td>.92</td>
<td>-560.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>470.</td>
<td>-.52</td>
<td>-280.</td>
<td>-.56</td>
<td>230.</td>
</tr>
<tr>
<td>Neck</td>
<td>9.</td>
<td>-1.10</td>
<td>-11.</td>
<td>-1.08</td>
<td>-12.</td>
</tr>
<tr>
<td>Upper back</td>
<td>-140.</td>
<td>.54</td>
<td>82.</td>
<td>.50</td>
<td>-290.</td>
</tr>
<tr>
<td>Lower back</td>
<td>98.</td>
<td>-.52</td>
<td>-82.</td>
<td>.50</td>
<td>220.</td>
</tr>
<tr>
<td>Right hip</td>
<td>-63.</td>
<td>.36</td>
<td>33.</td>
<td>.64</td>
<td>110.</td>
</tr>
<tr>
<td>Left hip</td>
<td>-42.</td>
<td>-.08</td>
<td>-54.</td>
<td>.52</td>
<td>100.</td>
</tr>
<tr>
<td>Right knee</td>
<td>-12.</td>
<td>.66</td>
<td>3.</td>
<td>.66</td>
<td>21.</td>
</tr>
<tr>
<td>Left knee</td>
<td>11.</td>
<td>.48</td>
<td>-2.</td>
<td>-.44</td>
<td>18.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subject 1, Trial 8</th>
<th>Joint</th>
<th>X-direction (N-m)</th>
<th>Y-direction (N-m)</th>
<th>Z-direction (N-m)</th>
<th>Magnitude (N-m)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mag</td>
<td>time</td>
<td>Mag</td>
<td>time</td>
<td>Mag</td>
</tr>
<tr>
<td>Right hand</td>
<td>-67.</td>
<td>.96</td>
<td>41.</td>
<td>-.38</td>
<td>-17.</td>
</tr>
<tr>
<td>Left hand</td>
<td>40.</td>
<td>-.12</td>
<td>-25.</td>
<td>-.42</td>
<td>-16.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>180.</td>
<td>.86</td>
<td>38.</td>
<td>-.38</td>
<td>-150.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>180.</td>
<td>-.10</td>
<td>-82.</td>
<td>-.12</td>
<td>-120.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>-380.</td>
<td>-.40</td>
<td>-210.</td>
<td>-.44</td>
<td>-400.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>380.</td>
<td>-.46</td>
<td>390.</td>
<td>-.10</td>
<td>200.</td>
</tr>
<tr>
<td>Neck</td>
<td>11.</td>
<td>-1.02</td>
<td>-13.</td>
<td>-1.04</td>
<td>-14.</td>
</tr>
<tr>
<td>Upper back</td>
<td>-160.</td>
<td>-.52</td>
<td>86.</td>
<td>.54</td>
<td>280.</td>
</tr>
<tr>
<td>Lower back</td>
<td>100.</td>
<td>-.52</td>
<td>-86.</td>
<td>.54</td>
<td>-230.</td>
</tr>
<tr>
<td>Right hip</td>
<td>-58.</td>
<td>.30</td>
<td>48.</td>
<td>.58</td>
<td>110.</td>
</tr>
<tr>
<td>Left hip</td>
<td>57.</td>
<td>-.28</td>
<td>-58.</td>
<td>.54</td>
<td>110.</td>
</tr>
<tr>
<td>Right knee</td>
<td>-11.</td>
<td>.60</td>
<td>2.1</td>
<td>.86</td>
<td>22.</td>
</tr>
<tr>
<td>Left knee</td>
<td>13.</td>
<td>.56</td>
<td>-2.3</td>
<td>.88</td>
<td>20.</td>
</tr>
<tr>
<td>Joint</td>
<td>X-direction (N-m)</td>
<td>Y-direction (N-m)</td>
<td>Z-direction (N-m)</td>
<td>Magnitude (N-m)</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mag</td>
<td>time</td>
<td>Mag</td>
<td>Time</td>
<td>Mag</td>
</tr>
<tr>
<td>Right hand</td>
<td>-44.</td>
<td>.68</td>
<td>13.</td>
<td>-.92</td>
<td>-19.</td>
</tr>
<tr>
<td>Left hand</td>
<td>87.</td>
<td>-.54</td>
<td>-39.</td>
<td>-.56</td>
<td>-51.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>-140.</td>
<td>-.54</td>
<td>-86.</td>
<td>.52</td>
<td>84.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>-260.</td>
<td>-.60</td>
<td>50.</td>
<td>-.88</td>
<td>-130.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>-250.</td>
<td>-.92</td>
<td>150.</td>
<td>-.96</td>
<td>-290.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>580.</td>
<td>-.58</td>
<td>160.</td>
<td>-.24</td>
<td>-370.</td>
</tr>
<tr>
<td>Neck</td>
<td>9.</td>
<td>-.22</td>
<td>7.</td>
<td>-1.20</td>
<td>20.</td>
</tr>
<tr>
<td>Upper back</td>
<td>110.</td>
<td>-.22</td>
<td>110.</td>
<td>-.10</td>
<td>-480.</td>
</tr>
<tr>
<td>Lower back</td>
<td>56.</td>
<td>-.80</td>
<td>-100.</td>
<td>-.10</td>
<td>-400.</td>
</tr>
<tr>
<td>Right hip</td>
<td>59.</td>
<td>.62</td>
<td>59.</td>
<td>.44</td>
<td>-140.</td>
</tr>
<tr>
<td>Left hip</td>
<td>-58.</td>
<td>-.10</td>
<td>59.</td>
<td>.46</td>
<td>-160.</td>
</tr>
<tr>
<td>Right knee</td>
<td>17.</td>
<td>-.12</td>
<td>6.</td>
<td>.44</td>
<td>-37.</td>
</tr>
<tr>
<td>Left knee</td>
<td>14.</td>
<td>.44</td>
<td>-5.</td>
<td>.44</td>
<td>-44.</td>
</tr>
<tr>
<td>Right hand</td>
<td>-62.</td>
<td>.80</td>
<td>11.</td>
<td>-.48</td>
<td>-15.</td>
</tr>
<tr>
<td>Left hand</td>
<td>60.</td>
<td>-.48</td>
<td>-19.</td>
<td>-.48</td>
<td>-31.</td>
</tr>
<tr>
<td>Right elbow</td>
<td>190.</td>
<td>.76</td>
<td>-79.</td>
<td>.78</td>
<td>85.</td>
</tr>
<tr>
<td>Left elbow</td>
<td>150.</td>
<td>-.16</td>
<td>-70.</td>
<td>-.14</td>
<td>-95.</td>
</tr>
<tr>
<td>R shoulder</td>
<td>-240.</td>
<td>-.82</td>
<td>170.</td>
<td>-.84</td>
<td>-470.</td>
</tr>
<tr>
<td>L shoulder</td>
<td>250.</td>
<td>-.88</td>
<td>150.</td>
<td>-.16</td>
<td>300.</td>
</tr>
<tr>
<td>Neck</td>
<td>11.</td>
<td>-.16</td>
<td>11.</td>
<td>-1.08</td>
<td>17.</td>
</tr>
<tr>
<td>Upper back</td>
<td>130.</td>
<td>-.14</td>
<td>71.</td>
<td>-.02</td>
<td>440.</td>
</tr>
<tr>
<td>Lower back</td>
<td>75.</td>
<td>.40</td>
<td>-70.</td>
<td>-.02</td>
<td>-380.</td>
</tr>
<tr>
<td>Right hip</td>
<td>-80.</td>
<td>.38</td>
<td>72.</td>
<td>.52</td>
<td>-130.</td>
</tr>
<tr>
<td>Left hip</td>
<td>76.</td>
<td>.38</td>
<td>-39.</td>
<td>.48</td>
<td>-150.</td>
</tr>
<tr>
<td>Right knee</td>
<td>-19.</td>
<td>.52</td>
<td>3.8</td>
<td>.52</td>
<td>-35.</td>
</tr>
<tr>
<td>Left knee</td>
<td>-12.</td>
<td>-.04</td>
<td>-2.6</td>
<td>-.04</td>
<td>-36.</td>
</tr>
</tbody>
</table>
shown in Figure 13 in the Methods section (inverted in a handstand position). The results were also computed in distal segment coordinates but, due to space limitations, these are not presented here.

From these tables, it can be seen that the Tkatchev is not completely symmetric. Subject 2, especially, exhibited a great deal of asymmetry in the hand forces. In trial 4, Subject 2 has hand forces four times higher at the right hand than at the left during the giant swing portion. The forces at the recatch are more balanced. Subject 1 shows some asymmetry at the hands at the time of recatch, although the giant swing is fairly symmetric. This recatch asymmetry may be due to catching one hand earlier than the other, or due to asymmetric body movement during flight. It was noted from the videotape that the gymnast caught the bar to the right of center, so the left hand deflected more easily than the right, since it was closer to the center of the bar. This may result in less force on the left hand at recatch.

The force peaks tended to coincide with four points during the performance. These points occurred before and after the bottom of the giant swing, before the release, and after the recatch. The forces during the giant swing tend to be the highest, probably due to gravity effects and due to trying to generate speed for the upswing. There were also some very large peaks at recatch, where the impact at
the hands alters the trajectory of the arms, stopping the movement of the hands. The upper back has the largest force peaks, as high as 5.6 times the body weight. Resultant forces of up to 3000 N. were found in both the x (anterior-posterior) and the y (longitudinal) directions. The spine is more stable under vertical loading, where muscles and ligaments run in the direction of the load, than under transverse loading, where the anterior and posterior longitudinal ligaments provide much of the support to prevent displacement of vertebrae (Kreighbaum and Barthels 1990). Because of this, the injury potential would probably be greater when the force is in the anterior-posterior direction. Kreighbaum and Barthels (1990) point out that if ligaments are relied upon for support on a long-term basis, there can eventually be stretching of the ligaments and damage to the discs, bodies, articular capsules and spinous processes. This indicates that strong muscles in the back and abdomen are necessary to prevent back injury in this move.

The calculated torque data was compared with the maximum torques that the gymnasts could voluntarily generate (see strength data). It was found the elbow torques sustained were about twice the established elbow strengths. The shoulder torques sustained were about seven times the strength for flexion/extension and five to six times the
strength values for abduction/adduction. The lower back torques sustained were similar to the strength data. The body can sustain torques much greater than it can generate because the combination of bone, ligament, tendon, and muscle can prevent or slow movement more readily than muscle can generate force. It can be seen by viewing the joint rotation data that much of the torques endured are done so isometrically (that is, there is no motion at the joints). This means that the joint can be stabilized using various muscle groups and anatomic structures.

The fact that the torques at the shoulders are so much higher than the strength data indicates that there may be increased potential for injury at the shoulder joint. This joint is highly dependent on soft tissue for its stability. If the gymnast were to err, he would not have the strength to correct the movement and overstretching of the shoulder soft tissue structures may occur.

Because the strength data does not correlate well with the torques experienced, it was decided to use the measured torque data from Subject 2, trial 9 as the reference standard. Torques calculated in the model that greatly exceed the peaks for this trial (that are at least 110% of the maximum measured torques) will generate warning messages. This approach, similar to that of Bentham (1987),
uses the measured torques as the standard that the body can endure.

Figures 18-21 show examples of the forces at the hands versus time for each subject. Note that there is a great deal of asymmetry in the hand forces. Subject 1 shows a pattern of using more force on the left hand at the top of the giant swing, more force on the right hand at the bottom of the giant swing, more force on the left hand at release, and more force on the right hand at recatch. The reasons for this asymmetry are not clear. It may be due to a weight shift to keep the center of gravity aligned or possibly due to range of motion considerations at release, since Subject 1 had less range of motion capability in his left elbow. The recatch asymmetry may result directly from the release asymmetry.

As a measure of asymmetry, Tables 13 and 14 were constructed. Three points in the move where peaks typically occur were chosen as comparison points. These were at the bottom of the giant swing, just before release, and at recatch. The ratio of left side to right side forces and torques were computed.

The ratio of the left to the right hand for Subject 1, trial 8 is shown in Figure 22. This figure shows that the left hand dominates at the release and the right at the bottom of the giant and at the recatch.
Figure 18: Left hand force magnitude for Subject 1, trial 4
Figure 19: Right hand force magnitude for Subject 1, trial 4

Figure 20: Left hand force magnitude for Subject 2, trial 9
Figure 21: Right hand force magnitude for Subject 2, trial 9
Table 13: Force ratios at peaks (Left/right)

<table>
<thead>
<tr>
<th>Subject 1, #4</th>
<th>Subject 1, #8</th>
<th>Subject 2, #4</th>
<th>Subject 2, #9</th>
</tr>
</thead>
<tbody>
<tr>
<td>bog rel rec</td>
<td>bog rel rec</td>
<td>bog rel rec</td>
<td>bog rel rec</td>
</tr>
<tr>
<td>Time -.5</td>
<td>-.18 .92</td>
<td>-.52 -.12 .94</td>
<td>-.60 -.14 .70</td>
</tr>
<tr>
<td>hand .75</td>
<td>1.6 .42</td>
<td>.66 1.57 .46</td>
<td>6.1 1.80 1.0 1.54 1.85 .59</td>
</tr>
<tr>
<td>elb .75</td>
<td>1.6 .42</td>
<td>.67 1.59 .47</td>
<td>6.1 1.84 1.02 1.53 1.83 .57</td>
</tr>
<tr>
<td>shld .75</td>
<td>1.5 .42</td>
<td>.71 1.62 .48</td>
<td>6.05 1.75 1.03 1.53 1.87 .57</td>
</tr>
<tr>
<td>hips .96</td>
<td>1.08 .98</td>
<td>1.09 1.10 .91</td>
<td>.96 1.15 1.17 1.04 0.86 1.02</td>
</tr>
</tbody>
</table>

| Abbreviations are as follows: bog: bottom of giant, rel: release, rec: recatch. |

Table 14: Moment ratios at peaks (Left/right)

<table>
<thead>
<tr>
<th>Subject 1, #4</th>
<th>Subject 1, #8</th>
<th>Subject 2, #4</th>
<th>Subject 2, #9</th>
</tr>
</thead>
<tbody>
<tr>
<td>bog rel rec</td>
<td>bog rel rec</td>
<td>bog rel rec</td>
<td>bog rel rec</td>
</tr>
<tr>
<td>Time -.5</td>
<td>-.18 .92</td>
<td>-.52 -.12 .94</td>
<td>-.60 -.14 .70</td>
</tr>
<tr>
<td>hand .77</td>
<td>1.57 .42</td>
<td>.66 1.57 .46</td>
<td>6.1 1.9 1.0 1.55 1.85 .59</td>
</tr>
<tr>
<td>elb .62</td>
<td>1.21 .31</td>
<td>.53 1.42 .25</td>
<td>2.15 .37 .92 .46 4.82 .22</td>
</tr>
<tr>
<td>shld .59</td>
<td>1.50 .29</td>
<td>.51 1.43 .26</td>
<td>2.29 .68 .91 .57 4.03 .23</td>
</tr>
<tr>
<td>hips 2.07</td>
<td>1.27 1.41</td>
<td>2.73 1.06 .66</td>
<td>.79 .99 1.39 .87 1.31 .88</td>
</tr>
</tbody>
</table>

| Abbreviations are as follows: bog: bottom of giant, rel: release, rec: recatch. |

Figure 22: Ratio of left hand to right hand force for Subject 1, trial 8
The following were noted about the peak forces and torques: The peak torques generally occur before release or at the recatch. The lower body (hips, knees and lower back) has high torques during the flight/recatch phase, probably because the legs are used to complete the rotation of the body over the bar. The peak torques in the arms occur primarily at the recatch, and the back has highest torques during the beat section of the giant swing. The shoulders have high torque peaks both at the recatch and at the release. The legs and neck are highest during the flight and recatch phases of the Tkatchev.

The peak forces in the lower body for both gymnasts occurred during the giant swing, probably due to the beating action of the legs. Subject 1 seemed to use the right side more during the recatch and the left side during the release and giant swing. Peak forces for Subject 1 occurred at the lower back at the recatch and during the giant swing. The peak forces at the recatch are primarily in the x-direction, while they are primarily in the y (axial) direction for the giant swing. The peak magnitude of the forces at the upper back joint is approximately five times the body weight of Subject 1. Peak forces at each hand are about 3.5 times the body weight. These results correlate well with the values found in previous studies of giant swing forces (Smith 1981, Yamashita et al. 1979, Kopp and Reid 1980).
Subject 2 had somewhat higher forces at the upper back during the giant swing than Subject 1, although the magnitude at recatch was similar. Subject 2 also displayed asymmetry in the loading of the body. The right side had the highest forces at the recatch, while the left side had higher forces during the giant swing and the release.

The highest forces and torques experienced by the gymnast during the Tkatchev are found primarily at the hands, elbows, shoulders and torso. This is different from movements in many other sports, which utilize the lower body for load bearing. The upper body is not designed for load bearing as is the lower body. The shoulders are less stable than the hip joint in order to allow maximum range of motion. The high forces at the shoulders require a great deal of strength to maintain stability. The highly developed muscular structure in the gymnasts' upper bodies attests to this fact.

The major torque peaks at the shoulders occur at the recatch for the abduction/adduction direction, at the bottom of the giant swing for the axial rotation direction, and there are four peaks in the flexion/extension direction correlating to before the bottom of the giant, after the bottom of the giant, release, and recatch. Figures 23-27 show these forces for the right shoulder of Subject 2, trial 9. At the recatch, the gymnast has extended and abducted shoulders. The impact with the bar causes the shoulders to flex and stops the
Figure 23: Right shoulder torques for Subject 2, trial 9: X-direction

Figure 24: Right shoulder torques for Subject 2, trial 9: Y-direction

Figure 25: Right shoulder torques for Subject 2, trial 9: Z-direction

Figure 26: Upper back torques for Subject 2, trial 9: Z-direction

Figure 27: Lower back torques for Subject 2, trial 9: Z-direction
rotation of the arms in the frontal plane. The arms move overhead and align with the torso for the transition into the next move. The muscles are working to restrict the movement at the shoulders initiated by the bar contact. This peak is of primary concern in the assessment of injury potential because it is a sharp peak and is less predictable because it is related to impact rather than a smooth transition within the swing that is seen at the bottom of the giant.

The upper back torque shows a peak at release that is caused by hyperextension of the back. This correlates with a large axial force peak on the spine. It appears that the shoulders absorption of high torques at recatch prevents excessively high values at the back.

The lower back has its primary torque peaks at the points of release and recatch. These are due to hyperextension of the spine. The gymnasts in this study did not exhibit excessive lower back motion, but it is probable that gymnasts with less shoulder strength may have higher forces at the lower back.

Center of gravity trajectory

Figures 28 through 31 show the path of the body’s center of gravity as calculated by the analysis program. Figure 32 shows the path of the body’s center of gravity for Subject 2, Trial 9 as computed by the Ariel software. It was noted that
there was some shifting of the center of gravity along the \( z \)-axis. This \( z \)-motion indicates that there was some force in the \( z \)-direction at the hands. However, comparing the motion along the \( z \)-axis with that of motion along the \( x \) and \( y \) axes, it is seen that the \( z \)-displacement is minor in comparison. The acceleration in the \( z \)-direction was calculated and compared to that in the \( y \)-direction (Figure 33). The \( z \)-acceleration appears to be negligible compared to the \( y \)-direction acceleration.

Subject 1 had a higher flight phase of the Tkatchev, while Subject 2 had a flatter trajectory during flight. Subject 2 displayed more \( z \)-displacement during the performance of the movement, but had similar acceleration values.

Comparison of theoretical bar deflections with measured bar deflections

Figures 34-37 show the comparison between the theoretical bar deflections (calculated as described in Theory section) and measured deflections (digitized on the Ariel system using 2-d data). It can be seen that there is significant agreement between the expected and measured deflections.

The calculated values for Subject 1, trial 8 appear to be oversmoothed and are in less agreement than those for Subject 2. This probably is due to oversmoothing of the displacement data prior to transformation using the Ariel system. Note
Figure 28: Center of gravity trajectory in the x-y plane: Subject 1, trial 4

Figure 29: Center of gravity trajectory in the x-y plane: Subject 2, trial 9

Figure 30: Center of gravity trajectory in the y-z plane: Subject 1, trial 4

Figure 31: Center of gravity trajectory in the y-z plane: Subject 2, trial 9
Figure 32: Center of gravity trajectory for Subject 2, trial 9 as calculated by Ariel system

Figure 33: Comparison of y and z center of gravity accelerations for Subject 2, trial 9
Figure 34: Bar deflections at the left hand for Subject 1, trial 8

Figure 35: Bar deflections at the right hand for Subject 1, trial 8

Figure 36: Bar deflections at the left hand for Subject 2, trial 9

Figure 37: Bar deflections at the right hand for Subject 2, trial 9
that in the calculated results for Subject 1, the peaks are smaller than the measured peaks and some of the smaller peaks are not present. This indicates that there was oversmoothing of the data.

The deflection calculations used the assumption of pinned ends for the bar. This assumption appears to be satisfactory, since the deflections agree so well. Therefore, the deflection/force model developed here was utilized in the simulation model to relate hand location and forces.

Comparison of analysis results with Ariel default calculations

The forces at the hands were measured in this analysis using strain gages and strength of materials theory. Since the measured deflections at the bar agreed closely with the theoretical deflections, there is reason to believe that the measured forces are accurate. Figures 38-41 show Ariel and strain gage force calculations for Subject 2, trial 9 in global coordinates. The forces computed by the Ariel program and the values measured in this study agree well, except for some peaks where Ariel's results are smaller. Since the exact algorithm used by Ariel is considered proprietary, it is impossible to resolve these differences completely. Some differences may be attributable to over-smoothing of the data for the Ariel defaults. It may also be due to differences in mass distribution algorithms, because if Ariel computes the
Figure 38: Right hand x- forces for Subject 2, trial 9: Ariel versus measured

Figure 39: Right hand y- forces for Subject 2, trial 9: Ariel versus measured

Figure 40: Left hand x- forces for Subject 2, trial 9: Ariel versus measured

Figure 41: Left hand y- forces for Subject 2, trial 9: Ariel versus measured
center of gravity location closer to the bar, it may reduce
the reaction force values. There may also be some differences
due to the fact that Ariel assumes a fixed ground point, and
the bar deflected up to 20 cm.

Comparison of kinematics with previously published data

Gervais and Tally (1993) published kinematic data for
nine Tkatchevs as performed by Canadian gymnasts during
competition. Table 15 summarizes some of the data they
presented along with data calculated in this analysis.
The values determined here are not in complete agreement, but
do have similar qualities. The hip range of motion was
similar, although the angles were about 20 degrees larger in
the current study. The radius of rotation was larger,
possibly because the gymnasts were taller than those in the
previous study or because of the human body model used and the
way in which the mass was distributed. This cannot be
confirmed, because anthropometric data was not provided in the
previous study. Some of the differences may also be
attributable to the three-dimensional aspect of this study,
since the joint rotations do not necessarily agree exactly
with the angles projected onto a plane.

Fink (1987) theorized that the rotation in the flight
phase of the Tkatchev was initiated by the transfer of
momentum from the legs to the entire body at the time of
Table 15: Comparison of kinematic data with previously published data

<table>
<thead>
<tr>
<th>Variable</th>
<th>Gervais &amp; Tally (average)</th>
<th>Present study (average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Height of center of mass at release (m)</td>
<td>0.57</td>
<td>0.73</td>
</tr>
<tr>
<td>Radius of rotation at release (m)</td>
<td>0.85</td>
<td>1.06</td>
</tr>
<tr>
<td>Maximum height of center of mass in flight (m)</td>
<td>0.85</td>
<td>1.17</td>
</tr>
<tr>
<td>Height of center of mass at regrasp (m)</td>
<td>0.24</td>
<td>0.03</td>
</tr>
<tr>
<td>Radius of rotation at regrasp (m)</td>
<td>0.69</td>
<td>1.11</td>
</tr>
<tr>
<td>Hip angle, start of beat (degrees)</td>
<td>-42</td>
<td>-23</td>
</tr>
<tr>
<td>Hip angle, end of beat (degrees)</td>
<td>61</td>
<td>75</td>
</tr>
</tbody>
</table>

release. The angular velocity data for Subject 2, trial 9 was viewed and Fink's theory was confirmed. At the release, the trunk rotation changed direction and increased, while the leg rotations almost stopped. This correlated with a hip torque peak of about -150 N-m. This indicates that muscle force stopped the rotation of the hips and the angular momentum was transferred to the entire body.
Results of the redundant equations for the force at the lower back and the moment at the upper back

The forces at the lower back were computed using the pelvis segment and again using the mid-trunk segment. The torques at the upper back were computed using the upper trunk segment and the mid-trunk segment. The differences in the two approaches are shown in Table 16. The differences appear large in some cases, which is disconcerting, especially when the direction of the force or torque is opposite. It must be noted that the calculations at the upper and lower back joints are done from the "chain" of the upper or lower body segments, and therefore are an accumulation of all the errors involved in the rotation matrices and the kinematics for all of the previous segments. The errors would therefore be expected to be the largest at these joints. The measured differences can therefore be seen as a measure of the maximum error involved. In future research, it may be preferred to use both approaches and have an overdetermined system that can be solved using least squares techniques. The problem with such an approach would be the magnitude and complexity involved in setting up the system of equations.

Because there was a great deal of blur and because the legs were toward the edges of the calibration space, it would be reasonable to give more weight to the results calculated using the upper body "chain" rather than the lower body "chain".
Table 16: Differences in back kinetics computed from upper body or lower body "chain"

<table>
<thead>
<tr>
<th></th>
<th>Subject 1, trial 4</th>
<th>Subject 1, trial 8</th>
<th>Subject 2, trial 4</th>
<th>Subject 2, trial 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>upper back torque</td>
<td>630 (N-m)</td>
<td>2400 (N-m)</td>
<td>680 (N-m)</td>
<td>2500 (N-m)</td>
</tr>
<tr>
<td>force</td>
<td>2600 (N)</td>
<td>1200 (N)</td>
<td>2500 (N)</td>
<td>1300 (N)</td>
</tr>
<tr>
<td>lower back torque</td>
<td>1300 (N-m)</td>
<td>4500 (N-m)</td>
<td>1800 (N-m)</td>
<td>3500 (N-m)</td>
</tr>
<tr>
<td>force</td>
<td>1300 (N)</td>
<td>4500 (N)</td>
<td>1700 (N)</td>
<td>4100 (N)</td>
</tr>
<tr>
<td>Maximum diff (time &amp; dir)</td>
<td>(X)</td>
<td>(X)</td>
<td>(Z)</td>
<td>(X)</td>
</tr>
<tr>
<td>Minimum diff (time &amp; dir)</td>
<td>(Y)</td>
<td>(Z)</td>
<td>(Y)</td>
<td>(Z)</td>
</tr>
<tr>
<td>Average x diff</td>
<td>100</td>
<td>280</td>
<td>150</td>
<td>310</td>
</tr>
<tr>
<td>Average y diff</td>
<td>67</td>
<td>17</td>
<td>90</td>
<td>23</td>
</tr>
<tr>
<td>Average z diff</td>
<td>2</td>
<td>100</td>
<td>77</td>
<td>113</td>
</tr>
<tr>
<td>Average magnitude diff</td>
<td>82</td>
<td>280</td>
<td>180</td>
<td>280</td>
</tr>
</tbody>
</table>

Figures 42-49 show the differences between the two methods of calculating lower back forces and upper back moments for Subject 2, Trial 9.

Modeling assumptions and errors involved

This experimental approach had a great deal of modeling involved. The human body was idealized using Hanavan's model. The accelerations of the segments were computed using
numerical methods. The forces at the hands were generated from strain gage data using estimates of hand location and the assumption of the direction of the force, as well as numerical smoothing of the resulting data. The moments at the hands were assumed to be primarily in the z-direction and proportional to the forces. The marker locations were assumed to be fixed on the segments and soft tissue motion was neglected. The segments were assumed to be rigid bodies, connected by ideal pin or ball joints, whose shapes and inertial properties were constant. They were assumed to be of constant density.

Each of these assumptions carries an inherent error factor. The cumulative effect of these errors can be large, especially if the assumptions are ill-founded.

The Ariel force computation and the comparison of y- and z- center of gravity accelerations show that the assumption of the forces at the hands not having a z-component results in minimal error, since the z acceleration of the center of gravity is much smaller than the x and y accelerations. The z forces at the hands would therefore be much smaller than the x or y forces.

The Hanavan model of the human body uses rigid bodies for each segment. Other researchers have attempted to use more complex models to reduce the error involved in the assumption
Figure 42: Lower back force (X) for Subject 2, trial 9: comparison of methods of computation

Figure 43: Lower back force (Y) for Subject 2, trial 9: comparison of methods of computation

Figure 44: Lower back force (Z) for Subject 2, trial 9: comparison of methods of computation

Figure 45: Lower back force (Magnitude) for Subject 2, trial 9: comparison of methods of computation
Figure 46: Upper back torque (X) for Subject 2, trial 9: comparison of computation methods

Figure 47: Upper back torque (Y) for Subj. 2, trial 9: comparison of computation methods

Figure 48: Upper back torque (Z) for Subject 2, trial 9: comparison of computation methods

Figure 49: Upper back torque (Magnitude) for Subject 2, trial 9: comparison of computation methods
of inertial properties (Yeadon 1990). While the inertial properties do improve, the time constraints of taking additional anthropometric measurements is daunting.

Ackland et al. (1988) found that the inertial properties of segments are minimally affected by the assumption of a constant density rigid body, therefore this assumption seems well-founded.

The literature shows that using more markers will reduce the error involved with soft tissue motion (Verstraete and Soutas-Little 1990), but it also increases the time and effort required for digitization. In addition, it is difficult to find locations for a large number of markers per segment that do not interfere with the athlete’s ability to perform. In addition, for moves such as the Tkatchev where the gymnast changes direction with respect to the camera, a great deal of estimation of marker location is required. If the digitizer is not an extremely skilled estimator, the errors involved in estimating marker locations may negate the improvements from an increased number of markers. Because of this, it seems that the three or four markers per segment is the most that is feasible to use in this experimental approach.

The assumption of ideal pinned or ball and socket joints is also false, since the joints of the human body actually all have six degrees of freedom (Kinzel and Gutkowski, 1983). The error involved in using these idealized joints has not been
reported. The problems involved in identifying motion for all six degrees of freedom are impressive, though, and no research was found that attempted a kinetic analysis using the six-degree of freedom model for any joint.

**Experimental problems and suggestions for further study**

There were several difficulties in the experimental process that limited the effectiveness and accuracy of the data. Of primary concern is the frame rate of the cameras was only 30 frames per second. This is quite slow for parts of this movement and there was considerable blur, especially at the bottom of the giant swing. This greatly reduced the accuracy of the digitization process. In addition, the rate at which the strain gage data was sampled, and adjusting the strain gage data to match the sample rate of the Ariel results probably clipped some force peaks, since the final frame period was only .02 seconds. It was impossible to use smaller time intervals because of the limitations of the Ariel system in the total number of frames it could store. It is possible that some of the error associated with the strain gage data sampling rate could be counteracted by the 12-20% overestimation of the forces discussed in the Methods section.

In addition, the lighting used for illuminating the markers was inadequate, resulting in good illumination for part of the move, and poor illumination in other parts of the
move. This was in part because of the large field of view for
the move and the focused lighting. Brighter lights with
larger fields of illumination would have provided much better
contrast between the markers and the background throughout the
move.

The cameras were set up to maximize the number of angles
available, but there were problems with the field of view
again and with the alignment of the cameras with the gymnast
(which reduced one camera to a 2-d view). Using some
different angles, such as from above, might improve the view
of the move. The preliminary investigations were done from a
position slightly above the bar, and the digitization process
was much easier. A wide angle lens may also improve the view,
if it does not distort the distances.

The calibration device was made by hand from PVC pipe.
It was too flexible, too difficult to align precisely with the
bar and did not have enough markers to provide really good
calibration. The problem in designing a calibration device
that was adequate to the task was to make a device that had
precise locations of markers and would cover a large space.
In addition, it was desirable for the device to be portable
and easily assembled, since some of the studies were done in a
competitive environment, where minimal interference was
required. The use of wood dowels would have allowed more
markers, but it may not have been as resilient, and it would
probably have had to be quite heavy in order to maintain straight dowels at distances of up to 5 feet from the bar. Better machining of the device may also have improved its properties, since it would have made it easier to put the device together in a repeatable manner.

Marker location on the segments was estimated using a combination of video data and anthropometry. It would have been better to videotape each subject in a static reference position and use that data as the reference frame data.

Limitations of the experimental approach

The experimental approach to determining the forces and torques in the Tkatchev release move is limited to determining the forces and torques under the exact laboratory conditions that existed during the experiment. In this experiment, the subjects were similar in build, strength and ability. It was impossible, therefore, to draw any conclusions about the effects of these variables on the kinetics experienced by the joints during the move. In addition, it would be of interest to determine if the joint forces and torques increased if the trajectory became too high or too low. It would be impossible to do this in a laboratory situation because of the difficulty of getting a gymnast to perform as instructed and also because of the risk of injury. The experimental approach is expensive in terms of equipment and of time. In order to obtain kinetic
data under conditions that may cause injury, one must take data for a large number of moves and "hope" for a bad one. This is expensive as well as dangerous. In addition, to obtain statistical data, one would have to test a large number of subjects. Since the digitization process for one move alone takes about three hours, the time associated with a statistical analysis would be unacceptable unless the project was undertaken by a large group of researchers with immense computing resources.

The simulation approach to experimentation allows the researcher and his or her computer to invent situations and test the effects on the performer without risk to the gymnast and without setting up video or film equipment. The parameters can be altered at the whim of the researcher and a number of situations may be tested. Although some real data is required as a basis for starting, the huge database required for experimental analysis is not needed.

The second part of this study developed a computer simulation model for the Tkatchev. The results from the simulation are discussed in the next section.

Part 2-Modeling and Simulation

Validation of model

The model was validated using the results of Subject 2, trial 9 as input. There were problems with using the exact
results of the trial because of significant rotation during the flight phase due to asymmetric movement of the arms and legs during flight. Since there were points during the flight where the arms moved more rapidly than could be tracked at 30 fps, it was assumed that there were errors in the data for one or both sides of the body. Given this, two "test" data sets were devised. The first set was symmetric about the body’s center line using the right side data as the basis and the left side as a mirror image of the right. The second set used the left side data with the right side a mirror image. Neither data set allowed axial or lateral trunk or head rotation.

During the course of the validation, it was found that the results were significantly closer to the analysis results when the hands were assumed fixed at the bar, rather than letting them deflect according to the forces. Because the model of the bar seemed to hinder, rather than aid, the simulation, the option to leave the bar fixed was allowed. This option was used in the validation results. Figure 50 shows the differences in left hand forces when the bar is fixed versus when bar deflections are calculated. In addition, the assumption of zero moment at the hands was used in the validation.
Figure 50: Left hand force magnitude: fixed bar versus deflecting bar results

Figure 51 shows the angular velocity of the mid-trunk segment, which is the variable from which all of the segment kinematics are derived, for the analysis, and for both simulation data sets. Both sets follow the analysis results reasonably well, with an underestimation of the large peak during the giant swing. Part of the difference can be explained by the fact that only one rotation is allowed, while rotations in all three directions were measured in the analysis. Also, the determination of the angular velocity was based on filtered data of the rotation of the mid-trunk, which was calculated from a least squares algorithm. Both the
filtering and the least squares method may have contributed to the errors.

Figures 52-54 compare the results of the analysis and the simulation for force magnitudes at the right shoulder, upper back and lower back. Figures 55-57 compare the torque magnitude results for these joints.

Tables 17 and 18 compare the validation results with the analysis results for all joints at the three peak points: giant, release and recatch. The data indicates the difference between the peak in the validation and the peak in the
Figure 52: Forces at the right shoulder: analysis versus simulation validation

Figure 53: Forces at the upper back: analysis versus simulation validation
Figure 54: Force magnitudes at the lower back: analysis versus simulation validation

Figure 55: Torques at the right shoulder: analysis versus simulation validation
Figure 56: Torques at the upper back: analysis versus simulation validation

Figure 57: Torque magnitudes at the lower back: analysis versus simulation validation
Table 17: Peak force magnitude differences between validation and analysis results

<table>
<thead>
<tr>
<th>Joint</th>
<th>Right side data</th>
<th>Left side data</th>
<th>Giant Release</th>
<th>Recatch</th>
<th>Giant Release</th>
<th>Recatch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Neck</td>
<td>14(10)</td>
<td>16(11)</td>
<td>109(97)</td>
</tr>
<tr>
<td>Right shoulder</td>
<td>-177</td>
<td>18</td>
<td>-1434</td>
<td>47</td>
<td>-27</td>
<td>-154</td>
</tr>
<tr>
<td></td>
<td>(12)</td>
<td>(2)</td>
<td>(70)</td>
<td>(2)</td>
<td>(18)</td>
<td>(53)</td>
</tr>
<tr>
<td>Left shoulder</td>
<td>-1023</td>
<td>-780</td>
<td>-509</td>
<td>-873</td>
<td>-952</td>
<td>178</td>
</tr>
<tr>
<td></td>
<td>(44)</td>
<td>(46)</td>
<td>(44)</td>
<td>(37)</td>
<td>(57)</td>
<td>(16)</td>
</tr>
<tr>
<td>Right elbow</td>
<td>326</td>
<td>71</td>
<td>-1340</td>
<td>20</td>
<td>-133</td>
<td>-1084</td>
</tr>
<tr>
<td></td>
<td>(22)</td>
<td>(8)</td>
<td>(64)</td>
<td>(1)</td>
<td>(14)</td>
<td>(52)</td>
</tr>
<tr>
<td>Left elbow</td>
<td>-504</td>
<td>-676</td>
<td>-341</td>
<td>-810</td>
<td>-880</td>
<td>-85</td>
</tr>
<tr>
<td></td>
<td>(22)</td>
<td>(40)</td>
<td>(31)</td>
<td>(35)</td>
<td>(53)</td>
<td>(8)</td>
</tr>
<tr>
<td>Upper back</td>
<td>-426</td>
<td>-306</td>
<td>-1963</td>
<td>78</td>
<td>-820</td>
<td>-1306</td>
</tr>
<tr>
<td></td>
<td>(12)</td>
<td>(15)</td>
<td>(68)</td>
<td>(2)</td>
<td>(40)</td>
<td>(45)</td>
</tr>
<tr>
<td>Lower back</td>
<td>-330</td>
<td>-107</td>
<td>-413</td>
<td>-256</td>
<td>-327</td>
<td>-162</td>
</tr>
<tr>
<td></td>
<td>(16)</td>
<td>(6)</td>
<td>(31)</td>
<td>(13)</td>
<td>(18)</td>
<td>(12)</td>
</tr>
<tr>
<td>Right hip</td>
<td>-169</td>
<td>-135</td>
<td>-160</td>
<td>-300</td>
<td>-213</td>
<td>-123</td>
</tr>
<tr>
<td></td>
<td>(18)</td>
<td>(20)</td>
<td>(34)</td>
<td>(32)</td>
<td>(31)</td>
<td>(26)</td>
</tr>
<tr>
<td>Left hip</td>
<td>-208</td>
<td>-50</td>
<td>-206</td>
<td>-339</td>
<td>-128</td>
<td>-169</td>
</tr>
<tr>
<td></td>
<td>(21)</td>
<td>(8)</td>
<td>(40)</td>
<td>(35)</td>
<td>(21)</td>
<td>(23)</td>
</tr>
<tr>
<td>Right knee</td>
<td>174</td>
<td>-19</td>
<td>-50</td>
<td>180</td>
<td>-49</td>
<td>-21</td>
</tr>
<tr>
<td></td>
<td>(64)</td>
<td>(7)</td>
<td>(28)</td>
<td>(67)</td>
<td>(18)</td>
<td>(12)</td>
</tr>
<tr>
<td>Left knee</td>
<td>123</td>
<td>33</td>
<td>-28</td>
<td>129</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>(38)</td>
<td>(15)</td>
<td>(18)</td>
<td>(40)</td>
<td>(1)</td>
<td>(.6)</td>
</tr>
<tr>
<td>Right hand</td>
<td>67</td>
<td>-16</td>
<td>-1300</td>
<td>48</td>
<td>-160</td>
<td>-1102</td>
</tr>
<tr>
<td></td>
<td>(4)</td>
<td>(2)</td>
<td>(62)</td>
<td>(3)</td>
<td>(17)</td>
<td>(53)</td>
</tr>
<tr>
<td>Left hand</td>
<td>-756</td>
<td>-757</td>
<td>-299</td>
<td>-775</td>
<td>-901</td>
<td>-101</td>
</tr>
<tr>
<td></td>
<td>(32)</td>
<td>(45)</td>
<td>(27)</td>
<td>(33)</td>
<td>(54)</td>
<td>(9)</td>
</tr>
<tr>
<td>Average (%)</td>
<td>24</td>
<td>17</td>
<td>47</td>
<td>27</td>
<td>26</td>
<td>28</td>
</tr>
<tr>
<td>Joint</td>
<td>Right side data</td>
<td>Left side data</td>
<td>Giant Release</td>
<td>Recatch</td>
<td>Giant Release</td>
<td>Recatch</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>----------------</td>
<td>---------------</td>
<td>----------</td>
<td>---------------</td>
<td>----------</td>
</tr>
<tr>
<td>Neck</td>
<td>17 (81)</td>
<td>5 (24)</td>
<td>20 (125)</td>
<td>9 (43)</td>
<td>4 (20)</td>
<td>4 (25)</td>
</tr>
<tr>
<td>Right shoulder</td>
<td>294 (85)</td>
<td>97 (41)</td>
<td>-142 (28)</td>
<td>204 (59)</td>
<td>102 (43)</td>
<td>-281 (56)</td>
</tr>
<tr>
<td>Left shoulder</td>
<td>217 (67)</td>
<td>3 (1)</td>
<td>15 (4)</td>
<td>227 (70)</td>
<td>8 (2)</td>
<td>-124 (36)</td>
</tr>
<tr>
<td>Right elbow</td>
<td>182 (130)</td>
<td>32 (30)</td>
<td>-119 (56)</td>
<td>91 (65)</td>
<td>43 (41)</td>
<td>-139 (66)</td>
</tr>
<tr>
<td>Left elbow</td>
<td>204 (172)</td>
<td>-29 (17)</td>
<td>-15 (14)</td>
<td>113 (96)</td>
<td>-18 (11)</td>
<td>-35 (33)</td>
</tr>
<tr>
<td>Upper back</td>
<td>205 (70)</td>
<td>-37 (8)</td>
<td>83 (25)</td>
<td>250 (86)</td>
<td>-59 (13)</td>
<td>140 (42)</td>
</tr>
<tr>
<td>Lower back</td>
<td>71 (26)</td>
<td>-140 (36)</td>
<td>32 (16)</td>
<td>-2 (1)</td>
<td>155 (67)</td>
<td>165 (83)</td>
</tr>
<tr>
<td>Right hip</td>
<td>19 (19)</td>
<td>-71 (53)</td>
<td>-12 (12)</td>
<td>34 (35)</td>
<td>43 (48)</td>
<td>8 (13)</td>
</tr>
<tr>
<td>Left hip</td>
<td>23 (24)</td>
<td>-86 (60)</td>
<td>7 (7)</td>
<td>38 (40)</td>
<td>58 (64)</td>
<td>13 (13)</td>
</tr>
<tr>
<td>Right knee</td>
<td>0 (0)</td>
<td>-19 (53)</td>
<td>0 (0)</td>
<td>11 (42)</td>
<td>-16 (44)</td>
<td>6 (26)</td>
</tr>
<tr>
<td>Left knee</td>
<td>0 (0)</td>
<td>-20 (55)</td>
<td>0 (0)</td>
<td>11 (42)</td>
<td>-17 (46)</td>
<td>6 (26)</td>
</tr>
<tr>
<td>Right hand</td>
<td>-45 (100)</td>
<td>-28 (100)</td>
<td>-62 (100)</td>
<td>-45 (100)</td>
<td>-28 (100)</td>
<td>-62 (100)</td>
</tr>
<tr>
<td>Left hand</td>
<td>-70 (100)</td>
<td>-50 (100)</td>
<td>-32 (100)</td>
<td>-28 (100)</td>
<td>-50 (100)</td>
<td>-32 (100)</td>
</tr>
<tr>
<td>Average (%)</td>
<td>67 (100)</td>
<td>44 (100)</td>
<td>37 (100)</td>
<td>60 (100)</td>
<td>46 (100)</td>
<td>47 (100)</td>
</tr>
</tbody>
</table>
analysis. Following the magnitude of the difference, the percentage error is indicated.

The validation results show that the model closely matches the analysis in angular velocity, forces, and moments. Some differences exist which may be attributed to the difficulty in modeling the deflections at the hand, the differences associated with the symmetric and asymmetric movements, and the errors involved in the analysis, which were discussed earlier. In addition, it was found that there was significant noise in the data for the forces and torques, especially at the neck and knee joints, where the angular and linear kinematics were dependent upon all the segments in the chain from the mid-trunk to the head and from the mid-trunk to the shanks. Post-filtering the results data provided a much closer fit with the experimental data (see Figure 58). In addition, since the moment at the hands was neglected in the validation, the moments in the upper body chain had a built-in "error". Since there was more error in the upper body than the lower body, it appears that the assumption of the moments at the hands can have a significant effect on the moments in the rest of the body.
Results of experiments with simulation program

The results of changing the simulation parameters on the angular velocity of the reference segment, the forces at the shoulder, and the forces at the lower back are shown in Table 19. These results are illustrative of the capabilities of the model. Further analysis of the results of multiple simulations would be required to determine any overall conclusions from this data. The parameters involved in each simulation were listed in the methods section.
Problems in simulation model and simulation program

Initially, the simulation was to use the joint angles versus time as input, along with initial conditions and compute the body position and orientation as described by Huston and Passerello (1971). This approach would entail the solving of six simultaneous second order differential equations using a numerical integration scheme over a time period exceeding two seconds. The forces at the hands were to be computed using the deflections of the bar as they related to the acceleration of the center of gravity. (See Appendix H for details of the solution). The problems encountered with this approach were many. First of all, the actual center of gravity location from the bar and the calculated distance differed greatly, since the calculated distance was based on a rigid body model with fixed joints and the shoulders in reality could "stretch" and rotate and the center of the joint at the shoulder, especially, could move. This difference resulted in the calculated deflections at the bar being much larger than in reality, these large deflections reflected large forces, which greatly altered the trajectory of the body. Using this approach resulted in an oscillation building up in the bar with the gymnast being "slingshotted" back and forth.
<table>
<thead>
<tr>
<th>Simulation</th>
<th>Effect on $\omega_i$</th>
<th>Effects on $F_{\text{Right}}$ Shoulder</th>
<th>Effects on $F_{\text{Lower Back}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NOKNEES</td>
<td>No significant difference</td>
<td>Increased release peak 30%</td>
<td>Increased peak at release 50%</td>
</tr>
<tr>
<td>NOLOBACK</td>
<td>Sharper peak during flight</td>
<td>Decreased release peak 30%</td>
<td>Increased peak at release 50%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Added small peaks during giant</td>
<td>Decreased recatch peaks 30%</td>
</tr>
<tr>
<td>NOHIPS</td>
<td>Multiple peaks during flight</td>
<td>Increased giant swing peaks 50%</td>
<td>Increased giant swing peaks 75%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Increased release peaks 30%</td>
<td>Decreased recatch peaks 25%</td>
</tr>
<tr>
<td>NOUPBACK</td>
<td>Slower rise on recatch</td>
<td>Increased release peak 30%</td>
<td>Increased giant swing peak 75%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Decreased recatch peaks 30%</td>
<td>Decreased recatch peaks</td>
</tr>
<tr>
<td>LONGLEGS</td>
<td>No significant difference</td>
<td>Increased release peak 30%</td>
<td>Increased giant swing peak 50%</td>
</tr>
<tr>
<td>FASTSWNG</td>
<td>Increased magnitude 50% No reversal at recatch</td>
<td>Increased giant swing peaks 125%</td>
<td>Increased giant swing peak 125%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Increased release peaks 100%</td>
<td></td>
</tr>
<tr>
<td>FASTWHIP</td>
<td>Increased magnitude 50% Sharp reversal on recatch</td>
<td>Increased giant swing peaks 125%</td>
<td>Increased giant swing peak 125%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Increased release peaks 100%</td>
<td></td>
</tr>
</tbody>
</table>
Since the primary purpose of this study was to determine the forces and torques at the joints, rather than to calculate the body orientation and trajectory per se, it was decided to predetermine the trajectory of the center of gravity during the hand contact phase. Defining the center of gravity angle and the joint angles defined the position of the center of gravity. The orientation of the body could then be determined and the forces and torques computed as in the analysis. Although this reduced the sophistication of the model, its results were equally valid in terms of kinetic results and there were many factors which could be used as experimental variables.

It was hoped that the deflection of the bar could be incorporated into the model and its effects determined. This did not seem to be possible, because the iterative process developed to compute the distance from the cg to the global origin developed large oscillations and became unstable. It is possible that a different approach to solving this may be more stable. The error involved in neglecting the bar movements is probably not large, since the Ariel results assumed a fixed bar and had similar force curves. The primary result would be that the small peaks seen right before release could not be duplicated.
In the simulation, the internal computer round-off errors also came into play. The out of plane angular velocities during flight, which should have been zero, were between .01 and .25 rad/sec during the flight phase. Dapena (1979) noted that this type of simulation was only valid (minimal angle of orientation error) for 0.6-0.8 seconds. After that time the errors in the computer round-off and the numerical integrations became too large. Since the flight phase in the Tkatchev is around .6 seconds, this is probably why the errors were seen.

Recommendations for further study

In order for simulations to be run, a set of initial data must exist that can be altered. For experimental simulations, it seems that a large number of "reference" Tkatchevs should be used to get a set of experimental data for further validation and to prevent the user from having to manufacture joint angle data. Instead, the user could say "give me Gymnast A using Gymnast D’s arm style during flight" and see what the effects would be. Further study is needed to provide a "library" of data for using in comparative studies.

It also became clear in the validation process, that the quality of the data is paramount to obtaining meaningful results. All numerical methods used have inherent errors. In addition, all computing adds errors to the data. If the data
is highly suspect early on, the results are almost meaningless after all the number crunching is done. In order to have good validation and simulation results, the data must be well defined at all points in the move, the input data must be smoothed before performing the computations, and experimental data must match the results of the simulation. In further data collection, frame rates of 100 frames per second or higher should be used in order to capture the rapid movements at the legs during the giant and at the arms during the flight phase.

In addition, it would be advantageous if a method were developed so that cameras could be focussed on different areas, rather than all on one large area in order to get better resolution of points. It is foreseeable that a future method will use six cameras, with two focussed on each segment of the working area. A program would then have to be written that could track points from one area to the next and correlate the data into one file. This would allow detailed study of the fine movements at the hands and wrists and better resolution of the gross movements in the lower body.

If this program were to be developed further, advances in the programming are needed. Maintaining the memory requirements of this program to the amount of memory available on a PC was difficult. A skilled computer programmer could probably reduce the memory requirements of this program, allow
use of the mouse or a light pen in making selections, improve the graphics and the dialog interfaces, provide better output for the user without having to use a spreadsheet program to get graphics and statistics, improve the speed of running the program, and allow more user options for running different simulations.

Further research and programming advances could also include allowing the release to be changed by a fraction of a frame to experiment with the optimal release point, to allow alteration of the center of gravity angle file (which is currently fixed input) to alter the trajectory for part of the move only. It would also be useful to be able to interactively change the angles of the joints in only one area of the move, rather than changing the overall range of motion.

Finally, although this program was written to simulate the Tkatchev release move, it is broad enough in scope to be used for any swinging or release move on the high bar. In order to do so, center of gravity information and joint angle information for other moves would have to be input into data files. After that, this program could be used as-is to experiment with a number of moves on the horizontal bar. Uneven bar moves that utilize a single bar could also be simulated using this program.
CONCLUSIONS

In this research, forces and torques at a gymnast’s joints during the performance of a Tkatchev release move on the horizontal bar were calculated using a three-dimensional rigid body model. Segments were modeled as rigid bodies jointed by frictionless ball and socket or hinge joints.

In the analysis section, two subjects were videotaped and the kinematics of their segments and the joint kinetics were calculated throughout the move. The highest forces were found in the back, shoulders and arms. The highest torques were found in the shoulders. The largest force and torque peaks occurred near the bottom of the giant swing, prior to the release and after the recatch. The highest force peaks were about five times the gymnast’s weight, and the peak torques were about 600 N-m.

In the simulation, the body kinematics and kinetics were calculated using the center of gravity trajectory and joint angles as input during hand contact phases and using the principle of conservation of angular momentum during the flight phase. In the validation of this model, an average error of 28% was found in joint force peaks and an average error of 50% was found in joint torque peaks. The overall error was much lower, since the peaks were affected by noise and anomalies in the data.
Difficulty in developing a working model was encountered due to deficiencies in the rigid body linkage model of the human body. This resulted in additional constraints being required to define the input.

This model of the Tkatchev release move has the potential to aid researchers, students, coaches and gymnasts in understanding the effects of various factors on the forces experienced during the performance of release moves. It can be used with various joint angle input to simulate other release moves, to determine the contribution of various joint motions to the completion of the move, to determine the effects of gymnast size on forces, to optimize the performance of the move and to establish factors that may contribute to gymnast injuries. It provides a basis for building a database of kinetics in gymnastics. Such a database would help sports medicine practitioners, coaches, and equipment manufacturers in reducing the stresses on the gymnast by altering the equipment or the performance characteristics. It also provides a start in determining the muscle groups that must be developed in order to perform different moves.
REFERENCES


Kopp, P.M. and J.G. Reid. 1980. A force and torque analysis


ACKNOWLEDGMENTS

I would like to acknowledge the contributions of the following people to this work. Without their assistance, I could never have completed this dissertation.

Dr. Cynthia Tant provided me with equipment for performing the experimental stage of this work, with technical assistance in taking the data, with suggestions for improved techniques and with a sports-related biomechanics view of the project. Several of her students also assisted in setting up and running the experiments. Dr. Loran Zachary provided me with expert advice of the use of strain gages and assisted in application of the strain gages. He also provided use of equipment for taking the strain gage data. Coach Dave Mickelson provided use of his practice gym and Coach Tom Davis of the University of Iowa permitted the two volunteer subjects to participate and accompanied them to Iowa State. I would also like to thank the two subjects for participating in this project. Dr. Jesus Dapena, (Department of Kinesiology, Indiana State University, Bloomington, IN 47405) developed the CYLBOD program used to obtain the computer animation of the gymnast. My stepson, Matthew Ketchum, gave up part of his summer vacation to help me take data and agreed to let me use him as a guinea pig in our test runs. Dr. Patrick Patterson provided moral support for finishing this dissertation and
patiently read and re-read all my drafts. Finally, I would like to thank my husband, Charles Ketchum, for never giving up on me, for making me work when I didn’t want to, and for making me take a break when I was on the verge of collapse. I would also like to acknowledge my son, Bradley Ketchum, whose first year of life was spent watching Mom work on the computer.
APPENDIX A

DERIVATION OF EQUATIONS FOR SIMULATION

The simulation program takes joint rotation angles versus time and the angular position of the center of gravity around the bar and calculates the center of gravity trajectory and the rotation angles of the reference segment (the mid-trunk) with respect to the fixed reference frame. All rotation angles derived in this appendix refer to a 3-1-2 sequence of right hand rotations from the proximal segment coordinates to the distal segment coordinates. From the center of gravity trajectory and the reference segment rotations, segment kinematics and joint kinetics are determined.

The input variables are the joint rotations $\alpha_{ij}$, $\beta_{ij}$, and $\gamma_{ij}$, where the $ij$ subscript refers to the rotation angles between segments $i$ and $j$ and $i$ is the segment proximal to $j$. The unknowns are $\alpha$, $\beta$, and $\gamma$ (without subscripts), which refer to the rotation angles between the fixed reference frame and the segment used as the body reference frame (in this case, the mid trunk).

For the simulation, the following equations need to be derived in terms of the unknown variables:

1) A method of computing the center of gravity path from the input data. This was done using $r-\theta$ coordinates as discussed earlier.

2) A method of computing the reference segment rotations
from the input data. This was done using the methods of Spoor and Veldpaus (1980) discussed earlier.

3) The angular velocity of each segment with respect to the fixed reference frame \( (\omega^N) \).

4) The angular acceleration of each segment with respect to the fixed reference frame \( (\alpha^N) \).

5) The location of the center of gravity of the body with respect to the origin of the reference segment \( (\overline{P}_c) \).

6) The location of the center of gravity of each segment with respect to the reference segment origin \( (\overline{q}_j) \) and with respect to the body center of gravity \( (\overline{P}_j) \).

7) The location of the point of force application (hands) with respect to the body center of gravity \( (\overline{r}_j) \).

8) Computation of the inertia dyadic of each segment in the reference segment coordinates \( (\overline{I}) \).

9) The dot products of the angular acceleration and angular velocity with the inertia dyadics and the cross products in the moment equation \( (\overline{I}^x, \overline{I}^x\overline{\omega}, \overline{r}_j^xF_j, \text{and } \overline{P}_j^x\overline{m}_j) \).

10) The first and second derivatives of the vectors from the origin of the reference segment to the center of gravity of each segment \( (q'\text{'s}) \) and of \( P_c \), the vector from the origin of the reference segment to the center of gravity of the entire body. The linear accelerations of the segments will be computed from this information.
Figure A-1 shows the body segments in the orientation where all segment coordinate systems are aligned with the fixed reference frame. Refer to this figure for segment numbering.

A note about terminology used in this derivation: The notation is loosely based on that of Kane and Levinson (1985). For vectors with a superscript, for instance, $\overline{\omega}^N_j$, the second superscript $j$, refers to the body and the first to the coordinate system the vector is referenced to. So $\overline{\omega}^N_j$ is the angular velocity of segment $j$ with respect to fixed reference frame $A$, and $\overline{\omega}^N_i$ would be the angular velocity of segment $j$ with respect to segment $i$.

![Figure A-1: Numbering of the segments](image_url)
Angular Velocity of the Segments

The first step in the process is to find the angular velocity of each segment in terms of the unknown variables.

The mid-trunk segment, segment 1, is the reference segment for this simulation. Given a 3-1-2 right hand rotation of reference segment 1 (whose basis vectors are $\vec{b}_i$, $\vec{b}_j$ and $\vec{b}_k$) through angles $\alpha$, $\beta$, and $\gamma$, with respect to the fixed reference frame A (whose basis vectors are $\vec{a}_i$, $\vec{a}_j$ and $\vec{a}_k$) then the angular velocity of segment 1 with respect to the fixed reference frame is:

$$\omega^{A_1} = (\beta c\alpha - \gamma c\beta s\alpha) \vec{a}_1 + (\beta s\alpha + \gamma c\beta c\alpha) \vec{a}_2 + (\dot{\alpha} + \gamma s\beta) \vec{a}_3$$  \hspace{1cm} (1)

in fixed reference frame coordinates and

$$\omega^{A_1} = (\beta c\gamma - \alpha c\beta s\gamma) \vec{b}_1 + (\dot{\alpha} s\beta + \gamma) \vec{b}_2 + (\dot{\alpha} c\beta c\gamma + \beta s\gamma) \vec{b}_3$$  \hspace{1cm} (2)

in the mid trunk reference frame.

For any two reference frames fixed in different segment, the rotation matrix for a 3-1-2 rotation from the proximal segment $i$ to the distal segment $j$ is shown below, where $n_{ik}$ and $n_{jk}$ ($k=1,3$) are the basis vectors for segments $i$ and $j$.

$$\begin{pmatrix} n_{i1} \\ n_{i2} \\ n_{i3} \end{pmatrix} = \begin{pmatrix} c\alpha_{ij} c\gamma_{ij} - s\alpha_{ij} s\beta_{ij} s\gamma_{ij} \\ - s\alpha_{ij} s\beta_{ij} c\gamma_{ij} + c\alpha_{ij} s\gamma_{ij} + s\alpha_{ij} s\beta_{ij} c\gamma_{ij} \\ c\beta_{ij} c\gamma_{ij} - c\alpha_{ij} s\beta_{ij} s\gamma_{ij} \\ s\gamma_{ij} c\beta_{ij} \\ s\beta_{ij} c\gamma_{ij} \end{pmatrix} \begin{pmatrix} n_{j1} \\ n_{j2} \\ n_{j3} \end{pmatrix}$$  \hspace{1cm} (3)
The rotation matrix between segments $i$ and $j$, $[R_{ij}]$, can be written in generalized terms as follows:

\[
\begin{bmatrix}
  n_{i1} \\
  n_{i2} \\
  n_{i3}
\end{bmatrix} =
\begin{bmatrix}
  R_{11}^{ij} & R_{12}^{ij} & R_{13}^{ij} \\
  R_{21}^{ij} & R_{22}^{ij} & R_{23}^{ij} \\
  R_{31}^{ij} & R_{32}^{ij} & R_{33}^{ij}
\end{bmatrix}
\begin{bmatrix}
  n_{j1} \\
  n_{j2} \\
  n_{j3}
\end{bmatrix}
\] (4)

The angular velocity $\omega^{Ab}$ of any segment $b$ relative to the fixed reference frame $A$, can be derived from the relationship:

\[ \omega^{Ab} = \omega^A + \omega^b \] (5)

so, for a series of linked segments, the angular velocity of the most distal segment can be written as:

\[ \omega^{ab} = \omega^{a1} + \omega^{12} + \omega^{23} + \ldots + \omega^{nb} \] (6)

From equation (4), the angular velocity of the mid-trunk segment can be written in reference segment coordinates as:

\[ \omega^{A1} = u_{11} \overline{b}_1 + u_{12} \overline{b}_2 + u_{13} \overline{b}_3 \]

where

\[ u_{11} = \beta c\gamma - \alpha c\beta s\gamma \] (7)

\[ u_{12} = \alpha s\beta + \gamma \]

\[ u_{13} = \alpha c\beta c\gamma + \beta s\gamma \]
For segment 2, the upper trunk, the angular velocity with respect to the mid-trunk can be written in reference segment (mid-trunk) coordinates using the relationships in equation (3):

\[ \omega_{12} = u_{21} \overrightarrow{b}_1 + u_{22} \overrightarrow{b}_2 + u_{23} \overrightarrow{b}_3 \]  

where

\[ u_{21} = \beta_{12} \alpha_{12} - \gamma_{12} c \beta_{12} s \alpha_{12} \]
\[ u_{22} = \beta_{12} s \alpha_{12} + \gamma_{12} c \beta_{12} c \alpha_{12} \]
\[ u_{23} = \alpha_{12} + \gamma_{12} s \beta_{12} \]

Then, from equation (8),

\[ \omega^x = (u_{11} + u_{21}) \overrightarrow{b}_1 + (u_{12} + u_{22}) \overrightarrow{b}_2 + (u_{13} + u_{23}) \overrightarrow{b}_3 \]  

For segment 3, the pelvis, the angular velocity with respect to the reference segment is:

\[ \omega^{13} = u_{31} \overrightarrow{b}_1 + u_{32} \overrightarrow{b}_2 + u_{33} \overrightarrow{b}_3 \]  

where

\[ u_{31} = \beta_{13} \alpha_{13} - \gamma_{13} c \beta_{13} s \alpha_{13} \]
\[ u_{32} = \beta_{13} s \alpha_{13} + \gamma_{13} c \beta_{13} c \alpha_{13} \]
\[ u_{33} = \alpha_{13} + \gamma_{13} s \beta_{13} \]
Then the angular velocity of the pelvis with respect to the fixed reference frame is:

\[ \omega^{A3} = (u_{11} + u_{31}) \vec{b}_1 + (u_{12} + u_{32}) \vec{b}_2 + (u_{13} + u_{33}) \vec{b}_3 \]  

(13)

For segment 4, the head, the angular velocity with respect to the upper trunk is written in upper trunk (segment 2) coordinates as follows:

\[ \vec{\omega}^{24} = u_{41} \vec{n}_{21} + u_{42} \vec{n}_{22} + u_{43} \vec{n}_{23} \]  

(14)

where

\[
\begin{align*}
    u_{41} &= \beta_{24} c\alpha_{24} - \gamma_{24} c\beta_{24} s\alpha_{24} \\
    u_{42} &= \beta_{24} s\alpha_{24} + \gamma_{24} c\beta_{24} c\alpha_{24} \\
    u_{43} &= \alpha_{24} + \gamma_{24} s\beta_{24}
\end{align*}
\]

(15)

Using the rotation matrix between segments 1 and 2, \( \vec{\omega}^{24} \) can be written in terms of the segment 1 basis vectors:

\[
\begin{align*}
    \omega^{24} &= (u_{41} R_{11}^{12} + u_{42} R_{12}^{12} + u_{43} R_{13}^{12}) \vec{b}_1 \\
    &+ (u_{41} R_{21}^{12} + u_{42} R_{22}^{12} + u_{43} R_{23}^{12}) \vec{b}_2 \\
    &+ (u_{41} R_{31}^{12} + u_{42} R_{32}^{12} + u_{43} R_{33}^{12}) \vec{b}_3
\end{align*}
\]

(16)

\[ \omega^{24} = u_{41} \vec{b}_1 + u_{42} \vec{b}_2 + u_{43} \vec{b}_3 \]  

(17)
Then the angular velocity of the head with respect to the fixed reference frame is:

\[ \bar{\omega}_4 = \bar{\omega}_1 + \bar{\omega}_2 + \bar{\omega}_3 \]

\[ = (u_{11} + u_{21} + u_{41}) \bar{e}_1 + (u_{12} + u_{22} + u_{42}) \bar{e}_2 + (u_{13} + u_{23} + u_{43}) \bar{e}_3 \]  

(18)

For segment 7, the right upper arm, the angular velocity with respect to the upper trunk (segment 2) can be computed in segment 2 coordinates using the relationship from equation (3).

\[ \bar{\omega}^{27} = u_{71} \bar{r}_1 + u_{72} \bar{r}_2 + u_{73} \bar{r}_3 \]  

(19)

where

\[ u_{71} = \dot{\beta}_{27} c \alpha_{27} - \dot{\gamma}_{27} c \beta_{27} S \alpha_{27} \]

\[ u_{72} = \dot{\beta}_{27} s \alpha_{27} + \dot{\gamma}_{27} c \beta_{27} c \alpha_{27} \]

\[ u_{73} = \dot{\alpha}_{27} + \dot{\gamma}_{27} s \beta_{27} \]

(20)

Using the rotation matrix between segment 2 and 1, the angular velocity of segment 7 with respect to segment 2 can be written in terms of segment 1 basis vectors as follows:

\[ \bar{\omega}^{27} = (u_{71} R_{11}^{12} + u_{72} R_{12}^{12} + u_{73} R_{13}^{12}) \bar{e}_1 \]

\[ + (u_{71} R_{21}^{12} + u_{72} R_{22}^{12} + u_{73} R_{23}^{12}) \bar{e}_2 \]

\[ + (u_{71} R_{31}^{12} + u_{72} R_{32}^{12} + u_{73} R_{33}^{12}) \bar{e}_3 \]  

(21)
\[ \overline{\omega}^{27} = u_{\gamma_1} \overline{\Omega}_1 + u_{\gamma_2} \overline{\Omega}_2 + u_{\gamma_3} \overline{\Omega}_3 \]  

(22)

And the angular velocity of segment 7 with respect to the fixed reference frame is:

\[ \overline{\omega}^{A7} = \overline{\omega}^{A1} + \overline{\omega}^{12} + \overline{\omega}^{27} \]
\[ \overline{\omega}^{A7} = (u_{11} + u_{21} + u_{\gamma_1}) \overline{\Omega}_1 + (u_{12} + u_{22} + u_{\gamma_2}) \overline{\Omega}_2 + (u_{13} + u_{23} + u_{\gamma_3}) \overline{\Omega}_3 \]  

(23)

Similarly, for the left upper arm, segment 8, the angular velocity with respect to the upper trunk, segment 2, is:

\[ \overline{\omega}^{28} = u_{g1} \overline{\Omega}_{21} + u_{g2} \overline{\Omega}_{22} + u_{g3} \overline{\Omega}_{23} \]  

(24)

where

\[ u_{g1} = \beta_{28} \mathcal{c}_{28} - \gamma_{28} \mathcal{b}_{28} \mathcal{c}_{28} \]

\[ u_{g2} = \beta_{28} \mathcal{s}_{28} + \gamma_{28} \mathcal{c}_{28} \mathcal{b}_{28} \]

\[ u_{g3} = \alpha_{28} + \gamma_{28} \mathcal{s}_{28} \beta_{28} \]  

(25)

Rotating from segment 2 coordinates to reference segment coordinates, the angular velocity of segment 8 with respect to segment 2 is:

\[ \overline{\omega}^{28} = (u_{g1} \mathcal{R}_{12}^{12} + u_{g2} \mathcal{R}_{22}^{12} + u_{g3} \mathcal{R}_{32}^{12}) \overline{\Omega}_1 \]
\[ + (u_{g1} \mathcal{R}_{21}^{12} + u_{g2} \mathcal{R}_{22}^{12} + u_{g3} \mathcal{R}_{32}^{12}) \overline{\Omega}_2 \]
\[ + (u_{g1} \mathcal{R}_{31}^{12} + u_{g2} \mathcal{R}_{32}^{12} + u_{g3} \mathcal{R}_{33}^{12}) \overline{\Omega}_3 \]  

(26)
The angular velocity of the left upper arm, segment 8, with respect to the fixed reference frame, $A$, is:

$$\bar{\omega}^{28} = \omega'_{81} \vec{b}_1 + \omega'_{82} \vec{b}_2 + \omega'_{83} \vec{b}_3$$

(27)

For the thigh segments, a similar analysis is done. The angular velocity of the right thigh, segment 9, with respect to the pelvis, segment 3, is written in segment 3 coordinates as:

$$\bar{\omega}^{39} = \omega_{91} \vec{r}_{31} + \omega_{92} \vec{r}_{32} + \omega_{93} \vec{r}_{33}$$

(29)

where

$$\begin{align*}
\omega_{91} &= \beta_{39} c_{39} - \gamma_{39} c_{39} s_{39} \\
\omega_{92} &= \beta_{39} s_{39} + \gamma_{39} c_{39} c_{39} \\
\omega_{93} &= \gamma_{39} s_{39} c_{39}
\end{align*}$$

(30)

Then, the angular velocity of segment 9 with respect to the pelvis can be written in terms of the segment 1 basis vectors using the rotation matrix between segment 1 and segment 3.
The angular velocity of segment 9, the right thigh, with respect to the fixed reference frame, A, is then written as follows:

\[
\vec{\omega}^{39} = (u_9 R_{11}^{13} + u_9 R_{12}^{13} + u_9 R_{13}^{13}) \vec{e}_1 + (u_9 R_{21}^{13} + u_9 R_{22}^{13} + u_9 R_{23}^{13}) \vec{e}_2 + (u_9 R_{31}^{13} + u_9 R_{32}^{13} + u_9 R_{33}^{13}) \vec{e}_3
\]  \hspace{1cm} (31)

\[
\vec{\omega}^{39} = u'_{91} \vec{e}_1 + u'_{92} \vec{e}_2 + u'_{93} \vec{e}_3
\]  \hspace{1cm} (32)

and the angular velocity of the left thigh, segment 10, with respect to the pelvis, segment 3, is written in segment 3 coordinates as follows:

\[
\vec{\omega}^{310} = (u_{11} + u_{31} + u'_{91}) \vec{e}_1 + (u_{12} + u_{32} + u'_{92}) \vec{e}_2 + (u_{13} + u_{33} + u'_{93}) \vec{e}_3
\]  \hspace{1cm} (33)

where

\[
u_{10,1} = \beta_{3,10} c\alpha_{3,10} - \gamma_{3,10} c\beta_{3,10} s\alpha_{3,10}
\]

\[
u_{10,2} = \beta_{3,10} s\alpha_{3,10} + \gamma_{3,10} c\beta_{3,10} c\alpha_{3,10}
\]

\[
u_{10,3} = \alpha_{3,10} + \gamma_{3,10} s\beta_{3,10}
\]  \hspace{1cm} (35)
Rotating from segment 3 to segment 1 coordinates, the angular velocity of segment 10 with respect to the pelvis is written in reference segment coordinates as:

\[
\bar{\omega}^{3,10} = (u_{10,1}R_{11}^{13} + u_{10,2}R_{12}^{13} + u_{10,3}R_{13}^{13}) \bar{E}_1 \\
+ (u_{10,1}R_{21}^{13} + u_{10,2}R_{22}^{13} + u_{10,3}R_{23}^{13}) \bar{E}_2 \\
+ (u_{10,1}R_{31}^{13} + u_{10,2}R_{32}^{13} + u_{10,3}R_{33}^{13}) \bar{E}_3
\]  

(36)

Then the angular velocity of the left thigh, segment 10, with respect to the fixed reference frame, \(A\), is written in mid-trunk coordinates as:

\[
\bar{\omega}^{A,10} = \bar{\omega}^{A1} + \bar{\omega}^{13} + \bar{\omega}^{3,10} \\
= (u_{11} + u_{21} + u_{10,1}) \bar{E}_1 + (u_{12} + u_{22} + u_{10,2}) \bar{E}_2 + (u_{13} + u_{23} + u_{10,3}) \bar{E}_3
\]  

(38)

Hinge joints were assumed at the elbows and at the knees. For these joints, the only rotation allowed was through an angle \(\alpha_i\) about the mediolateral axis (flexion/extension). The angular velocity of the right forearm, segment 5, is with respect to the right upper arm, segment 7, is then written in right upper arm coordinates as:

\[
\bar{\omega}^{75} = \bar{\omega}^{7573}
\]  

(39)
Using the rotation matrix between segments 2 and 7, the unit vector $\overline{n}_3$ can be rotated into upper trunk coordinates:

$$\overline{n}_3 = R_{13}^{27} \overline{n}_{21} + R_{23}^{27} \overline{n}_{22} + R_{33}^{27} \overline{n}_{23} \tag{40}$$

Then, rotating from segment 2 to segment 1 coordinates, the unit vector $\overline{n}_3$ is written as follows:

$$\overline{n}_3 = R_{13}^{27} (R_{13}^{12} \overline{D}_1 + R_{21}^{12} \overline{D}_2 + R_{33}^{12} \overline{D}_3)
+ R_{23}^{27} (R_{13}^{12} \overline{D}_1 + R_{22}^{12} \overline{D}_2 + R_{33}^{12} \overline{D}_3)
+ R_{33}^{27} (R_{13}^{12} \overline{D}_1 + R_{23}^{12} \overline{D}_2 + R_{33}^{12} \overline{D}_3) \tag{41}$$

$$= (R_{13}^{27} R_{11}^{12} + R_{23}^{27} R_{12}^{12} + R_{33}^{27} R_{13}^{12}) \overline{D}_1
+ (R_{13}^{27} R_{21}^{12} + R_{23}^{27} R_{22}^{12} + R_{33}^{27} R_{23}^{12}) \overline{D}_2
+ (R_{13}^{27} R_{31}^{12} + R_{23}^{27} R_{32}^{12} + R_{33}^{27} R_{33}^{12}) \overline{D}_3$$

So the angular velocity of the right forearm, segment 5, with respect to the right upper arm, segment 7, is written in reference segment coordinates as:

$$\overline{\omega}^{75} = \overline{\alpha}_7^{27} (R_{13}^{27} R_{11}^{12} + R_{23}^{27} R_{12}^{12} + R_{33}^{27} R_{13}^{12}) \overline{D}_1
+ \overline{\alpha}_7^{27} (R_{13}^{27} R_{21}^{12} + R_{23}^{27} R_{22}^{12} + R_{33}^{27} R_{23}^{12}) \overline{D}_2
+ \overline{\alpha}_7^{27} (R_{13}^{27} R_{31}^{12} + R_{23}^{27} R_{32}^{12} + R_{33}^{27} R_{33}^{12}) \overline{D}_3 \tag{42}$$
\[ \vec{\omega}^{57} = u_{51} \vec{b}_1 + u_{52} \vec{b}_2 + u_{53} \vec{b}_3 \]  

(43)

The angular velocity of segment 5 with respect to the fixed reference frame, A, is then:

\[ \vec{\omega}^{A5} = \vec{\omega}^{A1} + \vec{\omega}^{12} + \vec{\omega}^{27} + \vec{\omega}^{75} \]

\[ \vec{\omega}^{A5} = (u_{11} + u_{21} + u_{71} + u_{51}) \vec{b}_1 + (u_{12} + u_{22} + u_{72} + u_{52}) \vec{b}_2 + (u_{13} + u_{23} + u_{73} + u_{53}) \vec{b}_3 \]  

(44)

Similarly, for the left forearm, segment 6, the angular velocity with respect to the left upper arm, segment 8 is written in reference segment coordinates as:

\[ \vec{\omega}^{66} = \dot{\alpha}_{66} (R_{13} \dot{R}_{11} + R_{23} \dot{R}_{12} + R_{33} \dot{R}_{13}) \vec{b}_1 \]

\[ + \dot{\alpha}_{66} (R_{13} \dot{R}_{21} + R_{23} \dot{R}_{22} + R_{33} \dot{R}_{23}) \vec{b}_2 \]

\[ + \dot{\alpha}_{66} (R_{13} \dot{R}_{31} + R_{23} \dot{R}_{32} + R_{33} \dot{R}_{33}) \vec{b}_3 \]  

(45)

\[ = u_{61} \vec{b}_1 + u_{62} \vec{b}_2 + u_{63} \vec{b}_3 \]

and the angular velocity of segment 6 with respect to the fixed reference frame, A is:

\[ \vec{\omega}^{A6} = \vec{\omega}^{A1} + \vec{\omega}^{12} + \vec{\omega}^{28} + \vec{\omega}^{66} \]

\[ \vec{\omega}^{A6} = (u_{11} + u_{21} + u_{81} + u_{61}) \vec{b}_1 + (u_{12} + u_{22} + u_{82} + u_{62}) \vec{b}_2 + (u_{13} + u_{23} + u_{83} + u_{63}) \vec{b}_3 \]  

(46)
For the right shank, segment 11, and the left shank, segment 12, the angular velocity with respect to the right and left thighs (segments 9 and 10) are written as:

\[
\overline{\omega}^{9,11} = \dot{\alpha}_{9,11} \left( R_{39}^{13} R_{11}^{13} + R_{39}^{13} R_{12}^{13} + R_{39}^{13} R_{13}^{13} \right) \overline{B}_1 + \dot{\alpha}_{9,11} \left( R_{39}^{13} R_{31}^{13} + R_{39}^{13} R_{32}^{13} + R_{39}^{13} R_{33}^{13} \right) \overline{B}_2 + \dot{\alpha}_{9,11} \left( R_{39}^{13} R_{31}^{13} + R_{39}^{13} R_{32}^{13} + R_{39}^{13} R_{33}^{13} \right) \overline{B}_3 \]

\[
= u_{11,1} \overline{B}_1 + u_{11,2} \overline{B}_2 + u_{11,3} \overline{B}_3
\]

\[
\overline{\omega}^{10,12} = \dot{\alpha}_{10,12} \left( R_{310}^{11} R_{11}^{11} + R_{310}^{11} R_{12}^{11} + R_{310}^{11} R_{13}^{11} \right) \overline{B}_1 + \dot{\alpha}_{10,12} \left( R_{310}^{11} R_{31}^{11} + R_{310}^{11} R_{32}^{11} + R_{310}^{11} R_{33}^{11} \right) \overline{B}_2 + \dot{\alpha}_{10,12} \left( R_{310}^{11} R_{31}^{11} + R_{310}^{11} R_{32}^{11} + R_{310}^{11} R_{33}^{11} \right) \overline{B}_3 \]

\[
= u_{12,1} \overline{B}_1 + u_{12,2} \overline{B}_2 + u_{12,3} \overline{B}_3
\]

and the angular velocities of segments 11 and 12 with respect to the fixed reference frame, \(A\) are written in mid-trunk coordinates as:

\[
\overline{\omega}^{A,11} = \overline{\omega}^{A} + \overline{\omega}^{13} + \overline{\omega}^{9,11}
\]

\[
\overline{\omega}^{A,11} = (u_{11} + u_{31} + u_{91} + u_{11,1}) \overline{B}_1 + (u_{12} + u_{32} + u_{92} + u_{11,2}) \overline{B}_2 + (u_{13} + u_{33} + u_{93} + u_{11,3}) \overline{B}_3
\]
Angular Acceleration of Segments

To find the angular accelerations for each segment, one must take the derivatives of the angular velocities with respect to the fixed reference frame.

For any segment \(i\), the angular acceleration \(\alpha^i\) is:

\[
\overrightarrow{\alpha}^i = \overrightarrow{\omega}^i + \omega_{11} \frac{d}{dt} \overrightarrow{b}_1 + \omega_{12} \frac{d}{dt} \overrightarrow{b}_2 + \omega_{13} \frac{d}{dt} \overrightarrow{b}_3 + \omega_{21} \frac{d}{dt} \overrightarrow{b}_1 + \omega_{22} \frac{d}{dt} \overrightarrow{b}_2 + \omega_{23} \frac{d}{dt} \overrightarrow{b}_3
\]

(51)

where \(\omega_j\) is the \(j\)th component of the \(i\)th segment's angular velocity in reference segment coordinates.

For any vector \(\mathbf{v}\) fixed in body \(i\), the derivative in reference frame A can be found from the relationship:

\[
\frac{d}{dt} \overrightarrow{v} = \overrightarrow{\omega}^i \times \overrightarrow{v}
\]

(52)

Then the derivatives of the basis vectors of segment 1 are derived below:
The angular acceleration of segment 1 with respect to the fixed reference frame, A, is:

\[
\ddot{\alpha}^A = \ddot{u}_{11} \dot{\bar{B}}_1 + \ddot{u}_{12} \dot{\bar{B}}_2 + \ddot{u}_{13} \dot{\bar{B}}_3 \\
+ u_{11} \left( u_{12} \bar{B}_2 - u_{13} \bar{B}_3 \right) \\
+ u_{12} \left( u_{11} \bar{B}_1 - u_{13} \bar{B}_3 \right) \\
+ u_{13} \left( u_{12} \bar{B}_1 - u_{11} \bar{B}_2 \right) \\
= \dot{u}_{11} \bar{B}_1 + \dot{u}_{12} \bar{B}_2 + \dot{u}_{13} \bar{B}_3
\]
\[ \overline{\alpha}^{A2} = \omega_{21} \overline{b}_1 + \omega_{22} \overline{b}_2 + \omega_{23} \overline{b}_3 + \omega_{21} (\overline{\omega}^{A2} \times \overline{\omega}^{A2}) + \omega_{22} (\overline{\omega}^{A2} \times \overline{\omega}^{A2}) + \omega_{23} (\overline{\omega}^{A2} \times \overline{\omega}^{A2}) \]

\[ \overline{\alpha}^{A2} = (\dot{u}_{11} + \dot{u}_{21} - u_{11} u_{13} + u_{21} u_{13}) \overline{b}_1 + (\dot{u}_{12} + \dot{u}_{22} + u_{21} u_{13} - u_{22} u_{11}) \overline{b}_2 + (\dot{u}_{13} + \dot{u}_{23} + u_{22} u_{11} - u_{23} u_{12}) \overline{b}_3 \]  

(55)

where

\[ \dot{u}_{21} = \beta_{12} c_{12} - \beta_{12} \dot{a}_{12} s_{12} - \gamma_{12} c_{12} s_{12} + \gamma_{12} \beta_{12} s_{12} c_{12} s_{12} - \gamma_{12} \dot{a}_{12} c_{12} c_{12} s_{12} \]  

(56)

\[ \dot{u}_{22} = \beta_{12} s_{12} + \beta_{12} \dot{a}_{12} c_{12} + \gamma_{12} c_{12} c_{12} c_{12} s_{12} - \gamma_{12} \beta_{12} s_{12} s_{12} c_{12} - \gamma_{12} \dot{a}_{12} c_{12} s_{12} \]  

\[ \dot{u}_{23} = \dot{a}_{12} + \gamma_{12} s_{12} c_{12} c_{12} \]

and the angular acceleration of the pelvis, segment 3 with respect to the fixed reference frame is:

\[ \overline{\alpha}^{A3} = (\dot{u}_{11} + \dot{u}_{21} - u_{22} u_{11} + u_{23} u_{12}) \overline{b}_1 + (\dot{u}_{12} + \dot{u}_{22} + u_{21} u_{13} - u_{22} u_{11}) \overline{b}_2 + (\dot{u}_{13} + \dot{u}_{23} + u_{22} u_{11} - u_{23} u_{12}) \overline{b}_3 \]  

(57)

where

\[ \dot{u}_{31} = \beta_{13} c_{13} - \beta_{13} \dot{a}_{13} s_{13} - \gamma_{13} c_{13} s_{13} + \gamma_{13} \beta_{13} c_{13} s_{13} - \gamma_{13} \dot{a}_{13} c_{13} c_{13} c_{13} \]  

(58)

\[ \dot{u}_{32} = \beta_{13} s_{13} + \beta_{13} \dot{a}_{13} c_{13} + \gamma_{13} c_{13} c_{13} c_{13} c_{13} - \gamma_{13} \beta_{13} s_{13} s_{13} c_{13} - \gamma_{13} \dot{a}_{13} c_{13} s_{13} \]  

\[ \dot{u}_{33} = \dot{a}_{13} + \gamma_{13} s_{13} c_{13} c_{13} \]
The angular acceleration of the head, segment 4 with respect to the fixed reference frame is:

\[ \ddot{\alpha}_{44} = (\dot{u}_{11} + \dot{u}_{21} + \dot{u}_{22} + \dot{u}_{42} - (u_{22} + u_{42}) \cdot u_{43} + (u_{23} + u_{43}) \cdot u_{12}) \cdot \ddot{E}_1 + (\dot{u}_{12} + \dot{u}_{22} + \dot{u}_{21} + (u_{21} + u_{41}) \cdot u_{13} - (u_{23} + u_{43}) \cdot u_{11}) \cdot \ddot{E}_2 + (\dot{u}_{13} + \dot{u}_{23} + \dot{u}_{43} + (u_{22} + u_{42}) \cdot u_{11} - (u_{21} + u_{41}) \cdot u_{12}) \cdot \ddot{E}_3 \]  

(59)

where

\[ \dot{u}_{41} = u_{41} R_{11}^{12} + u_{41} R_{11}^{12} + u_{42} R_{12}^{12} + u_{42} R_{12}^{12} + u_{43} R_{13}^{12} + u_{43} R_{13}^{12} \]
\[ \dot{u}_{42} = u_{41} R_{21}^{12} + u_{41} R_{21}^{12} + u_{42} R_{22}^{12} + u_{42} R_{22}^{12} + u_{43} R_{23}^{12} + u_{43} R_{23}^{12} \]
\[ \dot{u}_{43} = u_{41} R_{31}^{12} + u_{41} R_{31}^{12} + u_{42} R_{32}^{12} + u_{42} R_{32}^{12} + u_{43} R_{33}^{12} + u_{43} R_{33}^{12} \]  

(60)

and

\[ \dot{u}_{41} = \beta_{24} c_{24} - \beta_{24} s_{24} + \gamma_{24} c_{24} s_{24} - \gamma_{24} c_{24} s_{24} + \gamma_{24} s_{24} - \gamma_{24} s_{24} c_{24} \]
\[ \dot{u}_{42} = \beta_{24} s_{24} + \beta_{24} c_{24} + \gamma_{24} c_{24} c_{24} + \gamma_{24} c_{24} c_{24} s_{24} - \gamma_{24} c_{24} c_{24} - \gamma_{24} c_{24} s_{24} c_{24} + \gamma_{24} c_{24} s_{24} c_{24} \]
\[ \dot{u}_{43} = \hat{a}_{24} + \hat{y}_{24} c_{24} + \hat{y}_{24} c_{24} c_{24} \]  

(61)

The derivatives of the elements of the rotation matrix for any two segments i and j are given in equations 64–66.
The angular accelerations of the right and left upper arms, segments 7 and 8, can be written as follows:
\[
\ddot{\alpha}^{A7} = (\dot{u}_{11} + \dot{u}_{21} + \dot{u}_{12} - (u_{22} + u_{12}) \dot{u}_{13} + (u_{23} + u_{13}) \dot{u}_{12}) \ddot{b}_1 \\
+ (\dot{u}_{12} + \dot{u}_{22} + \dot{u}_{21} - (u_{22} + u_{12}) \dot{u}_{11} + (u_{23} + u_{13}) \dot{u}_{11}) \ddot{b}_2 \\
+ (\dot{u}_{13} + \dot{u}_{23} + \dot{u}_{12} - (u_{22} + u_{12}) \dot{u}_{11} + (u_{23} + u_{13}) \dot{u}_{11}) \ddot{b}_3 
\]

(65)

\[
\ddot{\alpha}^{A8} = (\dot{u}_{11} + \dot{u}_{21} + \dot{u}_{81} - (u_{22} + u_{82}) \dot{u}_{13} + (u_{23} + u_{83}) \dot{u}_{12}) \ddot{b}_1 \\
+ (\dot{u}_{12} + \dot{u}_{22} + \dot{u}_{62} + (u_{22} + u_{62}) \dot{u}_{13} - (u_{23} + u_{63}) \dot{u}_{11}) \ddot{b}_2 \\
+ (\dot{u}_{13} + \dot{u}_{23} + \dot{u}_{63} + (u_{22} + u_{62}) \dot{u}_{11} - (u_{23} + u_{63}) \dot{u}_{11}) \ddot{b}_3 
\]

(66)

where \( u' \) derivatives are:

\[
\dot{u}_k = u_k \dot{R}_k^{12} + u_k \dot{R}_k^{12} + u_k \dot{R}_k^{12} + u_k \dot{R}_k^{12} + u_k \dot{R}_k^{12} 
\]

(67)

\( (k=7,8) \)

and \( u \) derivatives are:

\[
\dot{u}_k = \beta_{2k} \alpha_{2k} - \beta_{2k} \dot{a}_{2k} s_{a_{2k}} - \gamma_{2k} c_{\beta_{2k} s_{a_{2k}}} + \gamma_{2k} \beta_{2k} s_{\beta_{2k} s_{a_{2k}}} - \gamma_{2k} a_{2k} c_{\beta_{2k} c_{a_{2k}}} 
\]

(68)

\( (k=7,8) \)

The angular accelerations for the thighs (segments 9 and 10) which are connected by ball and socket joints to segment 3 (the pelvis) are similarly derived:
\[
\bar{\alpha}^{Ag} = \left( \dot{u}_{11} + \dot{u}_{31} + \dot{u}^{'}_{91} - (u_{32} + u^{'}_{92}) u_{13} + (u_{33} + u^{'}_{93}) u_{12} \right) \bar{b}_1
\]
\[
+ \left( \dot{u}_{12} + \dot{u}_{32} + \dot{u}^{'}_{92} + (u_{31} + u^{'}_{91}) u_{13} - (u_{33} + u^{'}_{93}) u_{11} \right) \bar{b}_2
\]
\[
+ \left( \dot{u}_{13} + \dot{u}_{33} + \dot{u}^{'}_{93} + (u_{32} + u^{'}_{92}) u_{11} - (u_{31} + u^{'}_{91}) u_{12} \right) \bar{b}_3
\]

\[
\bar{\alpha}^{Ag,10} = \left( \dot{u}_{11} + \dot{u}_{21} + \dot{u}^{'}_{10,1} - (u_{32} + u^{'}_{10,2}) u_{13} + (u_{33} + u^{'}_{10,3}) u_{12} \right) \bar{b}_1
\]
\[
+ \left( \dot{u}_{12} + \dot{u}_{22} + \dot{u}^{'}_{10,2} + (u_{31} + u^{'}_{10,1}) u_{13} - (u_{33} + u^{'}_{10,3}) u_{11} \right) \bar{b}_2
\]
\[
+ \left( \dot{u}_{13} + \dot{u}_{23} + \dot{u}^{'}_{10,3} + (u_{32} + u^{'}_{10,2}) u_{11} - (u_{31} + u^{'}_{10,1}) u_{12} \right) \bar{b}_3
\]

where \( u' \) and \( u \) derivatives are calculated as follows:

\[
\dot{u}^{'}_{91} = \ddot{u}_{91} R_{11}^{13} + \dot{u}_{91} \dot{R}_{11}^{13} + \dot{u}_{92} R_{12}^{13} + \ddot{u}_{92} \dot{R}_{12}^{13} + \ddot{u}_{93} R_{13}^{13} + \dot{u}_{93} \dot{R}_{13}^{13}
\]
\[
\dot{u}^{'}_{92} = \ddot{u}_{91} R_{21}^{13} + \dot{u}_{91} \dot{R}_{21}^{13} + \dot{u}_{92} R_{22}^{13} + \ddot{u}_{92} \dot{R}_{22}^{13} + \ddot{u}_{93} R_{23}^{13} + \dot{u}_{93} \dot{R}_{23}^{13}
\]
\[
\dot{u}^{'}_{93} = \ddot{u}_{91} R_{31}^{13} + \dot{u}_{91} \dot{R}_{31}^{13} + \dot{u}_{92} R_{32}^{13} + \ddot{u}_{92} \dot{R}_{32}^{13} + \ddot{u}_{93} R_{33}^{13} + \dot{u}_{93} \dot{R}_{33}^{13}
\]

\[
\dot{u}^{'}_{10,1} = \ddot{u}_{10,1} R_{11}^{13} + \dot{u}_{10,1} \dot{R}_{11}^{13} + \dot{u}_{10,2} R_{12}^{13} + \ddot{u}_{10,2} \dot{R}_{12}^{13} + \ddot{u}_{10,3} R_{13}^{13} + \dot{u}_{10,3} \dot{R}_{13}^{13}
\]
\[
\dot{u}^{'}_{10,2} = \ddot{u}_{10,1} R_{21}^{13} + \dot{u}_{10,1} \dot{R}_{21}^{13} + \dot{u}_{10,2} R_{22}^{13} + \ddot{u}_{10,2} \dot{R}_{22}^{13} + \ddot{u}_{10,3} R_{23}^{13} + \dot{u}_{10,3} \dot{R}_{23}^{13}
\]
\[
\dot{u}^{'}_{10,3} = \ddot{u}_{10,1} R_{31}^{13} + \dot{u}_{10,1} \dot{R}_{31}^{13} + \dot{u}_{10,2} R_{32}^{13} + \ddot{u}_{10,2} \dot{R}_{32}^{13} + \ddot{u}_{10,3} R_{33}^{13} + \dot{u}_{10,3} \dot{R}_{33}^{13}
\]

\[
\dot{u}_{91} = \beta_{39} \sigma_{39} - \beta_{39} \hat{a}_{39} s_{a_{39}} - \gamma_{39} \beta_{39} s_{b_{39}} s_{a_{39}} + \gamma_{39} \hat{a}_{39} c_{b_{39}} c_{a_{39}} - \gamma_{39} \beta_{39} \sigma_{a_{39}} - \gamma_{39} \hat{a}_{39} s_{c_{39}} c_{a_{39}} c_{b_{39}} c_{a_{39}}
\]
\[
\dot{u}_{92} = \beta_{39} s_{a_{39}} + \beta_{39} \hat{a}_{39} c_{a_{39}} + \gamma_{39} \beta_{39} c_{b_{39}} c_{a_{39}} - \gamma_{39} \hat{a}_{39} s_{b_{39}} s_{c_{39}} c_{a_{39}} - \gamma_{39} \beta_{39} \sigma_{c_{39}} - \gamma_{39} \hat{a}_{39} s_{c_{39}} s_{b_{39}} c_{a_{39}}
\]
\[
\dot{u}_{93} = \hat{a}_{39} + \gamma_{39} s_{c_{39}} + \gamma_{39} \beta_{39} c_{b_{39}} c_{a_{39}}
\]
Then, the forearm segments' angular accelerations are as follows:

\[
\ddot{u}_{10,1} = \beta_{3,10} \alpha_{3,10} - \beta_{3,10} \dot{\alpha}_{3,10} s_{3,10} - \gamma_{3,10} c_{3,10} \beta_{3,10} s_{3,10} \\
+ \dot{\gamma}_{3,10} \beta_{3,10} s_{3,10} - \gamma_{3,10} \dot{\alpha}_{3,10} c_{3,10} \\
\ddot{u}_{10,2} = \beta_{3,10} \alpha_{3,10} + \beta_{3,10} \dot{\alpha}_{3,10} c_{3,10} + \gamma_{3,10} \beta_{3,10} \alpha_{3,10} \\
- \gamma_{3,10} \dot{\beta}_{3,10} s_{3,10} - \gamma_{3,10} \dot{\alpha}_{3,10} c_{3,10} s_{3,10} \\
\ddot{u}_{10,3} = \dot{\alpha}_{3,10} + \dot{\gamma}_{3,10} \beta_{3,10} s_{3,10} + \gamma_{3,10} \beta_{3,10} c_{3,10}
\]

(74)

where the derivatives of \(u_{ij}\) and \(u_{ij}^\prime\) are as follows:

\[
\ddot{u}_{10,1} = \dot{\alpha}_{75} \left( R_{13} R_{12} + R_{23} R_{12} + R_{33} R_{12} \right) \\
+ \ddot{\alpha}_{75} \left( R_{13} R_{12} + R_{23} R_{12} + R_{33} R_{12} \right)
\]

(77)
\[
\dot{\alpha}_{61} = \dot{\alpha}_{66} (R_{13} R_{12} + R_{23} R_{22} + R_{33} R_{32}) \\
+ \dot{\alpha}_{66} (R_{13} R_{11} + R_{23} R_{12} + R_{33} R_{12} + R_{33} R_{13}) \\
+ \dot{\alpha}_{66} (R_{13} R_{12} + R_{23} R_{22} + R_{33} R_{23}) \\
+ \dot{\alpha}_{66} (R_{13} R_{22} + R_{23} R_{22} + R_{33} R_{23}) \\
+ \dot{\alpha}_{66} (R_{13} R_{23} + R_{23} R_{23} + R_{33} R_{23}) \\
+ \dot{\alpha}_{66} (R_{13} R_{32} + R_{23} R_{32} + R_{33} R_{32} + R_{33} R_{33}) \\
\]

(78)

Similarly, the angular accelerations of the shank segments (11 and 12) are given as follows:

\[
\ddot{\alpha}^{A,11} = [\ddot{u}_{11} + \ddot{u}_{31} + \dot{u}_{11} + \dot{u}_{11} - (u_{32} + u_{32} + u_{11,1}) u_{13} \\
+ (u_{33} + u_{33} + u_{11,1}) u_{13}] \mathcal{B}_1 \\
+ [\ddot{u}_{12} + \ddot{u}_{32} + \dot{u}_{12} + \dot{u}_{12} + (u_{31} + u_{31} + u_{11,1}) u_{13} \\
+ (u_{33} + u_{33} + u_{11,1}) u_{13}] \mathcal{B}_2 \\
+ [\ddot{u}_{13} + \ddot{u}_{33} + \dot{u}_{13} + \dot{u}_{13} - (u_{31} + u_{31} + u_{11,1}) u_{13} \\
+ (u_{33} + u_{33} + u_{11,1}) u_{13}] \mathcal{B}_3 \\
\]

(79)

\[
\ddot{\alpha}^{A,12} = [\ddot{u}_{11} + \ddot{u}_{31} + \dot{u}_{12,1} + \dot{u}_{12,1} - (u_{32} + u_{32} + u_{12,2}) u_{13} \\
+ (u_{33} + u_{33} + u_{12,2}) u_{13}] \mathcal{B}_1 \\
+ [\ddot{u}_{12} + \ddot{u}_{32} + \dot{u}_{12} + \dot{u}_{12} + (u_{31} + u_{31} + u_{12,2}) u_{13} \\
+ (u_{33} + u_{33} + u_{12,2}) u_{13}] \mathcal{B}_2 \\
+ [\ddot{u}_{13} + \ddot{u}_{33} + \dot{u}_{13} + \dot{u}_{13} - (u_{31} + u_{31} + u_{12,2}) u_{13} \\
+ (u_{33} + u_{33} + u_{12,2}) u_{13}] \mathcal{B}_3 \\
\]

(80)
where the derivatives of \( u_{1ij} \) and \( u_{12j} \) are as follows:

\[
\dot{u}_{11,1} = \ddot{a}_{9,11} (R_{13}^9 R_{11}^{13} + R_{23}^9 R_{12}^{13} + R_{33}^9 R_{13}^{13}) + \ddot{a}_{9,11} (R_{13}^9 R_{11}^{13} + R_{23}^9 R_{12}^{13} + R_{33}^9 R_{13}^{13})
\]

\[
\dot{u}_{11,2} = \ddot{a}_{9,11} (R_{12}^9 R_{21}^{12} + R_{22}^9 R_{22}^{12} + R_{32}^9 R_{23}^{12}) + \ddot{a}_{9,11} (R_{12}^9 R_{21}^{12} + R_{22}^9 R_{22}^{12} + R_{32}^9 R_{23}^{12})
\]

\[
\dot{u}_{11,3} = \ddot{a}_{9,11} (R_{13}^9 R_{31}^{13} + R_{23}^9 R_{32}^{13} + R_{33}^9 R_{33}^{13}) + \ddot{a}_{9,11} (R_{13}^9 R_{31}^{13} + R_{23}^9 R_{32}^{13} + R_{33}^9 R_{33}^{13})
\]

\[
\dot{u}_{12,1} = \ddot{a}_{10,12} (R_{13}^3 R_{11}^{13} + R_{23}^3 R_{12}^{13} + R_{33}^3 R_{13}^{13}) + \ddot{a}_{10,12} (R_{13}^3 R_{11}^{13} + R_{23}^3 R_{12}^{13} + R_{33}^3 R_{13}^{13})
\]

\[
\dot{u}_{12,2} = \ddot{a}_{10,12} (R_{12}^3 R_{21}^{12} + R_{22}^3 R_{22}^{12} + R_{32}^3 R_{23}^{12}) + \ddot{a}_{10,12} (R_{12}^3 R_{21}^{12} + R_{22}^3 R_{22}^{12} + R_{32}^3 R_{23}^{12})
\]

\[
\dot{u}_{12,3} = \ddot{a}_{10,12} (R_{13}^3 R_{31}^{13} + R_{23}^3 R_{32}^{13} + R_{33}^3 R_{33}^{13}) + \ddot{a}_{10,12} (R_{13}^3 R_{31}^{13} + R_{23}^3 R_{32}^{13} + R_{33}^3 R_{33}^{13})
\]

**Location of Center of Gravity of Body**

The position vector for the center of gravity of the entire body \((\bar{F}_G)\) with respect to the origin of the reference segment, was found using the relationship:
\[ \vec{p}_g = \sum_{i=1}^{12} \left( \frac{m_i}{M_{\text{tot}}} \right) \vec{q}_i \]  

(83)

where \( \vec{q}_i \) is the vector from the origin of the reference segment to the center of gravity of segment \( i \). The equations for \( \vec{q}_i \)'s are given below. Refer to Figure A-2 for definition of vectors and dimensions.

\[ \vec{q}_1 = 0 \]  

(84)

\[ \vec{q}_2 = -(L_1/2) \vec{b}_2 - (L_2/2) \vec{n}_{22} \]  

(85)

\[ \vec{q}_3 = (L_1/2) \vec{b}_2 + (L_3/2) \vec{n}_{32} \]  

(86)

\[ \vec{q}_4 = -(L_1/2) \vec{b}_2 - L_2 \vec{n}_{22} - (L_4/2) \vec{n}_{42} \]  

(87)

\[ \vec{q}_5 = -(L_1/2) \vec{b}_2 - L_2 \vec{n}_{22} - R_2 \vec{n}_{23} - L_7 \vec{n}_{72} - \eta_5 L_5 \vec{n}_{52} \]  

(88)

\[ \vec{q}_6 = -(L_1/2) \vec{b}_2 - L_2 \vec{n}_{22} + R_2 \vec{n}_{23} - L_6 \vec{n}_{62} - \eta_6 L_6 \vec{n}_{62} \]  

(89)

\[ \vec{q}_7 = -(L_1/2) \vec{b}_2 - L_2 \vec{n}_{22} - R_2 \vec{n}_{23} - \eta_7 L_7 \vec{n}_{72} \]  

(90)

\[ \vec{q}_8 = -(L_1/2) \vec{b}_2 - L_2 \vec{n}_{22} + R_2 \vec{n}_{23} - \eta_8 L_8 \vec{n}_{82} \]  

(91)
Figure A-2:  
(a) Dimensions of segments  
(b) Right side of body, q-vectors
The terms of each of the above equations are then rotated into reference segment coordinates \((b_1, b_2, b_3)\) using the rotation matrices between segments, as described in the angular velocity section.

**Location of the Hands with Respect to Center of Gravity**

The location of the hands with respect to the entire body's center of gravity was found in reference segment coordinates, using the location of the center of gravity of the forearm segments, segment 5 and 6, with respect to the center of gravity of the reference segment \((\bar{q}_5, \bar{q}_6)\), and the location of the hands with respect to the center of gravity of the forearm \((- (1-\eta_j) L_j)\) minus the location of the center of gravity of the entire body with respect to the origin of the reference segment \((\bar{F}_0)\). See Figure A-3 for a diagram.
Figure A-3: Diagram of hand location vector

\[-(1-n)h_6 \cdot n_{26} = hl_{en} \cdot n_{26}\]
The hand locations can be rewritten in terms of the reference segment coordinates using the transformation matrices between segments.

**Inertia Dyadic of Segments in Reference Segment Coordinates**

The inertia dyadic is known in the segment coordinates from the principal inertias. These can be converted to the reference segment coordinates (in which the angular velocity and angular acceleration are derived) as follows:

\[
\overline{I}_j = \overline{I}_{xx} \overline{n}_{j1} \overline{n}_{j1} + \overline{I}_{yy} \overline{n}_{j2} \overline{n}_{j2} + \overline{I}_{zz} \overline{n}_{j3} \overline{n}_{j3}
\]  

(97)

If \( \overline{n}_{j}(\beta=1,3) \) is related to segment 1 coordinates system by:

\[
\begin{bmatrix}
\overline{b}_1 \\
\overline{b}_2 \\
\overline{b}_3
\end{bmatrix} = [R^1]^J \begin{bmatrix}
\overline{n}_{j1} \\
\overline{n}_{j2} \\
\overline{n}_{j3}
\end{bmatrix}
\]  

(98)

where the terms of \([R^1]^J\) are functions of the known joint angles, then

\[
\overline{I}_j = I_{xx}(R_{11} \overline{b}_1 + R_{21} \overline{b}_2 + R_{31} \overline{b}_3)(R_{11} \overline{b}_1 + R_{21} \overline{b}_2 + R_{31} \overline{b}_3)
+ I_{yy}(R_{12} \overline{b}_1 + R_{22} \overline{b}_2 + R_{32} \overline{b}_3)(R_{12} \overline{b}_1 + R_{22} \overline{b}_2 + R_{32} \overline{b}_3)
+ I_{zz}(R_{13} \overline{b}_1 + R_{23} \overline{b}_2 + R_{33} \overline{b}_3)(R_{13} \overline{b}_1 + R_{23} \overline{b}_2 + R_{33} \overline{b}_3)
\]  

(99)
\[ I_j = (I_{xx}R_{11}^2 + I_{yy}R_{12}^2 + I_{zz}R_{13}^2) \overline{B_1B_2} + (I_{xx}R_{11}R_{21} + I_{yy}R_{12}R_{22} + I_{zz}R_{13}R_{23}) \overline{B_1B_2} + (I_{xx}R_{11}R_{31} + I_{yy}R_{12}R_{32} + I_{zz}R_{13}R_{33}) \overline{B_1B_3} \]  

(100)

\[ + (I_{xx}R_{21}^2 + I_{yy}R_{22}^2 + I_{zz}R_{23}^2) \overline{B_2B_2} + (I_{xx}R_{11}R_{21} + I_{yy}R_{12}R_{22} + I_{zz}R_{13}R_{23}) \overline{B_2B_1} + (I_{xx}R_{21}R_{31} + I_{yy}R_{22}R_{32} + I_{zz}R_{23}R_{33}) \overline{B_2B_3} \]  

(101)

\[ + (I_{xx}R_{31}^2 + I_{yy}R_{32}^2 + I_{zz}R_{33}^2) \overline{B_3B_3} + (I_{xx}R_{11}R_{31} + I_{yy}R_{12}R_{32} + I_{zz}R_{13}R_{33}) \overline{B_3B_1} + (I_{xx}R_{31}R_{31} + I_{yy}R_{32}R_{32} + I_{zz}R_{33}R_{33}) \overline{B_3B_2} \]  

(102)

If the inertia dyadic is written as:

\[ I = I_{11}\overline{B_1B_1} + I_{12}\overline{B_1B_2} + I_{13}\overline{B_1B_3} + I_{21}\overline{B_2B_1} + I_{22}\overline{B_2B_2} + I_{23}\overline{B_2B_3} + I_{31}\overline{B_3B_1} + I_{32}\overline{B_3B_2} + I_{33}\overline{B_3B_3} \]  

(103)

\((I_{0}=I_{ii}, j\neq i)\) then the dot products of the inertia dyadic with any vector, \(\rho = \rho_1\overline{B_1} + \rho_2\overline{B_2} + \rho_3\overline{B_3}\), can be computed as:
The angular acceleration for each segment can be written in the following form:

\[ \ddot{\alpha}^j = (\alpha_{11}^j + A_1 - Bu_{13} + Cu_{12}) \vec{E}_1 + (\alpha_{21}^j + A_2 + Du_{13} - Cu_{11}) \vec{E}_2 + (\alpha_{31}^j + A_3 + Bu_{11} - Du_{12}) \vec{E}_3 \]  

(105)

where \( \alpha_i \) and \( u_i \) are functions of the unknown angles and \( A_i \), \( B_i \), \( C_i \), and \( D_i \) are constants that depend on the known joint angles.

Then,

\[ I_{\ddot{\alpha}} = (I_{11} (\dot{u}_{11} + A_1 - Bu_{13} + Cu_{12}) + I_{12} (\dot{u}_{12} + A_2 + Du_{13} - Cu_{11}) + I_{13} (\dot{u}_{13} + A_3 + Bu_{11} - Du_{12})) \vec{E}_1 + (I_{21} (\dot{u}_{11} + A_1 - Bu_{13} + Cu_{12}) + I_{22} (\dot{u}_{12} + A_2 + Du_{13} - Cu_{11}) + I_{23} (\dot{u}_{13} + A_3 + Bu_{11} - Du_{12})) \vec{E}_2 + (I_{31} (\dot{u}_{11} + A_1 - Bu_{13} + Cu_{12}) + I_{32} (\dot{u}_{12} + A_2 + Du_{13} - Cu_{11}) + I_{33} (\dot{u}_{13} + A_3 + Bu_{11} - Du_{12})) \vec{E}_3 \]  

(106)
The angular velocity of each segment with respect to the global coordinates can be written as a combination of an unknown $\vec{\omega}_1^j = u_{1j} \mathbf{b}_1 + u_{12} \mathbf{b}_2 + u_{13} \mathbf{b}_3$ and a known $\vec{\omega}_1^j$. Using this, the dot product of the inertia and the angular velocity is:

$$I \vec{\omega} = [I_{11} (u_{11} + \omega_1^1) + I_{12} (u_{12} + \omega_2^1) + I_{13} (u_{13} + \omega_3^1)] \vec{b}_1$$

$$+ [I_{21} (u_{11} + \omega_1^2) + I_{22} (u_{12} + \omega_2^2) + I_{23} (u_{13} + \omega_3^2)] \vec{b}_2$$

$$+ [I_{31} (u_{11} + \omega_1^3) + I_{32} (u_{12} + \omega_2^3) + I_{33} (u_{13} + \omega_3^3)] \vec{b}_3$$

so

$$I \vec{\omega} \times \vec{\omega} = [(I_{21} (u_{11} + \omega_1^1) + I_{22} (u_{12} + \omega_2^1) + I_{23} (u_{13} + \omega_3^1)) (u_{13} + \omega_3^1) - (I_{31} (u_{11} + \omega_1^2) + I_{32} (u_{12} + \omega_2^2) + I_{33} (u_{13} + \omega_3^2)) (u_{12} + \omega_2^2)] \vec{b}_1$$

$$+ [(I_{31} (u_{11} + \omega_1^3) + I_{32} (u_{12} + \omega_2^3) + I_{33} (u_{13} + \omega_3^3)) (u_{13} + \omega_3^3) - (I_{11} (u_{11} + \omega_1^1) + I_{12} (u_{12} + \omega_2^1) + I_{13} (u_{13} + \omega_3^1)) (u_{12} + \omega_2^2)] \vec{b}_2$$

$$+ [(I_{11} (u_{11} + \omega_1^1) + I_{12} (u_{12} + \omega_2^1) + I_{13} (u_{13} + \omega_3^1)) (u_{13} + \omega_3^3) - (I_{21} (u_{11} + \omega_1^2) + I_{22} (u_{12} + \omega_2^2) + I_{23} (u_{13} + \omega_3^2)) (u_{12} + \omega_2^3)] \vec{b}_3$$

The above equations must then be put in terms of the unknowns and constants for use in the differential equations. The $u_{1i}$'s will also be used as unknowns to simplify the equations.

The right hand term $\Sigma I \cdot a$ can be written in terms of the unknown variables as follows:
This term can be separated into three components \((b_1, b_2, b_3)\) and the constant terms can be renamed as follows:

Then the ΣΙαα equation can be rewritten as follows:
\[ \begin{align*}
\Sigma I_{11} &= Q_1 \\
\Sigma I_{12} &= Q_2 \\
\Sigma I_{13} &= Q_3 \\
\Sigma I_{21} &= Q_2 \\
\Sigma I_{22} &= Q_4 \\
\Sigma I_{23} &= Q_5 \\
\Sigma I_{31} &= Q_3 \\
\Sigma I_{32} &= Q_5 \\
\Sigma I_{33} &= Q_6
\end{align*} \]

\[ \Sigma I_{11} A_1 + \Sigma I_{12} A_2 + \Sigma I_{13} A_3 = R_1 \]

\[ \Sigma I_{21} A_1 + \Sigma I_{22} A_2 + \Sigma I_{23} A_3 = R_2 \]

\[ \Sigma I_{31} A_1 + \Sigma I_{32} A_2 + \Sigma I_{33} A_3 = R_3 \]

\[ \begin{align*}
\Sigma I_{12} D - \Sigma I_{11} B &= S_1 \\
\Sigma I_{22} D - \Sigma I_{21} B &= S_2 \\
\Sigma I_{32} D - \Sigma I_{31} B &= S_3 \\
\Sigma I_{11} C - \Sigma I_{13} D &= T_1 \\
\Sigma I_{21} C - \Sigma I_{23} D &= T_2 \\
\Sigma I_{31} C - \Sigma I_{33} D &= T_3 \\
\Sigma I_{13} B - \Sigma I_{12} C &= V_1 \\
\Sigma I_{23} B - \Sigma I_{22} C &= V_2 \\
\Sigma I_{33} B - \Sigma I_{32} C &= V_3
\end{align*} \]

\[ \sum_{i=1}^{12} \vec{a} = (Q_1 u_{11} + Q_2 u_{12} + Q_3 u_{13} + R_1 + S_1 u_{13} + T_1 u_{12} + V_1 u_{11}) \vec{e}_1 
+ (Q_2 u_{11} + Q_4 u_{12} + Q_5 u_{13} + R_2 + S_2 u_{13} + T_2 u_{12} + V_2 u_{11}) \vec{e}_2 
+ (Q_3 u_{11} + Q_6 u_{12} + Q_6 u_{13} + R_3 + S_3 u_{13} + T_3 u_{12} + V_3 u_{11}) \vec{e}_3 \]  \hspace{1cm} (112)

Similarly, the \( \Sigma \vec{\omega} \times \vec{\omega} \) term can be written as follows:

\[ \sum_{i=1}^{12} \vec{a} \times \vec{a} = (w_{11} + w_{12} u_{11} + w_{13} u_{12} + w_{14} u_{13} + w_{15} u_{12} + Q_2 u_{11} u_{13} - Q_3 u_{11} u_{12} + Q_5 (u_{12} - u_{13})) \vec{e}_1 
+ (w_{21} + w_{22} u_{11} + w_{23} u_{12} + w_{24} u_{13} + w_{25} u_{12} + Q_2 u_{11} u_{13} - Q_3 u_{11} u_{12} + Q_3 (u_{12} - u_{13})) \vec{e}_2 
+ (w_{31} + w_{32} u_{11} + w_{33} u_{12} + w_{34} u_{13} + w_{35} u_{12} + Q_3 u_{11} u_{13} - Q_5 u_{11} u_{13} + Q_6 (u_{12} - u_{13})) \vec{e}_3 \]  \hspace{1cm} (113)

where
\[ W_{11} = \sum_{i=1}^{12} \left[ (I_{i2} - I_{i3}) (\omega_{i1}^2 \omega_{i2}^1) + I_{i3}^2 \omega_{i1}^2 \omega_{i3}^1 + I_{i2} \left( (\omega_{i1}^1)^2 - (\omega_{i2}^1)^2 \right) - I_{i1}^2 \omega_{i1}^2 \omega_{i2}^1 \right] \]
\[ W_{21} = \sum_{i=1}^{12} \left[ (I_{i3} - I_{i1}) (\omega_{i1}^1 \omega_{i3}^1) + I_{i2} \omega_{i1}^1 \omega_{i2}^1 + I_{i3} \left( (\omega_{i1}^2)^2 - (\omega_{i3}^2)^2 \right) - I_{i2}^2 \omega_{i1}^2 \omega_{i3}^1 \right] \]
\[ W_{31} = \sum_{i=1}^{12} \left[ (I_{i1} - I_{i2}) (\omega_{i1}^1 \omega_{i2}^2) + I_{i1} \omega_{i2}^2 \omega_{i3}^1 + I_{i2} \left( (\omega_{i2}^1)^2 - (\omega_{i1}^1)^2 \right) - I_{i2}^2 \omega_{i1}^1 \omega_{i2}^1 \right] \]

\[ W_{12} = \sum_{i=1}^{12} \left[ I_{i1}^2 \omega_{i3}^1 - I_{i1} \omega_{i2}^1 \right] \]
\[ W_{13} = \sum_{i=1}^{12} \left[ (I_{i2} - I_{i3}) \omega_{i3}^2 - 2I_{i2} \omega_{i2}^1 - I_{i3} \omega_{i1}^1 \right] \]
\[ W_{14} = \sum_{i=1}^{12} \left[ (I_{i2} - I_{i3}) \omega_{i2}^1 + 2I_{i2} \omega_{i3}^1 + I_{i3} \omega_{i1}^1 \right] \]
\[ W_{15} = \sum_{i=1}^{12} \left[ (I_{i2} - I_{i3}) \right] \]

\[ W_{22} = \sum_{i=1}^{12} \left[ (I_{i3} - I_{i1}) \omega_{i3}^1 + 2I_{i3} \omega_{i1}^1 + I_{i3} \omega_{i3}^1 \right] \]
\[ W_{23} = \sum_{i=1}^{12} \left[ I_{i2} \omega_{i1}^1 - I_{i2} \omega_{i2}^1 \right] \]
\[ W_{24} = \sum_{i=1}^{12} \left[ (I_{i3} - I_{i1}) \omega_{i1}^1 + 2I_{i1} \omega_{i3}^1 - I_{i1} \omega_{i2}^1 \right] \]
\[ W_{25} = \sum_{i=1}^{12} \left[ (I_{i3} - I_{i1}) \right] \]
\[ W_{32} = \sum_{i=1}^{12} \left[ (I_{i1} - I_{i2}) \omega_{i2}^1 - 2I_{i1} \omega_{i1}^1 - I_{i2} \omega_{i3}^1 \right] \]
\[ W_{33} = \sum_{i=1}^{12} \left[ (I_{i1} - I_{i2}) \omega_{i1}^1 + 2I_{i1} \omega_{i2}^1 + I_{i2} \omega_{i3}^1 \right] \]
\[ W_{34} = \sum_{i=1}^{12} \left[ I_{i1} \omega_{i2}^1 - I_{i2} \omega_{i2}^1 \right] \]
\[ W_{35} = \sum_{i=1}^{12} \left[ (I_{i1} - I_{i2}) \right] \]
Linear Acceleration of Segments

The linear acceleration of the segments must be calculated in segment 1 coordinates. These were derived using the relationship given in Huston and Passerello (1971):

\[ \dddot{a}_{ij} = \dddot{a}_g + \dddot{q}_j + \dddot{P}_g \]  \hfill (115)

To do this computation, the second derivatives of the q's and of \( P_g \) must be derived. Since \( \dddot{P}_g = \Sigma (m_i/M_i) \dddot{q}_i \), then

\[ (\dddot{P}_g/\dddot{t}^2) = \Sigma (m_i/M_i) (\dddot{q}_j/\dddot{t}^2). \]

This means that if the second derivatives of the \( q_i \)'s are known, the second derivative of \( \dddot{P}_g \) can be determined. Given: \( q_j = q_{j1} \overrightarrow{b}_1 + q_{j2} \overrightarrow{b}_2 + q_{j3} \overrightarrow{b}_3 \), the derivatives of \( \overrightarrow{q}_j \) can be determined as follows:

\[ \frac{d^2}{dt^2} q_j = \frac{d}{dt} (q_{j1} \overrightarrow{b}_1 + q_{j2} \overrightarrow{b}_2 + q_{j3} \overrightarrow{b}_3) \]

\[ = q_{j1} \frac{d}{dt} \overrightarrow{b}_1 + q_{j2} \frac{d^2}{dt^2} \overrightarrow{b}_1 + \dddot{q}_{j1} \overrightarrow{b}_1 + q_{j2} \frac{d}{dt} \overrightarrow{b}_2 + q_{j3} \frac{d^2}{dt^2} \overrightarrow{b}_2 + \dddot{q}_{j2} \overrightarrow{b}_2 + q_{j3} \frac{d^2}{dt^2} \overrightarrow{b}_3 \]

\[ + q_{j3} \overrightarrow{b}_3 + 2q_{j3} \frac{d}{dt} \overrightarrow{b}_3 + q_{j3} \frac{d^2}{dt^2} \overrightarrow{b}_3 \]  \hfill (117)

The first and second derivatives of \( q \) vs time could be determined analytically using the values of the joint rotations and their derivatives. It is easier, however, to use the \( q \)'s calculated using the equations in the previous
section and use the known joint rotations to numerically take their derivatives, rather than doing extensive calculations at each time point of interest. Therefore, the first and second derivatives of the $q$'s were calculated using the same cubic splines routine used to take the derivatives of the known joint rotations.

The first derivatives of the segment 1 basis vectors were computed earlier as:

\[
\begin{align*}
\frac{d}{dt} \overline{b}_1 &= u_{13} \overline{b}_2 - u_{12} \overline{b}_3 \\
\frac{d}{dt} \overline{b}_2 &= u_{11} \overline{b}_3 - u_{13} \overline{b}_1 \\
\frac{d}{dt} \overline{b}_3 &= u_{12} \overline{b}_1 - u_{11} \overline{b}_2
\end{align*}
\]

The second derivatives were found using:

\[
\frac{d^2}{dt^2} \overline{b}_i = \frac{d}{dt} (\frac{d}{dt} \overline{b}_i) = \frac{d}{dt} (\overline{b}_i) = \omega^A \times (\frac{d}{dt} \overline{b}_i)
\]

Then, the second derivative of each vector is given below:

\[
\begin{align*}
\frac{d^2}{dt^2} \overline{b}_1 &= -(u_{13}^2 + u_{12}^2) \overline{b}_1 + (\dot{u}_{13} + u_{12}u_{11}) \overline{b}_2 + (u_{12}u_{11} - \dot{u}_{12}) \overline{b}_3 \\
\frac{d^2}{dt^2} \overline{b}_2 &= (u_{11}u_{12} - \dot{u}_{13}) \overline{b}_1 - (u_{12}^2 + u_{13}^2) \overline{b}_2 + (\dot{u}_{11} + u_{13}u_{12}) \overline{b}_3 \\
\frac{d^2}{dt^2} \overline{b}_3 &= (\dot{u}_{12} + u_{11}u_{13}) \overline{b}_1 + (u_{12}u_{13} - \dot{u}_{11}) \overline{b}_2 - (u_{12}^2 + u_{11}^2) \overline{b}_3
\end{align*}
\]
Using this result, equation (117) can be rewritten as:

\[
\frac{d^2}{dt^2} \mathbf{q}_j = (\ddot{q}_{j1} - 2\dot{q}_{j2} u_{13} + 2\dot{q}_{j3} u_{12} - q_{j1} (u_{13}^2 + u_{12}^2) + q_{j2} (u_{11} u_{12} - \dot{s}_{12}) + q_{j3} (\dot{u}_{12} + u_{11} u_{13})) \mathbf{b}_1
\]

\[+ (\ddot{q}_{j2} + 2\dot{q}_{j1} u_{13} - 2\dot{q}_{j2} u_{11} - q_{j2} (u_{11}^2 + u_{13}^2) + q_{j1} (u_{13} + u_{11} u_{12})) \mathbf{b}_2
\]

\[+ (\ddot{q}_{j3} - 2\dot{q}_{j1} u_{12} + 2\dot{q}_{j2} u_{11} - q_{j3} (u_{12}^2 + u_{11}^2) + q_{j1} (u_{13} u_{11} - \dot{u}_{12})) \mathbf{b}_3
\]

Then, the second derivative of the \( \mathbf{P}_G \) vector is:

\[
\frac{d^2}{dt^2} \mathbf{P}_G = \sum_{i=1}^{12} \frac{m_i}{M_{tot}} \left( \frac{d^2}{dt^2} \mathbf{q}_i \right)
\]

which, substituting from equation (120), can be written as:

\[
\frac{d^2}{dt^2} \mathbf{P}_G = \sum_{i=1}^{12} \frac{m_i}{M_{tot}} \left[ (\ddot{q}_{j1} - 2\dot{q}_{j2} u_{13} + 2\dot{q}_{j3} u_{12} - q_{j1} (u_{13}^2 + u_{12}^2)
\]

\[+ q_{j2} (u_{11} u_{12} - \dot{u}_{13}) + q_{j3} (\dot{u}_{12} + u_{11} u_{13})) \mathbf{b}_1
\]

\[+ (\ddot{q}_{j2} + 2\dot{q}_{j1} u_{13} - 2\dot{q}_{j2} u_{11} - q_{j2} (u_{11}^2 + u_{13}^2) + q_{j1} (u_{13} + u_{11} u_{12})) \mathbf{b}_2
\]

\[+ (\ddot{q}_{j3} - 2\dot{q}_{j1} u_{12} + 2\dot{q}_{j2} u_{11} - q_{j3} (u_{12}^2 + u_{11}^2) + q_{j1} (u_{13} u_{11} - \dot{u}_{12})) \mathbf{b}_3 \right]
\]
The acceleration of the center of gravity of the entire body in terms of the unknowns is:

$$\ddot{\bar{a}}_g = \ddot{x}_1 \ddot{a}_1 + \ddot{x}_2 \ddot{a}_2 + \ddot{x}_3 \ddot{a}_3$$  \hspace{1cm} (124)$$

Rotating into segment 1 coordinates, this can be written as:

$$\ddot{\bar{a}}_g = (\ddot{x}_1 (cacy-sasb\, sy) + \ddot{x}_2 (sacy+casb\, sy) - \ddot{x}_3 (sycb)) \ddot{\bar{b}}_1$$

$$+ (\ddot{x}_1 (sacb) + \ddot{x}_2 (cacb) + \ddot{x}_3 (sbc)) \ddot{\bar{b}}_2$$

$$+ (\ddot{x}_1 (casy+sasbcy) + \ddot{x}_2 (sasy-casbcy) + \ddot{x}_3 (cbcy)) \ddot{\bar{b}}_3$$  \hspace{1cm} (125)$$

The linear acceleration terms of a single segment’s center of gravity can then be written as:

$$\ddot{\bar{a}}_{gj} = \ddot{\bar{a}}_g + \frac{d^2}{dt^2} \ddot{\bar{a}}_j + \frac{d^2}{dt^2} \ddot{\bar{P}}_g$$  \hspace{1cm} (126)$$

substituting the values from equations (120) and (122)

$$\ddot{\bar{a}}_{gj} = (\ddot{x}_1 (cacy-sasb\, sy) + \ddot{x}_2 (sacy+casb\, sy) - \ddot{x}_3 (sycb) + q_{11} \dot{\bar{b}}_1$$

$$- q_{12} \dot{u}_{13} + q_{13} \dot{u}_{12} - q_1 \dot{(u_{13}^2+u_{12}^2)} + q_2 \dot{(u_{11} u_{12}-u_{13})} + q_3 \dot{(u_{12}+u_{11})} \ddot{\bar{b}}_1$$

$$+ (\ddot{x}_1 (sacb) + \ddot{x}_2 (cacb) + \ddot{x}_3 (sbc)) + q_{21} \dot{\bar{b}}_2$$

$$+ q_1 \dot{(u_{13}+u_{12})} - q_2 \dot{(u_{13}^2+u_{12}^2)} + q_3 \dot{(u_{12}+u_{11})} \ddot{\bar{b}}_2$$

$$+ (\ddot{x}_1 (casy+sasbcy) + \ddot{x}_2 (sasy-casbcy) + \ddot{x}_3 (cbcy) + q_{31} \dot{\bar{b}}_3$$

$$+ q_1 \dot{(u_{13} u_{12}-u_{13})} + q_2 \dot{(u_{11}+u_{12})} - q_3 \dot{(u_{12}^2+u_{11}^2)} \ddot{\bar{b}}_3$$  \hspace{1cm} (127)$$
where for any segment \( j \),

\[
Q_{k1}^{j'} = \bar{q}_{jk} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} \bar{q}_{ik}
\]

\[
Q_{12}^{j'} = 2\bar{q}_{j2} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} (2\bar{q}_{ij2})
\]

\[
Q_{13}^{j'} = 2\bar{q}_{j3} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} (2\bar{q}_{ij3})
\]

\[
Q_{22}^{j'} = 2\bar{q}_{j1} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} (2\bar{q}_{ij1})
\]

\[
Q_{23}^{j'} = 2\bar{q}_{j2} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} (2\bar{q}_{ij2})
\]

\[
Q_{33}^{j'} = 2\bar{q}_{j3} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} (2\bar{q}_{ij3})
\]

and

\[
Q_{k}^{j'} = q_{jk} + \sum_{i=1}^{12} \frac{m_i}{M_{tot}} q_{ik}
\]

The \( P_j \) terms were derived previously, using \( \overline{P}_j = \bar{q}_j - \overline{P}_G \).

Then, skipping a few intermediate steps, the three components of the cross product term \( \Sigma \overline{P}_j x \overline{m}_j \overline{a}_j \) along the basis vectors for segment 1 (the reference segment) are:

\[
(\sum_{j=1}^{12} \overline{P}_j x \overline{m}_j \overline{a}_j)_1 = \bar{x}_1 (\gamma \bar{a} \beta + \bar{a} \gamma \beta \gamma \bar{a} \gamma) \Delta_{2,1} + \bar{x}_1 \gamma \bar{a} \beta \Delta_{3,1}
+
\bar{x}_2 (\gamma \bar{a} \gamma - \gamma \bar{a} \beta \gamma) \Delta_{2,1} - \bar{x}_2 \gamma \bar{a} \beta \Delta_{3,1} + \Delta_{1,7} + u_{11} \Delta_{1,8}
- u_{12} \Delta_{2,6} - u_{13} \Delta_{3,6} - (u_{13} + u_{12}) \Delta_{3,2} + (-u_{12} + u_{13} \Delta_{1,1}) \Delta_{2,2}
+ (u_{11} + u_{12} u_{13}) \Delta_{2,3} + (u_{11} u_{12} u_{13}) \Delta_{3,3} - (u_{11}^2 + u_{12}^2) \Delta_{2,4} - (-u_{11} + u_{13} \Delta_{1,2}) \Delta_{3,4}
\]

(130)
\frac{12}{i=1} (\Sigma \bar{p}_j \times m \bar{a}_j)_2 = \dot{x}_1 (c\alpha \gamma - s\alpha \beta \delta \gamma) \Delta_{3,1} - \dot{x}_1 (c\alpha \gamma + s\alpha \beta \delta \gamma) \Delta_{1,1} + \dot{x}_2 (s\alpha \gamma + c\alpha \beta \delta \gamma) \Delta_{3,1} - \dot{x}_2 (s\alpha \gamma - c\alpha \beta \delta \gamma) \Delta_{1,1} + u_{12} \Delta_{2,8} - u_{13} \Delta_{3,5} + (-\dot{u}_{13} + u_{12} u_{11}) \Delta_{3,3} - (\dot{u}_{12} + u_{13} u_{11}) \Delta_{3,4} - (u_{11} + u_{12} u_{13}) \Delta_{1,3} - (u_{11}^2 + u_{12}^2) \Delta_{1,4} - (u_{13}^2 + u_{12}^2) \Delta_{3,2} - (-\dot{u}_{12} + u_{13} u_{11}) \Delta_{1,2} (131)

\frac{12}{i=1} (\Sigma \bar{p}_j \times m \bar{a}_j)_2 = \dot{x}_1 (-s\alpha \beta) \Delta_{1,1} - \dot{x}_1 (c\alpha \gamma - s\alpha \beta \delta \gamma) \Delta_{2,1} + \dot{x}_2 (s\alpha \gamma + c\alpha \beta \delta \gamma) \Delta_{2,1} + \dot{x}_2 (s\alpha \gamma - c\alpha \beta \delta \gamma) \Delta_{1,1} + u_{13} \Delta_{2,8} + u_{12} \Delta_{3,5} + (\dot{u}_{13} + u_{12} u_{11}) \Delta_{2,3} - (\dot{u}_{13} + u_{12} u_{11}) \Delta_{3,3} + (\dot{u}_{11} + u_{12} u_{13}) \Delta_{1,4} - (u_{11}^2 + u_{12}^2) \Delta_{1,3} + (u_{12}^2 + u_{13}^2) \Delta_{2,3} - (-\dot{u}_{11} + u_{12} u_{13}) \Delta_{2,3} (132)

where the constant terms are defined as follows:

\Delta_{1,1} = \frac{12}{i=1} \Sigma m^i P_1^i \Delta_{1,2} = \frac{12}{i=1} \Sigma m^i P_2^i \Delta_{1,3} = \frac{12}{i=1} \Sigma m^i P_3^i \Delta_{2,1} = \frac{12}{i=1} \Sigma m^i P_1^i q_1^i \Delta_{2,2} = \frac{12}{i=1} \Sigma m^i P_2^i q_1^i \Delta_{2,3} = \frac{12}{i=1} \Sigma m^i P_2^i q_2^i \Delta_{3,1} = \frac{12}{i=1} \Sigma m^i P_3^i \Delta_{3,2} = \frac{12}{i=1} \Sigma m^i P_3^i q_1^i \Delta_{3,3} = \frac{12}{i=1} \Sigma m^i P_3^i q_2^i (133)
\[\Delta_{1,4} = \sum_{i=1}^{12} m^i P_1^i Q_3^i \quad \Delta_{1,5} = \sum_{i=1}^{12} m^i P_1^i Q_{23}^i \quad \Delta_{1,6} = \sum_{i=1}^{12} m^i P_2^i Q_{32}^i \]
\[\Delta_{2,4} = \sum_{i=1}^{12} m^i P_2^i Q_3^i \quad \Delta_{2,5} = \sum_{i=1}^{12} m^i P_2^i Q_{13}^i \quad \Delta_{2,6} = \sum_{i=1}^{12} m^i P_3^i Q_{33}^i \]
\[\Delta_{3,4} = \sum_{i=1}^{12} m^i P_3^i Q_3^i \quad \Delta_{3,5} = \sum_{i=1}^{12} m^i P_3^i Q_{12}^i \quad \Delta_{3,6} = \sum_{i=1}^{12} m^i P_3^i Q_{22}^i \] (134)

\[\Delta_{1,7} = \sum_{i=1}^{12} m^i (P_2^i Q_{31}^i - P_3^i Q_{21}^i) \quad \Delta_{1,8} = \sum_{i=1}^{12} m^i (P_2^i Q_{32}^i + P_3^i Q_{23}^i) \]
\[\Delta_{2,7} = \sum_{i=1}^{12} m^i (P_3^i Q_{11}^i - P_3^i Q_{31}^i) \quad \Delta_{2,8} = \sum_{i=1}^{12} m^i (P_3^i Q_{13}^i + P_3^i Q_{33}^i) \]
\[\Delta_{3,7} = \sum_{i=1}^{12} m^i (P_1^i Q_{21}^i - P_2^i Q_{12}^i) \quad \Delta_{3,8} = \sum_{i=1}^{12} m^i (P_1^i Q_{22}^i + P_2^i Q_{12}^i) \] (135)
APPENDIX B

INFORMED WRITTEN CONSENT FORM FOR PARTICIPANTS IN A STUDY OF
RELEASE MOVES ON THE HORIZONTAL BAR

The purpose of this study is to examine the biomechanics of the Tkatchev release move on the horizontal bar. Volunteer subjects will be videotaped performing the Tkatchev. The bar will be instrumented with strain gages to measure the forces and torques on the bar while the gymnast performs. Data obtained from the videotape and strain gages will provide a database to create a computer model of the skill from which joint forces and torques can be estimated. The study will be performed at Iowa State University.

1. I have freely consented to take part in a scientific study being conducted by Martha Nichols-Ketchum, a doctoral candidate in Biomedical Engineering at Iowa State University.

2. The study has been explained to me. I understand the explanation that has been given and what participation in the study will involve.

3. I understand that I am free to discontinue my participation in the study at any time.

4. I understand that I will be performing the Tkatchev release move as it has been taught by my coach and that proper safety procedures and matting will be provided. A spotting belt will be available, if needed.

5. Emergency treatment of any injuries that may occur as a direct result of participation in this research will be treated at the Iowa State University Student Health Services, Student Services Building, and/or referred to Mary Greeley Hospital or another physician. Compensation for treatment of any injuries that may occur as a direct result of participation in this research may or may not be paid by Iowa State University depending on the Iowa Tort Claims Act. Claims for compensation will be handled by the Iowa State University Vice President for Business and Finance.

6. I understand that my name will not be associated with any publication and/or presentation of the data collected in this study.

7. I understand that videotapes and other measurement data will be collected and may be used for demonstrations, instruction and/or study.
8. I understand that my participation in this study does not guarantee any beneficial results for me.

9. I understand that, at my request, I can receive additional explanation of the study at any time.

__________________________________________________________________________
Gymnast                        Date                        Witness                        Date
### APPENDIX C

**ANTHROPOMETRIC MEASUREMENTS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>years</td>
<td></td>
</tr>
<tr>
<td>Head circumference</td>
<td>cm</td>
<td>*Maximum circumference of the head above the brow ridges and parallel to the Frankfort plate</td>
</tr>
<tr>
<td>(HEADC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chest circumference</td>
<td>cm</td>
<td>*Circumference of chest with tape passing over the nipples and perpendicular to the long axis of the trunk</td>
</tr>
<tr>
<td>(CC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Axillary circumference</td>
<td>cm</td>
<td>*Perpendicular to long axis of upper arm and passing just below lowest point of axilla</td>
</tr>
<tr>
<td>(AXILC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Elbow circumference</td>
<td>cm</td>
<td>*With elbow flexed to about 125 degrees, tape passes over olecranon process of the ulna and into the crease of the elbow</td>
</tr>
<tr>
<td>(ELBC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Forearm circumference</td>
<td>cm</td>
<td>*Maximum circumference of forearm, tape perpendicular to long axis of forearm</td>
</tr>
<tr>
<td>(FAC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wrist circumference</td>
<td>cm</td>
<td>*Minimum circumference of wrist proximal to the radial and ulnar styloid processes</td>
</tr>
<tr>
<td>(WRISUC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fist circumference</td>
<td>cm</td>
<td>*Subject makes tight fist with thumb across end of fist. Pass tape over thumb and knuckles</td>
</tr>
<tr>
<td>(FISTC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thigh circumference</td>
<td>cm</td>
<td>*Circumference just below the lowest point in the gluteal furrow with tape in horiz. plane</td>
</tr>
<tr>
<td>(THIHC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knee circumference</td>
<td>cm</td>
<td>*Knee circumference at mid-patella level with tape in horiz. plane</td>
</tr>
<tr>
<td>(GKNEC)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parameter</td>
<td>Unit(s)</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Calf circumference (CFC)</td>
<td>cm</td>
<td>Maximum circumference of the calf</td>
</tr>
<tr>
<td>Ankle circumference (ANKC)</td>
<td>cm</td>
<td>Minimum circumference of the ankle</td>
</tr>
<tr>
<td>Iliac fat (IF)</td>
<td>cm</td>
<td>The thickness of the panniculus adiposus just superior to the crest of the right ilium: approximated by skinfold measurement using Iliac Crest (fat) = 0.78 * skinfold iliac crest - 0.27 (±2.01)</td>
</tr>
<tr>
<td>Chest breadth (CHESB)</td>
<td>cm</td>
<td>Using a beam caliper, measure the horizontal breadth of the chest at the level of the nipples during normal breathing</td>
</tr>
<tr>
<td>Chest depth (CHESD)</td>
<td>cm</td>
<td>Measure using anthropometer held horizontally on the subject's right side at the level of the nipples during normal breathing</td>
</tr>
<tr>
<td>Waist breadth (WAISB)</td>
<td>cm</td>
<td>Horizontal breadth of the body at the level of the omphalion</td>
</tr>
<tr>
<td>Waist depth (WAISD)</td>
<td>cm</td>
<td>Measure the anterior to posterior distance of the abdomen at the level of the most lateral indentation points with abdomen relaxed</td>
</tr>
<tr>
<td>Hip breadth (HIPB)</td>
<td>cm</td>
<td>Horizontal distance across the greatest lateral protrusion of the hips</td>
</tr>
<tr>
<td>Buttock depth (BUTTD)</td>
<td>cm</td>
<td>The depth of the buttocks at the level of greatest rearward protrusion</td>
</tr>
<tr>
<td><strong>Parameter</strong></td>
<td><strong>Unit(s)</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Upper arm length (UPARL)</td>
<td>cm</td>
<td>&quot;Distance along the axis of the upper arm between the acromion and the radiale when subject stands with arm extended to the side.</td>
</tr>
<tr>
<td>Forearm length (FOARL)</td>
<td>cm</td>
<td>&quot;Distance along the axis of the lower arm between the radiale and styliion when subject stands with arm extended to the side.</td>
</tr>
<tr>
<td>Wrist breadth (WRISB)</td>
<td>cm</td>
<td>&quot;With spreading caliper, measure the maximum distance between the radial and ulnar styliion processes exerting sufficient pressure to compress the tissue overlying the radius and ulna.</td>
</tr>
<tr>
<td>Hand breadth (HANDB)</td>
<td>cm</td>
<td>&quot;Measure the maximum breadth of the hand across the distal ends of metacarpal II and V.</td>
</tr>
<tr>
<td>Stature (STAT)</td>
<td>cm</td>
<td>&quot;The vertical distance from the floor to the top of the head with the head in the Frankfort plane.</td>
</tr>
<tr>
<td>Chin-neck interval (CNI)</td>
<td>cm</td>
<td>&quot;Distance from top of head to anterior intersection of the chin and neck.</td>
</tr>
<tr>
<td>Shoulder height (SHLDH)</td>
<td>cm</td>
<td>&quot;Distance from floor to the acromion with subject standing erect.</td>
</tr>
<tr>
<td>Substernal height (SUBH)</td>
<td>cm</td>
<td>&quot;Distance from the floor to the substernale point at the lower edge of the sternum with subject standing erect.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Unit(s)</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Trochanteric height (TROCH)</td>
<td>cm</td>
<td>^Distance from the floor to the trochanterion on the right side with subject standing erect</td>
</tr>
<tr>
<td>Sitting height (SITH)</td>
<td>cm</td>
<td>^Vertical distance from sitting surface to top of head. Subject sitting with head oriented in Frankfort position and feet resting on a surface so that knees are bent at approximately right angles</td>
</tr>
<tr>
<td>Tibiale height (TIBH)</td>
<td>cm</td>
<td>^Vertical distance from floor to the right tibiale. Subject stands erect with legs slightly apart.</td>
</tr>
<tr>
<td>Sphyrion height (SPHYH)</td>
<td>cm</td>
<td>^Vertical distance from floor to sphyrion. Subject stands erect with legs slightly apart. Measured with measuring block.</td>
</tr>
<tr>
<td>Foot length (FOOTL)</td>
<td>cm</td>
<td>^Length of the foot along the long axis. Subject stands with right foot in foot box just touching the side and rear walls with the long axis of the foot parallel to the side wall.</td>
</tr>
<tr>
<td>Weight (W)</td>
<td>kg</td>
<td>^Weight of minimally clothed subject measured on standard medical type scale.</td>
</tr>
</tbody>
</table>

\(^a\) Clauser et al., 1969  
\(^b\) Miller, 1970
This program takes the strain gage output values and uses cubic splines to interpolate to points that correspond to the time values for each frame of the move.

This program uses "canned" subroutines from Kahaner, Moler and Nash.

```fortran
real time(105),s1(105),s2(105),s3(105),s4(105)
real s5(105),s6(105),s7(105),s8(105),t1(150),wk(300)
real d(150),ss1(150),ss2(150),ss3(150),ss4(150)
real ss5(150),ss6(150),ss7(150),ss8(150),dval(150)
character*50 fnl,fn2

read in values from data file
print *,"Data file with strain data:  ", fnl
read *, fnl
print *,"File for strain output:  ", fn2
read *, fn2
open (1,file=fnl)

read in number of points
read (1,*) n
do 100 i=1,n
   read (1,*) time(i),s1(i),s2(i),s3(i),s4(i),s5(i),s6(i)
   print *,i
100 continue

do 200 i=1,n
   read (1,*) s7(i),s8(i)
   print *,i
200 continue

close(1)
print *, "Enter number of frames: ",nval
read *,nval
print *, "Enter time for frame 1: ", t0
do 300 i=1,nval
   t1(i)=t0+(i-1)*0.02
300 continue

call programs to interpolate using cubic splines

call pchez(n,time,s1,d,.true.,wk,300,ierr)
call pchev(n,time,s1,d,nval,t1,ss1,dval,ierr)
call pchez(n,time,s2,d,.true.,wk,300,ierr)
```
CALL PCHEV(N, TIME, S2, D, NVAL, T1, SS2, DVAL, IERR)
CALL PCHEZ(N, TIME, S3, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S3, D, NVAL, T1, SS3, DVAL, IERR)
CALL PCHEZ(N, TIME, S4, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S4, D, NVAL, T1, SS4, DVAL, IERR)
CALL PCHEZ(N, TIME, S5, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S5, D, NVAL, T1, SS5, DVAL, IERR)
CALL PCHEZ(N, TIME, S6, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S6, D, NVAL, T1, SS6, DVAL, IERR)
CALL PCHEZ(N, TIME, S7, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S7, D, NVAL, T1, SS7, DVAL, IERR)
CALL PCHEZ(N, TIME, S8, D, .TRUE., WK, 300, IERR)
CALL PCHEV(N, TIME, S8, D, NVAL, T1, SS8, DVAL, IERR)

C
C Send output to data file
C
OPEN (2, FILE=FN2)
DO 1000 I=1, NVAL
   WRITE (2, *) I, T1(I), SS1(I), SS2(I), SS3(I), SS4(I), SS5(I)
   & SS6(I), SS7(I), SS8(I)
1000 CONTINUE
END
APPENDIX E
ANALYSIS PROGRAM AND SUBROUTINES

Program FORT2
C This program computes the forces and torques frame by frame for Tkatchevs performed and recorded by Ariel system.
C
C Variables:
m=mass;ixx,iyy,izz=inertias;eta=proportion of segment length (h) to c.g.; r,rr=major and minor radii of segments;a=reference frame coordinates; pos,vel,acc=kinematic data; rm=rotation matrix
n=number of frames

Markers (for pos,vel,acc)
l=right ankle; 2=right knee; 3=right thigh; 4=right back thigh;
5=right trochanter; 6=right ASIS; 7=left ASIS
8=left trochanter; 9=left back thigh; 10=left thigh
11=left knee; 12=left ankle;
13=right side wrist;
14=right elbow; 15=right back upper arm; 16=right shoulder; 17=left shoulder; 18=left back upper arm;
19=left elbow; 20=left side wrist; 21=forehead
22=right ear; 23=left ear; 24=sternum; 25=substernal
26=neck; 27=vertebrae; 28=right waist; 29=left waist

joints: 1=neck; 2=right shoulder; 3=left shoulder;
4=right elbow; 5=left elbow; 6=upper spine; 7=lower back; 8=right hip; 9=left hip; 10=right knee; 11=left knee; 12=right hand; 13=left hand

segments: 1=head; 2=upper trunk; 3=mid-trunk;
4=pelvis; 5=r.forearm; 6=l.forearm; 7=r.upper arm;
8=l.upper arm; 9=r.thigh; 10=l.thigh; 11=r.shank; 12=l.shank

real m(12),ixx(12),iyy(12),izz(12),eta(12),h(12)
real r(12),rr(12),rdp(11,3,3),rxf(3),rf(3,3)
real pcg(12,3),cg(3),thetadot(150,4),thetaddot(150,4)
real a(12,4,3),pos(35,3),vel(35,3),acc(35,3)
real rm(12,3,3),fl(3),fr(3)
real theta(11,3),omega(12,3),alpha(12,3),rcg(12,3)
real alphaxr(3),omegaxr(3),wxwyr(3),aa(3),hg(12,3)
real f(13,3),mom(13,3),sw(12,3),acg(12,3),sg(8)
real rrl,temp(3,3),merror(3),ferror(3)
real fd(11,3),momd(11,3),f7(3),m6(3)
integer n, mark(12), subnum
character*16 dir
character*1 sub
character*6 prefix
character*41 refdat
character*42 posfile, velfile, accfile, fnjta, errorfile
character*42 stfile, fnstore, stfile2, KINFILE, cgfile
character*42 kinfile2
PI=3.14159
print *, "Enter subject number (1 or 2):"
read *, subnum
if (subnum.eq.1) then
    sub='1'
else if (subnum.eq.2) then
    sub='2'
endif

C First, get masses, inertias, etc. from anthropometric
C data
C
    call inertia(m, ixx, iyy, izz, h, eta, r, rr, sub)
C
    Determine the reference frame for each segment and
C enter reference coordinates
C
    dir='c:\fortran\data\'
    print *, "Which file do you want to analyze?"
    print *, "1 Subject 1 #4"
    print *, "2 Subject 1 #8"
    print *, "3 Subject 2 #4"
    print *, "4 Subject 2 #9"
    print *, "5 Another file, not listed above"
    print *, ""
    read *, numfile
    if (numfile.eq.5) then
        print *, "Enter the name of the file to be
    & analyzed:"
        print *, "Enter the name in quotes: ‘fn’"
        read *, prefix
    else if (numfile.eq.1) then
        prefix='tkad#4'
    else if (numfile.eq.2) then
        prefix='tkad#8'
    else if (numfile.eq.3) then
        prefix='tkmk#4'
    else if (numfile.eq.4) then
        prefix='tkmk#9'
    else
        print *, "Not a valid selection, please re-enter"
&      choice "
      goto 11
endif
refdat='c:\fortran\data\subject'//sub//'.ref'

C Reference data is the location of markers in global
C coordinates when the gymnast is in the reference
C position (all segments aligned with the global
C coordinate system)
C
open (1,file=refdat)
do 100 i=l,12
   do 100 j=l,4
      read(l,*)(a(i,j,k),k=l,3)
100  continue
close(l)
C Define the reference point and relative position
C vectors for each segment. Mark(i)=marker for segment
C i that is used as the reference marker for relative
C acceleration of c.g.
C rcg(i,j) is the position vector (cm) from c.g. to
C mark(i) for segment i in the j-direction, where the
C coordinate system is segment is segment coordinates
C
C Segment 1: Head, reference marker=right ear
   mark(1)=2
C Segment 2: Upper trunk, reference marker=neck
   mark(2)=3
C Segment 3: Mid trunk, reference marker=vertebrae
   mark(3)=1
C Segment 4: Pelvis, reference marker=right waist
   mark(4)=1
C Segment 5: Right forearm, reference marker=elbow
   mark(5)=2
C Segment 6: Left forearm, reference marker=l.elbow
   mark(6)=2
C Segment 7: Right upper arm, reference marker=r.elbow
   mark(7)=1
C Segment 8: Left upper arm, reference marker=l.elbow
   mark(8)=1
C Segment 9: Right thigh, reference marker=r.knee
   mark(9)=1
C Segment 10: Left thigh, reference marker=l.knee
   mark(10)=1
C Segment 11: Right shank, reference marker, r. knee
   mark(11)=2
C Segment 12: Left shank, reference marker, l. knee
   mark(12)=2
   do i=l,12
do j=1,3
   rcg(i,j)=a(i,mark(i),j)
enddo
enddo

Add 100 to each reference position to avoid zeros in rotation matrix computations

C
C do i=1,12
   do j=1,4
      do k=1,3
         a(i,j,k)=a(i,j,k)+100.
      enddo
   enddo
enddo

C
C Redefine mark(i) as marker to be used as the reference in acceleration computations and cg computations
C
mark(1)=23
mark(2)=26
mark(3)=27
mark(4)=28
mark(5)=14
mark(6)=19
mark(7)=14
mark(8)=19
mark(9)=2
mark(10)=11
mark(11)=2
mark(12)=11

C
C Begin process of force and torque computation
C
posfile=dir//prefix//'.pes'
velfile=dir//prefix//'.vel'
accfile=dir//prefix//'.acc'
open (1,file=posfile)

C
C Read in number of frames, release frame, recatch frame, frame for end of bar contact, and file for strain gage data
C
read (1,*) n
read (1,*) nrel
read (1,*) nrec
read (1,*) nlast
read (1,*) stfile
stfile2=dir//stfile
fnjta=dir//prefix//'.jta'
fnstore=dir//prefix//'.ftd'
KINFILE=DIR//PREFIX//' .KIN'
kinfile2=dir//prefix//'.kn2'
cgfile=dir//prefix//' .eg'
errorfile=dir//prefix//' .err'
open (2,file=velfile,status='old')
open (3, file=accf ile, status='old')
open (4, file=stfile2, status='old')
open (5, file=fnjta)
open (6, file=fnstore)
OPEN (8,FILE=errorfile)
OPEN(11,FILE=KINFILE)
open (7, file=kinfile2)
open (12, file=cgfile)
open (9, file=dir//prefix//'.fds')

C First, get joint angles and their derivatives at the knee and elbow joints. These will later be used in the angular velocity and acceleration calculations for the shank and forearm segments

do kk=1,n
   DO j=1,29
      read (1,*),frame,time,(pos(j,k),k=1,3)
   enddo
   call rot(a,pos,rm)
call segrot(nn,rdp)
C From rotation matrix rdp, determine angles between segments
DO jt=1,11
   theta(jt,1)=atan(-rdp(jt,1,2)/rdp(jt,2,2))
   IF ((theta(jt,2).gt.PI/2).or.(theta(jt,2).lt.-PI/2))
&      then
      flag=1
      c1=rdp(jt,2)/cos(theta(jt,2))
      IF ((c1.lt.0.)and.(theta(jt,1).lt.0.)) then
         theta(jt,1)=theta(jt,1)+PI
      ENDIF
      IF ((c1.lt.0.)and.(theta(jt,1).gt.0.)) then
         theta(jt,1)=theta(jt,1)-PI
      ENDIF
   endif
   open (20,file='kejtang.dat')
   write (20,*),kk,theta(4,1),theta(5,1),theta(10,1)
&      ,theta(11,1)
endo}
theta
endo
close(1)
close(20)
print *,n
call cubics(n,thetadot,thetaddot)
open(1,file=posfile)
read (1,*) n
read (1,*) nrel
read (1,*) nrec
read (1,*) nlast
read (1,*) stfile
do 200 fnuin=l,n
   DO j=l,29
      read (1,*) frame,time,(pos(j,k),k=l,3)
      read (2,*) frame,time,(vel(j,k),k=l,3)
      read (3,*) frame,time,(acc(j,k),k=l,3)
   enddo
   call rot(a,pos,rm)
   call segrot(rm,rdp)
   C
   C Determine rotation matrices between segments
   C
call segrot(rm,rdp)
   C
   C From rotation matrix rdp, determine angles between
   segments with respect to proximal segment. Store for
   later use in model. This assumes a 3-1-2 rotation
   about z,x',y'' axes where theta (joint,1) is about z
   axis, theta(joint,2) is x' and theta (joint,3) is
   about y''
   C
   DO 1400 jt=l,ll
      theta(jt,1)=atan(-rdp(jt,1,2)/rdp(jt,2,2))
      theta(jt,2)=atan(rdp(jt,3,2)/sqrt(rdp(jt,1,2)**2
          & +rdp(jt,2,2)**2))
      theta(jt,3)=atan(-rdp(jt,3,1)/rdp(jt,3,3))
      IF ((rdp(jt,3,2).gt.0.).and.(theta(jt,2).lt.0.))
          & then
         theta(jt,2)=theta(jt,2)+PI
      endif
      IF ((rdp(jt,3,2).lt.0.).and.(theta(jt,2).gt.0.))
          & then
         theta(jt,2)=theta(jt,2)-PI
      endif
      if ((theta(jt,2).gt.PI/2).or.
          & (theta(jt,2).lt.-PI/2)) then
         flag=1
      endif
      s3=-(rdp(jt,3,1)/sqrt(rdp(jt,1,2)**2
          & +rdp(jt,2,2)**2))
      if (flag.eq.1) then
         s3=-s3
      endif
199

IF ((s3.gt.0.).and.(theta(jt,3).lt.0.)) then
  theta(jt,3)=theta(jt,3)+PI
ENDIF
IF ((s3.lt.0.).and.(theta(jt,3).gt.0.)) then
  theta(jt,3)=theta(jt,3)-PI
ENDIF
cl=rdp(jt,2,2)/cos(theta(jt,2))
IF ((cl.lt.0.).and.(theta(jt,1).lt.0.)) then
  theta(jt,1)=theta(jt,1)+PI
ENDIF
IF ((cl.lt.0.).and.(theta(jt,1).gt.0.)) then
  theta(jt,1)=theta(jt,1)-PI
ENDIF
write(5,*) (theta(jt,k),',',k=1,3)
1400 continue

C
C compute center of gravity location
C
do i=1,12
  do j=1,3
    pcg(i,j)=0.
  enddo
  do j=1,3
    do k=1,3
      pcg(i,j)=pcg(i,j)+rm(i,j,k)*rcg(i,k)
    enddo
    pcg(i,j)=pos(mark(i),j)-pcg(i,j)
  enddo
  enddo
  sum1=0.
  sum2=0.
  sum3=0.
  mtot=0.
do i=1,12
  sum1=sum1+m(i)*pcg(i,1)
  sum2=sum2+m(i)*pcg(i,2)
  sum3=sum3+m(i)*pcg(i,3)
  mtot=mtot+m(i)
endo
cg(1)=sum1/mtot
cg(2)=sum2/mtot
cg(3)=sum3/mtot

C
C The matrix RM is defined as: [global]=[rm][segment]
C Most of the conversions we will do are from global to
C segment coordinates. Therefore, we will transpose [rm]
C and redefine so that [segment]=[rm][global]
C
do k=1,12
  do i=1,3

do j=1,3
    temprot(i,j)=rm(k,i,j)
enddo
enddo
call trans(temprot)
do i=1,3
doi j=1,3
    rm(k,i,j)=temprot(i,j)
endoenddo

C Find angular velocity and angular acceleration using
C Verstraete and Soutas-Little algorithm
C
num=fnum
print*,num
call angvel(pos,vel,acc,omega,alp,rm,thetadot
& ,thetaddot,num)
C
Find linear acceleration for each segment
a(cg)=a(known point)+alp x r ++omegaxomegaxr
C
DO 1450 i=1,12
C DO cross products
alphaxr(1)=alp(i,2)*rcg(i,3)-alp(i,3)*rcg(i,2)
alphaxr(2)=alp(i,3)*rcg(i,1)-alp(i,1)*rcg(i,3)
alphaxr(3)=alp(i,1)*rcg(i,2)-alp(i,2)*rcg(i,1)
omegaxr(1)=omeg(i,2)*rcg(i,3)-omeg(i,3)*rcg(i,2)
omegaxr(2)=omeg(i,3)*rcg(i,1)-omeg(i,1)*rcg(i,3)
omegaxr(3)=omeg(i,1)*rcg(i,2)-omeg(i,2)*rcg(i,1)
wxwxr(1)=omeg(i,2)*omegaxr(3)-omeg(i,3)*omegaxr(2)
wxwxr(2)=omeg(i,3)*omegaxr(1)-omeg(i,1)*omegaxr(3)
wxwxr(3)=omeg(i,1)*omegaxr(2)-omeg(i,2)*omegaxr(1)
C
C Rotate acceleration vector of marker for segment
C into local coordinates
C
DO 1500 j=1,3
    aa(j)=rm(i,j,1)*acc(mark(i),1)+rm(i,j,2)
& *acc(mark(i),2)+rm(i,j,3)*acc(mark(i),3)
1500 continue
C
Compute the center of gravity accelerations in local
C coordinates. Convert from cm/sec to m/sec
C
DO 1540 j=1,3
    acg(i,j)=(aa(j)+alphaxr(j)+wxwxr(j))/100
1540 continue
1450 continue
Determine momentum change for each segment (Units will be cm-m-kg/sec^2 (N-cm). Divide to get N-m.

```
DO 1550 i=1,12
    hg(i,1)=(ixx(i)*alpha(i,1)-(iyy(i)-izz(i)))*omega(i,2)*omega(i,3))/100
    hg(i,2)=(iyy(i)*alpha(i,2)-(izz(i)-ixx(i)))*omega(i,1)*omega(i,3))/100
    hg(i,3)=(ixx(i)*alpha(i,3)-(ixx(i)-iyy(i)))*omega(i,1)*omega(i,2))/100
1550 continue
```

Determine force vectors at hands using strain gage data

```
read (4,*) frame,time,(sg(i),i=1,8)
if ((fnum.gt.nrel).and.(fnum.it.nrec)) then
    DO 1555 i=1,3
        fl(i)=0.
        fr(i)=0.
1555 continue
elseif (fnum.gt.nlast) then
    DO 1560 i=1,3
        fl(i)=0.
        fr(i)=0.
1560 continue
else
    dc=pos(13,3)+100
    dd=pos(20,3)+100
    call stgage(sg,dc,dd,fl,fr)
endif
```

Compute moments at hands using the constant relationship Tz=K*Fmag

```
FLMAG=sqrt(fl(1)**2+fl(2)**2)
FRMAG=sqrt(fr(1)**2+fr(2)**2)
ML=-3.*FLMAG
MR=-3.*FRMAG
```

Rotate the vectors to forearm coordinates

```
f(12,1)=-(rm(5,1,1)*fr(1)+rm(5,1,2)*fr(2))
    &+rm(5,1,3)*fr(3))
f(12,2)=-(rm(5,2,1)*fr(1)+rm(5,2,2)*fr(2))
    &+rm(5,2,3)*fr(3))
f(12,3)=-(rm(5,3,1)*fr(1)+rm(5,3,2)*fr(2))
    &+rm(5,3,3)*fr(3))
f(13,1)=-(rm(6,1,1)*fl(1)+rm(6,1,2)*fl(2)
& \quad \text{+}\text{rm}(6,1,3)\text{*fl}(3)) \\
& f(13,2)=-(\text{rm}(6,2,1)\text{*fl}(1)+\text{rm}(6,2,2)\text{*fl}(2)) \\
& \quad \text{+}\text{rm}(6,2,3)\text{*fl}(3)) \\
& f(13,3)=-(\text{rm}(6,3,1)\text{*fl}(1)+\text{rm}(6,3,2)\text{*fl}(2)) \\
& \quad \text{+}\text{rm}(6,3,3)\text{*fl}(3)) \\
& \text{mom}(12,1)=-(\text{nti}(5,1,3)\text{*MR}) \\
& \text{mom}(12,2)=-(\text{rm}(5,2,3)\text{*MR}) \\
& \text{mom}(12,3)=-(\text{rm}(5,3,3)\text{*MR}) \\
& \text{mom}(13,1)=-(\text{rm}(6,1,3)\text{*ML}) \\
& \text{mom}(13,2)=-(\text{rm}(6,2,3)\text{*ML}) \\
& \text{mom}(13,3)=-(\text{rm}(6,3,3)\text{*ML}) \\

\begin{verbatim}
C Find weights of segments in segment coordinates
C
DO 1600 i=1,12
   DO 1600 j=1,3
      sw(i,j)=rm(i,j,2)*m(i)*9.81
1600 continue
C
C Find forces at elbows
C
DO 1650 i=1,3
   fd(4,i)=m(5)*acg(5,i)-f(12,i)-sw(5,i)
   fd(5,i)=m(6)*acg(6,i)-f(13,i)-sw(6,i)
1650 continue
C
C Find moments at elbows
C
   rxf(1)=-(1-eta(5))*h(5))*f(12,3)
   rxf(2)=0.
   rxf(3)=((1-eta(5))*h(5))*f(12,1)
   rxf2(1)=eta(5)*h(5)*fd(4,3)
   rxf2(2)=0.
   rxf2(3)=-eta(5)*h(5)*fd(4,1)
   do i=1,3
      momd(4,i)=hg(5,i)-rxf(i)-rxf2(i)-mom(12,i)
   enddo
   rxf(1)=-(1-eta(5))*h(5))*f(13,3)
   rxf(2)=0.
   rxf(3)=((1-eta(5))*h(5))*f(13,1)
   rxf2(1)=eta(5)*h(5)*fd(5,3)
   rxf2(2)=0.
   rxf2(3)=-eta(5)*h(5)*fd(5,1)
   do i=1,3
      momd(5,i)=hg(6,i)-rxf(i)-rxf2(i)-mom(13,i)
   enddo
C
C Rotate into upper arm coordinates
C
DO 1700 j=4,5
\end{verbatim}
DO 1800 i=1,3
  f(j,i)=-(rdp(j,i,1)*fd(j,1)+rdp(j,i,2)*fd(j,2)
  & +rdp(j,i,3)*fd(j,3))
  mom(j,i)=-(rdp(j,i,1)*momd(j,1)+rdp(j,i,2)
  & *momd(j,2)+rdp(j,i,3)*momd(j,3))
1800 continue
1700 continue

C Find forces at shoulders
C
DO 1850 i=1,3
  fd(2,i)=m(7)*acg(7,i)-f(4,i)-sw(7,i)
  fd(3,i)=m(8)*acg(8,i)-f(5,i)-sw(8,i)
1850 continue

C Find moments at shoulders
C
rxf(1)=-(1-eta(7))*h(7)*f(4,3)
  rxf(2)=0.
  rxf(3)=((1-eta(7))*h(7))*f(4,1)
  rxf2(1)=eta(7)*h(7)*fd(2,3)
  rxf2(2)=0.
  rxf2(3)=-eta(7)*h(7)*fd(2,1)
do i=1,3
  momd(i,2)=hg(7,i)-rxf(i)-rxf2(i)-momd(4,i)
endo
rxf(1)=-(1-eta(7))*h(7)*f(5,3)
  rxf(2)=0.
  rxf(3)=((1-eta(7))*h(7))*f(5,1)
  rxf2(1)=eta(7)*h(7)*fd(3,3)
  rxf2(2)=0.
  rxf2(3)=-eta(7)*h(7)*fd(3,1)
do i=1,3
  momd(i,3)=hg(8,i)-rxf(i)-rxf2(i)-momd(5,i)
endo

C Rotate into upper trunk coordinates
C
DO 1900 j=2,3
  DO 2000 i=1,3
    f(j,i)=-(rdp(j,i,1)*fd(j,1)+rdp(j,i,2)*fd(j,2)
    & +rdp(j,i,3)*fd(j,3))
    mom(j,i)=-(rdp(j,i,1)*momd(j,1)+rdp(j,i,2)
    & *momd(j,2)+rdp(j,i,3)*momd(j,3))
  2000 continue
1900 continue

C Find forces at neck
C
DO 2050 i=1,3
fd(1,i)=m(1)*acg(1,i)-sw(1,i)

continue

C Find moments at neck
rrl=h(1)/2.
rfx(1)=rrl*fd(1,3)
rfx(2)=0
rfx(3)=-rrl*fd(1,1)
DO 2055 i=1,3
  momd(1,i)=hg(1,i)-rfx(i)
2055 continue

C Rotate into upper trunk coordinates
DO i=1,3
  f(1,i)=-(rdp(1,i,1)*fd(1,1)+rdp(1,i,2)*fd(1,2)
  &+rdp(1,i,3)*fd(1,3))
  mom(1,i)=-(rdp(1,i,1)*momd(1,1)+rdp(1,i,2)
  &+momd(1,2)+rdp(1,i,3)*momd(1,3))
enddo

C Find forces at upper trunk "joint"
DO i=1,3
  fd(6,i)=m(2)*acg(2,i)-sw(2,i)-f(1,i)-f(2,i)
&-f(3,i)
enddo

C Rotate into mid-trunk coordinates
DO i=1,3
  f(6,i)=-(rdp(6,i,1)*fd(6,1)+rdp(6,i,2)*fd(6,2)
  &+rdp(6,i,3)*fd(6,3))
enddo

C Find forces at knees
DO i=1,3
  fd(10,i)=m(11)*acg(11,i)-sw(11,i)
  fd(11,i)=m(12)*acg(12,i)-sw(12,i)
enddo

C Find moments at knees: mknee=Hq-rxFknee
C
C Take cross product r x F, r=distance from cg to knee
C Right knee
rrl=-eta(11)*h(11)
rfx(1)=rrl*fd(10,3)
rfx(2)=0
rfx(3)=-rrl*fd(10,1)
C Compute moment
DO 2355 i=1,3
momd(10, i)=hg(11, i)-rfx(i)

continue
C Left knee
   rfx(1)=rrl*fd(11, 3)
   rfx(2)=0
   rfx(3)=-rrl*fd(11, 1)
C Compute moment
   DO 2360 i=1,3
      momd(11, i)=hg(12, i)-rfx(i)
   continue
C C Rotate into thigh coordinates
   C
   DO 2400 j=10,11
      DO 2500 i=1,3
         f(j,i)=-(rdp(j,i,1)*fd(j,1)+rdp(j,i,2)*fd(j,2)
&            +rdp(j,i,3)*fd(j,3))
         mom(j,i)=-(rdp(j,i,1)*momd(j, 1)+rdp(j,i,2)
&            *momd(j,2)+rdp(j,i,3)*momd(j,3))
      continue
   2400 continue
C C Find forces at hips
   C
   DO 2550 i=1,3
      fd(8,i)=m(9)*acg(9,i)-f(10,i)-sw(9,i)
      fd(9,i)=m(10)*acg(10,i)-f(11,i)-sw(10,i)
   2550 continue
C C Find moments at hips
   C C Take cross products r x F, r=-rcg
   C Right hip
   C Moment arising from knee force
      rrl=(1-eta(9))*h(9)
      r2=-eta(9)*h(9)
      rfx(1)=rrl*f(10, 3)
      rfx(2)=0
      rfx(3)=-rrl*f(10, 1)
C C Moment arising from hip force
   C rxf2(1)=r2*fd(8, 3)
   C rxf2(2)=0
   C rxf2(3)=-r2*fd(8, 1)
C C Compute moment
   DO 2555 i=1,3
      momd(8, i)=hg(9, i)-rfx(i)-rxf2(i)-mom(10, i)
   2555 continue
C C Left hip
C C Moment arising from knee force
   C rxf(1)=rrl*f(11, 3)
rxf(2)=0
rxf(3)=-rrl*f(11,1)

C Moment arising from hip force
rxf2(1)=r2*fd(9,3)
rxf2(2)=0
rxf2(3)=-r2*fd(9,1)

C Compute moment
DO 2560 i=1,3
    momd(9,i)=hg(10,i)-rxf(i)-rxf2(i)-mom(11,i)
2560 continue

C Rotate into pelvis coordinates
DO 2600 j=8,9
    DO 2700 i=1,3
        f(j,i)=-(rdp(j,i,1)*fd(j,1)+rdp(j,i,2)*fd(j,2)
                 +rdp(j,i,3)*fd(j,3))
        mom(j,i)=- (rdp(j,i,1)*momd(j,1)+rdp(j,i,2)
                     +momd(j,2))
    2700 continue
2600 continue

C Find forces at lower back "joint"
DO 2750 i=1,3
    fd(7,i)=m(4)*acg(4,i)-sw(4,i)-f(8,i)-f(9,i)
2750 continue

C Find moments at lower back joint
rr1=r(9)
r2=h(4)/2.

C Moment arising from right hip force
rxf(1)=r2*f(11,3)-rrl*f(11,2)
rxf(2)=-rrl*f(11,1)
rxf(3)=-r2*f(11,1)

C Moment arising from left hip force
rxf2(1)=r2*f(9,3)+rrl*f(9,2)
rxf2(2)=rrl*f(9,1)
rxf2(3)=-r2*f(9,1)

C Moment arising from lower back force
rxf3(1)=-r2*fd(7,3)
rxf3(2)=0
rxf3(3)=r2*fd(7,1)

C Compute moment
DO 2755 i=1,3
    momd(7,i)=hg(4,i)-rxf(i)-rxf2(i)-rxf3(i)
                 -mom(11,i)-mom(9,i)
2755 continue
C Rotate into mid-trunk coordinates
DO 2850 i=1,3
  f(7,i) = -(rdp(7,i,1)*fd(7,1) + rdp(7,i,2)*fd(7,2)
  & + rdp(7,i,3)*fd(7,3))
  mom(7,i) = -(rdp(7,i,1)*momd(7,1) + rdp(7,i,2)*momd(7,2)
  & + rdp(7,i,3)*momd(7,3))
2850 continue

C Upper spine "joint"

rr1 = h(3)/2.
C Moment arising from lower back force
  rxf(1) = rr1*f(7,3)
  rxf(2) = 0
  rxf(3) = -rr1*f(7,1)
C Moment arising from upper back force
  rxf2(1) = -rr1*f(6,3)
  rxf2(2) = 0
  rxf2(3) = rr1*f(6,1)

C Compute moment
DO 2855 i=1,3
  mom(6,i) = hg(3,i) - rxf(i) - rxf2(i) - mom(7,i)
2855 continue

C Print out forces and torques to data file
write (6,*) frame, time
DO 3000 i=1,13
  write (6,*) i, (f(i,j), j=1,3), (mom(i,j), j=1,3)
3000 continue
write (9,*) frame, time
DO i=1,11
  write (9,*) i, (fd(i,j), j=1,3), (momd(i,j), j=1,3)
enddo
write(9,*) (fr(i), i=1,3), mr
write (9,*) (fl(i), i=1,3), ml

C Write out kinematics to file for further analysis
DO I=1,12
  WRITE (11,*) I, (omega(I,J), J=1,3)
  & (ALPHA(I,J), J=1,3)
  WRITE (7,*) I, (acg(I,J), J=1,3)
ENDDO

C Write out center of gravity trajectory for plotting
write(12,*) frame,",",(cg(j),",",j=1,3)

C There will be some error involved. Compute the upper
C spine moment using the upper trunk segment, and
C subtract from the value found using the mid trunk
segment, print all three values to a file for further analysis

\[
rr1 = \frac{h(2)}{2}.
\]
\[
rr2 = r(2)
\]

Moment arising from neck force
\[
rxf(1) = -rr1*f(1,3)
\]
\[
rxf(2) = 0
\]
\[
rxf(3) = rr1*f(1,1)
\]

Moment arising from upper back force
\[
rxf2(1) = rr1*fd(6,3)
\]
\[
rxf2(2) = 0
\]
\[
rxf2(3) = -rr1*fd(6,1)
\]

Moments from left shoulder and right shoulder
\[
rxf3(1) = -rr1*(f(2,3)+f(3,3))+rr2*(f(2,2)-f(3,2))
\]
\[
rxf3(2) = rr2*(f(3,1)-f(2,1))
\]
\[
rxf3(3) = rr1*(f(2,1)+f(3,1))
\]

Compute moment in upper trunk coordinates
\[
DO \ i = 1, 3
\]
\[
\text{momd}(6,i) = hg(2,i) - rxf(i) - rxf2(i) - rxf3(i)
\]
\text{mom}(1,i) - mom(2,i) - mom(3,i)
\]
\]
\[
enddo
\]

Rotate into mid trunk coordinates
\[
DO \ i = 1, 3
\]
\[
m6(i) = - (rdp(6,i,1)*momd(6,1) + rdp(6,i,2)*momd(6,2) + rdp(6,i,3)*momd(6,3))
\]
\[
merror(i) = m6(i) - mom(6,i)
\]
\]
\[
enddo
\]

Compute the lower back force from the mid trunk segment and subtract from the value found using the pelvis. Write all three values to a file for further analysis.

Find forces at lower trunk "joint"
\[
DO \ i = 1, 3
\]
\[
f7(i) = m(3)*acg(3,i) - sw(3,i) - f(6,i)
\]
\[
ferror(i) = f7(i) - f(7,i)
\]
\]
\[
write(8,*), (f7(i),i=1,3),(f(7,i),i=1,3)
\]
\[
& (ferror(i),i=1,3)
\]
\[
write(8,*), (m6(i),i=1,3),(mom(6,i),i=1,3)
\]
\[
& (merror(i),i=1,3)
\]
\[
200 \ continue
\]
\[
close(1)
\]
\[
close(2)
\]
\[
close(3)
\]
close(4)
close(5)
close(6)
close(8)
close(11)
close(12)
close(7)
close(9)

C RE-order forces, moments, kinematic data, and joint angles and store in new files for use in spreadsheet applications

call jts(prefix,n)
call fts(prefix,n)
call kin(prefix,n)
call fds(prefix,n)
end

C C C Subroutine to transpose a matrix
C
C subroutine trans(m)
real mt(3,3),m(3,3)
DO j=1,3
   DO k=1,3
      mt(j,k)=m(k,j)
   ENDDO
ENDDO
DO j=1,3
   DO k=1,3
      m(j,k)=mt(j,k)
   ENDDO
ENDDO
return

Subroutine INERTIA
C
C Subroutine Inertia(m,ixx,iyy,izz,h,eta,r,rr,sub)
C This subroutine takes anthropometric data and calculates mass, center of gravity and moments of inertia for each segment based on Hanavan and Clauser’s equations
C
character*1 sub
character*35 fn
real m(12),ixx(12),iyy(12),izz(12),h(12),mu
real sigma,eta(12)
real bb,r(12),rr(12),if
pi=3.14159
Define segments as follows

1=Head
2=Upper trunk, 3=Middle Trunk, 4=Lower Trunk
5=Right forearm & hand, 6=Left forearm & hand
7=Right Upper Arm, 8=Left Upper Arm
9=Right thigh, 10=Left thigh
11=Right shank & foot, 12=Left shank & foot

Read in anthropometric data

fn='c:\fortran\data\subject\sub\'.dat'
print *,fn
open (1,FILE=fn)
read(1,4000) age
read(1,4000) headc
read(1,4000) cc
read(1,4000) axilc
read(1,4000) elbc
read(1,4000) fac
read(1,4000) Wrisc
read(1,4000) Fistc
read(1,4000) Thihc
read(1,4000) gknec
read(1,4000) Cfc
read(1,4000) Foarl
read(1,4000) Wrisb
read(1,4000) Handb
read(1,4000) Stat
read(1,4000) Cni
read(1,4000) Shldh
read(1,4000) Subh
read(1,4000) Troch
read(1,4000) Sith
read(1,4000) Wsth
read(1,4000) Ankc
read(1,4000) If
read(1,4000) Chesb
read(1,4000) Chesd
read(1,4000) Waisb
read(1,4000) Waisd
read(1,4000) Hipb
read(1,4000) Buttd
read(1,4000) Uparl
read(1,4000) Tibh
read(1,4000) Sphyh
read(1,4000) Footl
read(1,4000) W
close (1)
4000 format (f8.2)
Distribute weight among segments using Clauser's equations

\[ m(1) = 0.104 \times \text{headc} + 0.015 \times w - 2.189 \]
\[ \text{trunkm} = 0.349 \times w + 0.423 \times (\text{sith} - (\text{stat} - \text{shldh})) + 0.229 \times \text{cc} - 35.460 \]
\[ h(2) = \text{shldh} - \text{subh} \]
\[ h(3) = \text{subh} - \text{wsth} \]
\[ h(4) = (\text{sith} - (\text{stat} - \text{shldh})) - h(2) - h(3) \]
\[ m(2) = h(2)/(\text{sith} - (\text{stat} - \text{shldh})) \times \text{trunkm} \]
\[ m(3) = h(3)/(\text{sith} - (\text{stat} - \text{shldh})) \times \text{trunkm} \]
\[ m(4) = h(4)/(\text{sith} - (\text{stat} - \text{shldh})) \times \text{trunkm} \]
\[ m(5) = 0.029 \times \text{wrisc} + 0.075 \times \text{wrisb} + 0.031 \times \text{handb} - 0.746 + 0.081 \times \text{wrisc} \]
\[ & + 0.052 \times \text{fac} - 1.650 \]
\[ m(6) = m(5) \]
\[ m(7) = 0.007 \times w + 0.092 \times \text{axilc} + 0.095 \times \text{uparl} - 3.101 \]
\[ m(8) = m(7) \]
\[ m(9) = 0.074 \times w + 0.123 \times \text{thihc} + 0.027 \times \text{if} - 4.216 \]
\[ m(10) = m(9) \]
\[ m(11) = 0.111 \times \text{cfc} + 0.047 \times \text{tibh} + 0.074 \times \text{ankc} - 4.208 \]
\[ & + 0.003 \times w + 0.048 \times \text{ankc} + 0.027 \times \text{footl} - 0.8690 \]
\[ m(12) = m(11) \]

Distribute remaining weight over segments

\[ \text{sum} = 0 \]
\[ \text{do} 4100 \ i = 1,12 \]
\[ \text{sum} = \text{sum} + m(i) \]
\[ 4100 \text{ continue} \]
\[ \text{bal} = w - \text{sum} \]
\[ \text{do} 4200 \ i = 1,12 \]
\[ m(i) = m(i) + m(i)/\text{sum} \times \text{bal} \]
\[ 4200 \text{ continue} \]

Calculate principle inertias for each segment

For head
\[ h(1) = \text{stat} - \text{shldh} \]
\[ r(1) = 0.5 \times (\text{stat} - \text{shldh}) \]
\[ r(r(1)) = \text{headc}/2/\pi \]
\[ \text{ixx}(1) = 0.2 \times m(1) \times (r(1)^2 + rr(1)^2) \]
\[ \text{izz}(1) = \text{ixx}(1) \]
\[ \text{iyy}(1) = 0.4 \times m(1) \times rr(1)^2 \]

For trunk
\[ r(2) = 0.5 \times \text{chesb} \]
\[ r(3) = 0.5 \times \text{hipb} \]
\[ r(4) = r(3) \]
rr(2) = 0.25 * (chesh + waisd)
nr(3) = 0.25 * (waisd + buttd)
nr(4) = nr(3)
do 4400 i = 2, 4
   ix(i) = m(i) / 12 * (3 * nr(i)**2 + h(i)**2)
   izz(i) = m(i) / 12 * (3 * r(i)**2 + h(i)**2)
   iyy(i) = m(i) / 4 * (r(i)**2 + nr(i)**2)
4400 continue

C
C For arms and legs, r = large radius, rr = small radius
C
r(7) = axiec / 2 / pi
r(7) = elbc / 2 / pi
r(5) = rr(7)
r(5) = wrisc / 2 / pi
r(9) = thiic / 2 / pi
r(9) = gknecc / 2 / pi
r(11) = rr(9)
r(11) = ankc / 2 / pi
h(7) = uparl
h(5) = foarl
h(9) = stat - sith - tibh
h(11) = tibh - spyh

do 4500 i = 5, 11, 2
   rt = rr(i) / r(i)
   aa = 9. / 20. / pi * (1. + rt + rt**2 + rt**3 + rt**4)
   d = 3. * m(i) / h(i) / (r(i)**2 + rr(i) * r(i) + rr(i)**2) / pi
   ix(i) = aa * m(i)**2 / d / h(i) + bb * m(i) * h(i)**2
   izz(i) = ix(i)
   iyy(i) = 0.3 * m(i) * (r(i)**5 - rr(i)**5) / (r(i)**3 - rr(i)**3)
   ix(i+1) = ix(i)
   iyy(i+1) = iyy(i)
   izz(i+1) = izz(i)
4500 continue

C
C Locate center of gravity along y-axis
C
do 4600 i = 1, 4
   etai = 0.5
4600 continue

do 4700 i = 5, 11, 2
   mu = rr(i) / r(i)
   sigma = 1 + mu + mu**2
   etai = (1 + 2 * mu + 3 * mu**2) / 4 / sigma
   etai = etai
4700 continue
Subroutine ROT: Compute the rotation matrix from global to segment coordinates

```c
subroutine rot(aseg, pos, rm)
C subroutine to compute rotation matrix for each segment with respect to the global coordinate system
C real aseg(12, 4, 3), a(4, 3), p(4, 3), pos(35, 3), r(3, 3)
real rm(12, 3, 3), r2(3, 3)
integer n, seg
C Segments with four markers: thighs, pelvis
C Segments with three markers: head, upper arms, upper trunk, mid trunk
C Segments with two markers: forearms, shanks
C Head
seg=1
n=3
do 5000 i=1, 4
   do 5000 j=1, 3
      a(i, j) = aseg(seg, i, j)
   5000 continue
do 5050 i=1, 3
   p(1, i) = pos(21, i)
   p(2, i) = pos(23, i)
   p(3, i) = pos(22, i)
5050 continue
call rot1(a, p, r, n)
C Upper and mid trunk
C do 5150 seg=2, 3
   do i=1, 4
      do j=1, 3
         a(i, j) = aseg(seg, i, j)
      enddo
   enddo
   n=3
   do 5200 j=1, 3
      if (seg.eq.2) then
```
p(1,j)=pos(26,j)
p(2,j)=pos(24,j)
p(3,j)=pos(25,j)
endif
if (seg.eq.3) then
p(1,j)=pos(27,j)
p(2,j)=pos(28,j)
p(3,j)=pos(29,j)
endif
5200 continue
call rot1(a,p,r,n)
do 5250 j=1,3
do 5250 k=1,3
rin(seg,j,k)=r(j,k)
5250 continue
5150 continue
C
C Upper arm segments
C
do seg=7,8
n=3
do i=1,4
  do j=1,3
    a(i,j)=aseg(seg,i,j)
  enddo
endo
do j=1,3
  if (seg.eq.7) then
    p(1,j)=pos(14,j)
p(2,j)=pos(15,j)
p(3,j)=pos(16,j)
  endif
  if (seg.eq.8) then
    p(1,j)=pos(19,j)
p(2,j)=pos(18,j)
p(3,j)=pos(17,j)
  endif
endo
call rot1(a,p,r,n)
do j=1,3
do k=1,3
  rm(seg,j,k)=r(j,k)
endo
endo
C
C Segments with four markers
C
n=4
seg=4
do i=1,4
  do j=1,3
    a(i,j)=aseg(seg,i,j)
  enddo
enddo
do i=1,3
  p(1,i)=pos(28,i)
p(2,i)=pos(29,i)
p(3,i)=pos(6,i)
p(4,i)=pos(7,i)
enddo
call rotl(a,p,r,n)
do j=1,3
  do k=1,3
    rin(seg, j,k)=r(j,k)
  enddo
enddo
do seg=9,10
  n=4
  do i=1,4
    do j=1,3
      a(i,j)=aseg(seg,i,j)
    enddo
  enddo
do j=1,3
  if (seg.eq.9) then
    p(1,j)=pos(2,j)
p(2,j)=pos(3,j)
p(3,j)=pos(4,j)
p(4,j)=pos(5,j)
  endif
  if (seg.eq.10) then
    p(1,j)=pos(11,j)
p(2,j)=pos(10,j)
p(3,j)=pos(9,j)
p(4,j)=pos(8,j)
  endif
enddo
call rotl(a,p,r,n)
do 5400 j=1,3
  do 5400 k=1,3
    rin(seg,j,k)=r(j,k)
  enddo
5400 continue
enddo
C
C  Segments with two markers
C
do seg=5,6
  do i=1,4
    do j=1,3
DO J=1,3
if (seg.eq.5) then
  p(1,j)=pos(13,j)
p(2,j)=pos(14,j)
do i=1,3
  r(i,j)=rm(7,j,i)
enddo
endif
if (seg.eq.6) then
  p(1,j)=pos(20,j)
p(2,j)=pos(19,j)
do i=1,3
  r(i,j)=rm(8,j,i)
enddo
endif
ENDDO

call rot2(a,p,r,r2)
do i=1,3
  do j=1,3
    rm(seg,i,j)=r2(i,j)
  enddo
enddo
endo
do seg=1,12
  do i=1,4
    do j=1,3
      a(i,j)=aseg(seg,i,j)
    enddo
  enddo
do j=1,3
  if (seg.eq.11) then
    p(1,j)=pos(1,j)
p(2,j)=pos(2,j)
do i=1,3
      r(i,j)=rm(9,j,i)
    enddo
  endif
  if (seg.eq.12) then
    p(1,j)=pos(12,j)
p(2,j)=pos(11,j)
do i=1,3
      r(i,j)=rm(10,j,i)
    enddo
  endif
ENDDO

call rot2(a,p,r,r2)
do i=1,3
do j=1,3
   rm(seg,i,j)=r2(i,j)
enddo
enddo
return
end

subroutine rot1(a,p,r,n)
  real a(4,3),p(4,3),amean(3),pmean(3),mt(3,3),mag
  real PAT(4,3,3),PATmean(3,3),M(3,3),MTM(3,3)
  real d11,d22,d33
  real v(3,3),mv(3,3),vt(3,3),r(3,3),x,x1,x2,x3
  integer n
  do 7000 i=1,3
     amean(i)=0.
     pmean(i)=0.
  7000 continue
  do 7050 j=1,3
    do 7050 i=1,n
      amean(j)=amean(j)+a(i,j)
      pmean(j)=pmean(j)+p(i,j)
  7050 continue
C
C compute the average vectors
C
  do 7100 i=1,3
     amean(i)=amean(i)/n
     pmean(i)=pmean(i)/n
  7100 continue
C
C compute matrices PAT
  do 7150 i=1,n
    do 7150 j=1,3
      do 7150 k=1,3
         pat(i,j,k)=p(i,j)*a(i,k)
      7150 continue
    7150 continue
  do 7200 i=1,3
    do 7200 j=1,3
      patmean(i,j)=pmean(i)*amean(j)
  7200 continue
C
C Compute matrix M
C
  do 7250 i=2,n
    do 7250 j=1,3
      do 7250 k=1,3
         pat(1,j,k)=pat(1,j,k)+pat(i,j,k)
      7250 continue
    7250 continue
  do 7300 j=1,3
do 7300 k=1,3
    pat(l,j,k)=pat(l,j,k)/n
    m(j,k)=pat(l,j,k)-patmean(j,k)
7300 continue

C Find the matrix MTM
C First, transpose matrix M and put in matrix MT
C
C do 7350 j=1,3
C do 7350 k=1,3
    mt(j,k)=m(k,j)
7350 continue
C call mm(mt,m,mtm)
C find eigenvalues and eigenvectors for MTM
C Use Newton's method to find three roots for the cubic equation (to find eigenvalues)
C Using matrix MTM find the cubic equation's coefficients 1,a2,a1,a0
C
42=a2=-(mtm(1,1)+mtm(2,2)+mtm(3,3))
a1=mtm(1,1)*mtm(2,2)+mtm(1,1)*mtm(3,3)+mtm(3,3)
& *mtm(2,2)-mtm(1,2)**2-mtm(1,3)**2-mtm(3,2)**2
a0=mtm(1,3)**2*mtm(2,2)+mtm(1,2)**2*mtm(3,3)
& +mtm(3,2)**2*mtm(1,1)-mtm(1,1)*mtm(2,2)*mtm(3,3)
& -2*mtm(1,2)**2*mtm(3,3)*mtm(1,3)
x=1.
n=0
7400 x1=x-(x**3+a2*x**2+a1*x+40)/(3*x**2+2*a2*x+41)
    if (abs(x1-x).gt.1e-6) then
        n=n+1
    endif
    if (n.gt.1e3) goto 7450
    x=x1
    goto 7400
endif

C Find quadratic equation that remains after first root is found. Let coefficients be 1,b1,b0
C
b1=a2-x1
b0=a1-b1*x1
    IF (B1**2-4*B0.LE.0) THEN
        R1=-B1/2
        R2=-B1/2
    ELSE
        r1=(-b1+sqrt(b1**2-4*b0))/2
        r2=(-b1-sqrt(b1**2-4*b0))/2
C
Use r1 and r2 as first try at roots for cubic equation

x=r1
n=0
7500 x2=x-(x**3+a2*x**2+a1*x+a0)/(3*x**2+2*a2*x+al)
if (abs(x2-x).gt.1e-6) then
  x=x2
  n=n+1
  if (n.gt.1000) goto 7550
  go to 7500
endif
7550 x=r2
n=0
7600 x3=x-(x**3+a2*x**2+a1*x+a0)/(3*x**2+2*a2*x+al)
if (abs(x3-x).gt.1e-6) then
  x=x3
  n=n+1
  if (n.gt.1e3) goto 7650
  goto 7600
endif
C compute eigenvectors (orthonormal set)
C first, order the eigenvalues in order (largest to smallest)

7650 dll=x2
d22=x3
d33=x1
if (d33.gt.d22) then
d22=x1
d33=x3
endif
v(1,1)=1.
v(3,1)=(d11-mtm(1,1)+mtm(1,2)**2/(mtm(2,2)-dll))
& /((mtm(1,3)-mtm(1,2)*mtm(3,2)/(mtm(2,2)-dll))
v(2,1)=(d11-mtm(1,1)-mtm(1,3)*v(3,1))/mtm(1,2)
C normalize
mag=sqrt(v(1,1)**2+v(2,1)**2+v(3,1)**2)
v(1,1)=v(1,1)/mag
v(2,1)=v(2,1)/mag
v(3,1)=v(3,1)/mag
v(1,2)=1.
v(3,2)=(d22-mtm(1,1)+mtm(1,2)**2/(mtm(2,2)-d22))
& /((mtm(1,3)-mtm(1,2)*mtm(3,2)/(mtm(2,2)-d22))
v(2,2)=(d22-mtm(1,1)-mtm(1,3)*v(3,2))/mtm(1,2)
C normalize
mag=sqrt(v(1,2)**2+v(2,2)**2+v(3,2)**2)
v(l,2)=v(l,2)/mag
v(2,2)=v(2,2)/mag
v(3,2)=v(3,2)/mag

third eigenvector is cross product of 1 and 2

v(1,3)=v(2,1)*v(3,2)-v(3,1)*v(2,2)
v(2,3)=v(3,1)*v(1,2)-v(1,1)*v(3,2)
v(3,3)=v(1,1)*v(2,2)-v(2,1)*v(1,2)

Multiply matrices M and V to get MV

call mm(m,v,mv)

Rotation matrix is 1/d11*mv1, 1/d22*mv2, 1/d11/d22*mv1*mv2 multiplied by vt

d11=sqrt(d11)
d22=sqrt(d22)

Cross product of first two columns

mv(1,3)=mv(2,1)*mv(3,2)-mv(3,1)*mv(2,2)
mv(2,3)=mv(3,1)*mv(1,2)-mv(1,1)*mv(3,2)
mv(3,3)=mv(1,1)*mv(2,2)-mv(2,1)*mv(1,2)

Normalize magnitudes

do 7700 i=1,3
   mv(i,1)=mv(i,1)/d11
   mv(i,2)=mv(i,2)/d22
   mv(i,3)=mv(i,3)/d11/d22
continue

compute V transpose =vt

do 7750 j=1,3
   do 7750 k=1,3
     vt(j,k)=v(k,j)
continue

call mm(mv,vt,r)
return

subroutine rot2(a,p,r,r2)
real a(4,3),p(4,3),r(3,3),r2(3,3),r1(3,3)
real arel(3),prel(3)

do i=1,3
   prel(i)=p(1,i)-p(2,i)
   arel(i)=a(1,i)-a(2,i)
endo
pl=r(1,1)*prel(1)+r(1,2)*prel(2)+r(1,3)*prel(3)
p2=r(2,1)*prel(1)+r(2,2)*prel(2)+r(2,3)*prel(3)
p3=r(3,1)*prel(1)+r(3,2)*prel(2)+r(3,3)*prel(3)
stheta=(p2*arel(1)-pl*arel(2))/((arel(2)**2+arel(1)**2))
if (stheta.gt.1.) then
  stheta=1.
end if
if (stheta.lt.-1.) then
  stheta=-1.
end if
ctheta=sqrt(1-stheta**2)
rl(1,1)=ctheta
rl(1,2)=-stheta
rl(2,1)=stheta
rl(2,2)=ctheta
rl(1,3)=0.
rl(2,3)=0.
rl(3,1)=0.
rl(3,2)=0.
rl(3,3)=1.
call trans(r)
call inm(r,rl,r2)
return
end

Subroutine SEGROT: Computes rotation matrices between segments

subroutine segrot(rm,rdp)
  real rm(12,3,3),rdp(11,3,3),r1(3,3),r2(3,3),r21(3,3)
do i=1,3
  do j=1,3
    r1(i,j)=rm(2,i,j)
    r2(i,j)=rm(1,i,j)
  enddo
enddo
call trans(r1)
call mm(r1,r2,r21)
do 350 i=1,3
  do 350 j=1,3
    rdp(1,i,j)=r21(i,j)
  continue
  do 400 i=1,3
    do 400 j=1,3
      r1(i,j)=rm(3,i,j)
      r2(i,j)=rm(2,i,j)
    continue
  call trans(r1)
end subroutine segrot
call mm(r1,r2,r21)
do 450 i=1,3
  do 450 j=1,3
    rdp(6,i,j)=r21(i,j)
  continue
do 500 i=1,3
  do 500 j=1,3
    r2(i,j)=rin(4,i,j)
  continue
call mm(r1,r2,r21)
do 550 i=1,3
  do 550 j=1,3
    rdp(7,i,j)=r21(i,j)
  continue
do 600 i=1,3
  do 600 j=1,3
    r1(i,j)=rm(2,i,j)
    r2(i,j)=rm(7,i,j)
  continue
call trans(r1)
call mm(r1,r2,r21)
do 650 i=1,3
  do 650 j=1,3
    rdp(2,i,j)=r21(i,j)
  continue
do 700 i=1,3
  do 700 j=1,3
    r2(i,j)=rm(8,i,j)
  continue
call mm(r1,r2,r21)
do 750 i=1,3
  do 750 j=1,3
    rdp(3,i,j)=r21(i,j)
  continue
do 800 i=1,3
  do 800 j=1,3
    r1(i,j)=rm(7,i,j)
    r2(i,j)=rm(5,i,j)
  continue
call trans(r1)
call mm(r1,r2,r21)
do 850 i=1,3
  do 850 j=1,3
    rdp(4,i,j)=r21(i,j)
  continue
do 900 i=1,3
  do 900 j=1,3
    r1(i,j)=rm(8,i,j)
    r2(i,j)=rm(6,i,j)
  continue
call trans(r1)
call mm(r1,r2,r21)
do 950 i=1,3  
do 950 j=1,3  
rdf(5,i,j)=r21(i,j)
950 continue

do 1000 i=1,3  
do 1000 j=1,3  
r1(i,j)=rm(4,i,j)  
r2(i,j)=rm(9,i,j)
1000 continue

call trans(r1)
call mm(r1,r2,r21)
do 1050 i=1,3  
do 1050 j=1,3  
rdf(8,i,j)=r21(i,j)
1050 continue

do 1100 i=1,3  
do 1100 j=1,3  
r2(i,j)=rm(10,i,j)
1100 continue

call mm(r1,r2,r21)
do 1150 i=1,3  
do 1150 j=1,3  
rdf(9,i,j)=r21(i,j)
1150 continue

do 1200 i=1,3  
do 1200 j=1,3  
r1(i,j)=rm(10,i,j)  
r2(i,j)=rm(12,i,j)
1200 continue

call trans(r1)
call mm(r1,r2,r21)
do 1250 i=1,3  
do 1250 j=1,3  
rdf(11,i,j)=r21(i,j)
1250 continue

do 1300 i=1,3  
do 1300 j=1,3  
r1(i,j)=rm(9,i,j)  
r2(i,j)=rm(11,i,j)
1300 continue

call trans(r1)
call mm(r1,r2,r21)
do 1350 i=1,3  
do 1350 j=1,3  
rdf(10,i,j)=r21(i,j)
1350 continue
return
end
**Subroutine CUBICS: Use cubic splines to compute derivatives of angles**

This program takes the elbow and knee angle output values and uses cubic splines to determine first and second derivatives. This program uses "canned" subroutines from Kahaner, Moler and Nash.

```fortran
subroutine cubics(n,d,d1)
real time(150),s1(150),s2(150),s3(150),s4(150)
real t(150),wk(300)
real ss1(150),ss2(150),ss3(150),ss4(150)
real ss5(150),ss6(150),ss7(150),ss8(150)
real dl(150,4),d(150,4)
character*50 fnl

C read in values from data file

fnl='kejtang.dat'
open (20,file=fnl)
print *,n
read in number of points
do 100 i=1,n
   read (20,*1) time(i),s1(i),s2(i),s3(i),s4(i)
100 continue
close(20)

call programs to interpolate using cubic splines

call pchez(n,time,s1,ss1,.true.,wk,300,ierr)
call pchez(n,time,s2,ss2,.true.,wk,300,ierr)
call pchez(n,time,s3,ss3,.true.,wk,300,ierr)
call pchez(n,time,s4,ss4,.true.,wk,300,ierr)

C Second derivatives

call pchez(n,time,ss1,ss5,.true.,wk,300,ierr)
call pchez(n,time,ss2,ss6,.true.,wk,300,ierr)
call pchez(n,time,ss3,ss7,.true.,wk,300,ierr)
call pchez(n,time,ss4,ss8,.true.,wk,300,ierr)

C Send output to variables d and d1

do 1000 i=1,n
   d(i,1)=ss1(i)
   d(i,2)=ss2(i)
   d(i,3)=ss3(i)
   d(i,4)=ss4(i)
   dl(i,1)=ss5(i)
   dl(i,2)=ss6(i)
```

---

**Note:** The above code is a natural representation of the Fortran code snippet provided. It has been formatted to improve readability and maintain the logical structure of the original code.


```c

dl(i,3)=ss7(i)
dl(i,4)=ss8(i)

1000 continue
return
end

Subroutine ANGVEL: Computes the angular velocity and angular acceleration of each segment in segment coordinates

```
\[
\omega(\text{seg},i) = r_{m}(1,i,1)w(1) + r_{m}(1,i,2)w(2) + r_{m}(1,i,3)w(3)
\]
\[
\alpha(\text{seg},i) = r_{m}(1,i,1)\text{angacc}(1) + r_{m}(1,i,2)\text{angacc}(2) + r_{m}(1,i,3)\text{angacc}(3)
\]

**Upper Arm segments**

```fortran
8050 continue
C
C Upper Arm segments
C
do 8100 seg=7,8
  n=3
  do 8150 j=1,3
    if (seg.eq.7) then
      P(l,j)=POS(14,j)
v(l,j)=vel(14,j)
a(l,j)=acc(14,j)
p(2,j)=pos(15,j)
v(2,j)=vel(15,j)
a(2,j)=acc(15,j)
p(3,j)=pos(16,j)
v(3,j)=vel(16,j)
a(3,j)=acc(16,j)
p(4,j)=0.
v(4,j)=0.
a(4,j)=0.
    endif
    if (seg.eq.8) then
      P(1,j)=POS(17,j)
v(1,j)=vel(17,j)
a(1,j)=acc(17,j)
p(2,j)=pos(18,j)
v(2,j)=vel(18,j)
a(2,j)=acc(18,j)
p(3,j)=pos(19,j)
v(3,j)=vel(19,j)
a(3,j)=acc(19,j)
p(4,j)=0.
v(4,j)=0.
a(4,j)=0.
    endif
  end if
8150 continue
C
C Trunk segments
```

```fortran
8200 continue
  call angvell(p,v,a,w,angacc,n)
do 8200 i=1,3
  omega(\text{seg},i) = r_{m}(1,i,1)w(1) + r_{m}(1,i,2)w(2) + r_{m}(1,i,3)w(3)
  alpha(\text{seg},i) = r_{m}(1,i,1)\text{angacc}(1) + r_{m}(1,i,2)\text{angacc}(2) + r_{m}(1,i,3)\text{angacc}(3)
8200 continue
8100 continue
C
C Trunk segments
```
C Upper and mid trunk (segments 2 and 3):  n=3
  do 8250 seg=2,4
   n=3
   do 8300 j=1,3
     if (seg.eq.2) then
       p(1,j)=pos(24,j)
       v(1,j)=vel(24,j)
       a(1,j)=acc(24,j)
       p(2,j)=pos(25,j)
       v(2,j)=vel(25,j)
       a(2,j)=acc(25,j)
       p(3,j)=pos(26,j)
       v(3,j)=vel(26,j)
       a(3,j)=acc(26,j)
       p(4,j)=0.
       v(4,j)=0.
       a(4,j)=0.
     endif
     if (seg.eq.3) then
       p(1,j)=pos(27,j)
       v(1,j)=vel(27,j)
       a(1,j)=acc(27,j)
       p(2,j)=pos(28,j)
       v(2,j)=vel(28,j)
       a(2,j)=acc(28,j)
       p(3,j)=pos(29,j)
       v(3,j)=vel(29,j)
       a(3,j)=acc(29,j)
       p(4,j)=0.
       v(4,j)=0.
       a(4,j)=0.
     endif
     if (seg.eq.4) then
       n=4
       p(1,j)=pos(28,j)
       v(1,j)=vel(28,j)
       a(1,j)=acc(28,j)
       p(2,j)=pos(29,j)
       v(2,j)=vel(29,j)
       a(2,j)=acc(29,j)
       p(3,j)=pos(6,j)
       v(3,j)=vel(6,j)
       a(3,j)=acc(6,j)
       p(4,j)=pos(7,j)
       v(4,j)=vel(7,j)
       a(4,j)=acc(7,j)
     endif
   8300 continue
  call angvell(p,v,a,w,angacc,n)
do 8350 i=1,3  
omega(seg, i)=rm(l, i, 1)*w(1)+rm(l, i, 2)*w(2)  
& +rm(l, i, 3)*w(3)  
alpha(seg, i)=rm(l, i, 1)*angacc(l)+rm(l, i, 2)*angacc(2)+  
& rm(l, i, 3)*angacc(3)  
8350 continue  
8250 continue  
C  
Thigh segments  
C  
do 8400 seg=9,10  
n=4  
do 8450 j=1,3  
if (seg.eq.9) then  
p(1,j)=pos(2,j)  
v(1,j)=vel(2,j)  
a(1,j)=acc(2,j)  
p(2,j)=pos(3,j)  
v(2,j)=vel(3,j)  
a(2,j)=acc(3,j)  
p(3,j)=pos(4,j)  
v(3,j)=vel(4,j)  
a(3,j)=acc(4,j)  
p(4,j)=pos(5,j)  
v(4,j)=vel(5,j)  
a(4,j)=acc(5,j)  
endif  
if (seg.eq.10) then  
p(1,j)=pos(8,j)  
v(1,j)=vel(8,j)  
a(1,j)=acc(8,j)  
p(2,j)=pos(9,j)  
v(2,j)=vel(9,j)  
a(2,j)=acc(9,j)  
p(3,j)=pos(10,j)  
v(3,j)=vel(10,j)  
a(3,j)=acc(10,j)  
p(4,j)=pos(11,j)  
v(4,j)=vel(11,j)  
a(4,j)=acc(11,j)  
endif  
8450 continue  
call angvell(p, v, a, w, angacc,n)  
do 8500 i=1,3  
omega(seg, i)=rm(l, i, 1)*w(1)+rm(l, i, 2)*w(2)  
& +rm(l, i, 3)*w(3)  
alpha(seg, i)=rm(l, i, 1)*angacc(l)+rm(l, i, 2)*angacc(2)+  
& rm(l, i, 3)*angacc(3)  
8500 continue  
8400 continue
To compute the angular velocity of forearm and shank segments use the rotation matrix. Omega of the forearm = omega of the upper arm plus the relative angular velocity of the forearm to the upper arm.

```
do 444 seg=11,12

print *,fnum
print *,seg-8
relomega=tdot(fnum,seg-8)
omega(seg,1)=omega(seg-2,1)
omega(seg,2)=omega(seg-2,2)
omega(seg,3)=omega(seg-2,3)+relomega

```

Compute angular acceleration in proximal segment coordinates

```
alpha(seg,1)=alpha(seg-2,1)+relomega*omega(seg-2,2)
alpha(seg,2)=alpha(seg-2,2)-relomega*omega(seg-2,1)
relalpha=tddot(fnum,seg-8)
alpha(seg,3)=alpha(seg-2,3)+relalpha
```

Rotate into distal segment coordinates

```
do j=1,3
  temp(j)=omega(seg,j)
  temp4(j)=alpha(seg,j)
enddo
do i=1,3
  do j=1,3
    x(i,j)=rm(seg-2,j,i)
    xx(i,j)=rm(seg,i,j)
  enddo
enddo
```

call matvec(x,temp,temp2)
call matvec(x,temp4,temp5)
call matvec(xx,temp2,temp3)
call matvec(xx,temp5,temp6)
do i=1,3
  omega(seg,i)=temp3(i)
  alpha(seg,i)=temp6(i)
enddo
```

444 continue
```
do 445 seg=5,6
relomega=tdot(fnum,seg-4)
relalpha=tddot(fnum,seg-4)
omega(seg,1)=omega(seg+2,1)
omega(seg,2)=omega(seg+2,2)
omega(seg,3)=omega(seg+2,3)+relomega
```
Compute angular acceleration in proximal segment coordinates

\[
\alpha(\text{seg},1) = \alpha(\text{seg}+2,1) + \text{relomega} \cdot \omega(\text{seg}+2,2)
\]
\[
\alpha(\text{seg},2) = \alpha(\text{seg}+2,2) - \text{relomega} \cdot \omega(\text{seg}+2,1)
\]
\[
\alpha(\text{seg},3) = \alpha(\text{seg}+2,3) + \text{relalpha}
\]

Rotate into distal segment coordinates

\[
\text{do } j = 1, 3 \\
\quad \text{temp}(j) = \omega(\text{seg},j) \\
\quad \text{temp4}(j) = \alpha(\text{seg},j)
\]
\[
\text{do } i = 1, 3 \\
\quad \text{do } j = 1, 3 \\
\quad \quad x(i,j) = \text{rm}(\text{seg}+2,j,i) \\
\quad \quad xx(i,j) = \text{rm}(\text{seg},i,j)
\]
\[
\text{enddo}
\]
\[
\text{enddo}
\]
\[
\text{call matvec}(x, \text{temp}, \text{temp2})
\]
\[
\text{call matvec}(x, \text{temp4}, \text{temp5})
\]
\[
\text{call matvec}(xx, \text{temp2}, \text{temp3})
\]
\[
\text{call matvec}(xx, \text{temp4}, \text{temp6})
\]
\[
\text{do } i = 1, 3 \\
\quad \omega(\text{seg},i) = \text{temp3}(i) \\
\quad \alpha(\text{seg},i) = \text{temp6}(i)
\]
\[
\text{enddo}
\]
\[
\text{445 continue}
\]
\[
\text{close(10)}
\]
\[
\text{return}
\end
\]

subroutine angvell(pos, vel, acc, s, t, n)
real pos(4, 3), vel(4, 3), acc(4, 3), npos(6, 3)
real rpos(6, 3), rvel(6, 3), racc(6, 3), rmag(6)
real a(18), b(18), c(18), d(18), w(3), s(3)
real work(3), rcond, t(3), wxr(6, 3), wwr(6, 3)
integer iwork(3)

C Compute relative position, velocity, and accel. vectors
C
\[
\text{do } 9050 \ j = 1, 6 \\
\quad \text{do } 9050 \ i = 1, 3 \\
\quad \quad rpos(j,i) = 0.
\]
9050 continue
\[
\text{do } 9100 \ i = 1, 3 \\
\quad rpos(1,i) = pos(1,i) - pos(2,i) \\
\quad rpos(2,i) = pos(1,i) - pos(3,i)
\]
rpos(4,i)=pos(1,i)-pos(4,i)
rpos(3,i)=pos(2,i)-pos(3,i)
rpos(5,i)=pos(2,i)-pos(4,i)
rpos(6,i)=pos(3,i)-pos(4,i)
rvel(1,i)=vel(1,i)-vel(2,i)
rvel(2,i)=vel(1,i)-vel(3,i)
rvel(4,i)=vel(1,i)-vel(4,i)
rvel(3,i)=vel(2,i)-vel(3,i)
rvel(5,i)=vel(2,i)-vel(4,i)
rvel(6,i)=vel(3,i)-vel(4,i)
racc(1,i)=acc(1,i)-acc(2,i)
racc(2,i)=acc(1,i)-acc(3,i)
racc(4,i)=acc(1,i)-acc(4,i)
racc(3,i)=acc(2,i)-acc(3,i)
racc(5,i)=acc(2,i)-acc(4,i)
racc(6,i)=acc(3,i)-acc(4,i)

9100 continue

C Remove the portion of the relative velocity that is
C not perpendicular to the relative position vector
C (assuming that this is due to soft tissue motion
C

do i=1,6
   dot=0.
   rmag(i)=sqrt(rpos(i,1)**2+rpos(i,2)**2+rpos(i,3)**2)
   do j=1,3
      npos(i,j)=rpos(i,j)/rmag(i)
      dot=dot+rvel(i,j)*npos(i,j)
   enddo
   do j=1,3
      rvel(i,j)=rvel(i,j)-dot*npos(i,j)
   enddo
enddo

do 9150 i=1,9*(n-2),3
   a(i)=0.
   a(i+1)=-rpos((i+2)/3,3)
   a(i+2)=rpos((i+2)/3,2)
   b(i)=rpos((i+2)/3,3)
   b(i+1)=0.
   b(i+2)=-rpos((i+2)/3,1)
   c(i)=-rpos((i+2)/3,2)
   c(i+1)=rpos((i+2)/3,1)
   c(i+2)=0.
   d(i)=rvel((i+2)/3,1)
   d(i+1)=rvel((i+2)/3,2)
   d(i+2)=rvel((i+3)/3,3)

9150 continue

C Reduce to three simultaneous equations
C
do 9200 i=1,3
  do 9200 j=1,3
    w(i,j)=0.
    s(i)=0.
 9200 continue
do 9250 i=1,9*(n-2)
  w(l,l)=w(l,l)+a(i)**2
  w(l,2)=w(l,2)+a(i)*b(i)
  w(l,3)=w(l,3)+a(i)*c(i)
  w(2,l)=w(l,2)
  w(2,2)=w(2,2)+b(i)**2
  w(2,3)=w(2,3)+b(i)*c(i)
  w(3,l)=w(l,3)
  w(3,2)=w(2,3)
  w(3,3)=w(3,3)+c(i)**2
  s(l)=s(l)+a(i)*d(i)
  s(2)=s(2)+b(i)*d(i)
  s(3)=s(3)+c(i)*d(i)
9250 continue
call sgefs(w,3,3,s,1,ind,work,iwork,rcond)

Calculation of angular acceleration using Verstraete & Soutas-Little algorithm

First, compute vector d={a[i/j-[wx(wxri/j)]]}k

Find cross products: wxri/j

do 9300 i=1,6
  wxr(i,1)=s(2)*rpos(i,3)-s(3)*rpos(i,2)
  wxr(i,2)=s(3)*rpos(i,1)-s(1)*rpos(i,3)
  wxr(i,3)=s(1)*rpos(i,2)-s(2)*rpos(i,1)
9300 continue

Find cross products: wxwxr

do 9350 i=1,6
  wwr(i,1)=s(2)*wxr(i,3)-s(3)*wxr(i,2)
  wwr(i,2)=s(3)*wxr(i,1)-s(1)*wxr(i,3)
  wwr(i,3)=s(1)*wxr(i,2)-s(2)*wxr(i,1)
9350 continue

Calculate vector d

do 9400 i=1,9*(n-2),3
  d(i)=racc((i+2)/3,1)-wwr((i+2)/3,1)
  d(i+1)=racc((i+2)/3,2)-wwr((i+2)/3,2)
  d(i+2)=racc((i+2)/3,3)-wwr((i+2)/3,3)
9400 continue

do 9450 i=1,3
t(i)=0.

9450 continue
do 9500 i=1,9*(n-2)
  t(1)=t(1)+a(i)*d(i)
  t(2)=t(2)+b(i)*d(i)
  t(3)=t(3)+c(i)*d(i)
9500 continue

call sgefs(w,3,3,t,2,ind,work,iwork,rcond)
return
end

Subroutine STGAGE: Computes forces at the hands using strain
gage data and video data

subroutine stgage(st,c,d,fr,fl)
   
   ! This subroutine calculates forces at hands in global
   ! coordinates using data from strain gages and the
   ! locations of the backs of wrists along the bar
   
   real st(8),c,d,fr(3),fl(3),len,convfac,a,b
   real mxa,mxb,vlx,vrx,mlx,mrx,m1x,m2x
   real mya,myb,vly,vry,mly,mry,m2y

   ! Define constants
   !
   ! a=2.54
   ! b=33.02
   ! convfac=43448.0
   ! len=241

   ! First, do calculations for the x-direction
   !
   mxa=st(1)*convfac
   mxb=st(2)*convfac
   vlx=(mxb-mxa)/(b-a)
   mlx=mxb-vlx*b
   vrxy=convfac*(st(3)-st(4))/(b-a)
   mrx=st(3)*convfac-vrxy*b
   mlx=mlx+vlx*c
   m2x=mrx+vrxy*(len-d)
   fr(1)=vlx-(m2x-mlx)/(d-c)
   fl(1)=vrxy+vlx-fr(1)

   ! Second, do calculations for the y-direction
   !
   mya=st(8)*convfac
   myb=st(7)*convfac
   vly=(myb-mya)/(b-a)
\[ \text{mly} = \text{myb} - \text{vly} \times \text{b} \]
\[ \text{vry} = \text{convfac} \times (\text{st}(6) - \text{st}(5)) / (\text{b} - \text{a}) \]
\[ \text{mry} = \text{st}(6) \times \text{convfac} - \text{vry} \times \text{b} \]
\[ \text{mly} = \text{mly} + \text{vly} \times c \]
\[ \text{m2y} = \text{mry} + \text{vry} \times (\text{len} - \text{d}) \]
\[ \text{fr}(2) = \text{vly} - (\text{m2y} - \text{mly}) / (\text{d} - \text{c}) \]
\[ \text{f1}(2) = \text{vry} + \text{vly} - \text{fr}(2) \]

C

Now, the z-direction is assumed to be zero

C

\[ \text{fr}(3) = 0. \]
\[ \text{f1}(3) = 0. \]
return
end

Subroutine JTS: Sorts joint angles into a form usable in spreadsheets

C

THIS PROGRAM SORTS THE DATA FROM THE *.jta FILES AND
ORDERS IT BY JOINT SO THAT IT CAN BE GRAPHED
C

subroutine jts(prefix, n)
REAL angle(150,13,3)
character*6 prefix
character*33 fn, fn2
fn = 'c:\fortran\data'//prefix//''.jta'
open (1,file=fn)
do i=1,n
do j=1,11
read (1,*)(angle(i,j,k), k=1,3)
endo
endo
close (1)
fn2 = 'c:\fortran\data'//prefix//''.jts'
open (2,file=fn2)
do i=1,11
do j=1,n
write (2,FMT=100) j, (angle(j,i,k), k=1,3)
100 FORMAT((I3.1, ',', F10.2, ',', F10.2, ',', F10.2, ',', F10.2)
endo
endo
close(2)
end

Subroutine FTS: Sorts force and torque data into format usable in spreadsheet applications
C

THIS PROGRAM SORTS THE DATA FROM THE *.ftd FILES AND
ORDERS IT BY JOINT SO THAT IT CAN BE GRAPHED
C

subroutine fts(prefix,n)
REAL force(150,13,3),moment(150,13,3)
character*6 prefix
character*33 fn,fn2
fn='c:\fortran\data\'/prefix//' .ftd'
open (1,file=fn)
do i=1,n
   read (1,*) frame,time
do j=1,13
   read (1,*) joint,(force(i,j,k),K=1,3)
   ,(moment(i,j,k),K=1,3)
enddo
dndo
close (1)
fn2='c:\fortran\data\'/prefix//' .fts'
open (2,file=fn2)
do i=1,13
do j=1,n
   write (2,FMT=100) j,(force(j,i,k),K=1,3)
   ,(moment(j,i,k),K=1,3)
100 FORMAT(13.1, ' ', F10.2, ',', F10.2, ',', F10.2)
dndo
dndo
close(2)
end

Subroutine KIN: Sorts kinematic data into format useable with spreadsheet applications
C  THIS PROGRAM Sorts the data from the *.kin files and orders it by joint so that it can be graphed
C
C subroutine kin(prefix,n)
REAL ALPHA(150,12,3),OMEGA(150,12,3),joint
character*6 prefix
character*33 fn,fn2
fn='c:\fortran\data\'/prefix//' .kin'
open (1,file=fn)
do i=1,n
   read (1,*) joint,(omega(i,j,k),K=1,3)
   ,(ALPHA(i,j,k),K=1,3)
enddo
dndo
close (1)
fn2='c:\fortran\data\'/prefix//' .kns'
open (2,file=fn2)
do i=1,12
do j=1,n
   write (2,FMT=100) j,(omega(j,i,k),K=1,3)
& (ALPHA(j,i,k), k=1,3)

100 FORMAT(I3.1,',''F10.2,',''F10.2,',''F10.2,',''
     + '','F10.2,',''F10.2,',''F10.2)
endo
dndo
open (1, file='c:\fortran\data\'//prefix//'.kn2')
do i=1,n
   do j=1,12
      READ (1,*) joint,(OMEGA(I,J,K),K=1,3)
   endo
endo
close(1)
do i=1,12
do j=1,n
   write (2,fmt=200) j,(OMEGA(j,i,k),K=1,3)
200 FORMAT(I3.1,',''F10.2,',''F10.2,',''F10.2,',''
     + '','F10.2,',''F10.2,',''F10.2)
endo
dndo
close(2)
end

Subroutine FDS: Sorts forces and torques in distal segment coordinates into format usable with spreadsheet applications

C THIS PROGRAM SORTS THE DATA FROM THE *.ftd FILES AND
C ORDERS IT BY JOINT SO THAT IT CAN BE GRAPHED
C
subroutine fds(prefix,n)
REAL force(150,13,3),moment(150,13,3)
character*6 prefix
character*33 fn,fn2
fn='c:\fortran\data\'//prefix//'fds'
open (1, file=fn)
do i=1,n
read (1,*) frame,time
   do j=1,11
      read (1,*) joint,(force(i,j,k),K=1,3)
   endo
   read (1,*) (force(i,12,j),j=1,3),moment(i,12,3)
   read (1,*) (force(i,13,j),j=1,3),moment(i,13,3)
endo
   close (1)
fn2='c:\fortran\data\'//prefix//'fdd'
open (2, file=fn2)
do i=1,13
   do j=1,n
      write (2,fmt=100) j,(force(j,i,k),K=1,3)
   endo
100 FORMAT(I3.1,',''F10.2,',''F10.2,',''F10.2,',''
     + '','F10.2,',''F10.2,',''F10.2)
Subroutine MM: Multiplies two matrices together

C

subroutine mm(m1,m2,m3)
C
C This subroutine computes the product of matrices m1
C and m2 and returns the product in m3
C
real m1(3,3),m2(3,3),m3(3,3)
do 6000 i=1,3
do 6000 j=1,3
 m3(i,j)=m1(i,1)*m2(1,j)+m1(i,2)*m2(2,j)
 &   +m1(i,3)*m2(3,j)
6000 continue
return
end

Subroutine MATVEC: Multiplies a matrix and a vector together

subroutine matvec(r,v,product)
C
C This subroutine multiplies a 3x3 matrix r by a vector v
C
real r(3,3),v(3),product(3)
do i=1,3
   product(i)=0.
enddo
do i=1,3
   do j=1,3
      product(i)=product(i)+r(i,j)*v(j)
   enddo
endo
ternt
end
**APPENDIX F**

**THEORETICAL VALUES FOR MARKER LOCATIONS FROM HANAVAN MODEL USING THE SEGMENT CENTER OF MASS AS THE ORIGIN POINT**

<table>
<thead>
<tr>
<th>Segment</th>
<th>Marker Location</th>
<th>Formulas for Location in Global Coordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Left Shank</td>
<td>Left ankle</td>
<td>0</td>
</tr>
<tr>
<td>Left Shank</td>
<td>Left knee</td>
<td>0</td>
</tr>
<tr>
<td>Left Thigh</td>
<td>Left knee</td>
<td>0</td>
</tr>
<tr>
<td>Left Thigh</td>
<td>Left front thigh</td>
<td>$-(R_t-RR_t)/2$</td>
</tr>
<tr>
<td>Left Thigh</td>
<td>Left back thigh</td>
<td>$(R_t-RR_t)/2$</td>
</tr>
<tr>
<td>Left Thigh</td>
<td>Left trochanter</td>
<td>0</td>
</tr>
<tr>
<td>Right Shank</td>
<td>Right ankle</td>
<td>0</td>
</tr>
<tr>
<td>Right Shank</td>
<td>Right knee</td>
<td>0</td>
</tr>
<tr>
<td>Right thigh</td>
<td>Right knee</td>
<td>0</td>
</tr>
<tr>
<td>Right thigh</td>
<td>Right front thigh</td>
<td>$-(R_t-RR_t)/2$</td>
</tr>
<tr>
<td>Right thigh</td>
<td>Right back thigh</td>
<td>$(R_t-RR_t)/2$</td>
</tr>
<tr>
<td>Right thigh</td>
<td>Right trochanter</td>
<td>0</td>
</tr>
<tr>
<td>Pelvis</td>
<td>Left ASIS</td>
<td>$RR_p$</td>
</tr>
<tr>
<td>Pelvis</td>
<td>Right ASIS</td>
<td>$RR_p$</td>
</tr>
<tr>
<td>Pelvis</td>
<td>Left waist</td>
<td>$-.25*RR_p$</td>
</tr>
<tr>
<td>Pelvis</td>
<td>Right waist</td>
<td>$-.25*RR_p$</td>
</tr>
<tr>
<td>Mid trunk</td>
<td>Left waist</td>
<td>$-.25*RR_{mt}$</td>
</tr>
<tr>
<td>Mid trunk</td>
<td>Right waist</td>
<td>$-.25*RR_{mt}$</td>
</tr>
<tr>
<td>Mid trunk</td>
<td>Vertebrae</td>
<td>$-RR_{mt}$</td>
</tr>
<tr>
<td>Upper trunk</td>
<td>Substernal</td>
<td>$RR_{ut}$</td>
</tr>
<tr>
<td>Upper trunk</td>
<td>Sternum</td>
<td>$RR_{ut}$</td>
</tr>
<tr>
<td>Segment</td>
<td>Marker Location</td>
<td>Formulas for Location in Global Coordinates</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------</td>
<td>---------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td>Y</td>
</tr>
<tr>
<td>Upper trunk</td>
<td>Neck</td>
<td>-RR&lt;sub&gt;ut&lt;/sub&gt;</td>
</tr>
<tr>
<td>Left forearm</td>
<td>Left side wrist</td>
<td>0</td>
</tr>
<tr>
<td>Left forearm</td>
<td>Left elbow</td>
<td>0</td>
</tr>
<tr>
<td>Left upper arm</td>
<td>Left elbow</td>
<td>0</td>
</tr>
<tr>
<td>Left upper arm</td>
<td>Left back upper arm</td>
<td>((R_u-RR_u)/2)</td>
</tr>
<tr>
<td>Left upper arm</td>
<td>Left shoulder</td>
<td>0</td>
</tr>
<tr>
<td>Right forearm</td>
<td>Right side wrist</td>
<td>0</td>
</tr>
<tr>
<td>Right forearm</td>
<td>Right elbow</td>
<td>0</td>
</tr>
<tr>
<td>Right upper arm</td>
<td>Right elbow</td>
<td>0</td>
</tr>
<tr>
<td>Right upper arm</td>
<td>Right back upper arm</td>
<td>((R_u-RR_u)/2)</td>
</tr>
<tr>
<td>Right upper arm</td>
<td>Right shoulder</td>
<td>0</td>
</tr>
<tr>
<td>Head</td>
<td>Forehead</td>
<td>(R_h/4)</td>
</tr>
<tr>
<td>Head</td>
<td>Right ear</td>
<td>0</td>
</tr>
<tr>
<td>Head</td>
<td>Left ear</td>
<td>0</td>
</tr>
</tbody>
</table>

SYMBOLS: R=large radius of segment, RR=small radius, SL=segment length. Subscripts: f=forearm, u=upper arm, h=head, ut=upper trunk, m=mid trunk, p=pelvis, t=thigh, s=shank
APPENDIX G

COMPUTER PROGRAMS FOR SIMULATION

Program GYMASTDATA

PROGRAM GYMASTDATA

C
C THIS PROGRAM ALLOWS THE USER TO CHOOSE THE DATA TO BE
C USE IN THE SIMULATION.
C THE USER CAN SPECIFY A DATA SET TO BE USED AND THEN MAY
C ALTER THE FOLLOWING PARAMETERS:
C PROPORTIONALITY CONSTANT BETWEEN HAND FORCE AND
C TORQUE
C SPECIFY ALL JT ROTATIONS TO BE ZERO FOR ONE OR MORE
C JOINTS
C INCREASE OR DECREASE SPEED OF GIANT SWING
C INCREASE OR DECREASE RANGE OF MOTION AT A JT
C
C DECLARE VARIABLES
C
CHARACTER*25 DIR
CHARACTER*8 PREFIX
CHARACTER*14 JTNAME(ll)
CHARACTER*50 F1,F2
REAL PARAM(30),DT,DPS,A(240,11),B(240,11),G(240,11)
REAL A1(240),DUMMY(240),ANG(240)
REAL L,E,DIA,R,INERT
REAL DIST1,DIST2,T1,T2,T3,TD1,TD2,TD3,T(120),TT(240)
REAL MAX(11,3),MIN(11,3),WK(240),D(120),DVAL(240),KP
INTEGER YN,OPTION,YN1,N,FREL,RCF
lwk=240

C

print *," 
print *," 
print *,"Welcome to the RELEASE MOVE simulation program" 
print *,"This program can be run using default values 
& OR " 
print *,"the input data can be customized to perform a 
& number" 
print *,"of experiments" 
print *," 
print *," 
100 print *,"Do you want to run the default program (enter 
& 1)"
print *,"or do you want to run an experiment (enter 2)"
read *,yn
if ((yn.ne.1).and.(yn.ne.2)) then
goto 100
endif
if (yn.eq.1) then
dir='c:\fortran\data\'
prefix='subject2'
goto 2000
endif
print *,""
print *,"
111 print *,"The default directory for simulation data files & is"
print *,"C:\FORTRAN\DATA\"
print *,""
print *,"You may store the data files for your & experiment in"
print *,"the default directory or choose another & directory"
print *,"Do you want to use the default directory (enter & 1)"
print *,"or do you want to use another directory (enter & 2)"
read *,yn1
if (yn1.eq.1) then
dir='c:\fortran\data\'
elseif (yn1.eq.2) then
print *,"
print *,"Enter the directory name inside single quotes"
print *,"Enter the total length of the directory name must not & exceed"
print *,"25 characters"
else
goto 111
endif
print *,"
print *,"
print *,"Enter an 8 character identifier for the data & files"
print *,"Enter the name enclosed in single quotes"
read *,prefix
print *,"
print *,"
11 print *,"Which file do you want to use as the reference & file"
print *,"
print *,"1= 3 degrees of freedom, move not symmetric"
print *,"2= Move symmetric using right side data, 1 dof & in trunk"
print *,"3= Move symmetric using left side data, 1 dof & in trunk"
read *,number
if (number.eq.1) then
fl='c:\fortran\data\subject2.jvt'
f2='c:\fortran\data\subject2.cga'
endif
if (number.eq.2) then
fl='c:\fortran\data\sub2rss.jvt'
f2='c:\fortran\data\sub2rss.cga'
endif
if (number.eq.3) then
fl='c:\fortran\data\sub21ss.jvt'
f2='c:\fortran\data\sub21ss.cga'
endif
if ((number.lt.1).or.(number.gt.3)) then
print *, "Try again"
goto 11
endif

C
C ALLOW THE USER TO CHOOSE THE EXPERIMENTAL VARIABLES TO
C CHANGE
C
C print *, "There are several experimental variables that
C & you may "
print *, "alter in this simulation"
print *, "You may make changes in one or more of the
C & following"
print *, "categories"
print *, ""
print *, "> Change the anthropometry of the gymnast"
print *, "> Change the speed of the giant swing"
print *, "> Change the constant of proportionality
C & between"
print *, "the forces and moments at the hands"
print *, "> Eliminate some joint rotations"
print *, "> Increase or decrease the range of motion of a
C & joint"
print *, "> Redistribute the force between the two hands"
print *, "or"
print *, "> Change the release point"
print *, ""
print *, "Enter 1 to make changes, 0 to run as-is"
print *, "0 is only valid with a name that has already
C & been run"
print *, ""
print *, "Enter an option"
read *, option
if (option.eq.0) then
goto 2000
C
C ALLOW THE USER TO ALTER ANTHROPOMETRIC DATA
C
open (1, file='c:\fortran\data\subject2.dat')
do i=1,30
   read (1,4000) param(i)
enddo
close(1)
4000 format(f8.2)
print *, " "
print *, "The following are the anthropometric data for"
print *, "the default gymnast"
200 print *, "1=Head circ.: ",param(1)," 2=Chest & cir.
 & circ.: ",param(2)
print*, "3=axillary circ.:",param(3)," 4=elbow & cir.:",param(4)
print*,"5=forearm circ.:",param(5)," 6=wrist & cir.":",param(6)
print*,"7=thigh circ.:",param(7)," 8=knee & cir.":",param(8)
print*,"9=calf circ.:",param(9)," 10=forearm & len.":",param(10)
print*, "11=wrist breadth:" ,param(11)," 12=hand & br." ,param(12)
print*,"13=stature: ",param(13)," 14=shoulder & ht." ,param(14)
print*, "15=substernal ht." ,param(15)," 16=troch. & ht" ,param(16)
print*, "17=sitting ht." ,param(17)," 18=waist & height" ,param(18)
print*, "19=ankle circ." ,param(19)," 20=iliac & fat":" ,param(20)
print*, "21=chest br." ,param(21)," 22=chest & depth" ,param(22)
print*, "23=waist depth",param(23)," 24=hip & breadth",param(24)
print*, "25=buttock depth",param(25)," 26=upper arm & len." ,param(26)
print*, "27=tibiale ht." ,param(27)," 28=foot & len." ,param(28)
print*, "29=weight",param(29)," 30=sphyrion & ht" ,param(30)
print*, " "
print*, "All meas. in cm, except weight in kg."
print *, " "
print*, "Enter the # of the parameter you want to change"
print *, " "
print*, "0=go on to the next menu"
read *,n
if((n.gt.0).and.(n.le.30))then
print *,""
print *,"new value for parameter:"
read *,param(n)
goto 200
elseif(n.eq.0) then
goto 300
else
print *,"NOT A VALID RESPONSE!!!"
GOTO 200
endif
continue
open (1,file=dir(1:len_trim(dir))/prefix/'' .dat')
do i=1,30
  write (1,4000) param(i)
enddo

C ALLOW THE USER TO CHANGE THE SPEED OF THE GIANT
C THIS OPTION CHANGES THE RATE OF JOINT MOTION THROUGHOUT
C THE MOVE PROPORTIONATELY

print *,""
print *,"The average angular speed of the default giant & swing is"
print*,"approximately 210 degrees per second (3.665 & rad/sec)"
print *,""
print *,"Do you want to change the speed of the move?"
print *,"1=Yes, 2=No"
read *,ynl
if ((ynl.ne.1).and.(ynl.ne.2))then
goto 400
elseif (ynl.eq.1) then
print *,""
print *,"Enter the new average angular speed in degrees & per sec"
print *,"(Maximum allowed value=420 dps)"
read *,dps
C ALTER TIME INTERVAL BY DPS/210
C
dt=210./dps*.02
C TAKE THE JOINT ANGLE VS. TIME DATA AND CG ANGLE VS. TIME DATA
C AND INTERPOLATE TO GET DATA POINTS EVERY .02 SECONDS
C COMPUTE THE NUMBER OF FRAMES IN THE NEW MOVE
open (1, file=f1)
do i=1,119
   do j=1,11
      print *,i,j
      read (1,*)a(i,j)
      read (1,*)b(i,j)
      read (1,*)g(i,j)
   enddo
endo
close(1)
do i=1,119
   t(i)=i*dt
endo
nf=int(119.*dt/
0.02)
do i=1,nf
   tt(i)=i*.02
endo
do j=1,11
do i=1,119
   al(i)=a(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   a(i,j)=dummy(i)
endo
do i=1,119
   al(i)=b(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   b(i,j)=dummy(i)
endo
do i=1,119
   al(i)=g(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   g(i,j)=dummy(i)
endo
do i=1,119
   al(i)=a(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   g(i,j)=dummy(i)
endo
do i=1,119
   al(i)=b(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   g(i,j)=dummy(i)
endo
do i=1,119
   al(i)=g(i,j)
endo
call pchez(119,t,al,d,.true.,wk,lwk,ierr)
call pchev(119,t,al,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   g(i,j)=dummy(i)
endo
do i=1,119
   al(i)=a(i,j)
endo
open (1, file=dir(1:len_trim(dir))/prefix//'.jvt')
do i=1,nf
   do j=1,11
      write(1,*)a(i,j)
      write(1,*)b(i,j)
      write(1,*)g(i,j)
enddo
endif
enddo

close(1)
open(1,file=f2)
read(1,*) (ang(j),j=1,119)
close(l)
call pchez(119,t,ang,d,.true.,wk,lwk,ierr)
call pchev(119,t,ang,d,nf,tt,dummy,dval,ierr)
do i=1,nf
   ang(i)=dummy(i)
enddo
open(1,file=dir(l:len_trim(dir))//prefix//' .cga')
do j=1,nf
   write(1,*) ang(j)
enddo
close(1)
elseif (yn1.eq.2) then
   nf=119
open (1, file=f1)
do i=1,119
   do j=1,11
      read (1,*) a(i,j)
      read (1,*) b(i,j)
      read (1,*) g(i,j)
   enddo
enddo
close(1)
open(1,file=f2)
read(1,*) (ang(j),j=1,119)
close(1)
open (1, file=dir(l:len_trim(dir))//prefix//' .jvt')
do i=1,119
   do j=1,11
      write(1,*) a(i,j)
      write(1,*) b(i,j)
      write(1,*) g(i,j)
   enddo
enddo
close(1)
open(1, file=dir(l:len_trim(dir))//prefix//' .cga')
do i=1,nf
   write(1,*) ang(i)
enddo
close(1)
endif
fl=dir(l:len_trim(dir))//prefix//' .jvt'
f2=dir(l:len_trim(dir))//prefix//' .cga'
C
C ALLOW USER TO ALTER THE PROPORTIONALITY CONSTANT KPRIME
The default value of the constant relating the hand forces and moments is .03 Newton-meters per Newton.

Do you want to change this value?

Enter new value

kp=kp*100.
else
kp=3.
endif

ALLOW USER TO FIX JOINTS

In the default model, motion is allowed at the following joints:

Do you want to disallow motion at any of these joints?

How many joints do you want fixed?

Enter joint number you want fixed:
enddo
enddo
open (1, file=dir(1: len_trim(dir))/prefix/'. jvt')
do i=1, ll9
   do j=1, 11
      write(1,*) a(i, j)
      write(1,*) b(i, j)
      write(1,*) g(i, j)
   enddo
enddo
close(1)
endif

C
C ALLOW THE USER TO INCREASE OR DECREASE THE RANGE OF
C MOTION OF A JOINT
C
print *, " 
print *, " 
print *, "Would you like to modify the range of motion of one 
print *, "or more joints?"
print *, "1=Yes 2=No"
read *, ynl
if ((ynl.ne.1).and.(ynl.ne.2)) then
goto 700
elseif (ynl.eq.1) then
do j=1, 11
   do k=1, 3
      max(j, k)=0.
      min(j, k)=5000.
   enddo
endo
do i=1, nf
   do j=1, 11
      if(a(i, j).gt.max(j, 1)) then
         max(j, 1)=a(i, j)
      endif
      if(a(i, j).lt.min(j, 1)) then
         min(j, 1)=a(i, j)
      endif
      if(b(i, j).gt.max(j, 2)) then
         max(j, 2)=b(i, j)
      endif
      if(b(i, j).lt.min(j, 2)) then
         min(j, 2)=b(i, j)
      endif
      if(g(i, j).gt.max(j, 3)) then
         max(j, 3)=g(i, j)
      endif
      if(g(i, j).lt.min(j, 3)) then

\[ \min(j, 3) = \text{g}(i, j) \]

end if

end do

end do

\text{jtname}(1) = 'Neck'
\text{jtname}(2) = 'Right shoulder'
\text{jtname}(3) = 'Left shoulder'
\text{jtname}(4) = 'Right elbow'
\text{jtname}(5) = 'Left elbow'
\text{jtname}(6) = 'Upper back'
\text{jtname}(7) = 'Lower back'
\text{jtname}(8) = 'Right hip'
\text{jtname}(9) = 'Left hip'
\text{jtname}(10) = 'Right knee'
\text{jtname}(11) = 'Left knee'

print *, " "
print *, "The range of motion of the joints are"

\textbf{do i=1,11}

\textcolor{red}{\textbf{800}} print *, \text{jtname}(i)

\textbf{print *, "1-Flexion/Extension: \textbf{\text{max(i,1)}-\text{min(i,1)}}"}
\textbf{print *, "2-Lateral flexion: \textbf{\text{max(i,2)}-\text{min(i,2)}}"}
\textbf{print *, "3-Rotation: \textbf{\text{max(i,3)}-\text{min(i,3)}}"}

\textbf{print *, "If you want to change one of these ranges, enter the"}

\textbf{print *, "number of the rotation (0=no changes)"}

\textbf{print *, " "}

\textbf{read *, achg}

if (achg.eq.1) then

print *, "New range: "
read *, r1
\textbf{do j=1,nf}

\textcolor{red}{\textbf{a}(j,i) = a(j,i)*r1/\textbf{(max(i,1)-min(i,1))}}

enddo

elseif (achg.eq.2) then

print *, "New range: "
read *, r1
\textbf{do j=1,nf}

\textcolor{red}{\textbf{b}(j,i) = b(j,i)*r1/\textbf{(max(i,2)-min(i,2))}}

enddo

elseif (achg.eq.3) then

print *, "New range: "
read *, r1
\textbf{do j=1,nf}

\textcolor{red}{\textbf{g}(j,i) = g(j,i)*r1/\textbf{(max(i,3)-min(i,3))}}

enddo

elseif (achg.ne.0) then

print *, "NOT A VALID RESPONSE"
goto 800

endif
enddo
open (1, file=dir(1: len_trim(dir)) // prefix /// .jvt')
do i=1,119
   do j=1,11
      write (1,*), a(i,j)
      write (1,*), b(i,j)
      write (1,*), g(i,j)
   enddo
enddo
close (1)
endif
l=240
r=1.4
E=2e7

   C
   C ALLOW THE USER TO MODIFY THE RELEASE POINT
   C
   frel=75./119.*nf
   print *," "
   print *,"In the default giant swing, the release occurs & at"
   print *,"a point 61% of the way through the move"
   print *,"This will be at frame #",frel,"in your & simulation"
   print *,"You may change this frame number if you wish"
   print *,"1=Change it, 2=Keep the same release"
1200 read *,ynl
   if (ynl.eq.l) then
      print *,"Release frame"
      read *,frel
   elseif (ynl.ne.2) then
      print *,"NOT a VALID RESPONSE"
go to 1200
   endif
     C
     C ALLOW THE USER TO DIVIDE THE FORCES ASYMMETRICALLY B/W HANDS
     C
     print *," "
     print *,"In the default simulation, the hand forces are "
     print *,"divided equally between the hands"
     print *,"Do you want to change it for this simulation?"
800 print *,"1=yes, 2=no"
     read *,ynl
     if (ynl.eq.1) then
        print *," "
        print *," "

print *,"Enter the proportion of the force that the hand will take"
read *,div
if (div.gt.1.)then
print *,"That is not a reasonable response"
print *,"Please enter a number between 0 and 1"
go to 1250
endif
else
div=.5
endif
dist1=80.
dist2=160.
WRITE INITIAL CONDITIONS TO FILE
rcf=104./119.*nf
t1=-.35
t2=0.
t3=0.
td1=-.82
td2=0.
td3=0.
open (1,file=dir(1:trim(dir))//prefix//'.ic')
write(1,*)nf
write(1,*)dist1
write(1,*)dist2
write(1,*)t1
write(1,*)t2
write(1,*)t3
write(1,*)td1
write(1,*)td2
write(1,*)td3
write(1,*)frel
write(1,*)rcf
write(1,*)div
close(1)
INTERPOLATE JOINT ANGLE VS. TIME FILE AND STORE IN *.JT2
print *,"WRITING DATA TO FILES...."
call angint(prefix,dir,nf)
close(1)
return
end

**Subroutine ANGINT**

```fortran
SUBROUTINE ANGINT(PREFIX, DIR, NF)
REAL A(ll, 240), B(11, 240), G(11, 240)
REAL F(240)
REAL AL(ll), ALD(ll), ALDD(ll), BL(ll), BLD(ll), BLDD(ll)
REAL GL(ll), GLD(ll), GLDD(ll)
CHARACTER*50, FN1, FN2, FN
CHARACTER*25 DIR
CHARACTER*8 PREFIX
CHARACTER*2 NJT(ll), NJT1
njt(1)= ‘1’
njt(2)= ‘2’
njt(3)= ‘3’
njt(4)= ‘4’
njt(5)= ‘5’
njt(6)= ‘6’
njt(7)= ‘7’
njt(8)= ‘8’
njt(9)= ‘9’
njt(10)= ‘10’
njt(ll)= ‘11’
fn1=dir(1: len_trim(dir))//prefix/’ . jvt’
fn2=dir(1: len_trim(dir))//prefix/’ . jt2’
open (15, file=fn1)
do j=1, NF
  do i=1, ll
    read(15, *) a(i, j)
    read(15, *) b(i, j)
    read(15, *) g(i, j)
  enddo
enddo
close(15)
do i=1, ll
  print *, “Interpolating angles for joint number ”, njt(i)
  njt1=njt(i)
  open(20, file=’c:\fortran\jt’//njt1(1: len_trim(njt1))
  & //’ . dat’)
do j=1, NF
    f(j)=a(i, j)
  enddo
fn=’c:\fortran\dummy1.for’
call interpolatea(f, fn, nf, 2)
do j=1, NF
    f(j)=b(i, j)
  enddo
fn=’c:\fortran\dummy2.for’
```

```fortran
```
call interpolatea(f,fn,nf,2)
do j=1,nf
   f(j)=g(i,j)
enddo
fn='c:\fortran\dummy3.for'
call interpolatea(f,fn,nf,2)
open(21,file='c:\fortran\dummy1.for')
open(22,file='c:\fortran\dummy2.for')
open(23,file='c:\fortran\dummy3.for')
do k=1,nf*20
   read(21,*)ai
   read(21,*)adi
   read(21,*)addi
   read(22,*)bi
   read(22,*)bdi
   read(22,*)baddi
   read(23,*)gi
   read(23,*)gdi
   read(23,*)gaddi
   write(20,*)ai,adi,addi
   write(20,*)bi,bdi,baddi
   write(20,*)gi,gdi,gaddi
enddo
close(20)
close(21)
close(22)
close(23)
enddo
open (20,file=fn2)
open(21,file='c:\fortran\jt1.dat')
open(22,file='c:\fortran\jt2.dat')
open(23,file='c:\fortran\jt3.dat')
open(24,file='c:\fortran\jt4.dat')
open(25,file='c:\fortran\jt5.dat')
open(26,file='c:\fortran\jt6.dat')
open(27,file='c:\fortran\jt7.dat')
open(28,file='c:\fortran\jt8.dat')
open(29,file='c:\fortran\jt9.dat')
open(30,file='c:\fortran\jt10.dat')
open(31,file='c:\fortran\jt11.dat')
do i=1,nf*20
   read (21,*)al(1),ald(1),aldd(1)
   read(21,*)bl(1),bld(1),bldd(1)
   read (21,*)gl(1),gld(1),gldd(1)
   read (22,*)al(2),ald(2),aldd(2)
   read(22,*)bl(2),bld(2),bldd(2)
   read (22,*)gl(2),gld(2),gldd(2)
   read (23,*)al(3),ald(3),aldd(3)
   read(23,*)bl(3),bld(3),bldd(3)
   read (23,*)gl(3),gld(3),gldd(3)
read (24,*) al(4),ald(4),aldd(4)
read (24,*) bl(4),bld(4),bldd(4)
read (24,*) gl(4),gld(4),gldd(4)
read (25,*) al(5),ald(5),aldd(5)
read (25,*) bl(5),bld(5),bldd(5)
read (25,*) gl(5),gld(5),gldd(5)
read (26,*) al(6),ald(6),aldd(6)
read (26,*) bl(6),bld(6),bldd(6)
read (26,*) gl(6),gld(6),gldd(6)
read (27,*) al(7),ald(7),aldd(7)
read (27,*) bl(7),bld(7),bldd(7)
read (27,*) gl(7),gld(7),gldd(7)
read (28,*) al(8),ald(8),aldd(8)
read (28,*) bl(8),bld(8),bldd(8)
read (28,*) gl(8),gld(8),gldd(8)
read (29,*) al(9),ald(9),aldd(9)
read (29,*) bl(9),bld(9),bldd(9)
read (29,*) gl(9),gld(9),gldd(9)
read (30,*) al(10),ald(10),aldd(10)
read (30,*) bl(10),bld(10),bldd(10)
read (30,*) gl(10),gld(10),gldd(10)
read (31,*) al(11),ald(11),aldd(11)
read (31,*) bl(11),bld(11),bldd(11)
read (31,*) gl(11),gld(11),gldd(11)
do j=1,11
write (20,*) al(j),ald(j),aldd(j)
write (20,*) bl(j),bld(j),bldd(j)
write (20,*) gl(j),gld(j),gldd(j)
endo
do
close (20)
close (21)
close (22)
close (23)
close (24)
close (25)
close (26)
close (27)
close (28)
close (29)
close (30)
close (31)
return
end

Program SIMULATE
PROGRAM SIMULATE
REAL IXX(12),IYY(12),IZZ(12),IDY(12,3,3),M(12),MTOT
REAL H(12),ETA(12),R(12),RR(12),ROT(12,3,3),RELM(11,3,3)
REAL ALPHA(11), ALPHADOT(11), ALPHADDOT(11), BETA(11)
& , BETADOT(11)
REAL BETADDOT(11), GAMMA(11), GAMMADOT(11), GAMMADDOT(11)
REAL PG(3), P(12, 3), Q(12, 3), QDOT(12, 3), QDDOT(12, 3), H1(3)
REAL U(12, 3), UPRIME(12, 3), ANG(12, 3), OMEGA(12, 3), RW(12, 3)
REAL X(3), XDD, THETA(3), THETADOT(3), U1(3), U1DOT(3)
& , THETADDOT(3)
REAL
ACC(12, 3), FOR(13, 3), TOR(13, 3), MARK(22, 3), KP, CGANG(270)
REAL RH(270, 3), LH(270, 3), CGR(270), RBAR(270), TBAR(270)
REAL L, E, INERT, IC(16), PRH, PLH, ZH(2), FTOT(270, 2), LRH
REAL CGRAD, FTOTAL(2), ANGLE, XX(170, 3)
REAL GLH(3), GRH(3), LEFT(3), RIGHT(3), THETALAST(3)
REAL A(12, 3), B(12, 3), C(12), D(12), W, QQ, R1, S, T1, V, DELTA, GG
REAL XDOTREL(2), Y(6), FL, FR, RMATRIX(3, 3), AA(3, 3), VV(3, 3)
REAL ADUMMY(170, 3)
INTEGER NFRAMES, NRELEASE, RCF, N, RELF, RECF, FRAME, PIFLAG(3)
CHARACTER*25 DIR
CHARACTER*8 PREFIX
CHARACTER*50 FN, FN2
COMMON/COEFFS/W(3, 5), DELTA(3, 8), GG(4), QQ(6), R1(3), S(3),
& T1(3), V(3), PLH(3), PRH(3), KP, MTOT, FL(3), FR(3), XDD(3)
COMMON/CF2/LH, RH, FTOT, ZH, CGANG
COMMON/KINDAT/THETA, U1, U1DOT, RW, ROT, A, B, C, D, Q, QDOT, QDDOT,
& ANG, OMEGA, ACC
COMMON/ANGLES/ALPHA, BETA, GAMMA, ALPHADOT, BETADOT, GAMMADOT,
& ALPHADDOT, BETADDOT, GAMMADDOT, RELRM, U, UPRIME

SET ALL VARIABLES EQUAL TO ZERO

DATA IXX/12*0/, IYY/12*0/, IZZ/12*0/, IDY/108*0/
DATA M/12*0/, MTOT/0/, H/12*0/, ETA/12*0/, R/12*0/
DATA RR/12*0/, ROT/108*0/, RELRM/99*0/
DATA ALPHA/11*0/, ALPHADOT/11*0/, ALPHADDOT/11*0/
DATA BETA/11*0/, BETADOT/11*0/, BETADDOT/11*0/
DATA GAMMA/11*0/, GAMMADOT/11*0/, GAMMADDOT/11*0/
DATA
PG/3*0/, P/36*0/, Q/36*0/, QDOT/36*0/, QDDOT/36*0/, H1/3*0/
DATA U/36*0/, UPRIME/36*0/, ANG/36*0/, OMEGA/36*0/, RW/36*0/
DATA X/3*0/, XDD/3*0/, THETA/3*0/, THETADOT/3*0/, U1/3*0/
& , U1DOT/3*0/
DATA THETADDOT/3*0/, ACC/36*0/
DATA FOR/39*0/, TOR/39*0/, MARK/66*0/, KP/0/, CGANG/270*0/
DATA
RH/810*0/, LH/810*0/, CGR/270*0/, RBAR/270*0/, TBAR/270*0/
DATA L, E, INERT/3*0/, IC/16*0/, PRH, PLH/6*0/, ZH/2*0/
& , FTOT/540*0/
DATA LRH/0/, CGRAD/0/, FTOTAL/2*0/, ANGLE/0/, XX/510*0/
DATA
GLH/3*0/,GRH/3*0/,LEFT/3*0/,RIGHT/3*0/,THETALAST/3*0/
DATA A/36*0/,B/12*0/,C/12*0/,D/12*0/
DATA W,QQ,R1,S,T1,V,DELTA,GG/61*0/
DATA XDOTREL/2*0/,X/6*0/,FL,FR/6*0/,RMATRIX/9*0/,AA/9*0/ & VV/9*0/
DATA NFRAMES,NRELEASE,RCF,N,REL,F,RECF,FRAME/7*0/
DATA (PIFLAG(I),I=1,3)/3*0/
DATA OFFSET,DIST1,DIST2/3*0/

C
open (1,file='c:\fortran\data\filename.dat')
read (1,*) dir
read (1,*) prefix
print *,dir,prefix
close(1)

C
FN1=DIR(1:LEN_TRIM(DIR))//PREFIX//' .DAT'
!DATA FILE WITH ANTHROPOMETRIC INFO.
C FN2=DIR(1:LEN_TRIM(DIR))//PREFIX//' .IC'
!DATA FILE WITH INITIAL CONDITIONS
C FN3=DIR(1:LEN_TRIM(DIR))//PREFIX//' .JVT'
!DATA FILE WITH JOINT ANGLES VS. TIME
C FN4=DIR(1:LEN_TRIM(DIR))//PREFIX//' .QDT'
!DATA FILE WITH Q'S VERSUS TIME
C FN5=DIR(1:LEN_TRIM(DIR))//PREFIX//' .K1'
!CALCULATED IN QDOT
C FN6=DIR(1:LEN_TRIM(DIR))//PREFIX//' .K2'
!OUTPUT FILE FOR KINEMATIC DATA
C FN7=DIR(1:LEN_TRIM(DIR))//PREFIX//' .MKS'
!OUTPUT FILE FOR KINETIC DATA
C FN8=DIR(1:LEN_TRIM(DIR))//PREFIX//' .XY2'
!OUTPUT FILE WITH CG XY COORDINATES VS TIME
C FN9=DIR(1:LEN_TRIM(DIR))//PREFIX//' .THR'
!CALCULATED IN THIS PROGRAM
C FN10=DIR(1:LEN_TRIM(DIR))//PREFIX//' .CGA'
!DATA FILE WITH CG ANG VS TIME, RECATCH PHASE
C FN11=DIR(1:LEN_TRIM(DIR))//PREFIX//' .JT2'
!DATA FILE FOR INTERPOLATED JOINT ANGLES
C FN12=DIR(1:LEN_TRIM(DIR))//PREFIX//' .QD2'
!DATA FILE FOR INTERPOLATED Q'S
C FN17=DIR(1:LEN_TRIM(DIR))//PREFIX//' .FT2'
!DATA FILE WITH INTERPOLATED TOTAL FORCE
C
fn=dir(1:len_tr(im(dir))//prefix//' .Dat'
call inertia(m,ixx,iyy,izz,r,rr,h,eta,fn)
C
CALCULATE TOTAL BODY MASS
mtot=0.
Do i=1,12
   mtot=mtot+m(i)
enddo

C READ IN INITIAL CONDITIONS AND ASSIGN TO VARIABLES
C
open (2,file=dir(1:len_trim(dir))//prefix//' .ic')
Read(2,*) (ic(j),j=1,16)
close(2)
nframes=int(ic(1))*20
dist1=ic(2)
dist2=ic(3)
l=ic(4)
offset=1/2.-(dist1+dist2)/2.
E=ic(5)*100. !ADJUST UNITS TO GET RESULTS IN KG AND CM
!则 UNITS FOR E ARE KG-CM/S/S/CM/CM
inert=ic(6)
kp=ic(7)
theta(1)=ic(8)
theta(2)=ic(9)
theta(3)=ic(10)
thetadot(1)=ic(11)
thetadot(2)=ic(12)
thetadot(3)=ic(13)
nrelease=int(ic(14))*20
rcf=int(ic(15))*20
pct=ic(16)

C PART 2: CALCULATE THE SEGMENT CG VECTORS AND
DERIVATIVES
C IN REFERENCE SEGMENT COORDINATES—WRITE TO FN4
C
nf=nframes/20
print *,"number of frames",nf
fn=dir(1:len_trim(dir))//prefix//' .jvt'
fn2=dir(1:len_trim(dir))//prefix//' .qdt'
call qdots(fn,fn2,eta,h,r,nf,m)

C IN ORDER TO DO NUMERICAL INTEGRATION, NEED TO
INTERPOLATE
C DATA FOR SMALLER INTEGRATION STEPS. THIS SUBROUTINE
C INTERPOLATES THE Q'S SO THAT THERE IS A DATA POINT EVERY
C .001 SECONDS AND WRITES THE DATA TO FN12
C
print *,"Interpolating cg vectors, please wait...."
fn=dir(1:len_trim(dir))//prefix//' .qd2'
call interpolateq(fn2,fn,nf)
nframes=nf*80
PART 3A: GIANT SWING PHASE
FROM DATA FILE FN10, READ IN ANGLE OF CG VECTOR VS TIME WRT THE ORIGIN OF THE GLOBAL COORDINATE SYSTEM

C open(10, file=dir(1: len_trim(dir)) // prefix // '.cga')
N=nframes/20
read (10,*)(cgang(i), i=1, n)
close(10)

C GET RIGHT AND LEFT HAND POSITION VECTORS WITH RESPECT TO CG OF BODY IN SEGMENT 1 COORDINATES

n=nrelease/20
open(3, file=dir(1: len_trim(dir)) // prefix // '.jvt')
Do i=1, n
   do j=1, 11
      read(3,*)alpha(j)
      read(3,*)beta(j)
      read(3,*)gamma(j)
   enddo
   call rotation(alpha, beta, gamma, relrm)
   call rm(relrm, rot)
   call cg(r, eta, rot, m, q, h, pg, p)
   call handloc(plh, prh, p, rot, eta, h)
   do j=1, 3
      rh(i,j)=prh(j)
      lh(i,j)=plh(j)
   enddo
endo
close(3)
zh(1)=prh(3)+offset
zh(2)=plh(3)+offset

C GET THE RADIUS (DISTANCE FROM GLOBAL ORIGIN) TO CG OF BODY AT EACH FRAME (.02 SECONDS APART)

n=nrelease/20
Call getr(cgr,n,mtot,l,e,inert,rbar,tbar)

C COMPUTE THE X-Y COORDINATES OF THE CENTER OF GRAVITY (Z IS ASSUMED TO BE CONSTANT)

do i=1, n
   xx(i,1)=cgr(i)*cos(cgang(i))
   xx(i,2)=cgr(i)*sin(cgang(i))
   xx(i,3)=(zh(1)+zh(2))/2.
endo
USE THE INTERPOLATION ROUTINES TO COMPUTE X-Y COORDINATES AND THEIR DERIVATIVES EVERY .001 SECONDS, STORE IN FILES FOR READING IN LATER

```fortran
fn=dir(1:len_trim(dir))//prefix//' .xy2'
call interpolate!(xx,fn,n,xdotrel)
call filterl(fn,nrelease)
fn=dir(1:len_trim(dir))//prefix//' .ft2'
call interpolateb(ftot,fn,n)
call filterb(fn,nrelease)
close(3)
```

GIVEN R AND THETA AND BAR DEFLECTIONS, USE NUMERICAL METHODS TO SOLVE FOR SEGMENT 1 ROTATIONS IN GLOBAL COORDINATES USE THESE AND EQUATIONS DEVELOPED IN APPENDIX A TO CALCULATE KINEMATICS AND KINETICS DURING THE GIANT SWING

```fortran
open(5,file=dir(1:len_trim(dir))//prefix//' .mat')
!OUTPUT FOR KINEMATICS
open(6,file=dir(1:len_trim(dir))//prefix//' .tht')
!OUTPUT FOR KINETICS
open(7,file=dir(1:len_trim(dir))//prefix//' .mks')
!OUTPUT FOR ANIMATION DATA
open(8,file=dir(1:len_trim(dir))//prefix//' .xy2')
!INTERPOLATED XY AND 2ND DERIVATIVES
open(11,file=dir(1:len_trim(dir))//prefix//' .jt2')
!INTERPOLATED JOINT ANGLES
open(12,file=dir(1:len_trim(dir))//prefix//' .qd2')
!INTERPOLATED Q'S
open(17,file=dir(1:len_trim(dir))//prefix//' .ft2')
!INTERPOLATED FTOT
piflag=0
do frame=1,nrelease
  do i=1,12
    do j=1,3
      read(12,*)q(i,j),qdot(i,j),qddot(i,j)
    enddo
  enddo
  do i=1,11
    read(11,*)alpha(i),alphadot(i),alphaddot(i)
    read(11,*)beta(i),betadot(i),betaddot(i)
    read(11,*)gamma(i),gammadot(i),gammaddot(i)
  enddo
```

READ IN DATA FOR EACH FRAME
do i=1,3
read (8,*)x(i),xdd(i)
enddo
read(17,*)ftotal(1),ftotal(2)
xdd(3)=0.
If (mod(frame,20).Eq.0) Then
C
C SET UP EQUATIONS FOR USING SPOOR AND VELDPAUS METHODS TO
C COMPUTE THE ROTATION MATRIX BETWEEN SEGMENT 1 AND THE
C GLOBAL COORDINATES
C
C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN
ADJACENT
C SEGMENTS
C
call rotation(alpha,beta,gamma,relrm)
C
C CALCULATE THE ROTATION MATRIX OF EACH SEGMENT WITH
RESPECT
C TO THE REFERENCE SEGMENT
C
call rm(relrm,rot)
C
C LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE
SEGMENTS
C
call cg(r,eta,rot,m,q,h,pg,p)
call handloc(plh,prh,p,rot,eta,h)
C
C CALL ROT1 AND CALCULATE THE ROTATION MATRIX BETWEEN
GLOBAL
C AND S1 COORDINATES
C
do i=1,3
aa(1,i)=0.0
Aa(2,i)=plh(i)
aa(3,i)=prh(i)
vv(1,i)=x(i)
enddo
vv(2,1)=rbar(frame/20)*cos(tbar(frame/20))
vv(2,2)=rbar(frame/20)*sin(tbar(frame/20))
vv(2,3)=zh(2)
vv(3,1)=vv(2,1)
vv(3,2)=vv(2,2)
vv(3,3)=zh(1)
call rot1(aa,vv,rmatrix)
C
C CALCULATE THE ROTATION ANGLES FROM THE ROTATION MATRIX
C
pi=acos(-1.0)
theta(1) = atan2(-rmatrix(1,2), rmatrix(2,2))
theta(2) = asin(rmatrix(3,2))
theta(3) = atan2(-rmatrix(3,1), rmatrix(3,3))
if ((rmatrix(3,3).gt.0.).and.(Cos(theta(3)).lt.0.)) then
  theta(2) = Pi - theta(2)
elseif ((rmatrix(3,3).lt.0.).and.(Cos(theta(3)).gt.0.)) then
  theta(2) = Pi - theta(2)
endif

do i = l, 3
  if (theta(i) - thetalast(i).ge.6.) then
    piflag(i) = piflag(i) - 1
  endif
  if (theta(i) - thetalast(i).le.-6) then
    piflag(i) = piflag(i) + 1
  endif
  thetalast(i) = theta(i)
  theta(i) = theta(i) + 2*Pi*piflag(i)
endo

do i = l, 3
  write(5,*) (rmatrix(i,j), j = l, 3)
endo

write(6,*) frame, theta(1), theta(2), theta(3)

C
C   COMPUTE THE 3D POSITIONS OF LANDMARKS FOR ANIMATION
C
C   call landmark(h, eta, r, x, theta, p, rot, mark)
C
C   DRAW UPDATED FIGURE
C
C   call animate(mark)
C   do i = 1, 22
C       write(7,*) i, (mark(i,j), j = 1, 3)
C   enddo
C   endif
C   enddo
close(5)
close(6)
close(8)
close(11)
close(12)
close(17)
print *, "  
Print *, "Release frame"
print *, "please wait... computing kinematics and kinetics"
print *, "for giant swing phase"
print *, "...."
C
C   FROM THETAS, FILTER AND DIFFERENTIATE TO GET THE
C DERIVATIVES OF THE THETAS

fn=dir(1:len_trim(dir))/prefix//'.tht'
nf=nrelease/20
Call filtert(fn,nf)
call intt(fn,nf)

C LOOP TO FIND THE KINEMATICS AND KINETICS OF EACH SEGMENT/JOINT DURING THE GIANT SWING
open(5,file=dir(1:len_trim(dir))/prefix//'.K1'
!OUTPUT FOR KINEMATICS
open(6,file=dir(1:len_trim(dir))/prefix//'.K2'
!OUTPUT FOR KINETICS
open(8,file=dir(1:len_trim(dir))/prefix//'.xy2'
!INTERPOLATED XY AND 2ND DERIVATIVES
open(11,file=dir(1:len_trim(dir))/prefix//'.jt2'
!INTERPOLATED JOINT ANGLES
open(12,file=dir(1:len_trim(dir))/prefix//'.q2'
!INTERPOLATED Q'S
open(13,file=dir(1:len_trim(dir))/prefix//'.tht'
!INTERPOLATED THETAS
open(17,file=dir(1:len_trim(dir))/prefix//'.ft2'
!INTERPOLATED FTOT

do frame=1,nf

C C READ IN DATA FOR EACH FRAME

do icount=1,20
  do i=1,12
    do j=1,3
      read(12,*) q(i,j),qdot(i,j),qddot(i,j)
    enddo
  enddo
enddo

read (11,*) alpha(i),alphadot(i),alphaddot(i)
read (11,*) beta(i),betadot(i),betaddot(i)
read (11,*) gamma(i),gammadot(i),gammaddot(i)

endo
do i=1,3
read (8,*)x(i),xdd(i)
endo
read(17,*)ftotal(1),ftotal(2)
endo
do i=1,3
read(13,*)theta(i),thetadot(i),thetaddot(i)
endo

ul(1)=thetadot(2)*cos(theta(3))-thetadot(1)*cos(theta(2))
& \* \sin(\theta(3))
\begin{align*}
u_1(2) &= \dot{\theta}(1) \sin(\theta(2)) + \dot{\theta}(3) \\
u_1(3) &= \dot{\theta}(1) \cos(\theta(2)) \cos(\theta(3)) + \dot{\theta}(2) \sin(\theta(3))
\end{align*}

& + \dot{\theta}(2) \sin(\theta(3)) - \ddot{\theta}(1) \cos(\theta(2)) \sin(\theta(3)) + \dot{\theta}(1) \dot{\theta}(2) \sin(\theta(2)) \sin(\theta(3)) - \dot{\theta}(1) \dot{\theta}(3) \cos(\theta(2)) \cos(\theta(3)) + \ddot{\theta}(2) \sin(\theta(3)) + \dot{\theta}(2) \dot{\theta}(3) \cos(\theta(3))

C C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN ADJACENT SEGMENTS
C call rotation(alpha,beta,gamma,relm)
C C CALCULATE THE ROTATION MATRIX OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT
C call rt(rm,relm,rot)
C C LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE SEGMENTS
C call cg(r,eta,rot,m,q,h,pg,p)
call handloc(plh,prh,p,rot,eta,h)
C C CALCULATE THE TOTAL APPLIED MOMENT
C momtot=kp*sqrt(ftotal(1)**2+ftotal(2)**2)
C C CALCULATE ANGULAR VELOCITY OF EACH SEGMENT WITH RESPECT TO SEGMENT 1, STORE IN MATRIX RW
C call angv(rw)
C C CALCULATE THE TERMS OF THE RELATIVE ANGULAR ACCELERATIONS OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT
CALL ANGACC(A,B,C,D)

CALCULATE INERTIA DYADIC OF EACH SEGMENT WITH RESPECT TO
THE REFERENCE SEGMENT COORDINATES

CALL DYADIC(IXX,IYY,IZZ,ROT,IDY)

CALCULATE THE COEFFICIENTS OF THE I DOT W X W TERMS

CALL IWW(W,IDY,RW)

CALCULATE THE COEFFICIENTS OF THE I DOT ALPHA TERMS

CALL IDOTALPHA(IDY,A,B,C,D,QQ,R1,S,T1,V)

COMPUTE THE COEFFICIENTS OF THE PXMA TERMS

CALL PXMA(Q,QDOT,QDDOT,DELTA,M,MTOT,P)
CALL GAMS(QQ,DELTA,GG)
DO I=1,2
F1(I)=FTOTAL(I)*PCT
FR(I)=FTOTAL(I)*(1-PCT)
ENDDO

COMPUTE KINEMATICS

CALL KINEM(XDD,M)

COMPUTE KINETICS

RELF=NRLEAS
RECF=RCF
CALL FORTORQ(THETA,ANG,OMEGA,ACC,H,ETA,R,FOR,TOR,
& RELRM,ROT,KP,M,IXX,IYY,IZZ,F1,FR)

WRITE KINEMATIC AND KINETIC DATA TO FILES

DO I=1,12
WRITE(5,*) I
WRITE(5,*) (ANG(I,J),J=1,3)
WRITE(5,*) (OMEGA(I,J),J=1,3)
WRITE(5,*) (ACC(I,J),J=1,3)
ENDDO
IF (FRAME.NE.NF) THEN
DO I=1,13
WRITE(6,*) I,(FOR(I,J),J=1,3),(TOR(I,J),J=1,3)
ENDDO
ENDIF
ENDDO
CLOSE(13)
PART 3B: FLIGHT PHASE

VELOCITY AT RELEASE FROM THE LAST FRAME OF THE GIANT SWING PHASE (XDOTREL) WAS FOUND WHEN XY DATA WAS INTERPOLATED

\[
\begin{align*}
\text{xdd}(1) &= 0. \\
\text{xdd}(2) &= -981. \\
v_1 &= \text{xdotrel}(1) \\
v_2 &= \text{xdotrel}(2)
\end{align*}
\]

CALCULATE ANGULAR MOMENTUM OF THE BODY ABOUT ITS CENTER OF GRAVITY AT THE POINT OF RELEASE

call angmom(h1,rw,p,idy,m,u1)

PRINT *,"At the time of release"
PRINT *,"Angular Momentum in Segment 1 coordinates",H1
PRINT *,"Velocity ",x , v1, y , v2
PRINT *,"Expected Recatch Frame",RCF,"(" ,RCF/20,"")"
PRINT *,"Angular Velocity",Ul
PRINT *,"Initial Position",X
PRINT *,"Initial Orientation",theta
PRINT *,"DO YOU WANT TO ALLOW THE REFERENCE SEGMENT TO ROTATE?"
PRINT *,"ABOUT ITS LONG AXIS AND ITS AP AXIS DURING FLIGHT?"
PRINT *,"1=YES, 2=NO"
read *,igo
if (igo.eq.2) Then
h1(1)=0.
h1(2)=0.
theta(2)=0.
theta(3)=0.
endif

COMPUTE ANGULAR MOMENTUM IN GLOBAL COORDINATES

\[
\begin{align*}
r_{11} &= \cos(\theta(1)) \cos(\theta(3)) - \sin(\theta(1)) \sin(\theta(2)) \\
r_{12} &= -\sin(\theta(1)) \cos(\theta(2))
\end{align*}
\]
\[
\begin{align*}
    r_{13} &= \cos(\theta(1)) \sin(\theta(3)) + \sin(\theta(1)) \sin(\theta(2)) \cos(\theta(3)) \\
    r_{21} &= \sin(\theta(1)) \cos(\theta(3)) + \cos(\theta(1)) \sin(\theta(2)) \sin(\theta(3)) \\
    r_{22} &= \cos(\theta(1)) \cos(\theta(2)) \\
    r_{23} &= \sin(\theta(1)) \sin(\theta(3)) - \cos(\theta(1)) \sin(\theta(2)) \cos(\theta(3)) \\
    r_{31} &= -\sin(\theta(3)) \cos(\theta(2)) \\
    r_{32} &= \sin(\theta(2)) \\
    r_{33} &= \cos(\theta(2)) \cos(\theta(3)) \\
\end{align*}
\]

\[
y(1) = h_1(1) * r_{11} + h_1(2) * r_{12} + h_1(3) * r_{13} \\
y(2) = h_1(1) * r_{21} + h_1(2) * r_{22} + h_1(3) * r_{23} \\
y(3) = h_1(1) * r_{31} + h_1(2) * r_{32} + h_1(3) * r_{33}
\]

PRINT *, "ANGULAR MOMENTUM IN GLOBAL COORDINATES:"
PRINT *, "X (Horizontal axis perpendicular to bar) ", Y(1)
PRINT *, "Y (Vertical axis) ", Y(2)
PRINT *, "Z (Horizontal axis parallel to bar) ", Y(3)
PRINT *, "Do you want to change these values?"
PRINT *, "1-yes, 2-no"
read *, go
if (go.eq.1) then
    PRINT *, "Do you want to change the momentum about the X & axis?"
    PRINT *, "1-yes, 2-no"
    read *, gox
    if (gox.eq.1) then
        PRINT *, "Enter new angular momentum about the X axis"
        read *, Y(1)
    endif
    PRINT *, "Do you want to change the momentum about the Y & axis?"
    PRINT *, "1-yes, 2-no"
    read *, goy
    if (goy.eq.1) then
        PRINT *, "Enter new angular momentum about the Y axis"
        read *, Y(2)
    endif
    PRINT *, "Do you want to change the momentum about the Z & axis?"
    PRINT *, "1-yes, 2-no"
    read *, goz
    if (goz.eq.1) then
        PRINT *, "Enter new angular momentum about the Z axis"
        read *, Y(3)
    endif
endif

C COMPUTE THE FORCE AT THE HANDS REQUIRED TO PRODUCE THIS CHANGE IN ANGULAR MOMENTUM (AROUND Z-AXIS)
\[ \delta_3 = \frac{(y(3) - h_1(1)*r_{31} + h_1(2)*r_{32} + h_1(3)*r_{33})}{0.02} \]

\[ \text{Radius} = \sqrt{x(1)^2 + x(2)^2} \]

\[ \text{force} = \frac{\delta_3}{\text{Radius}} \]

\[ \text{tt} = \text{atan2}(x(2), x(1)) \]

\[ \text{forcex} = \frac{\text{abs} \left( \text{force} \times \sin \text{tt} \right)}{100}. \]

\[ \text{forcey} = \frac{\text{abs} \left( \text{force} \times \cos \text{tt} \right)}{100}. \]

\[ \begin{align*}
&\text{for}(12,1) = \frac{\text{forcex}}{2}. \\
&\text{for}(13,1) = \frac{\text{forcex}}{2}. \\
&\text{for}(12,2) = \frac{\text{forcey}}{2}. \\
&\text{for}(13,2) = \frac{\text{forcey}}{2}. \\
&\text{for}(12,3) = 0. \\
&\text{for}(13,3) = 0.
\end{align*} \]

\[ \text{do i=1,13} \]
\[ \quad \text{write(6,*) i,(for(i,j),j=1,3),(tor(i,j),j=1,3)} \]
\[ \text{enddo} \]

\[ \]

\[ \text{C THESE FORCES CHANGE THE LINEAR VELOCITY, ALSO} \]

\[ \]

\[ \delta v_1 = \text{forcex} \times 100. \times 0.02 / \text{mtot} \]

\[ \delta v_2 = \text{forcey} \times 100. \times 0.02 / \text{mtot} \]

\[ v_1 = \delta v_1 + v_1 \]

\[ v_2 = \delta v_2 + v_2 \]

\[ \text{PRINT *,'"CHANGING THE ANGULAR MOMENTUM CHANGED THE LINEAR"} \]

\[ \text{PRINT *,"VELOCITY TO: ",v1,"","v2} \]

\[ \text{PRINT *,"ENTER 1 TO CONTINUE, ENTER 2 TO CHANGE THESE VALUES"} \]

\[ \text{read *,go} \]

\[ \text{if (go.eq.2) then} \]

\[ \text{PRINT *,"ENTER VX"} \]

\[ \text{read *,v1} \]

\[ \text{PRINT *,"ENTER VY"} \]

\[ \text{read *,v2} \]

\[ \text{endif} \]

\[ \text{endif} \]

\[ \text{print *,'"do you want to force arms to be symmetric during"} \]

\[ \text{& flight?"} \]

\[ \text{print *,"1-yes,2-no"} \]

\[ \text{read *,iflag} \]

\[ \text{do frame=nrelease+1,rcf} \]

\[ \text{do i=1,12} \]

\[ \quad \text{do j=1,3} \]

\[ \quad \text{read(12,*)q(i,j),qdot(i,j),qddot(i,j)} \]

\[ \quad \text{enddo} \]

\[ \text{enddo} \]
do i=1,11
   read(11,*)alpha(i),alphadot(i),alphaddot(i)
   read(11,*)beta(i),betadot(i),betaddot(i)
   read(11,*)gamma(i),gammadot(i),gammaddot(i)
   if(abs(alphadot(i)).gt.1000.) then
      alphadot(i)=0.
      alphaddot(i)=0.
   endif
   if(abs(betadot(i)).gt.1000.) then
      betadot(i)=0.
      betaddot(i)=0.
   endif
   if(abs(gammadot(i)).gt.1000.) then
      gammadot(i)=0.
      gammaddot(i)=0.
   endif
enddo

C USE CONSERVATION OF LINEAR MOMENTUM IN THE HORIZONTAL DIRECTION TO COMPUTE THE X POSITION OF THE CG DURING FLIGHT

x(1)=x(1)+v1*.001

C USE CONSTANT ACCELERATION FORMULAS (GRAVITY) TO DETERMINE Y POSITION OF THE CG DURING FLIGHT

x(2)=x(2)+v2*.001-.5*981*.001**2
   v2=v2-981*.001

C EXTERNAL MOMENTS AND FORCES ARE ZERO (EXCEPT GRAVITY)

ftotal(1)=0.
ftotal(2)=0.
do i=1,2
   fr(i)=0.
   fl(i)=0.
enddo
   momtot=0.

C SOLVE FOR THE ANGULAR VELOCITY U1 USING DAPENA'S ALGORITHM

C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN ADJACENT SEGMENTS

if (iflag.eq.1) then
alpha(2)=alpha(3)
beta(2)=-beta(3)
gamma(2)=-gamma(3)
endif

C call rotation(alpha,beta,gamma,relrm)
C
C CALCULATE THE ROTATION MATRIX OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT
C call rm(rlrm,rot)
C
C LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE SEGMENTS
C call cg(r,eta,rot,m,q,h,pg,p)
call handloc(plh,prh,p,rot,eta,h)
C
C CALCULATE ANGULAR VELOCITY OF EACH SEGMENT WITH RESPECT TO SEGMENT 1, STORE IN MATRIX RW
C
call angv(rw)
do i=1,12
do j=1,3
if (abs(rw(i,j)).gt.15) then
  if (rw(i,j).lt.0) then
    rw(i,j)=-15.
  else
    rw(i,j)=-15.
  endif
endif
enddo
endo
cono
if (iflag.eq.1) then
rw(5,1)=-rw(6,1)
rw(5,2)=-rw(6,2)
rw(5,3)=rw(6,3)
rw(7,1)=-rw(8,1)
rw(7,2)=-rw(8,2)
rw(7,3)=rw(8,3)
endif
C
C CALCULATE INERTIA DYADIC OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT COORDINATES
C call dyadic(ixx,iyy,izz,rot,idy)
C
C CALCULATE ANGULAR VELOCITY OF THE REFERENCE SEGMENT
C
    call findul(idy,rw,ul,m,qdot,q,h1)
C    SET UP THE NUMERICAL INTEGRATION TO SOLVE FOR THE
C    ROTATION
C    ANGLES OF THE REFERENCE SEGMENT
C
    call integ(theta,ul)
C
    FROM NEW THETAS, CALCULATE H1 IN TERMS OF THE
    COORDINATES
C    OF THE REFERENCE SEGMENT
C
    r11=cos(theta(1))*cos(theta(3)) - sin(theta(1))*sin(theta(2))
    & *sin(theta(3))
    r12=-sin(theta(1))*cos(theta(2))
    r13=cos(theta(1))*sin(theta(3)) + sin(theta(1))*sin(theta(2))
    & *cos(theta(3))
    r21=sin(theta(1))*cos(theta(3)) + cos(theta(1))*sin(theta(2))
    & *sin(theta(3))
    r22=cos(theta(1))*cos(theta(2))
    r23=sin(theta(1))*sin(theta(3)) - cos(theta(1))*sin(theta(2))
    & *cos(theta(3))
    r31=-sin(theta(3))*cos(theta(2))
    r32=sin(theta(2))
    r33=cos(theta(2))*cos(theta(3))
    h1(1)=y(1)*r11+y(2)*r21+y(3)*r31
    h1(2)=y(1)*r12+y(2)*r22+y(3)*r23
    h1(3)=y(1)*r13+y(2)*r23+y(3)*r33
C
    CALCULATE THE TERMS OF THE RELATIVE ANGULAR
    ACCELERATIONS
C    OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT
C
    call angacc(a,b,c,d)
C
    CALCULATE THE COEFFICIENTS OF THE I DOT W X W TERMS
C
    call iww(w,idy,rw)
C
    CALCULATE THE COEFFICIENTS OF THE I DOT ALPHA TERMS
C
    call idotalpha(idy,a,b,c,d,qq,r1,s,t1,v)
C
    COMPUTE THE COEFFICIENTS OF THE PXMA TERMS
call pxma(q,qdot,qddot,delta,m,mtot,p)

REARRANGE THE EQUATIONS TO BE IN THE FORM OF YDOT=F(Y,T)
call gams(qq,delta,gg)

EVERY .02 SECONDS, COMPUTE KINEMATICS AND KINETICS
if (mod(frame,20).Eq.0) Then

GET DERIVATIVES OF U1
call fl(theta,u1,u1dot)

COMPUTE KINEMATICS
call kinem(xdd,m)

COMPUTE KINETICS

fl(1)=0.
fl(2)=0.
fr(1)=0.
fr(2)=0.
relf=nrelease
recf=rcf
call fortorq(theta,ang,omega,acc,h,eta,r,for,tor,
& relrm,rot,kp,m,ixx,iyy,izz,fl,fr)

COMPUTE THE 3D POSITIONS OF LANDMARKS FOR ANIMATION
call landmark(h,eta,r,x,theta,p,rot,mark)

DRAW UPDATED FIGURE
call animate(mark)

WRITE KINEMATIC AND KINETIC DATA TO FILES

do i=1,12
  write(5,*i)
  write(5,*)(ang(i,j),j=1,3)
  write(5,*)(omega(i,j),j=1,3)
  write(5,*)(acc(i,j),j=1,3)
enddo
do i=1,13
  write(6,* i,(for(i,j),j=1,3),(tor(i,j),j=1,3)
enddo
do i=1,22
  write(7,*i,(mark(i,j),j=1,3)
enddo
endif
enddo

C C PART 3C: RECATCH PHASE
C CALCULATE THE INITIAL BODY POSITION AT THE POINT OF
C RECATCH
C
500 angle=atan2(x(2),x(1))
cgrad=sqrt(x(2)**2+x(1)**2)

C CHECK TO SEE IF HANDS ARE NEAR THE BAR
C
C IN GLOBAL COORDINATES, COMPUTE THE VECTOR FROM THE CG TO
C THE HANDS
C
glh(l)=plh(l)*rll+plh(2)*rl2+plh(3)*rl3
glh(2)=plh(l)*r21+plh(2)*r22+plh(3)*r23
glh(3)=plh(l)*r31+plh(2)*r32+plh(3)*r33
grh(1)=prh(l)*rll+prh(2)*rl2+prh(3)*rl3
grh(2)=prh(l)*r21+prh(2)*r22+prh(3)*r23
grh(3)=prh(l)*r31+prh(2)*r32+prh(3)*r33

C COMPUTE THE HAND LOCATIONS IN GLOBAL COORDINATES
C
do i=1,3
left(i)=x(i)+glh(i)
right(i)=x(i)+grh(i)
enddo
lrh=sqrt(left(l)**2+left(2)**2)
rrh=sqrt(right(1)**2+right(2)**2)

C SEE IF BOTH HANDS MISSED
C IF SO, THEN CONTINUE FLIGHT PHASE
C
if((lrh.Gt.15.).And.(rrh.Gt.15.)) Then
if (frame.gt.nframes) then
ntot=nframes/20
open(1, file='c:\fortran\ntot.dat')
write(1,*)ntot
close(1)
goto 2222
endif
endif

C IF THE BAR IS BETWEEN THE HANDS AND THE CG, THEN ASSUME
C RECATCH
C
if ((left(1)/x(1).Lt.0).And.(Left(2)/x(2).Lt.0)) Then
goto 3333
endif
do i=1,12
do j=1,3
   read(12,*)q(i,j),qdot(i,j),qddot(i,j)
endo
enddo
doi=1,11
read(11,*)alpha(i),alphadot(i),alphaddot(i)
read(11,*)beta(i),betadot(i),betaddot(i)
read(11,*)gamma(i),gammadot(i),gammaddot(i)
if(abs(alphadot(i)).gt.1000.) then
   alphadot(i)=0.
   alphaddot(i)=0.
endif
if(abs(betadot(i)).gt.1000.) then
   betadot(i)=0.
   betaddot(i)=0.
endif
if(abs(gammadot(i)).gt.1000.) then
   gammadot(i)=0.
   gammaddot(i)=0.
endif
Enddo

C USE CONSERVATION OF LINEAR MOMENTUM IN THE HORIZONTAL DIRECTION TO COMPUTE THE X POSITION OF THE CG DURING FLIGHT
C
   x(1)=x(1)+v1*.001

C USE CONSTANT ACCELERATION FORMULAS (GRAVITY) TO DETERMINE Y POSITION OF THE CG DURING FLIGHT
C
   x(2)=x(2)+v2*.001-.5*981*.001**2
   V2=v2-981*.001

C EXTERNAL MOMENTS AND FORCES ARE ZERO (EXCEPT GRAVITY)
C
   ftotal(1)=0.
   ftotal(2)=0.
   do i=1,2
      fr(i)=0.
      fl(i)=0.
   enddo
   momtot=0.

C SOLVE FOR THE ANGULAR VELOCITY U1 USING DAPENA'S ALGORITHM
C
C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN ADJACENT
          SEGMENTS
          
          if (iflag.eq.1) then
          alpha(2)=alpha(3)
beta(2)=-beta(3)  
gamma(2)=-gamma(3)
endif
          call rotation(alpha,beta,gamma,relrm)
          
          CALL THE ROTATION MATRIX OF EACH SEGMENT WITH
          RESPECT
          TO THE REFERENCE SEGMENT
          
          call rm(relrm,rot)
          
          LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE
          SEGMENTS
          
          call cg(r,eta,rot,m,q,h,pg,p)
          
          CALCULATE ANGULAR VELOCITY OF EACH SEGMENT WITH RESPECT
          TO
          SEGMENT 1, STORE IN MATRIX RW
          
          call angv(rw)
          do i=1,12
          do j=1,3
          if (abs(rw(i,j)).gt.15)then
              if (rw(i,j).lt.0)then
                  rw(i,j)=-15.
              else
                  rw(i,j)=-15.
              endif
          endif
          enddo
          enddo
          
          if (iflag.eq.1) then
          rw(5,l)=-rw(6,l)
rw(5,2)=-rw(6,2)
rw(5,3)=rw(6,3)
rw(7,l)=-rw(8,l)
rw(7,2)=-rw(8,2)
rw(7,3)=rw(8,3)
endif
          
          CALL THE INERTIA DYADIC OF EACH SEGMENT WITH RESPECT TO
          THE REFERENCE SEGMENT COORDINATES
          
          call dyadic(ixx,iyy,izz,rot,idy)
C CALCULATE ANGULAR VELOCITY OF THE REFERENCE SEGMENT
C call findul(idy,rw,u1,m,qdot,q,h1)
C SET UP THE NUMERICAL INTEGRATION TO SOLVE FOR THE
C ANGLES OF THE REFERENCE SEGMENT
C call integ(theta,u1)
C FROM NEW THETAS, CALCULATE H1 IN TERMS OF THE
C COORDINATES
C OF THE REFERENCE SEGMENT
C
r11=cos(theta(1))*cos(theta(3))-sin(theta(1))*sin(theta(2))
 & *sin(theta(3))
r12=-sin(theta(1))*cos(theta(2))
r13=cos(theta(1))*sin(theta(3))+sin(theta(1))*sin(theta(2))
 & *sin(theta(3))
r21=sin(theta(1))*cos(theta(3))+cos(theta(1))*sin(theta(2))
 & *sin(theta(3))
r22=cos(theta(1))*cos(theta(2))
r23=sin(theta(1))*sin(theta(3))-cos(theta(1))*sin(theta(2))
 & *cos(theta(3))
r31=-sin(theta(3))*cos(theta(2))
r32=sin(theta(2))
r33=cos(theta(2))*cos(theta(3))
h1(1)=y(1)*r11+y(2)*r21+y(3)*r31
h1(2)=y(1)*r12+y(2)*r22+y(3)*r32
h1(3)=y(1)*r13+y(2)*r23+y(3)*r33
C CALCULATE THE TERMS OF THE RELATIVE ANGULAR
ACCELERATIONS
C OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT
C call angacc(a,b,c,d)
C CALCULATE THE COEFFICIENTS OF THE I DOT W X W TERMS
C call iww(w,idy,rw)
C CALCULATE THE COEFFICIENTS OF THE I DOT ALPHA TERMS
C call idotalpha(idy,a,b,c,d,qq,r1,s,t1,v)
C COMPUTE THE COEFFICIENTS OF THE PXMA TERMS
CALL PXMA(Q,QDOT,QDDOT,DDELTA,M,MTOT,P)

REARRANGE THE EQUATIONS TO BE IN THE FORM OF YDOT=F(Y,T)

CALL GAMS(QQ,DDELTA,GG)

EVERY .02 SECONDS, COMPUTE KINEMATICS AND KINETICS

IF (MOD(FRAME,20).EQ.0) THEN
  PRINT *,"****HANDS ARE NOT NEAR THE BAR!!!! FLIGHT CONTINUES****"
  PRINT *,FRAME/20
  PRINT *,LRH,RRH
  PRINT *,"LEFT HAND POSITION",LEFT
  PRINT *,"RIGHT HAND POSITION",RIGHT
  PRINT *,"ENTER 1 TO CONTINUE"
  READ *,GO

GET DERIVATIVES OF U1

CALL FL(THETA,UL,ULDOT)

COMPUTE KINEMATICS

CALL KINEM(XDD,M)

COMPUTE KINETICS

F1(1)=0.
F1(2)=0.
FR(1)=0.
FR(2)=0.
RELF=NRELEASE
RECFC=RCF
CALL FORTORQ(THETA,ANG,OMEGA,ACC,H,ETA,R,FOR,TOR,
&   RELRM,ROT,KP,M,IXX,IIY,IZZ,F1,FR)

COMPUTE THE 3D POSITIONS OF LANDMARKS FOR ANIMATION

CALL LANDMARK(H,ETA,R,X,THETA,P,ROT,MARK)

DRAW UPDATED FIGURE

CALL ANIMATE(MARK)

WRITE KINEMATIC AND KINETIC DATA TO FILES

DO I=1,12
write(5,*),i
write(5,*),(ang(i,j),j=1,3)
write(5,*),(omega(i,j),j=1,3)
write(5,*),(acc(i,j),j=1,3)

enddo
do i=1,13
write(6,*),i,(for(i,j),j=1,3),(tor(i,j),j=1,3)
enddo
do i=1,22
write(7,*),i,(mark(i,j),j=1,3)
endif
frame=frame+1
go to 500
endif

C IF BOTH HANDS DIDN'T MISS, ASSUME THEY BOTH CATCH
C
3333 close(8)
PRINT *, "RECATCH!!!"
n=frame/20
print *,"recatch frame",n
print *,"enter 1 to continue"
read *,go

C CALCULATE ANGLE VS. TIME USING THE CGANGLE DERIVATIVES
C
fn=dir(1:len_trim(dir))/prefix//'.cga'
fn2=dir(1:len_trim(dir))/prefix//'.thr'
call findang(nframes/20,n,angle,fn,fn2)
open(9,file=fn2)
n1=(nframes-frame)/20
ntot=n1+frame/20
open(1,file='c:\fortran\ntot.dat')
write(1,*),ntot
close(1)
read (9,*)(cgang(i),i=1,n1)
close(9)

C GET RIGHT AND LEFT HAND POSITION VECTORS WITH RESPECT TO
CG
C OF BODY IN SEGMENT 1 COORDINATES
C
open(3,file=dir(1:len_trim(dir))/prefix//'.jvt')
do i=1,n
do j=1,11
read(3,*),alpha(j)
read(3,*),beta(j)
read(3,*),gamma(j)
enddo
enddo

do i=1,nl
   do j=1,11
      read(3,*)alpha(j)
      read(3,*)beta(j)
      read(3,*)gamma(j)
   enddo
   call rotation(alpha,beta,gamma,relrm)
   call rm(relrm,rot)
   call cg(r,eta,rot,m,q,h,pg,p)
   call handloc(plh,prh,p,rot,eta,h)
   do j=1,3
      rh(i,j)=prh(j)
      lh(i,j)=plh(j)
   enddo
enddo

close(3)
call getz(theta,plh,prh,zh)
x(3)=0.5*(zh(1)+zh(2))

C GET THE RADIUS (DISTANCE FROM GLOBAL ORIGIN) TO CG OF BODY AT EACH FRAME (.02 SECONDS APART)
call getr(cgr,nl,mtot,1,e,inert,rbar,tbar)

C COMPUTE THE X-Y COORDINATES OF THE CENTER OF GRAVITY (Z IS ASSUMED TO BE CONSTANT)
do i=1,nl
   xx(i,1)=cgr(i)*cos(cgang(i))
   xx(i,2)=cgr(i)*sin(cgang(i))
   xx(i,3)=(zh(1)+zh(2))/2.
endo
open(1,file=dir(l:len_trim(dir))//prefix//' .mat')
open(2,file=dir(l:len_trim(dir))//prefix//' .tht')
do frame=l,nl

C READ IN DATA FOR EACH FRAME

do k=1,20
   do i=1,12
      do j=1,3
         read(12,*)q(i,j),qdot(i,j),qddot(i,j)
      enddo
   enddo
endo
do i=1,11
   read(11,*)alpha(i),alphadot(i),alphaddot(i)
   read(11,*)beta(i),betadot(i),betaddot(i)
   read(11,*)gamma(i),gammadot(i),gammaddot(i)
enddo
enddo
x(1)=xx(frame,1)
x(2)=xx(frame,2)

C SET UP EQUATIONS FOR USING SPOOR AND VELDPAUS METHODS TO
C COMPUTE THE ROTATION MATRIX BETWEEN SEGMENT 1 AND THE
C GLOBAL COORDINATES
C
C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN
ADJACENT
C SEGMENTS
C
call rotation(alpha,beta,gamma,relrm)
C
C CALCULATE THE ROTATION MATRIX OF EACH SEGMENT WITH
RESPECT
C TO THE REFERENCE SEGMENT
C
call nti(relrm,rot)
C
C LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE
SEGMENTS
C
call cg(r,eta,rot,m,q,h,pg,p)
call handloc(plh,prh,p,rot,eta,h)
C
C CALL ROT1 AND CALCULATE THE ROTATION MATRIX BETWEEN
GLOBAL
C AND S1 COORDINATES
C
do i=1,3
   aa(1,i)=0.0
   aa(2,i)=plh(i)
   aa(3,i)=prh(i)
   vv(1,i)=xx(frame,i)
endo
vv(2,1)=rbar(frame)*cos(tbar(frame))
vv(2,2)=rbar(frame)*sin(tbar(frame))
vv(2,3)=zh(2)
vv(3,1)=vv(2,1)
vv(3,2)=vv(2,2)
vv(3,3)=zh(1)
call rot1(aa,vv,rmatrix)
C
C CALCULATE THE ROTATION ANGLES FROM THE ROTATION MATRIX
C
pi=acos(-1.0)
theta(1)=atan2(-rMatrix(1,2),rmatrix(2,2))
theta(2)=asin(rmatrix(3,2))
theta(3) = \text{atan2}(-r_{matrix}(3,1), r_{matrix}(3,3))
if ((r_{matrix}(3,3) > 0.) \text{ and } (\text{Cos}(\theta(3)) < 0.)) then
  \theta(2) = \pi - \theta(2)
elsif ((r_{matrix}(3,3) < 0.) \text{ and } (\text{Cos}(\theta(3)) > 0.)) then
  \theta(2) = \pi - \theta(2)
end if
\text{do } i=1,3
  \text{write}(1,*)(r_{matrix}(i,j), j=1,3)
\text{enddo}
\text{write}(2,*)(\text{frame}, \theta(1), \theta(2), \theta(3))

\text{C COMPUTE THE 3D POSITIONS OF LANDMARKS FOR ANIMATION}
\text{C}
\text{call landmark(h, eta, r, x, theta, p, rot, mark)}
\text{C DRAW UPDATED FIGURE}
\text{C}
\text{call animate(mark)}
\text{do } i=1,22
  \text{write}(7,*)(\text{mark}(i,j), j=1,3)
\text{enddo}
\text{close(1)}
\text{close(2)}
\text{close(11)}
\text{close(12)}

\text{C FROM THETAS, FILTER AND DIFFERENTIATE TO GET THE}
\text{C DERIVATIVES OF THE THETAS}
\text{C}
\text{fn=dir(1:len_trim(dir))/prefix//'.tht'}
\text{nf=n1}
\text{IN ORDER TO SMOOTH THE RECATCH DATA, INTERPOLATE 20}
\text{FRAMES BETWEEN THE DATA FOR THE ANGLES DURING THE .02 SECOND}
\text{FRAMES, THEN FILTER AND THEN TAKE DERIVATIVES AT THE}
\text{POINTS NEEDED}
\text{C}
\text{open(1, file=fn)}
\text{do } i=1,nf
  \text{read (1,*)(dummy1, (adummy(i,j), j=1,3)}
\text{enddo}
\text{close(1)}
\text{call interpolatec(adummy, fn, nf)}
\text{call filtert2(fn, nf*20)}
\text{call intt2(fn, nf*20)}
C LOOP TO FIND THE KINEMATICS AND KINETICS OF EACH
C SEGMENT/JOINT DURING THE SWING
C
open(11, file=dir(1: len_trim(dir))//prefix//'.jt2')
!INTERPOLATED JOINT ANGLES
open(12, file=dir(1: len_trim(dir))//prefix//'.qd2')
!INTERPOLATED Q'S
open(13, file=dir(1: len_trim(dir))//prefix//'.Tht')
!INTERPOLATED THETAS

do icount=1, n
  do i=1, 12
    do j=1, 3
      read(12, *) q(i, j), qdot(i, j), qddot(i, j)
    enddo
  enddo
  do i=1, 11
    read(11, *) alpha(i), alphadot(i), alphaddot(i)
    read(11, *) beta(i), betadot(i), betaddot(i)
    read(11, *) gamma(i), gammadot(i), gammaddot(i)
  enddo
enddo

do frame=1, nf
C
C READ IN DATA FOR EACH FRAME
C
do icount=1, 20
  do i=1, 12
    do j=1, 3
      read(12, *) q(i, j), qdot(i, j), qddot(i, j)
    enddo
  enddo
  do i=1, 11
    read(11, *) alpha(i), alphadot(i), alphaddot(i)
    read(11, *) beta(i), betadot(i), betaddot(i)
    read(11, *) gamma(i), gammadot(i), gammaddot(i)
  enddo
  do i=1, 3
    x(i)=x*{frame, i}
  enddo
  do i=1, 2
    ftotal(i)=ftot(frame, i)
  enddo
  xdd(1)=ftotal(1)/mtot
  xdd(2)=ftotal(2)/mtot-981
  xdd(3)=0.
enddo

C
C
read(13,*) theta(i), thetadot(i), thetaddot(i)
enddo
enddo

ul(1)=thetadot(2)*cos(theta(3))-thetadot(1)*cos(theta(2))
&   +*sin(theta(3))
ul(2)=thetadot(1)*sin(theta(2))+thetadot(3)
ul(3)=thetadot(1)*cos(theta(2))*cos(theta(3))
&   +thetadot(2)*sin(theta(3))

uldot(1)=thetaddot(2)*cos(theta(3))-thetadot(2)*thetadot(3)
&   *sin(theta(3))-thetaddot(1)*cos(theta(2))*sin(theta(3))
&   +thetadot(1)*thetadot(2)*sin(theta(2))*sin(theta(3))
&   -thetadot(1)*thetadot(3)*cos(theta(2))*cos(theta(3))
uldot(2)=thetaddot(1)*sin(theta(2))+thetadot(1)
uldot(3)=thetaddot(1)*cos(theta(2))*cos(theta(3))
&   -thetadot(1)*thetadot(2)*sin(theta(2))*cos(theta(3))
&   -thetadot(1)*thetadot(3)*cos(theta(2))*sin(theta(3))
&   +thetaddot(2)*sin(theta(3))+thetadot(2)*thetadot(3)*
&   cos(theta(3))

C
C CALCULATE THE RELATIVE ROTATION MATRICES BETWEEN
ADJACENT
C SEGMENTS
C
call rotation(alpha,beta,gamma,relrm)
C
C CALCULATE THE ROTATION MATRIX OF EACH SEGMENT WITH
RESPECT
C TO THE REFERENCE SEGMENT
C
call rm(relrm,rot)
C
C LOCATE THE CENTER OF GRAVITY WITH RESPECT TO THE
SEGMENTS
C
call cg(r,eta,rot,m,q,h,pg,p)
call handloc(plh,prh,p,rot,eta,h)
C
C CALCULATE THE TOTAL APPLIED MOMENT
C
momtot=kp*sqrt(ftotal(1)**2+ftotal(2)**2)
C
C CALCULATE ANGULAR VELOCITY OF EACH SEGMENT WITH RESPECT
TO
C SEGMENT 1, STORE IN MATRIX RW
C
call angv(rw)
CALCULATE THE TERMS OF THE RELATIVE ANGULAR ACCELERATIONS
OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT

CALL ANGACC(A,B,C,D)

CALCULATE INERTIA DYADIC OF EACH SEGMENT WITH RESPECT TO THE REFERENCE SEGMENT COORDINATES

CALL DYADIC(IXX, IYY, IZZ, ROT, IDY)

CALCULATE THE COEFFICIENTS OF THE I DOT W X W TERMS

CALL IWW(W, IDY, RW)

CALCULATE THE COEFFICIENTS OF THE I DOT ALPHA TERMS

CALL IDOTALPHA(IDY, A, B, C, D, QQ, R1, S, T1, V)

COMPUTE THE COEFFICIENTS OF THE PXMA TERMS

CALL PXMA(Q, QDOT, QDDOT, DELTA, M, MTOT, P)
CALL GAMS(QQ, DELTA, GG)

DO I=1,2
FL(I)=FTOTAL(I)*PCT
FR(I)=FTOTAL(I)*(1-PCT)
ENDDO

COMPUTE KINEMATICS

CALL KINEM(XDD, M)

COMPUTE KINETICS

RELF=NRLEASE
REC=RCF
CALL FORTORQ(THETA, ANG, OMEGA, ACC, H, ETA, R, FOR, TOR, 
&   RELRM, ROT, KP, M, IXX, IYY, IZZ, FL, FR)

WRITE KINEMATIC AND KINETIC DATA TO FILES

DO I=1,12
   WRITE(5,*) I
   WRITE(5,*) (ANG(I,J), J=1,3)
   WRITE(5,*) (OMEGA(I,J), J=1,3)
   WRITE(5,*) (ACC(I,J), J=1,3)
ENDDO
DO I=1,13


write(6,*) i,(for(i,j),j=1,3),(tor(i,j),j=1,3)
enddo
enddo  
2222 close(5)
close(6)
close(7)
close(11)
close(12)
close(13)
end

Subroutine Inertia

SUBROUTINE INERTIA(M,IXX,IYY,IZZ,R,RR,H,ETA,FN)
C THIS SUBROUTINE TAKES ANTHROPOMETRIC DATA AND CALCULATES
C MASS, CENTER OF GRAVITY AND MOMENTS OF INERTIA FOR EACH
C SEGMENT BASED ON HANAVAN AND CLAUSER'S EQUATIONS
CHARACTER*50 FN
REAL
M(12),IXX(12),IYY(12),IZZ(12),H(12),MU,SIGMA,ETA(12)
REAL BB,R(12),RR(12),IF
pi=3.14159
C
C DEFINE SEGMENTS AS FOLLOWS
C 1=MID TRUNK
C 2=UPPER TRUNK, 3=LOWER TRUNK, 4=HEAD
C 5=RIGHT FOREARM & HAND, 6=LEFT FOREARM & HAND
C 7=RIGHT UPPER ARM, 8=LEFT UPPER ARM
C 9=RIGHT THIGH, 10=LEFT THIGH
C 11=RIGHT SHANK & FOOT, 12=LEFT SHANK & FOOT
C
C READ IN ANTHROPOMETRIC DATA
C
open (1,file=fn)
read(1,4000) headc
read(1,4000) cc
read(1,4000) axilc
read(1,4000) elbc
read(1,4000) fac
read(1,4000) wrisc
read(1,4000) thihc
read(1,4000) gk nec
read(1,4000) cfc
read(1,4000) foarl
read(1,4000) wrisb
read(1,4000) handb
read(1,4000) stat
read(1,4000) shldh
read(1,4000) subh
read(1,4000) troch
read(1,4000) sith
read(1,4000) wsth
read(1,4000) ankc
read(1,4000) if
read(1,4000) chesb
read(1,4000) chesd
read(1,4000) waisd
read(1,4000) hipb
read(1,4000) buttd
read(1,4000) uparl
read(1,4000) tibh
read(1,4000) footl
read(1,4000) w
read(1,4000) sphyh
close (1)

4000 format (f8.2)
C
C Distribute weight among segments using Clauer’s equations
C
m(4)=.104*headc+.015*w-2.189
trunkm=.349*w+.423*(sith-(stat-shldh))+.229*cc-35.460
h(2)=shldh-subh
h(1)=subh-wsth
h(3)=(sith-(stat-shldh))-h(2)-h(1)
if ((sith-(stat-shldh))*trunkm.Eq. 0) Then
  m(2)=0
  m(i)=0
  m(3)=0
else
  m(2)=h(2)/(sith-(stat-shldh))*trunkm
  m(1)=h(1)/(sith-(stat-shldh))*trunkm
  m(3)=h(3)/(sith-(stat-shldh))*trunkm
end if
m(5)=.029*wrisc+.075*wrisb+.031*handb-.746+.081*wrisc
& +.052*fac-1.650
m(6)=m(5)
m(7)=.007*w+.092*axilc+.095*uparl-3.101
m(8)=m(7)
m(9)=.074*w+.123*thihc+.027*if-4.216
m(10)=m(9)
m(11)=.111*cfc+.047*tibh+.074*ankc-4.208
& +.003*w+.048*ankc+.027*footl-.8690
m(12)=m(11)
C
C Distribute remaining weight over segments
C

        sum=0
        do 4100 i=1,12
            sum=sum+m(i)
        4100 continue
        bal=w-sum
        do 4200 i=1,12
            if (sum*bal .Eq. 0) Then
                m(i)=0
            else
                m(i)=m(i)+m(i)/sum*bal
            end if
        4200 continue

        C
        C CALCULATE PRINCIPLE INERTIAS FOR EACH SEGMENT
        C
        C FOR HEAD
        C
        h(4)=stat-shldh
        r(4)=.5*(stat-shldh)
        if (pi.Eq.0) Then
            rr(4) = 0
        else
            rr(4)=headc/2/pi
        end if
        ixx(4)=0.2*m(4)*(r(4)**2+rr(4)**2)
        izz(4)=ixx(4)
        iyy(4)=0.4*m(4)*rr(4)**2

        C
        C FOR TRUNK
        C
        r(2)=.5*chesb
        r(1)=.5*hipb
        r(3)=r(1)
        rr(2)=.25*(chesd+waisd)
        rr(1)=.25*(waisd+buttd)
        rr(3)=rr(1)
        do 4400 i=1,3
            ixx(i)=m(i)/12*(3*rr(i)**2+h(i)**2)
            izz(i)=m(i)/12*(3*r(i)**2+h(i)**2)
            iyy(i)=m(i)/4*(r(i)**2+rr(i)**2)
        4400 continue

        C
        C FOR ARMS AND LEGS, R=LARGE RADIUS, RR=SMALL RADIUS
        C
        if (pi .Eq. 0) Then
            r(7)=0
            rr(7)=0
        else
            r(7)=axilc/2/pi
rr(7)=elbc/2/pi
end if
r(5)=rr(7)
if (pi .Eq. 0) Then
rr(5)=0
r(9)=0
rr(9)=0
else
rr(5)=wrisc/2/pi
r(9)=thihc/2/pi
rr(9)=gknec/2/pi
end if
r(11)=rr(9)
if (pi .Eq. 0) Then
rr(11)=0
else
rr(11)=ankc/2/pi
end if
h(7)=uparl
h(5)=foarl
h(9)=stat-sith-tibh
h(11)=tibh-sphyh
doi=5,11,2
r(i+1)=r(i)
rr(i+1)=rr(i)
h(i+1)=h(i)
enddo
do 4500 i=5,11,2
if (r(i) .Eq. 0) Then
rt=0
aa=0
bb=0
else
rt=rr(i)/r(i)
if (pi .Eq. 0) Then
aa=0
else
aa=9./20. /pi*(1.+rt+rt**2+rt**3+rt**4)/
& (1.+rt+rt**2)**2
end if
bb=3./80.* (1.+4.*rt+10.*rt**2+4.*rt**3+rt**4)
&/(1.+rt+rt**2)**2
end if
d=r(i)**2+rr(i)*r(i)+rr(i)**2
if (h(i) .Eq. 0 .Or. Pi .Eq. 0 .Or. D .Eq. 0) Then
d=0
else
d=3.*m(i)/h(i)/(r(i)**2+rr(i)*r(i)+rr(i)**2)/pi
End if
if (d .Eq. 0 .Or. H(i)+bb*m(i)*h(i)**2 .Eq. 0) Then
288

ixx(i)=0
else
  ixx(i)=aa*m(i)**2/d/h(i)+bb*m(i)*h(i)**2
end if
izz(i)=ixx(i)
if (r(i)**3-rr(i)**3 .Eq. 0) Then
  iyy(i)=0
else
  iyy(i)=.3*m(i)*(r(i)**5-rr(i)**5)/(r(i)**3-rr(i)**3)
End if
ixx(i+1)=ixx(i)
ixy(i+1)=ixy(i)
izz(i+1)=izz(i)
4500 continue
C
C  LOCATE CENTER OF GRAVITY ALONG Y-AXIS
C
do 4600 i=1,4
  eta(i)=.5
4600 continue
C
do 4700 i=5,11,2
  if (r(i) .Eq. 0) Then
    mu=0
  else
    mu=rr(i)/r(i)
  end if
  sigma=1+mu+mu**2
  if (sigma .Eq. 0) Then
    eta(i)=0
  else
    eta(i)=(1+2*mu+3*mu**2)/4/sigma
  end if
  eta(i+1)=eta(i)
4700 continue
return
end

Subroutine QDOTS

  SUBROUTINE QDOTS(FNIN,FNOUT,ETA,H,R,NFRAMES,M)

C
C        THIS SUBROUTINE READS IN THE JOINT ROTATION ANGLES VS.
C        TIME AND COMPUTES THE DERIVATIVES OF THE Q'S WHICH ARE
C        THE DISTANCES FROM THE CENTER OF MASS OF SEGMENT 1 TO
C        THE CENTER OF MASS OF SEGMENT 2
C
C        REAL WK(480), ETA(12), H(12),R(12),ALPHA(11)
C        REAL BETA(11),
C        GAMMA(11),Q(12,3),QDOT(240,12,3),P(12,3)
C        REAL QDDOT(240,12,3),Q1(240,12,3),T(240),M(12),PG(3)
REAL Y(240),QD(240),RELRM(11,3,3),ROT(12,3,3)
CHARACTER*50 FNIN,FNOUT
INTEGER NFRAMES

C READ IN JOINT ROTATION ANGLES FROM FILE
C
open (1,file=fnin)
do j=1,nframes
  do i=1,11
    read(1,*)alpha(i)
    read(1,*)beta(i)
    read(1,*)gamma(i)
  enddo
C CALCULATE Q'S FOR EACH SEGMENT
C
call rotation(alpha,beta,gamma,relrm)
call rm(relrm,rot)
call cg(r,eta,rot,m,q,h,pg,p)
C STORE Q'S IN ARRAY
C
do i=1,12
  do k=1,3
    ql(j,i,k)=q(i,k)
  enddo
enddo
C AFTER ALL Q'S ARE CALCULATED, USE CUBIC SPLINES TO
C DETERMINE THE DERIVATIVES
C
do i=1,12
  do k=1,3
    do j=1,nframes
      t(j)=j*.02
      y(j)=ql(j,i,k)
    enddo
    call pchez(nframes,t,y,qd,.true.,wk,480,ierr)
call pchez(nframes,t,qd,y,.true.,wk,480,ierr)
do j=1,nframes
  qdot(j,i,k)=qd(j)
quddot(j,i,k)=y(j)
enddo
C WRITE THE DERIVATIVES TO A DATA FILE FOR READING INTO
C THE PROGRAM
C
  open(1,file=fnout)
do i=1,nframes
    do j=1,12
      do k=1,3
        write(1,*)
        q1(i,j,k),qdot(i,j,k),qddot(i,j,k)
      enddo
    enddo
  enddo
  close(1)
return
end

Subroutine CG
C
C   THIS SUBROUTINE COMPUTES THE CENTER OF GRAVITY OF THE
C   BODY WITH RESPECT TO THE ORIGIN OF THE REFERENCE SEGMENT
C   OF THE BODY.
C
SUBROUTINE CG(R,ETA,ROT,M,Q,H,PG,P)
REAL Q(12,3),H(12),VECTOR(3),V2(3),ROT(12,3,3)
REAL RRM(3,3),V3(3),V4(4)
REAL R(12),ETA(12),M(12),MTOT,PG(3),P(12,3)

C   DEFINE VARIABLES:Q(I,J)-LOCATION OF CENTER OF GRAVITY
C   OF SEGMENT I WITH RESPECT TO THE ORIGIN OF SEGMENT 1
C   THE C.G. OF THE BODY IS FOUND USING PG=THE SUM OF
C   (MI/M*QI)
C   FIRST, COMPUTE Q'S
C
mtot=0.
do i=1,12
  mtot=mtot+m(i)
enddo

C   SEGMENT 1: MID-TRUNK
C
  do i=1,3
    q(1,i)=0
  enddo

C   SEGMENT 2: UPPER TRUNK
C
  vector(1)=0.
  vector(2)=-h(2)/2.
  vector(3)=0.
do i=1,3
do j=1,3
    rrm(i,j)=rot(2,i,j)
enddo
enddo
call matvec(rrm,vector,v2)
q(2,1)=v2(1)
q(2,2)=v2(2)-h(1)/2.
q(2,3)=v2(3)

C SEGMENT 3: PELVIS
C
vector(1)=0.
vector(2)=h(3)/2.
vector(3)=0.
do i=1,3
    do j=1,3
        rrm(i,j)=rot(3,i,j)
    enddo
enddo
enddo
call matvec(rrm,vector,v2)
q(3,1)=v2(1)
q(3,2)=v2(2)+h(1)/2.
q(3,3)=v2(3)

C SEGMENT 4: HEAD
C
vector(1)=0.
vector(2)=-h(4)/2.
vector(3)=0.
do i=1,3
    do j=1,3
        rrm(i,j)=rot(4,i,j)
    enddo
enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=-h(2)
vector(3)=0.
do i=1,3
    do j=1,3
        rrm(i,j)=rot(2,i,j)
    enddo
enddo
call matvec(rrm,vector,v3)
q(4,1)=v2(1)+v3(1)
q(4,2)=v2(2)+v3(2)-h(1)/2.
q(4,3)=v2(3)+v3(3)

C SEGMENT 5: RIGHT FOREARM
C
vector(1)=0.
vector(2)=-eta(5)*h(5)
vector(3)=0.
do i=1,3
do j=1,3
    rrm(i,j)=rot(5,i,j)
enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=-h(7)
vector(3)=0.
do i=1,3
do j=1,3
    rrm(i,j)=rot(7,i,j)
enddo
enddo
call matvec(rrm,vector,v3)
vector(1)=0.
vector(2)=-h(2)
vector(3)=-r(2)
do i=1,3
do j=1,3
    rrm(i,j)=rot(2,i,j)
enddo
enddo
call matvec(rrm,vector,v4)
q(5,1)=v2(1)+v3(1)+v4(1)
q(5,2)=v2(2)+v3(2)+v4(2)-h(1)/2.
q(5,3)=v2(3)+v3(3)+v4(3)

C SEGMENT 6: LEFT FOREARM

C

vector(1)=0.
vector(2)=-eta(6)*h(6)
vector(3)=0.
do i=1,3
do j=1,3
    rrm(i,j)=rot(6,i,j)
enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=-h(8)
vector(3)=0.
do i=1,3
do j=1,3
    rrm(i,j)=rot(8,i,j)
enddo
enddo
call matvec(rrm,vector,v3)
vector(1)=0.
vector(2)=-h(2)
vector(3)=r(2)
do i=1,3
  do j=1,3
    rrm(i,j)=rot(2,i,j)
  enddo
enddo
call matvec(rrm,vector,v4)
q(6,1)=v2(1)+v3(1)+v4(1)
q(6,2)=v2(2)+v3(2)+v4(2)-h(1)/2.
q(6,3)=v2(3)+v3(3)+v4(3)
C
C SEGMENT 7: RIGHT UPPER ARM
C
vector(1)=0.
vector(2)=-eta(7)*h(7)
vector(3)=0.
do i=1,3
  do j=1,3
    rrm(i,j)=rot(7,i,j)
  enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=-h(2)
vector(3)=-r(2)
do i=1,3
  do j=1,3
    rrm(i,j)=rot(2,i,j)
  enddo
enddo
call matvec(rrm,vector,v3)
q(7,1)=v2(1)+v3(1)
q(7,2)=v2(2)+v3(2)-h(1)/2.
q(7,3)=v2(3)+v3(3)
C
C SEGMENT 8: LEFT UPPER ARM
C
vector(1)=0.
vector(2)=-eta(8)*h(8)
vector(3)=0.
do i=1,3
  do j=1,3
    rrm(i,j)=rot(8,i,j)
  enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=-h(2)
vector(3)=r(2)
do i=1,3
doo j=1,3
    rrm(i,j)=rot(2,i,j)
endo
dodo
  call matvec(rrm,vector,v3)
q(8,1)=v2(1)+v3(1)
q(8,2)=v2(2)+v3(2)-h(1)/2.
q(8,3)=v2(3)+v3(3)
endo
C
C SEGMENT 9: RIGHT THIGH SEGMENT
C
vector(1)=0.
vector(2)=eta(9)*h(9)
vector(3)=0.
do i=1,3
doo j=1,3
    rrm(i,j)=rot(9,i,j)
endo
dodo
  call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=h(3)
vector(3)=-r(9)
do i=1,3
doo j=1,3
    rrm(i,j)=rot(3,i,j)
endo
dodo
  call matvec(rrm,vector,v3)
q(9,1)=v2(1)+v3(1)
q(9,2)=v2(2)+v3(2)+h(1)/2.
q(9,3)=v2(3)+v3(3)
C
C SEGMENT 10: LEFT THIGH
C
vector(1)=0.
vector(2)=eta(10)*h(10)
vector(3)=0.
do i=1,3
doo j=1,3
    rrm(i,j)=rot(10,i,j)
endo
dodo
  call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=h(3)
vector(3)=r(9)
do i=1,3
   do j=1,3
      rrm(i,j)=rot(3,i,j)
   enddo
enddo
call matvec(rrm,vector,v3)
q(10,1)=v2(1)+v3(1)
q(10,2)=v2(2)+v3(2)+h(1)/2.
q(10,3)=v2(3)+v3(3)

SEGMENT 11: RIGHT SHANK

vector(1)=0.
vector(2)=eta(11)*h(11)
vector(3)=0.
do i=1,3
   do j=1,3
      rrm(i,j)=rot(11,i,j)
   enddo
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=h(9)
vector(3)=0.
do i=1,3
   do j=1,3
      rrm(i,j)=rot(9,i,j)
   enddo
enddo
call matvec(rrm,vector,v3)
vector(1)=0.
vector(2)=h(3)
vector(3)=-r(9)
do i=1,3
   do j=1,3
      rrm(i,j)=rot(3,i,j)
   enddo
enddo
call matvec(rrm,vector,v4)
q(11,1)=v2(1)+v3(1)+v4(1)
q(11,2)=v2(2)+v3(2)+v4(2)+h(1)/2.
q(11,3)=v2(3)+v3(3)+v4(3)

SEGMENT 12: LEFT SHANK

vector(1)=0.
vector(2)=eta(12)*h(12)
vector(3)=0.
do i=1,3
   do j=1,3


rrm(i,j)=rot(12,i,j)
enddo
do i=1,3
do j=1,3
   rrm(i,j)=rot(10,i,j)
enddo
call matvec(rrm,vector,v2)
vector(1)=0.
vector(2)=h(10)
vector(3)=0.
do i=1,3
do j=1,3
   rrm(i,j)=rot(10,i,j)
enddo
call matvec(rrm,vector,v3)
vector(1)=0.
vector(2)=h(3)
vector(3)=r(10)
do i=1,3
do j=1,3
   rrm(i,j)=rot(3,i,j)
enddo
call matvec(rrm,vector,v4)
q(12,1)=v2(1)+v3(1)+v4(1)
q(12,2)=v2(2)+v3(2)+v4(2)+h(1)/2.
q(12,3)=v2(3)+v3(3)+v4(3)
do j=1,3
   pg(j)=0.
enddo
do i=1,3
do j=1,12
   pg(i)=pg(i)+m(j)/Mtot*q(j,i)
enddo
do i=1,12
   do j=1,3
      p(i,j)=q(i,j)-pg(j)
   enddo
endo
return
end

Subroutine ROTATION
THIS SUBROUTINE SETS UP THE ROTATION MATRICES BETWEEN SEGMENTS FOR THE SIMULATION PROGRAM. THE INPUT IS JOINT ANGLES. THE OUTPUT IS THE ROTATION MATRICES.

SUBROUTINE ROTATION(ALPHA, BETA, GAMMA, RELRM)

JOINTS ARE NUMBERED AS FOLLOWS: 1-NECK, 2-R SHOULDER, 3-L SHOULDER, 4-R ELBOW, 5-L ELBOW, 6-UPPER BACK, 7-LOWER BACK, 8-RIGHT HIP, 9-LEFT HIP, 10-RIGHT KNEE, 11-LEFT KNEE

REAL RELRM(11,3,3), ALPHA(11), BETA(11), GAMMA(11)

DO I=1,11
   RELRM(i,1,1) = COS(ALPHA(i)) * COS(GAMMA(i)) - SIN(ALPHA(i)) * SIN(BETA(i)) * SIN(GAMMA(i))
   RELRM(i,1,2) = -SIN(ALPHA(i)) * COS(BETA(i))
   RELRM(i,1,3) = COS(ALPHA(i)) * SIN(GAMMA(i)) + SIN(ALPHA(i)) * SIN(BETA(i)) * COS(GAMMA(i))
   RELRM(i,2,1) = SIN(ALPHA(i)) * COS(GAMMA(i)) + COS(ALPHA(i)) * SIN(BETA(i)) * SIN(GAMMA(i))
   RELRM(i,2,2) = COS(ALPHA(i)) * COS(BETA(i))
   RELRM(i,2,3) = SIN(ALPHA(i)) * SIN(GAMMA(i)) - COS(ALPHA(i)) * SIN(BETA(i)) * COS(GAMMA(i))
   RELRM(i,3,1) = -SIN(GAMMA(i)) * COS(BETA(i))
   RELRM(i,3,2) = SIN(BETA(i))
   RELRM(i,3,3) = COS(BETA(i)) * COS(GAMMA(i))
ENDO

RETURN
END

SUBROUTINE RM

SUBROUTINE TO COMPUTE THE ROTATION MATRIX OF EACH SEGMENT RELATIVE TO SEGMENT 1

SUBROUTINE RM(RELRM, ROT)

REAL RELRM(11,3,3), ROT(12,3,3), A(3,3), B(3,3), C(3,3)

SEGMENT 1: MID TRUNK

DO I=1,3
   DO J=1,3
      IF (I.EQ.J) THEN
         ROT(1,I,J) = 1.
      ELSE
         ROT(1,I,J) = 0.
      ENDIF
   ENDDO
enddo

C SEGMENT 2:  UPPER TRUNK
C
do i=1,3
    do j=1,3
        rot(2,i,j)=relrm(6,i,j)
    enddo
endo
c
C SEGMENT 3:  PELVIS
C
do i=1,3
    do j=1,3
        rot(3,i,j)=relrm(7,i,j)
    enddo
endo
c
C SEGMENT 4:  HEAD
C
do i=1,3
    do j=1,3
        a(i,j)=relrm(6,i,j) !UPPER BACK
        b(i,j)=relrm(1,i,j) !NECK
    enddo
endo
c
call mm(a,b,c)
do i=1,3
    do j=1,3
        rot(4,i,j)=c(i,j)
    enddo
endo
c
C SEGMENT 7:  RIGHT UPPER ARM
C
do i=1,3
    do j=1,3
        a(i,j)=relrm(6,i,j) !UPPER BACK
        b(i,j)=relrm(2,i,j) !RIGHT SHOULDER
    enddo
endo
c
call mm(a,b,c)
do i=1,3
    do j=1,3
        rot(7,i,j)=c(i,j)
    enddo
endo
c
C SEGMENT 8:  LEFT UPPER ARM
C
do i=1,3
    do j=1,3
        a(i,j)=relrm(6,i,j) !UPPER BACK
        b(i,j)=relrm(3,i,j) !LEFT SHOULDER
    enddo
enddo
call mm(a,b,c)
do i=1,3
    do j=1,3
        rot(8,i,j)=c(i,j)
    enddo
enddo

SEGMENT 9: RIGHT THIGH

do i=1,3
    do j=1,3
        a(i,j)=relrm(7,i,j) !LOWER BACK
        b(i,j)=relrm(8,i,j) !RIGHT HIP
    enddo
enddo
call mm(a,b,c)
do i=1,3
    do j=1,3
        rot(9,i,j)=c(i,j)
    enddo
enddo

SEGMENT 10: LEFT THIGH

do i=1,3
    do j=1,3
        a(i,j)=relrm(7,i,j) !LOWER BACK
        b(i,j)=relrm(9,i,j) !LEFT HIP
    enddo
enddo
call mm(a,b,c)
do i=1,3
    do j=1,3
        rot(10,i,j)=c(i,j)
    enddo
enddo

SEGMENT 5: RIGHT FOREARM

do i=1,3
    do j=1,3
        a(i,j)=rot(7,i,j) !RIGHT UPPER ARM
        b(i,j)=relrm(4,i,j) !RIGHT ELBOW
    enddo
enddo
call mm(a,b,c)
do i=1,3
do j=1,3
   rot(5,i,j)=c(i,j)
enddo
enddo

C C SEGMENT 6: LEFT FOREARM C
C
do i=1,3
do j=1,3
   a(i,j)=rot(8,i,j) !LEFT UPPER ARM
   b(i,j)=relrm(5,i,j) !left elbow
enddo
enddo
call mm(a,b,c)
do i=1,3
do j=1,3
   rot(6,i,j)=c(i,j)
enddo
enddo

C C SEGMENT 11: RIGHT SHANK C
C
do i=1,3
do j=1,3
   a(i,j)=rot(9,i,j) !RIGHT THIGH
   b(i,j)=relrm(10,i,j) !RIGHT KNEE
enddo
enddo
call mm(a,b,c)
do i=1,3
do j=1,3
   rot(11,i,j)=c(i,j)
enddo
enddo

C C SEGMENT 12: LEFT SHANK C
C
do i=1,3
do j=1,3
   a(i,j)=rot(10,i,j) !LEFT THIGH
   b(i,j)=relrm(11,i,j) !LEFT KNEE
enddo
enddo
call mm(a,b,c)
do i=1,3
do j=1,3
   rot(12,i,j)=c(i,j)
Subroutine MATVEC

SUBROUTINE MATVEC(R,V,PRODUCT)

C

C THIS SUBROUTINE MULTIPLIES A 3X3 MATRIX R BY A VECTOR V
C
REAL R(3,3),V(3),PRODUCT(3)
do i=1,3
    product(i)=0.
enddo
do i=1,3
do j=1,3
    product(i)=product(i)+r(i,j)*v(j)
enddo
return
end

Subroutine MM

SUBROUTINE MM(M1,M2,M3)

C

C THIS SUBROUTINE COMPUTES THE PRODUCT OF MATRICES M1 AND M2
C AND RETURNS THE PRODUCT IN M3
C
REAL M1(3,3),M2(3,3),M3(3,3)
do 6000 i=1,3
do 6000 j=1,3
    m3(i,j)=m1(i,1)*m2(1,j)+m1(i,2)*m2(2,j)+m1(i,3)*m2(3,j)
6000 continue
return
end

Subroutine INTERPOLATEQ

SUBROUTINE INTERPOLATEQ(FN1, FN2, NF)

REAL A(12,4), ADD(12,4), B(12,4)
REAL BDD(12,4), G(12,4), GDD(12,4), AI(12,80)
REAL BI(12,80), GI(12,80), ADI(12,80), BDI(12,80), D(4)
REAL GDI(12,80), ADDI(12,80), BDDI(12,80), GDDI(12,80)
REAL T(4), WK(8), T1(80), F1(80), F(4), D1(80)
CHARACTER*50, FN1, FN2
INTEGER NF
open(20, file=fn2)
f=fn/4
open (15, file=fn1)
do k=1,nf

do j=1,4
  t(j) = .02*(j-1)
doi=1,12
  read(15,*)a(i,j),dummy,add(i,j)
  read(15,*)b(i,j),dummy,bdd(i,j)
  read(15,*)g(i,j),dummy,gdd(i,j)
dendo
endo
doi=1,80
  t1(i) = .001*(i-1)
dendo
doi=1,12
  do j=1,4
    f(j) = a(i,j)
dendo
call pchez(4,t,f,d,.true.,wk,8,ierr)
call pchev(4,t,f,d,80,t1,f1,d1,ierr)
do j=1,80
  ai(i,j) = f1(j)
  ad(i,j) = d1(j)
dendo
doi=1,4
  f(j) = b(i,j)
dendo
call pchez(4,t,f,d,.true.,wk,8,ierr)
call pchev(4,t,f,d,80,t1,f1,d1,ierr)
do j=1,80
  bi(i,j) = f1(j)
  bdi(i,j) = d1(j)
dendo
doi=1,4
  f(j) = g(i,j)
dendo
call pchez(4,t,f,d,.true.,wk,8,ierr)
call pchev(4,t,f,d,80,t1,f1,d1,ierr)
do j=1,80
  gi(i,j) = f1(j)
  gdi(i,j) = d1(j)
dendo
doi=1,4
  f(j) = add(i,j)
dendo
call pchez(4,t,f,d,.true.,wk,8,ierr)
call pchev(4,t,f,d,80,t1,f1,d1,ierr)
do j=1,80
  addi(i,j) = f1(j)
dendo
doi=1,4
  f(j) = bdd(i,j)
enddo
call pchez(4,t,f,d,.true.,wk,8,ierr)
call pchev(4,t,f,d,80,t1,f1,d1,ierr)
do j=1,80
 bddi(i,j)=f1(j)
enddo
do j=1,4
 f(j)=gdd(i,j)
enddo
enddo
enddo
enddo
enddo
enddo
enddo
enddo
enddo
enddo
close(15)
close(20)
return
dern

Subroutine GETR
C
C THIS SUBROUTINE USES AN ITERATIVE PROCESS TO CALCULATE
THE
RADIUS OF THE CENTER OF GRAVITY VECTOR FROM THE ORIGIN
OF
THE GLOBAL COORDINATE SYSTEM
C
SUBROUTINE GETR(R,NFRAMES,M,L,E,INERT,RB,TB)

C VARIABLES USED:
C INPUT:  CGANG=ANGLE OF THE CENTER OF GRAVITY FROM THE
C ORIGIN
C LH=VECTOR FROM THE CENTER OF GRAVITY TO THE LEFT
C HAND IN REFERENCE SEGMENT COORDINATES
C RH=VECTOR FROM THE CG TO THE RIGHT HAND IN R.S.
C COORD.
C ZH=DISTANCE OF EACH HAND FROM THE CENTER OF THE
BAR
C (WHICH IS THE ORIGIN OF THE GLOBAL COORDINATE
C SYSTEM)
C NFRAMES=THE NUMBER OF FRAMES IN THE GIANT SWING
PHASE

M = THE TOTAL MASS OF THE BODY
L = THE LENGTH OF THE BAR
E = YOUNG'S MODULUS OF THE BAR
INERT = AREA MOMENT OF INERTIA OF THE BAR

OUTPUT: R = THE RADIUS OF THE C.G. VECTOR FROM THE ORIGIN
FTOT = THE TOTAL FORCE OF BOTH HANDS ON THE BAR
RB = THE DISTANCE OF THE CENTER OF THE BAR FROM THE ORIGIN OF THE GLOBAL COORDINATE SYSTEM
(= MAGNITUDE OF DEFLECTION)

VARIABLES USED IN COMPUTATIONS:
RH1 = THE XY DISTANCE FROM THE CG TO THE RIGHT HAND
RH2 = THE XY DISTANCE FROM THE CG TO THE LEFT HAND
RH = THE XY DISTANCE FROM THE CG TO THE CENTER OF THE BAR
(TAKEN AS THE AVERAGE OF THE LEFT AND RIGHT HAND DISTANCES)
RLAST = TEMPORARY STORAGE ARRAY FOR THE PREVIOUS ITERATION OF R
WK, IERR = ARRAYS AND VALUES FOR THE CUBIC SPLINES ROUTINE
T = TIME
TDOT = THE FIRST TIME DERIVATIVE OF CGANG
TDDOT = THE SECOND TIME DERIVATIVE OF CGANG
RDCT = THE FIRST TIME DERIVATIVE OF R
RDDOT = THE SECOND TIME DERIVATIVE OF R
T1 = THE ANGLE BETWEEN R AND RB
T2 = THE ANGLE BETWEEN RB AND RH
T3 = THE ANGLE BETWEEN R AND RH

DECLARE VARIABLES

REAL WK(540), M, L, E, INERT
REAL CGANG(270), LH(270, 3), RH(270, 3), R(270), FTOT(270, 2) & ZH(2)
REAL RH1, RH2, RRH(270), RB(270), RLAST(270), T(270), TDOT(270) & TB(270)
REAL TDDOT(270), RDOT(270), RDDOT(270), T1(270), T2(270), T3(270) & PI
COMMON/CF2/LH, RH, FTOT, ZH, CGANG
pi=acos(-1.)

C FIRST, FIND RH AND USE IT AS THE FIRST ESTIMATE OF R

open (33, file='c:\for\data\handloc.dat')
do i=1,nframes
write(33,*)rh(i,1),rh(i,2),lh(i,1),lh(i,2)
enddo
close(33)
do i=1,nframes
    rh1=sqrt(rh(i,1)**2+rh(i,2)**2+rh(i,3)**2-zh(l)**2)
    rh2=sqrt(lh(i,1)**2+lh(i,2)**2+lh(i,3)**2-zh(2)**2)
    r(i)=r(i)=(rh1+rh2)/2.
enddo

C NEXT, SET UP TIME ARRAY AND GET DERIVATIVES OF CGANG

do i=1,nframes
    t(i)=i*0.02
enddo
call pchez(nframes,t,cgang,tdot,.true.,wk,270,ierr)
call pchez(nframes,t,tdot,tddot,.true.,wk,270,ierr)

C NOW, CALCULATE R

C GET DERIVATIVES OF R

call pchez(nframes,t,r,rdot,.true.,wk,270,ierr)
call pchez(nframes,t,rdot,rddot,.true.,wk,270,ierr)

C CALCULATE TOTAL FORCE AT THE HAND FROM THE ACCELERATIONS
FROM THESE R,CGANG VALUES USING EQUATIONS FOR
ACCELERATION

open (33, file='c:\for\data\r.dat')
do i=1,nframes

ftot(i,1)=m*(rddot(i)-r(i)*tdot(i)**2)-m*981*sin(cgang(i))
ftot(i,2)=m*(r(i)*tddot(i)+2*rdot(i)*tdot(i))
& -m*981*cos(cgang(i))

C USE A SIMPLY SUPPORTED BEAM MODEL, AND CALCULATE THE BAR
DEFLECTION ASSOCIATED WITH THESE FORCES

rb(i)=sqrt(ftot(i,1)**2+ftot(i,2)**2)*L**3/(48.*E*Inert)

C LET THERE BE A MAXIMUM BAR DEFLECTION OF 25 CM, ASSUME
LARGER DEFLECTIONS ARE ABERRATIONS
if (rb(i).Gt.20) Then
  rb(i)=20
endif

    t1(i)=abs(atan2(f tot(i,2),f tot(i,1))-cgang(i))
if (t1(i).gt.pi) then
  t1(i)=2.*pi-t1(i)
endif

write(33,*)Ftot(i,1),f tot(i,2),r(i),rdot(i),rddot(i),rb(i)
write(33,*)cgang(i),tdot(i),tddot(i)
enddo

close(33)

C CALCULATE THE NEW R=RH+RB (USING VECTOR GEOMETRY)

errsum=0.
do i=1,nframes
  rlast(i)=r(i)
  if(((rb(i)*sin(t1(i))/rrh(i)).gt.1.) then
    t3(i)=pi/2.
  else
    t3(i)=asin(rb(i)*sin(t1(i))/rrh(i))
  endif
  t2(i)=pi-t1(i)-t3(i)
  if (sin(t3(i)).eq.0) then
    if (t1(i).eq.0) then
      r(i)=rrh(i)-rb(i)
    else
      r(i)=rrh(i)+rb(i)
    endif
  endif
  if (sin(t3(i)).eq.0) then
    r(i)=rb(i)
  else
    r(i)=rb(i)*sin(t2(i))/sin(t3(i))
  endif
endo
do i=1,nframes
  tb(i)=cgang(i)-t1(i)
endo

C CHANGE FORCES FROM POLAR TO RECTANGULAR COORDINATES

do i=1,nframes
  fx=f tot(i,1)*cos(cgang(i))-f tot(i,2)*sin(cgang(i))
  fy=f tot(i,1)*sin(cgang(i))+f tot(i,2)*cos(cgang(i))
  f tot(i,1)=fx
  f tot(i,2)=fy
endo
Subroutine HANDLOC

SUBROUTINE TO COMPUTE LOCATION OF FORCE APPLICATION (HANDS) WITH RESPECT TO BODY CENTER OF GRAVITY (IN B COORDINATES)

SUBROUTINE HANDLOC(PLH, PRH, P, ROT, ETA, H)

REAL PLH(3), PRH(3), P(12, 3), VECTOR(3), V2(3), ROT(12, 3, 3)
REAL RRM(3, 3), ETA(12), H(12), HLEN
hlen=9.04

LOCATION OF HANDS=LOCATION OF FOREARM C.G. + VECTOR FROM FOREARM C.G. TO HAND

RIGHT FOREARM IS SEGMENT 5, LEFT FOREARM IS SEGMENT 6

DEFINE THE VECTOR FROM THE CG OF THE FOREARM TO THE HAND

vector(1)=0.
vector(2)=-(1-eta(5))*h(5)-hlen
vector(3)=0.

ROTATE VECTORS INTO MID-TRUNK COORDINATES

do i=1,3
  do j=1,3
    rrm(i, j)=rot(5, i, j)
  enddo
enddo
call matvec(rrm, vector, v2)

DEFINE P

do i=1,3
  prh(i)=v2(i)+p(5, i)
endo

REPEAT FOR LEFT HAND LOCATION VECTOR

DEFINE THE VECTOR FROM THE CG OF THE FOREARM TO THE HAND

vector(1)=0.
vector(2)=-(1-eta(6))*h(6)-hlen
vector(3)=0.
C ROTATE VECTORS INTO MID-TRUNK COORDINATES

C do i=1,3
  do j=1,3
    rrm(i,j)=rot(6,i,j)
  enddo
enddo

call matvec(rrm,vector,v2)

C DEFINE P

C do i=1,3
  plh(i)=v2(i)+p(6,i)
enddo

return
end

Subroutine INTERPOLATE1

SUBROUTINE INTERPOLATE1(X,FN8,N,XDOTREL)
REAL X(170,3),XX1,XX2,XX3
REAL XDD1,XDD2,XDD3,XDOTREL(2)
REAL A(170),T(170),D(170),WK(340)

CHARACTER*50 FN8,FN1,FN2
lwlc=340

do i=1,n
  a(i)=x(i,1)
  t(i)=.02*i
enddo

call pchez(n,t,a,d,.true.,wk,lwk,ierr)

xdotrel(1)=d(n)

fn1='c:\fortran\dummy1.fil'
fn2='c:\fortran\dummy2.fil'

call interpolate(a,fn1,n,1)
do i=1,n
  a(i)=x(i,2)
endo

call pchez(n,t,a,d,.true.,wk,lwk,ierr)

xdotrel(2)=d(n)
call interpolate(a,fn2,n,1)

xx3=x(1,3)
xd3=0.

Open(20,file=fn8)
open(21,file=fn1)
open(22,file=fn2)
do i=1,n*20
  read (21,*) xx1,xdd1
  read (22,*) xx2,xdd2
  write(20,*) xx1,xdd1
  write(20,*) xx2,xdd2
  write(20,*) xx3,xdd3
endo
enddo
close(20)
close(21)
close(22)
return
end

Subroutine INTERPOLATEA

SUBROUTINE INTERPOLATEA(X,FN,N,FLAG)
INTEGER FLAG
REAL X(170),XX1(3400),T(170),T1(3400),D(170),WK(170) & DVAL(3400)
REAL XDD(3400),DD(170)
CHARACTER*50 FN
lwk=340
do i=l,n
   t(i)=.02*(I-1)
endo
do i=1,n*20
   t1(i)=.001*(I-1)
endo
call pchez(n,t,x,d,.True.,WK,lwk,ierr)
call pchev(n,t,x,d,n*20,t1,xx1,dval,ierr)
if (flag.eq.1) then
call pchez(n,t,d,dd,.true.,WK,lwk,ierr)
call pchez(n,t,dd,d,.true.,WK,lwk,ierr)
call pchev(n,t,dd,d,n*20,t1,xdd,dval,ierr)
open(50,file=fn)
do i=1,n*20
   write(50,*) xx1(i),xdd(i)
endo
close(50)
elseif (flag.eq.2) then
open(50,file=fn)
do i=1,n*20
   xdd(i)=dval(i)
endo
call pchez(n,t,d,dd,.true.,WK,lwk,ierr)
call pchev(n,t,d,dd,n*20,t1,xdd,dval,ierr)
do i=1,n*20
   write(50,*) xx1(i)
   write(50,*) xdd(i)
   write(50,*) dval(i)
endo
close(50)
else
open(50,file=fn)
do i=1,n*20
   write(50,*) xx1(i)
endo
close(50)
close(50)
endif
return
end

Subroutine INTERPOLATEB
SUBROUTINE INTERPOLATEB(X,FN,N)
REAL X(170,2),XX1,XX2
REAL A(170)
CHARACTER*50 FN,FN1
do i=1,n
  a(i)=x(i,1)
enddo
fnl='c:\fortran\duininyl.fil'
call interpolatea(a,fnl,n,0)
do i=1,n
  a(i)=x(i,2)
enddo
fnl='c:\fortran\duiiuny2. f il'
call interpolatea(a,fnl,n,0)
open(20,file=fn)
open(21,file='c:\fortran\duitiitiyl. fil')
open(22, file='c; \fortran\duininy2. f il')
do i=1,n*20
  read(21,*)xx1
  read(22,*)xx2
  write(20,*) xx1,xx2
enddo
close(20)
close(21)
close(22)
return
end

Subroutine ANGV
C THIS SUBROUTINE FINDS THE ANGULAR VELOCITY OF EACH SEGMENT
C RELATIVE TO THE MID-TRUNK SEGMENT IN MID-TRUNK (SEGMENT 1)
C COORDINATES.
C SUBROUTINE ANGV(RW)
C SEGMENTS ARE NUMBERED AS FOLLOWS: 1-MID TRUNK, 2-UPPER TRUNK, 3-PELVIS, 4-HEAD, 5-RIGHT FOREARM, 6-LEFT FOREARM,
C 7-RIGHT UPPER ARM, 8-LEFT UPPER ARM, 9-RIGHT THIGH, 10-LEFT THIGH, 11-RIGHT SHANK, 12-LEFT SHANK.
C
REAL
ALPHA(ll), BETA(ll), GAMMA(ll), ALPHADOT(ll), BETADOT(ll)
REAL GAMMADOT(ll), ALPHADDOT(ll), BETADDOT(ll), GAMMADDOT(ll)
REAL RELRM(11,3,3), U(12,3), UPRIME(12,3), RW(12,3)
REAL RRM(3,3), UROT(3), UBACK(3)
INTEGER SEG, JT
C
COMMON/ANGLES/ALPHA, BETA, GAMMA, ALPHADOT, BETADOT, GAMMADOT,
& ALPHADDOT, BETADDOT, GAMMADDOT, RELRM, U, UPRIME
C
C SEGMENT 1
C
do i=1,3
   rw(1,i)=0.0
   u(1,i)=0.0
endo
C
C SEGMENT 2-UPPER TRUNK
C
call ucalc(6, alpha, alphadot, beta, betadot, gammadot, u, 2)
do i=1,3
   rw(2,i)=u(2,i)
endo
C
C SEGMENT 3-PELVIS
C
call ucalc(7, alpha, alphadot, beta, betadot, gammadot, u, 3)
do i=1,3
   rw(3,i)=u(3,i)
endo
C
C SEGMENT 4-HEAD
C IN TERMS OF UPPER TRUNK COORDINATES
C
call ucalc(1, alpha, alphadot, beta, betadot, gammadot, u, 4)
C
C ROTATE TO MID-TRUNK COORDINATES
C
seg=4
jt=6
do i=1,3
   urot(i)=u(seg,i)
do j=1,3
   rrm(i,j)=relrm(jt, i, j)
endo
endo
call matvec(rrm,urot,uback)
do i=1,3
   uprime(seg,i)=uback(i)
   rw(seg,i)=uprime(seg,i)+u(2,i)
endo

C C SEGMENT 7—RIGHT UPPER ARM
C IN UPPER TRUNK COORDINATES
C
seg=7
jt=2
call ucalc(jt, alpha, alphadot, beta, betadot, gammadot, u, seg)
C C ROTATE INTO MID TRUNK COORDINATES
C
do i=1,3
   urot(i)=u(seg,i)
endo
call matvec(rrm,urot,uback)
do i=1,3
   uprime(seg,i)=uback(i)
   rw(seg,i)=uprime(seg,i)+u(2,i)
endo

C C SEGMENT 8—LEFT UPPER ARM
C IN UPPER TRUNK COORDINATES
C
seg=8
jt=3
call ucalc(jt, alpha, alphadot, beta, betadot, gammadot, u, seg)
C C ROTATE INTO MID TRUNK COORDINATES
C
do i=1,3
   urot(i)=u(seg,i)
endo
call matvec(rrm,urot,uback)
do i=1,3
   uprime(seg,i)=uback(i)
   rw(seg,i)=uprime(seg,i)+u(2,i)
endo

C C SEGMENT 9—RIGHT THIGH
C IN PELVIS COORDINATES
C
seg=9
jt=8
call ucalc(jt, alpha, alphadot, beta, betadot, gammadot, u, seg)
C
C ROTATE INTO MID TRUNK COORDINATES
C
  do i=1,3
    urot(i)=u(seg,i)
    do j=1,3
      rrm(i,j)=relrm(7,i,j)
    enddo
  enddo
call matvec(rrm, urot, uback)
  do i=1,3
    uprime(seg,i)=uback(i)
    rw(seg,i)=uprime(seg,i)+u(3,i)
  enddo
C
C SEGMENT 10-LEFT THIGH
C IN PELVIS COORDINATES
C
seg=10
t=9
call ucalc(jt, alpha, alphadot, beta, betadot, gammadot, u, seg)
C
C ROTATE INTO MID TRUNK COORDINATES
C
  do i=1,3
    urot(i)=u(seg,i)
  enddo
call matvec(rrm, urot, uback)
  do i=1,3
    uprime(seg,i)=uback(i)
    rw(seg,i)=uprime(seg,i)+u(3,i)
  enddo
C
C SEGMENT 5-RIGHT FOREARM
C
  do i=1,3
    urot(i)=relrm(2,i,3)
    do j=1,3
      rrm(i,j)=relrm(6,i,j)
    enddo
  enddo
call matvec(rrm, urot, uback)
  do i=1,3
    u(5,i)=alphadot(4)*uback(i)
    rw(5,i)=u(5,i)+rw(7,i)
  enddo
C SEGMENT 6-LEFT FOREARM

do i=1,3
    urot(i)=relrm(3,i,3)
enddo
call matvec(rrm,urot,uback)
do i=1,3
    u(6,i)=alphadot(5)*uback(i)
    rw(6,i)=u(6,i)+rw(8,i)
enddo

C SEGMENT 11-RIGHT SHANK

do i=1,3
    urot(i)=relrm(8,i,3)
    do j=1,3
        rrm(i,j)=relrm(7,i,j)
    enddo
call matvec(rrm,urot,uback)
do i=1,3
    u(11,i)=uback(i)*alphadot(10)
    rw(11,i)=u(11,i)+rw(9,i)
enddo

C SEGMENT 12-LEFT SHANK

do i=1,3
    urot(i)=relrm(9,i,3)
enddo
call matvec(rrm,urot,uback)
do i=1,3
    u(12,i)=alphadot(11)*uback(i)
    rw(12,i)=u(12,i)+rw(10,i)
enddo
return
end

Subroutine UCALC

SUBROUTINE UCALC(JT,A,ADOT,B,BDOT,GDOT,U,SEG)
REAL A(11),ADOT(11),B(11),BDOT(11),GDOT(11),U(12,3)
INTEGER SEG,JT

u(seg,1)=bdot(jt)*cos(a(jt)) - gdot(jt)*cos(b(jt))*sin(a(jt))
u(seg,2)=bdot(jt)*sin(a(jt)) + gdot(jt)*cos(b(jt))*cos(a(jt))
u(seg,3)=adot(jt) + gdot(jt)*sin(b(jt))
return
end
Subroutine ANGAC

C THIS SUBROUTINE COMPUTES TERMS OF THE ANGULAR
C ACCELERATION OF EACH SEGMENT IN SEGMENT 1 (MID-TRUNK)
C SEGMENT COORDINATES
C EACH SEGMENT'S ANGULAR VELOCITY CAN BE WRITTEN IN THE
C FOLLOWING FORM:
C ANGACC= (ANGACC11+A1-B*U(1,3)+C*U(1,2))B1+(ANGACC12+A2
C +D*U(1,3)-C*U(1,1))B2+(ANGACC13+A3+B*U(1,1)-D*U(1,2))B3
C ANGACC11'S AND U(1,1)'S ARE IN TERMS OF UNKNOWN
C ALPHA, BETA, GAMMA. ALL OTHER TERMS ARE CONSTANTS
C (A,B,C,D) THIS SUBROUTINE FINDS A-D FOR EACH SEGMENT

SUBROUTINE ANGACC(A,B,C,D)
REAL U(12,3), ALPHA(11), BETA(11), GAMMA(11), ALPHADOT(11)
REAL BETADOT(11), GAMMADOT(11), ALPHADDOT(11), BETADDOT(11)
REAL GAMMADDOT(11), RRMDOT(11,3,3), A(12,3), B(12)
REAL C(12), D(12), UDOT(12,3), UDP(12,3), UPRIME(12,3)
REAL RELRM(11,3,3)
INTEGER JT, SEG

COMMON/ANGLES/ ALPHA, BETA, GAMMA, ALPHADOT, BETADOT, GAMMADOT,
& ALPHADDOT, BETADDOT, GAMMADDOT, RELRM, U, UPRIME

C IF THE ANGLE'S SECOND DERIVATIVE IS HIGH, IT PROBABLY
C CHANGED SIGN, I.E. WENT FROM POSITIVE TO NEGATIVE PI.
C
C COMPUTE DERIVATIVES OF ROTATION MATRIX TERMS
C
DO I=1,11
IF (ALPHADDOT(I).GT.100) THEN
ALPHADDOT(I)=0.
ENDIF
IF (BETADDOT(I).GT.100) THEN
BETADDOT(I)=0.
ENDIF
IF (GAMMADDOT(I).GT.100) THEN
GAMMADDOT(I)=0.
ENDIF
END DO

DO I=1,11
RRMDOT(I,1,1)=-ALPHADOT(I)*SIN(ALPHA(I))
& *COS(GAMMA(I))-GAMMADOT(I)*COS(ALPHA(I))
& *SIN(GAMMA(I))-ALPHADOT(I)*COS(ALPHA(I))
& *SIN(BETA(I))*SIN(GAMMA(I))-BETADOT(I)
& *SIN(ALPHA(I))*COS(BETA(I))*SIN(GAMMA(I))
& \ -gammadot(I) *\sin(alpha(I)) *\sin(beta(I)) \\
& \ \ *\cos(gamma(I)) \\
rrmdot(i,1,2)=\-alphadot(I) *\cos(alpha(I)) \\
& \ \ *\cos(beta(I)) +betadot(I) *\sin(alpha(I)) \\
& \ \ *\sin(beta(I)) \\
rrmdot(i,1,3)=\-alphadot(I) *\sin(alpha(I)) \\
& \ \ *\sin(gamma(I)) +gammadot(I) *\cos(alpha(I)) \\
& \ \ *\cos(gamma(I)) +alphadot(I) *\cos(alpha(I)) \\
& \ \ *\sin(beta(I)) *\sin(gamma(I)) \\
& \ \ +betadot(I) *\sin(alpha(I)) *\cos(beta(I)) \\
& \ \ *\cos(gamma(I)) -gammadot(I) *\sin(alpha(I)) \\
& \ \ *\sin(beta(I)) *\sin(gamma(I)) +betadot(I) *\cos(alpha(I)) *\cos(beta(I)) *\sin(gamma(I)) +gammadot(I) *\cos(alpha(I)) *\sin(beta(I)) \\
& \ \ *\cos(gamma(I)) \\
rrmdot(i,2,1)=alphadot(i) *\cos(alpha(I)) \\
& \ \ *\sin(gamma(I)) -alphadot(I) *\sin(alpha(I)) \\
& \ \ *\cos(alpha(I)) +alphadot(I) *\sin(alpha(I)) \\
& \ \ *\sin(beta(I)) *\cos(gamma(I)) -betadot(I) *\cos(alpha(I)) *\cos(beta(I)) *\cos(gamma(I)) +gammadot(I) *\cos(alpha(I)) *\sin(beta(I)) \\
rrmdot(i,2,2)=alphadot(I) *\sin(alpha(I)) \\
& \ \ *\cos(beta(I)) -betadot(I) *\cos(alpha(I)) \\
& \ \ *\sin(beta(I)) \\
rrmdot(i,2,3)=alphadot(i) *\sin(alpha(I)) \\
& \ \ *\cos(beta(I)) -betadot(I) *\cos(alpha(I)) \\
& \ \ *\sin(beta(I)) \\
rrmdot(i,3,1)=betadot(I) *\sin(beta(I)) *\sin(gamma(I)) \\
& \ \ -gammadot(I) *\cos(beta(I)) *\cos(gamma(I)) \\
rmdot(i,3,2)=betadot(I) *\cos(beta(I)) \\
& \ \ -gammadot(I) *\cos(beta(I)) *\sin(gamma(I)) \\
rrmdot(i,3,3)=-betadot(i) *\sin(beta(I)) *\cos(gamma(I)) \\
& \ \ -gammadot(I) *\cos(beta(I)) *\sin(gamma(I)) \\
enddo

C
C COMPUTE ANGULAR ACCELERATION TERMS SEGMENT BY SEGMENT
C
C SEGMENT 2 UPPER TRUNK
C
seg=2
jt=6

call uudot(jt, alpha, alphadot, alphaddot, beta, betadot \\
&betaddot, gammadot, gammaddot, udot, seg)
a(2,1)=udot(2,1) 
a(2,2)=udot(2,2) 
a(2,3)=udot(2,3) 
b(2)=u(2,2)
c(2)=u(2,3)
d(2)=u(2,1)

SEGMENT 3 PELVIS

seg=3
jt=7
call uudot(jt, alpha, alphadot, alphaddot, beta, betadot &
, betaddot, gammadot, gammaddot, udot, seg)
A(3,1)=udot(3,1)
A(3,2)=udot(3,2)
a(3,3)=udot(3,3)
b(3)=u(3,2)
c(3)=u(3,3)
d(3)=u(3,1)

SEGMENT 4 HEAD

seg=4
jt=1
call uudot(jt, alpha, alphadot, alphaddot, beta, betadot &
, betaddot, gammadot, gammaddot, udot, seg)
do i=1,3
  udp(seg, i)=udot(seg, 1)*relrm(jt, i, 1)+u(seg, 1)
  & *rrmdot(jt, i, 1)+udot(seg, 2)*relrm(jt, i, 2)
  & +u(seg, 2)*rrmdot(jt, i, 2)+udot(seg, 3)*relrm(jt, i, 3)
  & +u(seg, 3)*rrmdot(jt, i, 3)
enddo
a(seg, 1)=udot(2, 1)+udp(seg, 1)
a(seg, 2)=udot(2, 2)+udp(seg, 2)
a(seg, 3)=udot(2, 3)+udp(seg, 3)
b(seg)=u(2, 2)+uprime(seg, 2)
c(seg)=u(2, 3)+uprime(seg, 3)
d(seg)=u(2, 1)+uprime(seg, 1)

SEGMENT 7 RIGHT UPPER ARM

seg=7
jt=2
call uudot(jt, alpha, alphadot, alphaddot, beta, betadot &
, betaddot, gammadot, gammaddot, udot, seg)
do i=1,3
  udp(seg, i)=udot(seg, 1)*relrm(jt, i, 1)+u(seg, 1)
  & *rrmdot(jt, i, 1)+udot(seg, 2)*relrm(jt, i, 2)
  & +u(seg, 2)*rrmdot(jt, i, 2)+udot(seg, 3)*relrm(jt, i, 3)
  & +u(seg, 3)*rrmdot(jt, i, 3)
enddo
a(seg, 1)=udot(2, 1)+udp(seg, 1)
a(seg, 2)=udot(2, 2)+udp(seg, 2)
a(seg, 3)=udot(2, 3)+udp(seg, 3)
\[ b(\text{seg}) = u(2,2) + \text{uprime}(\text{seg},2) \]
\[ c(\text{seg}) = u(2,3) + \text{uprime}(\text{seg},3) \]
\[ d(\text{seg}) = u(2,1) + \text{uprime}(\text{seg},1) \]

SEGMENT 8 LEFT UPPER ARM

\[ \text{seg}=8 \]
\[ \text{jt}=3 \]
\[ \text{call uudot(jt, alpha, alphadot, alphaddot, beta, betadot, betaddot, gammadot, gammaddot, udot, seg)} \]
\[ \text{do } i=1,3 \]
\[ \quad \text{udp}(\text{seg},i) = \text{udot}(\text{seg},1) * \text{relrm}(\text{jt},i,1) + u(\text{seg},1) \]
\[ \quad \text{& } * \text{rrmdot}(\text{jt},i,1) + \text{udot}(\text{seg},2) * \text{relrm}(\text{jt},i,2) \]
\[ \quad \text{& } + u(\text{seg},2) * \text{rrmdot}(\text{jt},i,2) + \text{udot}(\text{seg},3) * \text{relrm}(\text{jt},i,3) \]
\[ \quad \text{& } + u(\text{seg},2) * \text{rrmdot}(\text{jt},i,3) \]
\[ \text{endo} \]
\[ a(\text{seg},1) = \text{udot}(2,1) + \text{udp}(\text{seg},1) \]
\[ a(\text{seg},2) = \text{udot}(2,2) + \text{udp}(\text{seg},2) \]
\[ a(\text{seg},3) = \text{udot}(2,3) + \text{udp}(\text{seg},3) \]
\[ b(\text{seg}) = u(2,2) + \text{uprime}(\text{seg},2) \]
\[ c(\text{seg}) = u(2,3) + \text{uprime}(\text{seg},3) \]
\[ d(\text{seg}) = u(2,1) + \text{uprime}(\text{seg},1) \]

SEGMENT 9 RIGHT THIGH

\[ \text{seg}=9 \]
\[ \text{jt}=8 \]
\[ \text{call uudot(jt, alpha, alphadot, alphaddot, beta, betadot, betaddot, gammadot, gammaddot, udot, seg)} \]
\[ \text{do } i=1,3 \]
\[ \quad \text{udp}(\text{seg},i) = \text{udot}(\text{seg},1) * \text{relrm}(\text{jt},i,1) + u(\text{seg},1) \]
\[ \quad \text{& } * \text{rrmdot}(\text{jt},i,1) + \text{udot}(\text{seg},2) * \text{relrm}(\text{jt},i,2) \]
\[ \quad \text{& } + u(\text{seg},2) * \text{rrmdot}(\text{jt},i,2) + \text{udot}(\text{seg},3) * \text{relrm}(\text{jt},i,3) \]
\[ \quad \text{& } + u(\text{seg},2) * \text{rrmdot}(\text{jt},i,3) \]
\[ \text{endo} \]
\[ a(\text{seg},1) = \text{udot}(3,1) + \text{udp}(\text{seg},1) \]
\[ a(\text{seg},2) = \text{udot}(3,2) + \text{udp}(\text{seg},2) \]
\[ a(\text{seg},3) = \text{udot}(3,3) + \text{udp}(\text{seg},3) \]
\[ b(\text{seg}) = u(3,2) + \text{uprime}(\text{seg},2) \]
\[ c(\text{seg}) = u(3,3) + \text{uprime}(\text{seg},3) \]
\[ d(\text{seg}) = u(3,1) + \text{uprime}(\text{seg},1) \]

SEGMENT 10 LEFT THIGH

\[ \text{seg}=10 \]
\[ \text{jt}=9 \]
\[ \text{call uudot(jt, alpha, alphadot, alphaddot, beta, betadot, betaddot, gammadot, gammaddot, udot, seg)} \]
\[ \text{do } i=1,3 \]
\[ \quad \text{udp}(\text{seg},i) = \text{udot}(\text{seg},1) * \text{relrm}(\text{jt},i,1) + u(\text{seg},1) \]
& *rrmdot(jt,i,1)+udot(seg,2)*relm(jt,i,2)
& +u(seg,2)*rrmdot(jt,i,2)+udot(seg,3)*relm(jt,i,3)
& +u(seg,2)*rrmdot(jt,i,3)
enddo
a(seg,1)=udot(3,1)+udp(seg,1)
a(seg,2)=udot(3,2)+udp(seg,2)
a(seg,3)=udot(3,3)+udp(seg,3)
b(seg)=u(3,2)+uprime(seg,2)
c(seg)=u(3,3)+uprime(seg,3)
d(seg)=u(3,1)+uprime(seg,1)

SEGMENT 5 RIGHT FOREARM

seg=5
jt=4
do i=1,3
udot(seg,i)=alphaddot(jt)*relm(2,1,3)*relm(6,i,1)
&+relm(2,2,3)*relm(6,i,2)+relm(2,3,3)*relm(6,i,3)
&+alphadot(jt)*(rrmdot(2,1,3)*relm(6,i,1)+relm(2,1,3))
&+rrmdot(6,i,1)+rrmdot(2,2,3)*relm(6,i,2)
&+relm(2,2,3)*rrmdot(6,i,2)+rrmdot(2,3,3)*relm(6,i,3)
&+relm(2,3,3)*rrmdot(6,i,3)
enddo
a(seg,1)=udot(2,1)+udp(seg+2,1)+udot(seg,1)
a(seg,2)=udot(2,2)+udp(seg+2,2)+udot(seg,2)
a(seg,3)=udot(2,3)+udp(seg+2,3)+udot(seg,3)
b(seg)=u(2,2)+uprime(seg+2,2)+u(seg,2)
c(seg)=u(2,3)+uprime(seg+2,3)+u(seg,3)
d(seg)=u(2,1)+uprime(seg+2,1)+u(seg,1)

SEGMENT 6 LEFT FOREARM

seg=6
jt=5
do i=1,3
udot(seg,i)=alphaddot(jt)*relm(3,1,3)
&+relm(6,i,2)+relm(3,3,3)*relm(6,i,3)
&+relm(3,1,3)*relm(6,i,1)+relm(3,1,3)*rrmdot(6,i,1)
&+rrmdot(3,2,3)*relm(6,i,2)
&+relm(3,2,3)*rrmdot(6,i,2)+rrmdot(3,3,3)*relm(6,i,3)
&+relm(3,3,3)*rrmdot(6,i,3)
enddo
a(seg,1)=udot(2,1)+udp(seg+2,1)+udot(seg,1)
a(seg,2)=udot(2,2)+udp(seg+2,2)+udot(seg,2)
a(seg,3)=udot(2,3)+udp(seg+2,3)+udot(seg,3)
b(seg)=u(2,2)+uprime(seg+2,2)+u(seg,2)
c(seg)=u(2,3)+uprime(seg+2,3)+u(seg,3)
d(seg)=u(2,1)+uprime(seg+2,1)+u(seg,1)
C SEGMENT 11 RIGHT SHANK

seg=11
jt=10
do i=1,3
udot(seg,i)=alphaddot(jt)*(relrm(8,1,3)
& *relrm(7,i,1)+relrm(8,2,3)
& *relrm(7,i,2)+relrm(8,3,3)*relrm(7,i,3)) +alphadot(jt)
& *(rrmdot(8,1,3)*relrm(7,i,1)+relrm(8,1,3)*rrmdot(7,i,1)
& +rrmdot(8,2,3)*relrm(7,i,2)
& +relrm(8,2,3)*rrmdot(7,i,2)+rrmdot(8,3,3)*relrm(7,i,3)
& +relrm(8,3,3)*rrmdot(7,i,3))
enddo
a(seg,1)=udot(3,1)+udp(seg-2,1)+udot(seg,1)
a(seg,2)=udot(3,2)+udp(seg-2,2)+udot(seg,2)
a(seg,3)=udot(3,3)+udp(seg-2,3)+udot(seg,3)
b(seg)=u(3,2)+uprime(seg-2,2)+u(seg,2)
c(seg)=u(3,3)+uprime(seg-2,3)+u(seg,3)
d(seg)=u(3,1)+uprime(seg-2,1)+u(seg,1)

C SEGMENT 12 LEFT SHANK

seg=12
jt=11
do i=1,3
udot(seg,i)=alphaddot(jt)*(relrm(9,1,3)
& *relrm(7,i,1)+relrm(9,2,3)
& *relrm(7,i,2)+relrm(9,3,3)*relrm(7,i,3)) +alphadot(jt)
& *(rrmdot(9,1,3)*relrm(7,i,1)+relrm(9,1,3)*rrmdot(7,i,1)
& +rrmdot(9,2,3)*relrm(7,i,2)
& +relrm(9,2,3)*rrmdot(7,i,2)+rrmdot(9,3,3)*relrm(7,i,3)
& +relrm(9,3,3)*rrmdot(7,i,3))
enddo
a(seg,1)=udot(3,1)+udp(seg-2,1)+udot(seg,1)
a(seg,2)=udot(3,2)+udp(seg-2,2)+udot(seg,2)
a(seg,3)=udot(3,3)+udp(seg-2,3)+udot(seg,3)
b(seg)=u(3,2)+uprime(seg-2,2)+u(seg,2)
c(seg)=u(3,3)+uprime(seg-2,3)+u(seg,3)
d(seg)=u(3,1)+uprime(seg-2,1)+u(seg,1)
return
end

Subroutine UUDOT

SUBROUTINE UUDOT(JT, A, ADOT, ADDOT, B, BDOT, BDDOT, GDOT &
, GDDOT, UDOT, SEG)
REAL UDOT(12,3), A(11), B(11), ADOT(11), BDOT(11), GDOT(11)
REAL ADDOT(11), BDDOT(11), GDDOT(11)
INTEGER JT, SEG

udot(seg,1)=bdot(jt)*cos(a(jt))
& \ -\text{bdot}(\text{jt})*\text{adot}(\text{jt})*\sin(\text{a(Jt)}) \\
& \ -\text{gddot}(\text{jt})*\cos(\text{b(jt)})*\sin(\text{a(Jt)}) \\
& \ +\text{gdot}(\text{jt})*\text{bdot}(\text{jt})*\sin(\text{b(jt)})*\sin(\text{a(jt)}) \\
& \ -\text{gdot}(\text{jt})*\text{adot}(\text{jt})*\cos(\text{b(jt)})*\cos(\text{a(jt)}) \\
\text{udot}(\text{seg,2})=\text{bddot}(\text{jt})*\sin(\text{a(jt)}) \\
& \ +\text{bdot}(\text{jt})*\text{adot}(\text{jt})*\cos(\text{a(jt)}) \\
& \ +\text{gddot}(\text{jt})*\cos(\text{b(jt)})*\cos(\text{a(jt)}) \\
& \ -\text{gdot}(\text{jt})*\text{bdot}(\text{jt})*\sin(\text{b(jt)})*\cos(\text{a(jt)}) \\
& \ -\text{gdot}(\text{jt})*\text{adot}(\text{jt})*\cos(\text{b(jt)})*\sin(\text{a(jt)}) \\
& \ +\text{gdot}(\text{jt})*\text{bdot}(\text{jt})*\cos(\text{b(jt)}) \\
return 
end

Subroutine DYADIC
C SUBROUTINE TO COMPUTE THE INERTIA DYADIC OF A SEGMENT IN 
REFERENCE SEGMENT COORDINATES 
C
SUBROUTINE DYADIC(IXX,IYY,IZZ,R,IDY)
REAL IXX(12),IYY(12),IZZ(12),R(12,3,3),IDY(12,3,3)
C
do i=1,12
  do j=1,3
    idy(i,j,j)=ixx(i)*r(i,j,1)**2+iyy(i)*r(i,j,2)**2

& \ +izz(i)*r(i,j,3)**2 
  enddo 
idy(i,1,2)=ixx(i)*r(i,1,1)*r(i,2,1)+iyy(i)*r(i,1,2) \\
& *r(i,2,2)+izz(i)*r(i,1,3)*r(i,2,3) 
idy(i,1,3)=ixx(i)*r(i,1,1)*r(i,3,1)+iyy(i)*r(i,1,2) \\
& *r(i,3,2)+izz(i)*r(i,1,3)*r(i,3,3) 
idy(i,2,1)=idy(i,1,2) 
idy(i,2,3)=idy(i,1,3) 
idy(i,3,2)=idy(i,2,3) 
enddo 
return 
end 

Subroutine IWW
C
C THIS SUBROUTINE COMPUTES THE COEFFICIENTS OF THE ANGULAR 
VELOCITY TERM I DOT OMEGA CROSS OMEGA OF THE 
DIFFERENTIAL 
C
SUBROUTINE IWW(W,IDY,RW)
REAL W(3,5),IDY(12,3,3),RW(12,3)
do i=1,3
  do j=1,5
enddo

do i=1,12
w(1,1)=w(1,1)+(idy(i,2,2)-idy(i,3,3))*rw(i,2)*rw(i,3)
& +idy(i,2,1)*rw(i,1)*rw(i,3)+idy(i,2,3)*((rw(i,3)**2)
& -rw(i,2)**2)-idy(i,3,1)*rw(i,1)*rw(i,2)
& -rw(i,2)**2)-idy(i,3,1)*rw(i,1)*rw(i,2)
w(2,1)=w(2,1)+(idy(i,3,3)-idy(i,1,1))*rw(i,1)*rw(i,3)
& +idy(i,3,2)*rw(i,1)*rw(i,2)+idy(i,3,1)*((rw(i,1)**2)
& +idy(i,3,2)*rw(i,1)*rw(i,2)+idy(i,3,1)*((rw(i,1)**2)
& -rw(i,3)**2)+idy(i,2,1)*rw(i,3)-idy(i,3,1)*rw(i,2)
w(3,1)=w(3,1)+(idy(i,1,1)-idy(i,2,2))*rw(i,1)*rw(i,2)
& +idy(i,1,3)*rw(i,3)*rw(i,2)+idy(i,1,2)*((rw(i,2)**2)
& +idy(i,1,3)*rw(i,3)*rw(i,2)+idy(i,1,2)*((rw(i,2)**2)
& -rw(i,1)**2)-idy(i,2,3)*rw(i,3)-2*idy(i,2,3)*rw(i,2)
& +idy(i,2,2)-idy(i,3,3))*rw(i,3)
& +2*idy(i,2,3)*rw(i,3)+idy(i,2,1)*rw(i,1)
w(1,5)=w(1,5)+idy(i,1,2)-idy(i,1,3)
w(2,2)=w(2,2)+(idy(i,3,3)-idy(i,1,1))*rw(i,3)
& +2*idy(i,3,1)*rw(i,1)+idy(i,3,2)*rw(i,2)
w(2,3)=w(2,3)+idy(i,3,2)*rw(i,1)-idy(i,1,2)*rw(i,3)
w(2,4)=w(2,4)+(idy(i,3,3)-idy(i,1,1))*rw(i,1)
& +2*idy(i,1,3)*rw(i,3)-idy(i,1,2)*rw(i,2)
w(2,5)=w(2,5)+idy(i,1,3)-idy(i,1,1)
w(3,2)=w(3,2)+(idy(i,1,1)-idy(i,2,2))*rw(i,2)
& +2*idy(i,1,2)*rw(i,1)-idy(i,2,3)*rw(i,3)
w(3,3)=w(3,3)+(idy(i,1,1)-idy(i,2,2))*rw(i,1)
& +2*idy(i,1,2)*rw(i,1)-idy(i,1,3)*rw(i,3)
w(3,4)=w(3,4)+idy(i,1,3)*rw(i,2)-idy(i,2,3)*rw(i,1)
w(3,5)=w(3,5)+idy(i,1,1)-idy(i,2,2)
enddo
return
end

Subroutine IDOTA
C
C THIS SUBROUTINE COMPUTES THE COEFFICIENTS OF THE I DOT
C ALPHA TERM OF THE DIFFERENTIAL EQUATION
C
SUBROUTINE IDOTALPHA(IDY,A,B,C,D,Q,R,S,T,V)
REAL Q(6),R(3),S(3),T(3),V(3),A(12,3),B(12),C(12),D(12)
REAL IDY(12,3,3)
do i=1,6
q(i)=0
enddo
do i=1,3
r(i)=0
s(i)=0
t(i)=0
enddo

do i=1,12
    q(1)=q(1)+idy(i,1,1)
    q(2)=q(2)+idy(i,1,2)
    q(3)=q(3)+idy(i,1,3)
    q(4)=q(4)+idy(i,2,2)
    q(5)=q(5)+idy(i,2,3)
    q(6)=q(6)+idy(i,3,3)
    do j=1,3
        r(j)=r(j)+idy(i,j,1)*a(i,1)+idy(i,j,2)*a(i,2)
        & +idy(i,j,3)*a(i,3)
        s(j)=s(j)+idy(i,j,2)*d(i)-idy(i,j,1)*b(i)
        t(j)=t(j)+idy(i,j,1)*c(i)-idy(i,j,3)*d(i)
        v(j)=v(j)+idy(i,j,3)*b(i)-idy(i,j,2)*c(i)
    enddo
endo
return
end

Subroutine PXMA
C THIS SUBROUTINE COMPUTES THE COEFFICIENTS OF THE P X MA
C TERM OF THE DIFFERENTIAL EQUATION
C
SUBROUTINE PXMA (Q, QDOT, QDDOT, DELTA, M, MTOT, P)
REAL Q(12,3), QDOT(12,3), QDDOT(12,3), DELTA(3,8)
REAL M(12), MTOT
REAL QP(12,3,3), QPP(12,3), P(12,3)

C FIRST, GET THE TERMS OF THE ACCELERATION OF EACH SEGMENT
C QP AND QPP, WHERE QP'S ARE THE CAPITAL Q PRIMES IN
C APPENDIX A, AND QPP'S ARE THE SMALL Q PRIMES IN APPENDIX
C
C
do i=1,12
    do j=1,3
        do k=1,3
            qp(i,j,k)=0.
        enddo
    enddo
    qpp(i,j)=0.
endo
endo

C
C
do i=1,8
    do j=1,3
        delta(j,i)=0
    enddo
endo

C
C
do i=1,12
    do j=1,12
        do k=1,3
if (mtot .Eq. 0) Then
  qp(i,k,1)=0
  qpp(i,k)=0
else
  qp(i,k,1)=qp(i,k,1)+m(j)/mtot*qddot(j,k)
  qpp(i,k)=qpp(i,k)+m(j)/mtot*q(j,k)
end if
enddo
if (mtot .Eq. 0) Then
  qp(i,3,3)=0
  qp(i,1,2)=0
else
  qp(i,3,3)=qp(i,3,3)+m(j)/mtot*2.*qdot(j,2)
  qp(i,1,2)=qp(i,1,2)+m(j)/mtot*2.*qdot(j,2)
end if
if (mtot .Eq. 0) Then
  qp(i,1,3)=0
  qp(i,2,3)=0
else
  qp(i,1,3)=qp(i,1,3)+m(j)/mtot*2.*qdot(j,3)
  qp(i,2,3)=qp(i,2,3)+m(j)/mtot*2.*qdot(j,3)
end if
enddo
doi=1,12
do k=1,3
  qpp(i,k)=q(i,k)-qpp(i,k)
  qp(i,k,1)=qddot(i,k)-qp(i,k,1)
dendo
qp(i,1,2)=-qp(i,1,2)+2.*qdot(i,2)
qp(i,2,2)=-qp(i,2,2)+2.*qdot(i,1)
qp(i,3,2)=-qp(i,3,2)+2.*qdot(i,1)
qp(i,1,3)=-qp(i,1,3)+2.*qdot(i,3)
qp(i,2,3)=-qp(i,2,3)+2.*qdot(i,3)
qp(i,3,3)=-qp(i,3,3)+2.*qdot(i,2)
dendo
doi=1,3
do j=1,12
  delta(i,1)=delta(i,1)+m(j)*p(j,i)
do k=2,4
  delta(i,k)=delta(i,k)+m(j)*p(j,i)*qpp(j,k-1)
dendo
enddo  
enddo  
do j=1,12  
delta(1,5)=delta(1,5)+m(j)*p(j,1)*qp(j,2,3)  
delta(1,6)=delta(1,6)+m(j)*p(j,1)*qp(j,3,2)  
delta(1,7)=delta(1,7)+m(j)*(p(j,2) &  
*qp(j,3,1)-p(j,3)*qp(j,2,1))  
delta(1,8)=delta(1,8)+m(j)*p(j,3)  
&  
*qp(j,3,2)+p(j,3)*qp(j,2,3))  
delta(2,5)=delta(2,5)+m(j)*p(j,2)*qp(j,1,3)  
delta(2,6)=delta(2,6)+m(j)*p(j,2)*qp(j,3,3)  
delta(2,7)=delta(2,7)+m(j)*(p(j,1) &  
*qp(j,1,1)-p(j,1)*qp(j,3,1))  
delta(2,8)=delta(2,8)+m(j)*p(j,3)  
&  
*qp(j,3,3)+p(j,3)*qp(j,1,3))  
delta(3,5)=delta(3,5)+m(j)*p(j,3)*qp(j,2,3)  
delta(3,6)=delta(3,6)+m(j)*p(j,3)*qp(j,5,3)  
delta(3,7)=delta(3,7)+m(j)*(p(j,1) &  
*qp(j,1,1)-p(j,1)*qp(j,3,1))  
delta(3,8)=delta(3,8)+m(j)*p(j,3)  
&  
*qp(j,3,3)+p(j,3)*qp(j,1,3))  
enddo  
return  
end

Subroutine GAMS

SUBROUTINE GAMS(Q,DELTA,GG)
C
C THIS SUBROUTINE COMPUTES THE GAMMA TERMS USED IN KINEMATICS
C
C INPUT VARIABLES: Q AND DELTA
C OUTPUT GG: GAMMAS
C
REAL Q(6), DELTA(3,8), GG(4)

gg(1)=(q(1)+delta(2,3)+delta(3,4))*(q(6)+delta(2,3) &  
+delta(1,2))*(q(3)-delta(3,2))*(delta(1,4)-q(3))  
gg(2)=(q(2)-delta(2,1))*q(6)+delta(2,3)+delta(1,2))  
&  
+(q(3)-delta(3,2))*(delta(3,3)-q(5))  
gg(3)=(q(2)-delta(1,3))*(q(6)+delta(2,3)+delta(1,2))  
&  
+(delta(1,4)-q(3))*(q(5)-delta(3,3))  
gg(4)=(q(4)+delta(3,4)+delta(1,2))*(q(6)+delta(2,3) &  
+delta(1,2))+(q(5)-delta(3,3))*(delta(3,3)-q(5))  
return  
end

Subroutine KINEM

SUBROUTINE KINEM(XDD,M)
THIS SUBROUTINE CALCULATES THE LINEAR AND ANGULAR ACCELERATION OF EACH SEGMENT AND THE ANGULAR VELOCITY OF EACH SEGMENT

REAL ANG(12,3), OMEGA(12,3), ACC(12,3)
REAL RW(12,3), ROT(12,3,3), A(12,3), B(12), C(12)
REAL Q(12,3), D(12), QPP(12,3)
REAL QP(12,3,3), O(3), VEC(3), QDOT(12,3)
REAL QDDOT(12,3), RR(3,3)
REAL ACG(3), M(12), MTOT
REAL XDD(3), U1(3), U1DOT(3), THETA(3)

COMMON/KINDAT/THETA, U1, U1DOT, RW, ROT, A, B, C, D, Q, QDOT, QDDOT,
& ANG, OMEGA, ACC

CALCULATE TOTAL BODY MASS

mtot=0.
do i=1,12
   mtot=mtot+m(i)
endo

COMPUTE THE ANGULAR VELOCITY OF EACH SEGMENT IN SEGMENT 1 COORDINATES

do i=1,12
   do j=1,3
      omega(i,j)=U1(J)+rw(i,j)
   enddo
endo

ROTATE INTO SEGMENT COORDINATES

do i=1,12
   do j=1,3
      o(j)=omega(i,j)
      do k=1,3
         rr(j,k)=rot(i,k,j)
      enddo
   enddo
   call matvec(rr,o,vec)
do j=1,3
   omega(i,j)=vec(j)
endo

COMPUTE ANGULAR ACCELERATION IN SEGMENT 1 COORDINATES

do i=1,12
   ang(i,1)=U1DOT(1)+a(i,1)-b(i)*u1(3)+c(i)*u1(2)
\[
\begin{align*}
\text{ang}(i,2) &= \dot{u} + a(i,2) + d(i) \cdot u(3) - c(i) \cdot u(1) \\
\text{ang}(i,3) &= \dot{u} + a(i,3) + b(i) \cdot u(1) - d(i) \cdot u(2)
\end{align*}
\]

enddo

C

ROTATE INTO SEGMENT COORDINATES

do i=1,12
    do j=1,3
        o(j)=ang(i,j)
        do k=1,3
            rr(j,k)=rot(i,k,j)
            enddo
        enddo
    endo
    call matvec(rr,o,vec)
    do j=1,3
        ang(i,j)=vec(j)
    endo
endo

C

COMPUTE LINEAR ACCELERATION IN SEGMENT 1 COORDINATES

acg(1)=\( \dot{x} \cdot (\cos(\theta(1)) \cdot \cos(\theta(3)) - \sin(\theta(1)) \cdot \sin(\theta(2)) \cdot \sin(\theta(3))) \) + \( \dot{x} \cdot (\sin(\theta(1)) \cdot \cos(\theta(3)) + \cos(\theta(1)) \cdot \sin(\theta(2)) \cdot \sin(\theta(3))) \) - \( \dot{x} \cdot \sin(\theta(3)) \cdot \cos(\theta(2)) \)

acg(2)=-\( \dot{x} \cdot \sin(\theta(1)) \cdot \cos(\theta(2)) \) + \( \dot{x} \cdot \cos(\theta(1)) \cdot \cos(\theta(2)) \) + \( \dot{x} \cdot \sin(\theta(2)) \)

acg(3)=\( \dot{x} \cdot (\cos(\theta(1)) \cdot \sin(\theta(3)) + \sin(\theta(1)) \cdot \sin(\theta(2)) \cdot \cos(\theta(3))) \) + \( \dot{x} \cdot (\sin(\theta(1)) \cdot \sin(\theta(3)) - \cos(\theta(1)) \cdot \cos(\theta(2)) \cdot \sin(\theta(3))) \) + \( \dot{x} \cdot \cos(\theta(2)) \cdot \cos(\theta(3)) \)

C

FIRST, GET THE TERMS OF THE ACCELERATION OF EACH SEGMENT QP AND QPP

do i=1,12
    do j=1,3
        do k=1,3
            qp(i,j,k)=0.
        enddo
        qpp(i,j)=0
    endo
endo

do i=1,12
    do j=1,12
        do k=1,3
            if (mtot .Eq. 0) Then
                qp(i,k,1)=0
            else
                qpp(i,k)=0
            endif
        endo
    endo
enddo
qp(i, k, l) = qp(i, k, l) + m(j) / mtot * qddot(j, k)
end if
if (mtot . Eq. 0) Then
  qp(i, k) = 0
else
  qp(i, k) = qp(i, k) + m(j) / mtot * q(j, k)
end if
enddo
if (mtot . Eq. 0) Then
  qp(i, 3, 3) = 0
  qp(i, 1, 2) = 0
else
  qp(i, 3, 3) = qp(i, 3, 3) + m(j) / mtot * 2.0 * qdot(j, 2)
  qp(i, 1, 2) = qp(i, 1, 2) + m(j) / mtot * 2.0 * qdot(j, 2)
end if
if (mtot . Eq. 0) Then
  qp(i, 2, 2) = 0
  qp(i, 3, 2) = 0
else
  qp(i, 2, 2) = qp(i, 2, 2) + m(j) / mtot * 2.0 * qdot(j, 1)
  qp(i, 3, 2) = qp(i, 3, 2) + m(j) / mtot * 2.0 * qdot(j, 1)
end if
if (mtot . Eq. 0) Then
  qp(i, 3, 3) = 0
  qp(i, 2, 3) = 0
else
  qp(i, 3, 3) = qp(i, 3, 3) + m(j) / mtot * 2.0 * qdot(j, 3)
  qp(i, 2, 3) = qp(i, 2, 3) + m(j) / mtot * 2.0 * qdot(j, 3)
end if
endo
enddo
  do i=1,12
    acc(i,1)=acg(1)+qp(i,1,1)-qp(i,1,2)*ul(3)+qp(i,1,3)*
      &     ul(2)-qp(1,1)*(ul(3)**2+ul(2)**2)+qp(1,2)*
      &     (ul(1)*ul(2)-uldot(3))+qp(1,3)*(uldot(2)
      &     +ul(1)*ul(3))
    acc(i,2)=acg(2)+qp(i,2,1)+qp(i,2,2)*ul(3)-qp(i,2,3)*
      &     ul(1)+qp(1,1)*(uldot(3)+ul(2)*ul(1))-qp(1,2)*
      &     (ul(1)**2+ul(3)**2)+qp(1,3)*(ul(2)*ul(3)
      &     -uldot(1))
    acc(i,3)=acg(3)+qp(i,3,1)+qp(i,3,2)*ul(1)-qp(i,3,3)*
      &     ul(2)+qp(1,1)*(ul(3)*ul(1)-uldot(2))+qp(1,2)*
      &     (uldot(1)+ul(3)*ul(2)) -qp(i,3)*(ul(2)**2
      &     +ul(1)**2)
  enddo

C ROTATE INTO SEGMENT COORDINATES
C
  do i=1,12
    do j=1,3
      o(j)=acc(i,j)
      do k=1,3
        rr(j,k)=rot(i,k)
      enddo
    enddo
    call matvec(rr,o,vec)
    do j=1,3
      acc(i,j)=vec(j)
    enddo
  enddo
  return
  end

Subroutine FORTORQ
SUBROUTINE FORTORQ(THETA, ANG, OMEGA, ACC, H, ETA, R, FOR, 
  & TOR, RELRM, ROT, KPRIME, MASS, IXX, IYY, IZZ, FL, FR)
C THIS SUBROUTINE COMPUTES THE FORCES AND TORQUES AT EACH 
C JOINT USING THE KINEMATICS AND INERTIAL PROPERTIES OF 
C THE SEGMENTS 
C THE FORCES AND TORQUES ARE CALCULATED IN PROXIMAL 
C SEGMENT COORDINATES 
C
C INPUT:
C THETA=ROTATION ANGLES OF THE REFERENCE SEGMENT WRT GLOBAL 
C COORD.
C ANG=ANGULAR ACCELERATION OF THE SEGMENTS
OMEGA=ANGULAR VELOCITY OF SEGMENTS
ACC=LINEAR ACCELERATION OF SEGMENTS
H=SEGMENT LENGTHS
ETA,R,RR=SEGMENT DIMENSIONS
RELRM=ROTATION MATRICES BETWEEN SEGMENTS
RELF=RELEASE FRAME, RECF=RECATCH FRAME
KPRIME=PROPORTIONALITY CONSTANTS BETWEEN HAND FORCES AND TORQUES
N=CURRENT FRAME NUMBER
ROTA=ROTATION MATRICES BETWEEN EA. SEGMENT AND SEGMENT 1
FL,FR=FORCES AT HANDS

OUTPUT:
FOR=FORCE VECTORS AT JOINTS IN PROXIMAL SEGMENT COORDINATES
TOR=TORQUE VECTORS AT JOINTS IN PROXIMAL SEGMENT COORDINATES

DECLARE VARIABLES

REAL THETA(3),FL(3),FR(3)
REAL ANG(12,3),OMEGA(12,3),ACC(12,3),H(12)
REAL ETA(12),R(12),FOR(13,3),TOR(13,3),RELRM(11,3,3)
REAL ML,MR,FRMAG,FLMAG,MASS(12),FD(3),TD(3),KPRIME
REAL S1ROT(3,3),ROT(12,3,3),RFROT(3,3)
REAL LFROT(3,3),ROTRF(3,3)
REAL ROTL(3,3),VECTOR(3),V2(3),V3(3)
REAL WT(12,3),SEGROT(3,3)
REAL ROTSEG(3,3),HG(12,3),RXF(3)
REAL RXF2(3),RXF3(3),IXX(12)
REAL IYY(12),IZZ(12)

PUT ACCELERATIONS INTO M/SEC

DO I=1,12
  DO J=1,3
    ACC(I,J)=ACC(I,J)/100.
  END DO
END DO

SET UP ROTATION MATRIX BETWEEN SEGMENT 1 COORDINATES AND GLOBAL COORDINATES

S1ROT(1,1)=COS(THETA(1))*COS(THETA(3))
& -SIN(THETA(1))*SIN(THETA(2))*SIN(THETA(3))
S1ROT(1,2)=-SIN(THETA(1))*COS(THETA(2))
S1ROT(1,3)=COS(THETA(1))*SIN(THETA(3))
& +SIN(THETA(1))*SIN(THETA(2))*COS(THETA(3))
S1ROT(2,1)=SIN(THETA(1))*COS(THETA(3))
& +COS(THETA(1))*SIN(THETA(2))*SIN(THETA(3))
slrot(2,2)=cos(theta(1))*cos(theta(2))
slrot(2,3)=sin(theta(1))*sin(theta(3)) & -cos(theta(1))*sin(theta(2))*cos(theta(3))
slrot(3,1)=-sin(theta(3))*cos(theta(2))
slrot(3,2)=sin(theta(2))
slrot(3,3)=cos(theta(2))*cos(theta(3))

START WITH FORCES AT THE HANDS

ROTATE FROM GLOBAL COORDINATES INTO FOREARM COORDINATES

FROM GLOBAL TO SEGMENT 1:
call trans(slrot)

FROM SEGMENT 1 TO FOREARM:

do i=1,3
   do j=1,3
      rotrf(i,j)=rot(5,j,i)
      rotlf(i,j)=rot(6,j,i)
   enddo
endo
call mm(rotrf, slrot, rfrot)
call mm(rotlf, slrot, lfrot)

MULTIPLY HAND FORCES W/ MATRIX TO GET FOREARM COORDINATES (CHANGE FORCES INTO NEWTONS)

vector(1)=fr(1)/100.
vector(2)=fr(2)/100.
vector(3)=0.
call matvec(rfrot, vector, v2)
vector(1)=f1(1)/100.
vector(2)=f1(2)/100.
vector(3)=0.
call matvec(lfrot, vector, v3)
do i=1,3
   for(12,i)=v2(i)
   for(13,i)=v3(i)
endo

MOMENTS AT HANDS

flmag=sqrt(for(13,1)**2+for(13,2)**2+for(13,3)**2)
frmag=sqrt(for(12,1)**2+for(12,2)**2+for(12,3)**2)
mr=kprime*frmag
ml=kprime*flmag

ROTATE INTO FOREARM COORDINATES
vector(1)=0.
vector(2)=0.
vector(3)=mr
call matvec(rfrot,vector,v2)
vector(1)=0.
vector(2)=0.
vector(3)=ml
call matvec(lfrot,vector,v3)
do i=1,3
    tor(12,i)=v2(i)
    tor(13,i)=v3(i)
enddo

C FIND WEIGHT OF EACH SEGMENT IN SEGMENT COORDINATES
C GET ROTATION MATRIX FROM GLOBAL TO SEGMENT COORDINATES FOR EACH SEGMENT
C
do i=2,12
do j=1,3
do kk=1,3
    rotseg(j,kk)=rot(i,kk,j)
enddo
dondocall iimi(rotseg,slrot,segrot)
C DO MULTIPLICATION OF MATRIX WITH WEIGHT VECTOR
C
vector(1)=0.
vector(2)=mass(i)*9.81
vector(3)=0.
call matvec(segrot,vector,v2)
do j=1,3
    wt(i,j)=v2(j)
enddo
dendo
vector(1)=0.
vector(2)=mass(1)*9.81
vector(3)=0.
call matvec(s1rot,vector,v2)
do j=1,3
    wt(1,j)=v2(j)
enddo

C FIND THE CHANGE IN MOMENTUM FOR EACH SEGMENT IN SEGMENT COORDINATES
C
do i=1,12
hg(i,1)=(ixx(i)*ang(i,1)-(iyy(i)-izz(i))*omega(i,2)
& \quad \quad \quad \quad \quad *\omega(i,3))/100.

\text{hg}(i,2) = (\text{iyy}(i) * \text{ang}(i,2) - (\text{izz}(i) - \text{ixx}(i)) * \omega(i,1)
& \quad \quad \quad \quad \quad *\omega(i,3))/100.

\text{hg}(i,3) = (\text{izz}(i) * \text{ang}(i,3) - (\text{ixx}(i) - \text{iyy}(i)) * \omega(i,2)
& \quad \quad \quad \quad \quad *\omega(i,1))/100.
\text{enddo}

\text{C}
\text{C} \quad \text{CALCULATE THE ELBOW FORCES AND TORQUES IN FOREARM}
\text{C} \quad \text{COORDINATES}
\text{C}
\text{do i}=1,3
\quad \text{fd}(i) = \text{inass}(5) * \text{acc}(5,i) - \text{for}(12,i) - \text{wt}(5,i)
\text{enddo}
\text{rxf}(1) = -(1 - \eta(5)) * h(5) * \text{for}(12,3)
\text{rxf}(2) = 0.
\text{rxf}(3) = (1 - \eta(5)) * h(5) * \text{for}(12,1)
\text{rxf2}(1) = \eta(5) * h(5) * \text{fd}(3)
\text{rxf2}(2) = 0.
\text{rxf2}(3) = -\eta(5) * h(5) * \text{fd}(1)
\text{do i}=1,3
\quad \text{td}(i) = \text{hg}(5,i) - \text{rxf}(i) - \text{rxf2}(i) - \text{tor}(12,i)
\text{enddo}
\text{C}
\text{C} \quad \text{ROTATE INTO UPPER ARM COORDINATES}
\text{C}
\text{do i}=1,3
\quad \text{for}(4,i) = -(\text{relrn}(4, i, 1) * \text{fd}(1) + \text{relrn}(4, i, 2) * \text{fd}(2)
& \quad \quad \quad \quad \quad + \text{relrm}(4, i, 3) * \text{fd}(3))
\quad \text{tor}(4,i) = -(\text{relrn}(4, i, 1) * \text{td}(1) + \text{relrm}(4, i, 2) * \text{td}(2)
& \quad \quad \quad \quad \quad + \text{relrm}(4, i, 3) * \text{td}(3))
\text{enddo}
\text{do i}=1,3
\quad \text{fd}(i) = \text{mass}(6) * \text{acc}(6,i) - \text{for}(13,i) - \text{wt}(6,i)
\text{enddo}
\text{rxf}(1) = -(1 - \eta(6)) * h(6) * \text{for}(13,3)
\text{rxf}(2) = 0.
\text{rxf}(3) = (1 - \eta(6)) * h(6) * \text{for}(13,1)
\text{rxf2}(1) = \eta(6) * h(6) * \text{fd}(3)
\text{rxf2}(2) = 0.
\text{rxf2}(3) = -\eta(6) * h(6) * \text{fd}(1)
\text{do i}=1,3
\quad \text{td}(i) = \text{hg}(6,i) - \text{rxf}(i) - \text{rxf2}(i) - \text{tor}(13,i)
\text{enddo}
\text{C}
\text{C} \quad \text{ROTATE INTO UPPER ARM COORDINATES}
\text{C}
\text{do i}=1,3
\quad \text{for}(5,i) = -(\text{relrn}(5, i, 1) * \text{fd}(1) + \text{relrn}(5, i, 2) * \text{fd}(2)
& \quad \quad \quad \quad \quad + \text{relrm}(5, i, 3) * \text{fd}(3))
& relrm(5,i,3)*fd(3))
  tor(5,i)=-(relrm(5,i,1)*td(1)+relrm(5,i,2)*td(2)
& +relrm(5,i,3)*td(3))
endo

C FIND THE FORCES AND MOMENTS AT THE SHOULDERS IN UPPER
C ARM COORDINATES

do i=1,3
  fd(i)=mass(7)*acc(7,i)-wt(7,i)-for(4,i)
endo
rxf(1)=-(1-eta(7))*h(7))*for(4,3)
rxf(2)=0.
rxf(3)=((1-eta(7))*h(7))*for(4,1)
rxf2(1)=eta(7)*h(7)*fd(3)
rxf2(2)=0.
rxf2(3)=-eta(7)*h(7)*fd(1)
do i=1,3
  td(i)=hg(7,i)-rxf(i)-rxf2(i)-tor(4,i)
endo

C ROTATE INTO UPPER TRUNK COORDINATES

do i=1,3
  for(2,i)=-(relrm(2,i,1)*fd(1)+relrm(2,i,2)*fd(2)
 & +relrm(2,i,3)*fd(3))
  tor(2,i)=-(relrm(2,i,1)*td(1)+relrm(2,i,2)*td(2)
 & +relrm(2,i,3)*td(3))
endo
do i=1,3
  fd(i)=mass(8)*acc(8,i)-wt(8,i)-for(5,i)
endo
rxf(1)=(-(1-eta(8))*h(8))*for(5,3)
rxf(2)=0.
rxf(3)=((1-eta(8))*h(8))*for(5,1)
rxf2(1)=eta(8)*h(8)*fd(3)
rxf2(2)=0.
rxf2(3)=-eta(8)*h(8)*fd(1)
do i=1,3
  td(i)=hg(8,i)-rxf(i)-rxf2(i)-tor(5,i)
endo

C ROTATE INTO UPPER TRUNK COORDINATES

do i=1,3
  for(3,i)=-(relrm(3,i,1)*fd(1)+relrm(3,i,2)*fd(2)
 & +relrm(3,i,3)*fd(3))
  tor(3,i)=-(relrm(3,i,1)*td(1)+relrm(3,i,2)*td(2)
 & +relrm(3,i,3)*td(3))
endo
FIND FORCES AND MOMENTS AT NECK JOINT

do i=1,3
   fd(i)=mass(4)*acc(4,i)-wt(4,i)
endo
dfd(1)=h(4)/2.*fd(3)
dfd(2)=0.
dfd(3)=-h(4)/2.*fd(1)
do i=1,3
dfd(i)=HG(4,i)-rxf(i)
endo

ROTATE INTO UPPER TRUNK COORDINATES

do i=1,3
dfd(i)=-(relrm(1,i,1)*fd(1)+relrm(1,i,2)*fd(2)
&+relrm(1,i,3)*fd(3))
dtd(1)=-(relrm(1,i,1)*td(1)+relrm(1,i,2)*td(2)
&+relrm(1,i,3)*td(3))
endo

FIND FORCES AT THE UPPER TRUNK

do i=1,3
   fd(i)=mass(2)*acc(2,i)-for(2,i)-for(3,i)-for(1,i)
&-wt(2,i)
endo

ROTATE INTO MID TRUNK COORDINATES

do i=1,3
   for(6,i)=-(relrm(6,i,1)*fd(1)+relrm(6,i,2)*fd(2)
&+relrm(6,i,3)*fd(3))
endo

FIND FORCES AT THE KNEES

do i=1,3
   fd(i)=mass(11)*acc(11,i)-wt(11,i)
endo

FIND MOMENTS AT KNEES

rrl=-eta(11)*h(11)
dfd(1)=rrl*fd(3)
dfd(2)=0.
dfd(3)=-rrl*fd(1)
do i=1,3
dtd(i)=HG(11,i)-rxf(i)
enddo

C ROTATE INTO THIGH COORDINATES
C
doi=1,3

for(10,i)=-(relrm(10,i,1)*fd(1)+relrm(10,i,2)*fd(2)
 & +relrm(10,i,3)*fd(3))
tor(10,i)=-(relrm(10,i,1)*td(1)+relrm(10,i,2)*td(2)
 & +relrm(10,i,3)*td(3))
enddo
do i=1,3
fd(i)=mass(12)*acc(12,i)-wt(12,i)
enddo
rrl=-eta(12)*h(12)
rxf(1)=rrl*fd(3)
rxf(2)=0
rxf(3)=-rrl*fd(1)

C COMPUTE MOMENT
C
do i=1,3
td(i)=hg(12,i)-rxf(i)
enddo

C ROTATE INTO THIGH COORDINATES
C
doi=1,3

for(11,i)=-(relrm(11,i,1)*fd(1)+relrm(11,i,2)*fd(2)
 & +relrm(11,i,3)*fd(3))
tor(11,i)=-(relrm(11,i,1)*td(1)+relrm(11,i,2)*td(2)
 & +relrm(11,i,3)*td(3))
enddo

C FIND FORCES AT HIPS
C
do i=1,3
fd(i)=mass(9)*acc(9,i)-for(10,i)-wt(9,i)
enddo

C FIND MOMENTS AT HIPS
C
C MOMENT ARISING FROM KNEE FORCE
C
rrl=(1-eta(9))*h(9)
r2=-eta(9)*h(9)
rxf(1)=rrl*for(10,3)
rxf(2)=0
rxf(3)=-rr1*for(10,1)

MOMENT ARISING FROM HIP FORCE

rxf2(1)=r2*fd(3)
rxf2(2)=0
rxf2(3)=-r2*fd(1)

COMPUTE MOMENT

do i=1,3
   td(i)=hg(9,i)-rxf(i)-rxf2(i)-tor(10,i)
enddo

ROTATE INTO PELVIS COORDINATES

do i=1,3
   for(8,i)=-(relrm(8,i,1)*fd(l)+relrm(8,i,2)*fd(2)
&   +relrm(8,i,3)*fd(3))
&   tor(8,i)=-(relnti(8,i,1)*td(l)+relrin(8,i,2)*td(2)
&   +relrin(8,i,3)*td(3))
enddo

LEFT HIP

do i=1,3
   fd(i)=inass(10)*acc(10,i)-for(11,i)-wt(10,i)
enddo

MOMENT ARISING FROM KNEE FORCE

rr1=(1-eta(10))*h(10)
r2=-eta(10)*h(10)
rxf(1)=rr1*for(11,3)
rxf(2)=0
rxf(3)=-rr1*for(11,1)

MOMENT ARISING FROM HIP FORCE

rxf2(1)=r2*fd(3)
rxf2(2)=0
rxf2(3)=-r2*fd(1)

COMPUTE MOMENT

do i=1,3
   td(i)=hg(10,i)-rxf(i)-rxf2(i)-tor(11,i)
enddo

ROTATE INTO PELVIS COORDINATES
C

do i=1,3
  for(9,i)=-(relm(9,i,1)*fd(1)+relm(9,i,2)*fd(2)
  &
    +relm(9,i,3)*fd(3))
  tor(9,i)=-(relm(9,i,1)*td(1)+relm(9,i,2)*td(2)
  &
    +relm(9,i,3)*td(3))
  enddo

C

  FIND FORCES AT LOWER BACK "JOINT"

  do i=1,3
    fd(i)=mass(3)*acc(3,i)-wt(3,i)-for(8,i)-for(9,i)
  enddo

C

  FIND MOMENTS AT LOWER BACK JOINT

  rrl=r(9)
  r2=h(3)/2.

  MOMENT ARISING FROM RIGHT HIP FORCE

  rxf(1)=r2*for(8,3)-rrl*for(8,2)
  rxf(2)=-rrl*for(8,1)
  rxf(3)=-r2*for(8,1)

  MOMENT ARISING FROM LEFT HIP FORCE

  rxf2(1)=r2*for(9,3)+rrl*for(9,2)
  rxf2(2)=rrl*for(9,1)
  rxf2(3)=-r2*for(9,1)

  MOMENT ARISING FROM LOWER BACK FORCE

  rxf3(1)=-r2*fd(3)
  rxf3(2)=0
  rxf3(3)=r2*fd(1)

  COMPUTE MOMENT

  do i=1,3
    td(i)=hg(3,i)-rxf(i)-rxf2(i)-rxf3(i)-tor(8,i)
  &
    tor(9,i)
  enddo

  ROTATE INTO MID-TRUNK COORDINATES

  do i=1,3
    for(7,i)=-(relm(7,i,1)*fd(1)+relm(7,i,2)*fd(2)
  &
    +relm(7,i,3)*fd(3))
    tor(7,i)=-(relm(7,i,1)*td(1)+relm(7,i,2)*td(2)

& +relrm(7,i,3)*td(3))
enddo

C
C UPPER SPINE "JOINT"
C
rr1=h(1)/2.
C
MOMENT ARISING FROM LOWER BACK FORCE
C
rxf(1)=rr1*for(7,3)
rxf(2)=0
rxf(3)=-rr1*for(7,1)
C
MOMENT ARISING FROM UPPER BACK FORCE
rxf2(1)=-rr1*for(6,3)
rxf2(2)=0
rxf2(3)=rr1*for(6,1)
C
COMPUTE MOMENT
C
DO i=1,3
   tor(6,i)=hg(1,i)-rxf(i)-rxf2(i)-tor(7,i)
endo
return
end

Subroutine LANDMARK
SUBROUTINE LANDMARK(H,ETA,R,CG,Y,P,ROT,MARK)
C
C THIS SUBROUTINE COMPUTES 3-D LOCATIONS OF 21 LANDMARKS IN
C GLOBAL COORDINATES FOR USE IN THE ANIMATION SUBROUTINE
C
REAL Y(6),P(12,3),ROT(12,3,3),MARK(22,3),CG(3)
REAL RMAIN(3,3),RR(3,3),V(3),V1(3),V2(3)
REAL H(12),ETA(12),R(12)
C
C LOCATE THE FOLLOWING LANDMARKS: (1) TOP OF HEAD (2) CHIN
C (3) SUPRASTERNALE (4) RIGHT SHOULDER (5) RIGHT ELBOW
C (6) RIGHT WRIST (7) RIGHT KNUCKLE III (SET = WRIST)
C (8) LEFT SHOULDER (9) LEFT ELBOW (10) LEFT WRIST (11) LEFT
C KNUCKLE III (SET = WRIST) (12) RIGHT HIP (13) RIGHT KNEE
C (14) RIGHT ANKLE (15)-(16) RIGHT FOOT (SET = ANKLE)
C (17) LEFT HIP (18) LEFT KNEE (19) LEFT ANKLE (20)-(21)
C LEFT FOOT (SET = ANKLE)
C
C CENTER OF GRAVITY LOCATION
C
   do i=1,3
      mark(22,i)=cg(i)
endo
ROTATION MATRIX FROM SEGMENT 1 TO GLOBAL COORDINATES

\[
\begin{align*}
\text{rmain}(1,1) &= \cos(y(1)) \cos(y(3)) - \sin(y(1)) \sin(y(2)) \\
&\quad \times \sin(y(3)) \\
\text{rmain}(1,2) &= -\sin(y(1)) \cos(y(2)) \\
\text{rmain}(1,3) &= \cos(y(1)) \sin(y(3)) + \sin(y(1)) \sin(y(2)) \\
&\quad \times \cos(y(3)) \\
\text{rmain}(2,1) &= \sin(y(1)) \cos(y(3)) + \cos(y(1)) \sin(y(2)) \\
&\quad \times \sin(y(3)) \\
\text{rmain}(2,2) &= \cos(y(1)) \cos(y(2)) \\
\text{rmain}(2,3) &= \sin(y(1)) \sin(y(3)) - \cos(y(1)) \sin(y(2)) \\
&\quad \times \cos(y(3)) \\
\text{rmain}(3,1) &= -\sin(y(3)) \cos(y(2)) \\
\text{rmain}(3,2) &= \sin(y(2)) \\
\text{rmain}(3,3) &= \cos(y(2)) \cos(y(3))
\end{align*}
\]

(1) & (2)

v(1)=0. \\
v(2)=.5*h(4) \\
v(3)=0. \\
do i=1,3 \\
do j=1,3 \\
\text{rr}(i,j)=\text{rot}(4,i,j) \\
enddo \\
enddo \\
call matvec(rr,v,v2) \\
do i=1,3 \\
v(i)=v2(i)+p(4,i) \\
v1(i)=-v2(i)+p(4,i) \\
enddo \\
call matvec(rmain,v1,v2) \\
do i=1,3 \\
mark(1,i)=v2(i)+cg(i) \\
enddo \\
call matvec(rmain,v,v2) \\
do i=1,3 \\
mark(2,i)=v2(i)+cg(i) \\
enddo

(3)

v(1)=0. \\
v(2)=-.5*h(2) \\
v(3)=0. \\
do i=1,3 \\
do j=1,3 \\
\text{rr}(i,j)=\text{rot}(2,i,j)
enddo
call matvec(rr,v,v2)
do i=1,3
  v(i)=v2(i)+p(2,i)
endo
call matvec(rmain,v,v2)
do i=1,3
  mark(3,i)=v2(i)+cg(i)
endo
v(1)=0.
v(2)=0.
v(3)=-r(2)
call matvec(rr,v,v2)
call matvec(rmain,v2,v1)
do i=1,3
  mark(4,i)=v1(i)+mark(3,i)
  mark(8,i)=-v1(i)+mark(3,i)
endo
v(1)=0.
v(2)=-(1-eta(7))*h(7)
v(3)=0.
doi=1,3
do j=1,3
  rr(i,j)=rot(7,i,j)
endo
do i=1,3
do j=1,3
  rr(i,j)=rot(7,i,j)
endo
call matvec(rr,v,v2)
doi=1,3
  v(i)=v2(i)+p(7,i)
endo
call matvec(rmain,v,v2)
doi=1,3
  mark(5,i)=v2(i)+cg(i)
endo
v(1)=0.
v(2)=-(1-eta(5))*h(5)
v(3)=0.
doi=1,3
do j=1,3
  rr(i,j)=rot(5,i,j)
endo
do i=1,3
do j=1,3
  rr(i,j)=rot(5,i,j)
endo
call matvec(rr,v,v2)
doi=1,3
  v(i)=v2(i)+p(5,i)
endo
call matvec(rmain,v,v2)
doi=1,3
  mark(6,i)=v2(i)+cg(i)
enddo
do i=1,3
  mark(7, i) = mark(6, i)
enddo
v(1) = 0.
v(2) = -(1 - eta(8)) * h(8)
v(3) = 0.
do i = 1, 3
do j = 1, 3
  rr(i, j) = rot(8, i, j)
enddo
call matvec(rr, v, v2)
do i = 1, 3
  v(i) = v2(i) + p(8, i)
enddo
call matvec(rmain, v, v2)
do i = 1, 3
  mark(9, i) = v2(i) + cg(i)
enddo
v(1) = 0.
v(2) = -(1 - eta(6)) * h(6)
v(3) = 0.
do i = 1, 3
do j = 1, 3
  rr(i, j) = rot(6, i, j)
enddo
call matvec(rr, v, v2)
do i = 1, 3
  v(i) = v2(i) + p(6, i)
enddo
call matvec(rmain, v, v2)
do i = 1, 3
  mark(10, i) = v2(i) + cg(i)
enddo
do i = 1, 3
  mark(11, i) = mark(10, i)
enddo
v(1) = 0.
v(2) = .5 * h(3)
v(3) = - r(9)
do i = 1, 3
do j = 1, 3
  rr(i, j) = rot(3, i, j)
enddo
call matvec(rr, v, v2)
do i = 1, 3
  v(i) = v2(i) + p(3, i)
enddo
enddo
call matvec(rmain,v,v2)
do i=1,3
mark(12,i)=v2(i)+c(i)
endo
v(1)=0.
v(2)=(1-eta(9))*h(9)
v(3)=0.
do i=1,3
do j=1,3
rr(i,j)=rot(9,i,j)
endo
call matvec(rr,v,v2)
do i=1,3
v(i)=v2(i)+p(9,i)
endo
call matvec(rmain,v,v2)
do i=1,3
mark(13,i)=v2(i)+c(i)
endo
v(1)=0.
v(2)=(1-eta(11))*h(11)
v(3)=0.
do i=1,3
do j=1,3
rr(i,j)=rot(11,i,j)
endo
call matvec(rr,v,v2)
do i=1,3
v(i)=v2(i)+p(11,i)
endo
call matvec(rmain,v,v2)
do i=1,3
mark(14,i)=v2(i)+c(i)
endo
do i=1,3
mark(15,i)=mark(14,i)
mark(16,i)=mark(14,i)
endo
v(1)=0.
v(2)=.5*h(3)
v(3)=r(10)
do i=1,3
do j=1,3
rr(i,j)=rot(3,i,j)
endo
call matvec(rr,v,v2)
do i=1,3
  v(i)=v2(i)+p(3,i)
enddo

call matvec(rmain,v,v2)
do i=1,3
  mark(17,i)=v2(i)+cg(i)
enddo

v(1)=0.
v(2)=(1-eta(10))*h(10)
v(3)=0.
do i=1,3
do j=1,3
  rr(i,j)=rot(10,i,j)
endo
dono

call matvec(rr,v,v2)
do i=1,3
  v(i)=v2(i)+p(10,i)
enddo

call matvec(rmain,v,v2)
do i=1,3
  mark(18,i)=v2(i)+cg(i)
enddo

v(1)=0.
v(2)=(1-eta(12))*h(12)
v(3)=0.
do i=1,3
do j=1,3
  rr(i,j)=rot(12,i,j)
endo
dono

call matvec(rr,v,v2)
do i=1,3
  v(i)=v2(i)+p(12,i)
enddo

call matvec(rmain,v,v2)
do i=1,3
  mark(19,i)=v2(i)+cg(i)
enddo

do i=1,3
  mark(20,i)=mark(19,i)
endo

do i=1,3
  mark(21,i)=mark(19,i)
enddo

return
end

Subroutine ANIMATE
C  THIS SUBROUTINE USES DAPENA'S ALGORITHMS TO ANIMATE
C  THE GYMNAST
C  ADAPTED FROM MAINJMP (DAPENA, SEE TEXT FOR REFERENCE)
C
INCLUDF 'FGRAPH.FI'
SUBROUTINE ANIMATE(MARK)
INCLUDF 'FGRAPH.FD'
DIMENSION COOR(21,3),CM(3),Z1(3),Z2(3),Z3(3),Y1(3)
REAL LM(3),MARK(22,3)
COMMON/ESCHER1/SCALH,SCALV,OGH,OGV
COMMON/ESCHER2/PRESH,PRESV
COMMON/RETINA/XMAT,Z1
COMMON/PROPERTY/STAT,XMASS,MF
C
C SETTING THE MASS AND THE STANDING HEIGHT OF THE SUBJECT (IN
C KILOGRAMS AND IN METERS, RESPECTIVELY).
C
stat=1.97
Xmass=70.3
C
C SETTING THE SEX OF THE SUBJECT (MALE: MF=1; FEMALE: MF=2).
C
MF=1
xlong=0.
xlat=90.
call initplt
do j=1,21
  coor(j,1)=mark(j,1)/100.
  coor(j,2)=mark(j,2)/100.
  coor(j,3)=mark(j,3)/100.
endo
do j=1,3
  cm(j)=mark(22,j)/100.
endo
C
C SETTING THE HORIZONTAL AND VERTICAL SCALES FOR THE DRAWING
C (SCALH AND SCALV, RESPECTIVELY, IN INCHES OF THE GRAPH PER
C REAL-LIFE METER).
C
SCALH=1.5
SCALV=SCALH
CALL MATVIEW(XLONG,XLAT,XMAT)
C
C CALCULATION OF THE R' COORDINATES OF A "CENTRAL" POINT OF
C THE GRAPH (Z1).
C
do i=1,3
  lm(i)=0.
undo
do i=1,3
CALL VIEW(XMAT,Z2,Z1)

C SETTING THE X' AND Y' COORDINATES OF THE LOWER LEFT CORNER OF THE GRAPH

OGH=-1.5
OGV=-2.0

C CALCULATION OF THE R' COORDINATES OF THE BODY LANDMARKS (COORTF) RELATIVE TO THE R' POSITION OF THE CENTRAL POINT.

DO 200 J=1,21
   DO 201 K=1,3
      Z3(K)=COOR(J,K)
   201 CONTINUE
   CALL SEE(Z3,Y1)
   DO 101 K=1,3
      COORTF(J,K)=Y1(K)
   101 CONTINUE
200 CONTINUE

C DRAWING THE ATHLETE.

CALL CYLBOD(COORTF)

C ENDING THE PLOT.

call plot(0,0,998)
return
END

Subroutines CLOSPL,MOVE,OPENPL,ERASE,CONT and SPACE

C THESE SUBROUTINES ADAPT DAPENA'S PROGRAM TO MICROSOFT FORTRAN GRAPHICS COMMANDS

SUBROUTINE CLOSPL
   INCLUDE 'FGRAP.FD'
   dummy=setvideomode($defaultmode)
   return
end

SUBROUTINE MOVE(WX,WY)
   INCLUDE 'FGRAP.FD'
   DOUBLE PRECISION WX,WY
   RECORD/WXYCOORD/S
   call moveto_w(wx,wy,s)
SUBROUTINE OPENPL
  INCLUDE 'FGRAPH.FD'
  INTEGER*2 MODESTATUS, MAXX, MAXY
  RECORD/VIDEOCONFIG/MYSCREEN
  COMMON MAXX MAXY
  modestatus=setvideomode($MAXRESMODE)
  if (modestatus.eq.0) stop 'error: cannot set graphics & mode'
  call getvideoconfig(myscreen)
  maxx=myscreen.numxpixels-1
  maxy=myscreen.numypixels-1
  return
end

SUBROUTINE ERASE
  INCLUDE 'FGRAPH.FD'
  CALL clearscreens($GCLEARSSCREEN)
  return
end

SUBROUTINE CONT(wx,wy)
  INCLUDE 'FGRAPH.FD'
  DOUBLE PRECISION WX, WY
  status=lineto_w (wx, wy)
  return
end

SUBROUTINE SPACE (IXLL, IYLL, IXUR, IYUR)
  INCLUDE 'FGRAPH.FD'
  DOUBLE PRECISION IXLL, IYLL, IXUR, IYUR
  status=setwindow(.true., ixll, iyur, ixur, iyll)
  return
end

SUBROUTINE FILTERA (FN, NF)
  CHARACTER*50 FN
  INTEGER NF
  REAL Q(8), QP
  n=7
  open (50, file=fn)
  open (51, file='c:\fortran\dummy1.fil')
  do k=1, n
    read (50,*) q(k)
    qp=q(k)
write(51,*)qp
enddo
do i=n+1,nf
read(50,*)q(n+1)
qp=0.
do k=1,n+1
    qp=qp+q(k)/n
enddo
write(51,*)qp
do k=1,n
q(k)=q(k+1)
enddo
do i=n+1,nf
read(50,*)q(n+1),q2(n+1)
qp=0.
qp2=0.
do k=1,n+1
    qp=qp+q(k)/n
    qp2=qp2+q2(k)/n
enddo
write(51,*)qp,qp2
enddo
close(50)
close(51)
open (50, file=’c:\fortran\ dummy1. fil’)
open (51, file=fn)
do i=1,nf
read(50,*)qp
write(51,*)qp
enddo
close(50)
close(51)
return
end

Subroutine FILTERB
SUBROUTINE FILTERB(FN,NF)
CHARACTER*50 FN
INTEGER NF
REAL Q(8),Q2(8),QP,QP2
n=7
open (50, file=fn)
open (51, file=’c:\fortran\ dummy1. fil’)
do k=1,n
read (50, *) q(k),q2(k)
qp=q(k)
write(51,*)qp,qp2
enddo
do i=n+1,nf
read(50,*)q(n+1),q2(n+1)
qp=0.
qp2=0.
do k=1,n+1
    qp=qp+q(k)/n
    qp2=qp2+q2(k)/n
enddo
write(51,*)qp,qp2
do k=1,n
q(k)=q(k+1)
q(k+1)
Subroutine ROT1

SUBROUTINE ROT1(A,P,R)
REAL A(3,3), P(3,3), AMEAN(3), PMEAN(3), MT(3,3), MAG
REAL PAT(3,3,3), PATMEAN(3,3), M(3,3), MTM(3,3), D11,D22,D33
REAL V(3,3), MV(3,3), VT(3,3), R(3,3), X, X1, X2, X3
n=3
   do 7000 i=1,3
      amean(i)=0.
      pmean(i)=0.
   7000 continue
   do 7050 j=1,3
      do 7050 i=1,n
         amean(j)=amean(j)+a(i,j)
         pmean(j)=pmean(j)+p(i,j)
   7050 continue
C compute the average vectors
   do 7100 i=1,3
      amean(i)=amean(i)/n
      pmean(i)=pmean(i)/n
   7100 continue
C compute matrices PAT
   do 7150 i=1,n
      do 7150 j=1,3
         do 7150 k=1,3
            pat(i,j,k)=p(i,j)*a(i,k)
         7150 continue
      7150 continue
   do 7200 i=1,3
      do 7200 j=1,3
         patmean(i,j)=pmean(i)*amean(j)
   7200 continue
Compute matrix M

```
do 7250 i=2,n
   do 7250 j=1,3
      do 7250 k=1,3
         pat(1,j,k)=pat(1,j,k)+pat(i,j,k)
    continue
   do 7300 j=1,3
      do 7300 k=1,3
         pat(1,j,k)=pat(1,j,k)/n
         m(j,k)=pat(1,j,k)-patmean(j,k)
    continue
```

Find the matrix MTM

First, transpose matrix M and put in matrix MT

```
do 7350 j=1,3
   do 7350 k=1,3
      mt(j,k)=m(k,j)
    continue
call mm(mt,m,mtm)
```

find eigenvalues and eigenvectors for MTM

Use Newton’s method to find three roots for the cubic equation (to find eigenvalues)

Using matrix MTM find the cubic equation’s coefficients 1,a2,a1,a0

```
a2=-(mtm(1,1)+mtm(2,2)+mtm(3,3))
a1=mtm(1,1)*mtm(2,2)+mtm(1,1)*mtm(3,3)+mtm(3,3)
&   *mtm(2,2)-mtm(1,2)**2-mtm(1,3)**2-mtm(3,2)**2
a0=mtm(1,3)**2*mtm(2,2)+mtm(1,2)**2*mtm(3,3)
&   +mtm(3,2)**2*mtm(1,1)-mtm(1,1)*mtm(2,2)*mtm(3,3)
&   -2*mtm(1,2)*mtm(3,2)*mtm(1,3)
x=1.
n=0
7400 x1=x-(x**3+a2*x**2+a1*x+a0)/(3*x**2+2*a2*x+a1)
if (abs(x1-x).gt.1e-6) then
   n=n+1
if (n.gt.1e3) goto 7450
   x=x1
   go to 7400
endif
```

Find quadratic equation that remains after first root
found. Let coefficients be 1, b1, b0

7450

b1=a2-x1
b0=a1-b1*x1
if (b1**2-4*b0.Le.0) Then
r1=-b1/2
r2=-b1/2
else
r1=(-b1+sqrt(b1**2-4*b0))/2
r2=(-b1-sqrt(b1**2-4*b0))/2
endif

Use r1 and r2 as first try at roots for cubic equation

x=r1
n=0

7500

x2=x-((x**3+a2*x**2+a1*x+a0)/(3*x**2+2*a2*x+a1)
if (abs(x2-x).gt.1e-6) then
x=x2
n=n+1
if (n.gt.1000) goto 7550
go to 7500
endif

7550 x=r2
n=0

7600

x3=x-((x**3+a2*x**2+a1*x+a0)/(3*x**2+2*a2*x+a1)
if (abs(x3-x).gt.1e-6) then
x=x3
n=n+1
if (n.gt.1e3) goto 7650
goto 7600
endif

compute eigenvectors (orthonormal set)

first, order the eigenvalues in order (largest to smallest)

7650
d11=x2
d22=x3
d33=x1
if (d33.gt.d22) then
d22=x1
d33=x3
endif

First, look for rotation about only the z-axis (likely in simulation)

if ((mtm(3,1).eq.0).and.(mtm(3,2).eq.0)) then
use the upper left hand part of the matrix to solve for $x_1$ and $x_2$

\[
\begin{align*}
    v(1,1) &= 1. \\
    v(2,1) &= \frac{(\text{mtm}(1,1) - d11)}{\text{mtm}(1,2)} \\
    v(3,1) &= 0. \\
    v(1,2) &= 1. \\
    v(2,2) &= \frac{\text{mtm}(2,1)}{\text{mtm}(2,2) - d22} \\
    v(3,2) &= 0. \\
    \text{flag} &= 1
\end{align*}
\]

else

\[
\begin{align*}
    \text{factor1} &= \text{mtm}(2,2) - d11 \\
    \text{factor2} &= (\text{mtm}(2,2) - d11) \times \text{mtm}(1,3) - \text{mtm}(1,2) \times \text{mtm}(2,3) \\
    \text{if } &((\text{factor1}. \text{Eq}.0) \text{ or } (\text{factor2}. \text{Eq}.0)) \text{ Then} \\
    \text{factor3} &= (\text{mtm}(2,1) \times \text{mtm}(1,3) - \text{mtm}(2,3) \times (\text{mtm}(1,1) - d11)) \\
    \text{if } &((\text{factor3}. \text{Eq}.0) \text{ then} \\
    v(3,1) &= 1. \\
    v(1,1) &= \frac{(\text{mtm}(1,3) \times \text{mtm}(3,2) - (\text{mtm}(3,3) - d11) \times \text{mtm}(1,2))}{((\text{mtm}(1,1) - d11) \times \text{mtm}(3,2) - \text{mtm}(3,1) \times \text{mtm}(1,2))} \\
    \text{endif} \\
    v(2,1) &= 1. \\
    v(3,1) &= -v(3,1) \times \text{mtm}(2,3) / \text{mtm}(2,1) \\
    \text{else} \\
    v(1,1) &= 1. \\
    v(3,1) &= (d11 - \text{mtm}(1,1)) + v(1,2) \times 2 / (\text{mtm}(2,2) - d11) \\
    \text{endif} \\
    \text{if } ((\text{mtm}(2,2) - d22). \text{Eq}.0) \text{ Then} \\
    v(2,1) &= 1. \\
    v(3,1) &= -\text{mtm}(1,2) \times \text{mtm}(2,1) / (\text{mtm}(1,3) - \text{mtm}(2,3)) \\
    \text{else} \\
    v(1,2) &= 1. \\
    v(3,2) &= ((d22 - \text{mtm}(1,1)) + v(1,2) \times 2 / (\text{mtm}(2,2) - d22)) \\
    \text{endif} \\
    \text{endif}
\end{align*}
\]

normalize

\[
\begin{align*}
    \text{mag} &= \sqrt{v(1,1)^2 + v(2,1)^2 + v(3,1)^2} \\
    v(1,1) &= v(1,1) / \text{mag} \\
    v(2,1) &= v(2,1) / \text{mag} \\
    v(3,1) &= v(3,1) / \text{mag} \\
    \text{mag} &= \sqrt{v(1,2)^2 + v(2,2)^2 + v(3,2)^2}
\end{align*}
\]
\[ v(l,2) = \frac{v(l,2)}{\text{mag}} \]
\[ v(2,2) = \frac{v(2,2)}{\text{mag}} \]
\[ v(3,2) = \frac{v(3,2)}{\text{mag}} \]

The third eigenvector is the cross product of 1 and 2:

\[ v(1,3) = v(2,1) \times v(3,2) - v(3,1) \times v(2,2) \]
\[ v(2,3) = v(3,1) \times v(1,2) - v(1,1) \times v(3,2) \]
\[ v(3,3) = v(1,1) \times v(2,2) - v(2,1) \times v(1,2) \]

Multiply matrices \( M \) and \( V \) to get \( MV \)

```
call mm(m,v,mv)
```

Rotation matrix is \( \frac{1}{d_{11}} m_{v1}, \frac{1}{d_{22}} m_{v2}, \frac{1}{d_{11}d_{22}} m_{v1xv2} \)

multiplied by \( v_t \)

\[ d_{11} = \sqrt{d_{11}} \]
\[ d_{22} = \sqrt{d_{22}} \]

Cross product of the first two columns

\[ m_{v}(1,3) = m_{v}(2,1) \times m_{v}(3,2) - m_{v}(3,1) \times m_{v}(2,2) \]
\[ m_{v}(2,3) = m_{v}(3,1) \times m_{v}(1,2) - m_{v}(1,1) \times m_{v}(3,2) \]
\[ m_{v}(3,3) = m_{v}(1,1) \times m_{v}(2,2) - m_{v}(2,1) \times m_{v}(1,2) \]

Normalize magnitudes

```
    do 7700 i=1,3
        mv(i,1)=mv(i,1)/d_{11}
        mv(i,2)=mv(i,2)/d_{22}
        mv(i,3)=mv(i,3)/d_{11}/d_{22}
    7700 continue
```

Compute \( V \) transpose = \( v_t \)

```
    do 7750 j=1,3
        do 7750 k=1,3
            vt(j,k)=v(k,j)
        7750 continue
```

Call \( mm(mv,vt,r) \)

```
if (flag.eq.1) then
    r(3,3)=1.
    r(3,1)=0.
    r(3,2)=0.
    r(1,3)=0.
    r(2,3)=0.
endif
```
Subroutine INTT

SUBROUTINE INTT(FN1,NF)
REAL A(200), B(200), G(200)
REAL AD(200), BD(200), GD(200), ADD(200), BDD(200), GDD(200)
REAL T(200), WK(400)
CHARACTER*50, FN1
INTEGER NF
open (15, file=fn1)
do k=1, nf
   t(k)=.02*(k-1)
   read(15,*) A(k), B(k), G(k)
enddo
close(15)
call pchez(nf, t, a, ad, .true., wk, 400, ierr)
call pche2(nf, t, ad, add, .true., wk, 400, ierr)
call pchez(nf, t, b, bd, .true., wk, 400, ierr)
call pche2(nf, t, bd, bdd, .true., wk, 400, ierr)
call pchez(nf, t, g, gd, .true., wk, 400, ierr)
call pche2(nf, t, gd, gdd, .true., wk, 400, ierr)
open (15, file=fn1)
do i=1, nf
   write(15, *) A(i), ad(i), add(i)
   write(15, *) B(i), bd(i), bdd(i)
   write(15, *) G(i), gd(i), gdd(i)
endo
close(15)
return
end

Subroutine FILTER1

SUBROUTINE FILTER1(FN, NF)
CHARACTER*50, FN
INTEGER NF
REAL Q(3, 8), Q2(3, 8), QP(3), QP2(3)
n=7
open (50, file=fn)
open (51, file='c:\forTRAN\dummy1.fil')
do k=1, n
   do j=1, 3
      read (50,*) q(j, k), q2(j, k)
      qp(j)=q(j, k)
      qp2(j)=q2(j, k)
      write(51,*) qp(j), qp2(j)
endo
endo
do i=n+1, nf
   do j=1, 3
   endo
read(50,*), q(j, n+1), q2(j, n+1)
qp(j)=0.
qp2(j)=0.
do k=1,n+1
   qp(j)=qp(j)+q(j,k)/n
   qp2(j)=qp2(j)+q2(j,k)/n
enddo
write(51,*), qp(j), qp2(j)
do k=1,n
   q(j,k)=q(j,k+1)
   q2(j,k)=q2(j,k+1)
enddo
enddo
close(50)
close(51)
open(50, file='c:\fortran\dummy1.fil')
open(51, file=fn)
do i=1,nf
   do j=1,3
      read(50,*), dummy, q(k, 1), q(k, 2), q(k, 3)
      do i=1,3
         qp(i)=q(k,i)
      enddo
      write(51,*), qp(i)
   enddo
enddo
close(50)
close(51)
return
end

Subroutine FILTERT
SUBROUTINE FILTERT(FN, NF)
CHARACTER*50 FN
INTEGER NF
REAL Q(8, 3)
REAL QP(3)
n=7
open(50, file=fn)
open(51, file='c:\fortran\dummy1.fil')
do k=1,n
   read(50,*), dummy, q(k, 1), q(k, 2), q(k, 3)
   do i=1,3
      qp(i)=q(k,i)
   enddo
   write(51,*), qp(i)
enddo
do i=n+1,nf
   read(50,*), dummy, q(n+1, 1), q(n+1, 2), q(n+1, 3)
   do m=1,3
      qp(m)=0.
   endo
\[ q_{p}(m) = q_{p}(m) + \frac{q(k,m)}{n} \]

enddo
write(51,*) q_{p}(m)
do k=1,n
q(k,m) = q(k+1,m)
enddo
enddo
enddo

close(50)
close(51)
open(50, file='c:\fortran\dummy1.fil')
open(51, file=fn)
do i=1,nf
read(50,*) q(1), q(2), q(3)
write(51,*) q(1), q(2), q(3)
enddo
close(50)
close(51)
return
end

Subroutine FILTERT2

SUBROUTINE FILTERT2(FN,NF)
CHARACTER*50 FN
INTEGER NF
REAL Q(8,3)
REAL QP(3)

n=7
open (50, file=fn)
open (51, file='c:\fortran\dummy1.fil')
do k=1,n
read (50,*) q(k,1), q(k,2), q(k,3)
do i=1,3
qp(i) = q(k,i)
write(51,*) qp(i)
enddo
doto i=n+1,nf
read(50,*) q(n+1,1), q(n+1,2), q(n+1,3)
do m=1,3
qp(m) = 0.
do k=1,n+1
qp(m) = qp(m) + q(k,m)/(n+1)
enddo
do k=1,n
q(k,m) = q(k+1,m)
enddo
write(51,*)(qp(m),m=1,3)
enddo
close(50)
close(51)
open(50,file='c:\fortran\dummy1.fil')
open(51,file=fn)
do i=1,nf
read(50,*)qp(1),qp(2),qp(3)
write(51,*)qp(1),qp(2),qp(3)
endo
close(50)
close(51)
return
end

Subroutine ANGMOM
SUBROUTINE ANGMOM(H,RW,P,IDY,M,U1)
C
C THIS SUBROUTINE CALCULATES THE TOTAL ANGULAR MOMENTUM OF
C THE BODY AT THE POINT OF RELEASE
C
REAL RW(12,3),P(12,3),PDOT(12,3),IDY(12,3,3),M(12)
REAL H(3),U1(3),AMOM(12,3),PXPDOT(12,3),OMEGA(12,3)
REAL MAT(3,3),V(3),V2(3)
C
C CALCULATE THE PRODUCT OF THE INERTIA TENSOR AND THE
ANGULAR
C VELOCITY
C
do i=1,12
do j=1,3
omega(i,j)=u1(j)+rw(i,j)
v2(j)=omega(i,j)
endo
do j=1,3
do k=1,3
mat(j,k)=idy(i,j,k)
endo
endo
call matvec(mat,v2,v)
do j=1,3
amom(i,j)=v(j)
endo
endo

C
C CALCULATE PDOT=OMEGA X P
C
do i=1,12
   pdot(i,1)=omega(i,2)*p(i,3)-omega(i,3)*p(i,2)
pdot(i,2)=omega(i,3)*p(i,1)-omega(i,1)*p(i,3)
pdot(i,3) = omega(i,1)*p(i,2) - omega(i,2)*p(i,1)
enddo

C CALCULATE THE CROSS PRODUCT OF P X PDOT
C
do i=1,12
  pxpdot(i,1) = p(i,2)*pdot(i,3) - p(i,3)*pdot(i,2)
  pxpdot(i,2) = p(i,3)*pdot(i,1) - p(i,1)*pdot(i,3)
  pxpdot(i,3) = p(i,1)*pdot(i,2) - p(i,2)*pdot(i,1)
enddo

C TOTAL ANGULAR MOMENTUM IS THE SUM OF THE ANGMOMS AND M*THE CROSS PRODUCTS
C
do i=1,3
  H(i) = 0.
enddo
do i=1,12
do j=1,3
  H(j) = H(j) + amom(i,j) + m(i)*pxpdot(i,j)
enddo
dendo
return
end

Subroutine FINDU1
C THIS SUBROUTINE USES THE METHODS OF DAPENA (1979) TO COMPUTE THE ANGULAR VELOCITY OF THE REFERENCE SEGMENT
C
SUBROUTINE FINDU1(IDY,RW,U1,M,QDOT,Q,H)
REAL IDY(12,3,3),RW(12,3),U1(3),M(12),QDOT(12,3),Q(12,3)
REAL MAT(3,3),H(3),SUM(3),A(3,3),B(3),V(3),V2(3),V3(3)
REAL A1,A2,A3,B11,B12,B211,B212,B213,C1,D,E11,E12,E13
  & ,B12,B221,B223
REAL C2,E21,E22,E23,B13,B231,B232,B233,C3,E32,E33
  & ,P1,P2,P3,Q1
REAL Q2,Q3,R1,R2,R3,S1,S2,S3
REAL WORK(3)
INTEGER IWORK(3)
  a1 = idy(1,1,1)
  a2 = idy(1,2,2)
  a3 = idy(1,3,3)
  b11 = 0
  b12 = 0
  b13 = 0
  do i=2,12
do j=1,3
  v2(j) = rw(i,j)
enddo
do k=1,3
do j=1,3
 mat(j,k)=idy(i,j,k)
 enddo
 enddo
 call matvec(mat,v2,v)
 b11=b11+v(1)
 b12=b12+v(2)
 b13=b13+v(3)
 enddo
 b211=0
 b212=0
 b213=0
 b221=0
 b222=0
 b223=0
 b231=0
 b232=0
 b233=0
 do i=2,12
 b211=b211+idy(i,1,1)
 b212=b212+idy(i,1,2)
 b213=b213+idy(i,1,3)
 b221=b221+idy(i,2,1)
 b222=b222+idy(i,2,2)
 b223=b223+idy(i,2,3)
 b231=b231+idy(i,3,1)
 b232=b232+idy(i,3,2)
 b233=b233+idy(i,3,3)
 enddo
 c1=0
 c2=0
 c3=0
 mtot=0
 do i=1,3
 sum(i)=0
 enddo
 do i=1,12
 mtot=mtot+m(i)
 enddo
 do i=2,12
 do j=1,3
 sum(j)=m(i)*q(i,j)
 enddo
 enddo
 do i=2,12
 do j=1,3
 v(j)=mtot*q(i,j)-sum(j)
 v2(j)=m(i)/mtot*qdot(i,j)
 enddo
 v3(1)=v(2)*v2(3)-v(3)*v2(2)
\[ v_3(2) = v(3) \cdot v_2(1) - v(1) \cdot v_2(3) \]
\[ v_3(3) = v(1) \cdot v_2(2) - v(2) \cdot v_2(1) \]
\[ c_1 = c_1 + v_3(1) \]
\[ c_2 = c_2 + v_3(2) \]
\[ c_3 = c_3 + v_3(3) \]

\[ \text{do } d = 0 \]
\[ \text{do } i = 1, 3 \]
\[ \text{sum}(i) = 0 \]
\[ \text{enddo} \]
\[ \text{do } i = 1, 12 \]
\[ \text{do } j = 1, 3 \]
\[ \text{sum}(j) = \text{sum}(j) + m(i) / m_{\text{tot}} \cdot q(i, j) \]
\[ \text{enddo} \]
\[ \text{enddo} \]
\[ \text{do } i = 2, 12 \]
\[ \text{do } j = 1, 3 \]
\[ v(j) = m(i) \cdot (q(i, j) - \text{sum}(j)) \cdot q(i, j) \]
\[ d = d + v(j) \]
\[ \text{enddo} \]
\[ \text{enddo} \]
\[ e_{11} = 0 \]
\[ e_{12} = 0 \]
\[ e_{13} = 0 \]
\[ e_{21} = 0 \]
\[ e_{22} = 0 \]
\[ e_{23} = 0 \]
\[ e_{31} = 0 \]
\[ e_{32} = 0 \]
\[ e_{33} = 0 \]
\[ \text{do } i = 2, 12 \]
\[ e_{11} = e_{11} + (m(i) \cdot \text{sum}(i) - q(i, 1)) \cdot q(i, 1) \]
\[ e_{12} = e_{12} + (m(i) \cdot \text{sum}(2) - q(i, 2)) \cdot q(i, 1) \]
\[ e_{13} = e_{13} + (m(i) \cdot \text{sum}(3) - q(i, 3)) \cdot q(i, 1) \]
\[ e_{21} = e_{21} + (m(i) \cdot \text{sum}(1) - q(i, 1)) \cdot q(i, 2) \]
\[ e_{22} = e_{22} + (m(i) \cdot \text{sum}(2) - q(i, 2)) \cdot q(i, 2) \]
\[ e_{23} = e_{23} + (m(i) \cdot \text{sum}(3) - q(i, 3)) \cdot q(i, 2) \]
\[ e_{31} = e_{31} + (m(i) \cdot \text{sum}(1) - q(i, 1)) \cdot q(i, 3) \]
\[ e_{32} = e_{32} + (m(i) \cdot \text{sum}(2) - q(i, 2)) \cdot q(i, 3) \]
\[ e_{33} = e_{33} + (m(i) \cdot \text{sum}(3) - q(i, 3)) \cdot q(i, 3) \]
\[ \text{enddo} \]
\[ p_1 = a_1 + b_{211} + e_{11} + d \]
\[ p_2 = b_{221} + e_{21} \]
\[ p_3 = b_{231} + e_{31} \]
\[ q_1 = b_{212} + e_{11} \]
\[ q_2 = a_2 + b_{222} + e_{22} + d \]
\[ q_3 = b_{232} + e_{32} \]
\[ r_1 = b_{213} + e_{13} \]
\[ r_2 = b_{223} + e_{23} \]
\[ r_3 = a_3 + b_{233} + e_{33} + d \]
CALL SOLUTION OF THE MATRIX-VECTOR SYSTEM

\[
\begin{align*}
s_1 &= H(1) - (b_11 + c_1) \\
s_2 &= H(2) - (b_12 + c_2) \\
s_3 &= H(3) - (b_13 + c_3)
\end{align*}
\]

a(1,1) = p1  
a(1,2) = q1  
a(1,3) = r1  
a(2,1) = p2  
a(2,2) = q2  
a(2,3) = r2  
a(3,1) = p3  
a(3,2) = q3  
a(3,3) = r3  
b(1) = s1  
b(2) = s2  
b(3) = s3  
lda = 3  
n = 3  

\text{itask} = 1  
call sgefs(a,3,3,b,lda,work,iwork,rcond)  
\text{do } i = 1, 3  
ul(i) = b(i)  
\text{enddo}  
return
end

\textit{Subroutine F1}

\texttt{SUBROUTINE F1(THETA,U1,U1DOT)}
\texttt{COMMON/COEFFS/W(3,5),DELTA(3,8),GG(4),QQ(6),R1(3),S(3),}
\texttt{& T1(3),V(3),PLH(3),FRH(3),KP,MTOT,FL(3),FR(3),XDD(3)}

\texttt{REAL THETA(3),U1(3),U1DOT(3)}
\texttt{REAL DELTA,QQ,R1,S,T1,V,KP,PLH,W}
\texttt{REAL PRH,MTOT,XDD}
\texttt{REAL GG,PXMAR(3)}
\texttt{REAL IWW(3),FXN(3),IDOTAR(3)}

\texttt{a1=THETA(1)}
\texttt{a2=THETA(2)}
\texttt{a3=THETA(3)}

\texttt{C} \texttt{C CROSS PRODUCT OF PXMA FOR EACH SEGMENT MINUS THE UDOT TERMS}
\texttt{C} \texttt{(N-CM)}
pxmar(1)=(delta(1,7)+ul(1)*delta(1,8)-ul(2)
& *delta(2,6)-ul(3)*delta(3,6)
& -ul(1)*ul(2)*delta(3,2)+ul(1)*ul(3)*delta(2,2)
& +ul(2)*ul(3)*(delta(2,3)-delta(3,4))
& +ul(1)**2*(delta(3,3)-delta(2,4))-ul(2)**2*delta(2,4)
& +ul(3)**2*(delta(3,3)))
pxmar(2)=(delta(2,7)-ul(3)*delta(3,5)+ul(2)*delta(2,8)
& -ul(1)*delta(1,6)
& +ul(1)*ul(2)*delta(3,3)-ul(2)*ul(3)*delta(1,3)
& +ul(1)*ul(3)*(delta(3,4)-delta(1,2))
& +ul(1)**2*(delta(1,4)-delta(3,2))+ul(1)**2*delta(1,4)
& -ul(3)**2*(delta(3,2)))
pxmar(3)=(delta(3,7)+ul(3)*delta(3,8)-ul(2)*delta(2,5)
& -ul(1)*delta(1,5)+ul(1)*ul(2)*(delta(1,2)-delta(2,3))
& +ul(2)**2*delta(1,4)
& +ul(1)*ul(3)*delta(3,3)-ul(1)**2*delta(1,3)+ul(2)**2*
& delta(2,2)+ul(3)**2*(delta(2,2)-delta(1,3))

C I DOT ALPHA MINUS THE U DOT TERMS (N-CM)
do i=1,3
idotar(i)=(r1(i)+s(i)*u1(3)+t1(i)*u1(2)+v(i)*u1(1))
enddo

C I DOT OMEGA CROSS OMEGA
iww(1)=(w(1,1)+w(1,2)*u1(1)+w(1,3)*u1(2)+w(1,4)*u1(3)
& +w(1,5)*u1(3)*u1(2)+qq(2)*u1(1)*u1(3)-qq(3)*u1(1)*u1(2)
& +qq(5)*(u1(3)**2-u1(2)**2))
iww(2)=(w(2,1)+w(2,2)*u1(1)+w(2,3)*u1(2)+w(2,4)*u1(3)
& +w(2,5)*u1(1)*u1(3)+qq(5)*u1(1)*u1(2)-qq(2)*u1(2)*u1(3)
& +qq(3)*(u1(2)**2-u1(3)**2))
iww(3)=(w(3,1)+w(3,2)*u1(1)+w(3,3)*u1(2)+w(3,4)*u1(3)
& +w(3,5)*u1(1)*u1(2)+qq(3)*u1(2)*u1(3)-qq(5)*u1(1)*u1(3)
& +qq(2)*(u1(2)**2-u1(1)**2))

C COMPUTE THE FUNCTION WHICH IS EQUAL TO THE U DOT TERMS
C SUM OF MOMENTS + SUM OF PXF = IDOTA+PXMA-IWW
C SUM OF MOMENTS+SUM OF PXF=(IDOTAR+PXMAR+FXN)-IWW
C SUM OF MOMENTS+PXF-IDOTAR-PXMAR+IWW=FXN
C (N-CM)
do i=1,3
\[ \text{fxn}(i) = (i \cdot w(i) - p\times xmar(i) - idotar(i)) \]

enddo

WRITE THE DIFFERENTIAL EQUATIONS

if \( ((g_1g_4 - g_2g_3) \neq 0) \) Then
    \( u_1\dot{d}(4) = 0 \)
else
    \( u_1\dot{d}(1) = \frac{(g_4 \cdot \text{fxn}(1) - g_2 \cdot \text{fxn}(2)) \cdot (q_6 + \delta (2,3) + \delta (1,2)) + \text{fxn}(3) \cdot ((\delta (3,2) - q_3) \cdot g_4 - (\delta (3,3) - q_5) \cdot g_2)}{(g_1 \cdot g_4 - g_2 \cdot g_3)} \)
end if

if \( ((g_2g_3 - g_1g_4) \neq 0) \) then
    \( u_1\dot{d}(2) = 0 \)
    \( u_1\dot{d}(3) = 0 \)
else
    \( u_1\dot{d}(2) = \frac{(g_3 \cdot \text{fxn}(1) - g_1 \cdot \text{fxn}(2)) \cdot (q_6 + \delta (2,3) + \delta (1,2)) + \text{fxn}(3) \cdot ((\delta (3,2) - q_3) \cdot g_3 - (\delta (3,3) - q_5) \cdot g_1)}{(g_2 \cdot g_3 - g_1 \cdot g_4)} \)
    \( \text{if } ((q_6 + \delta (2,3) + \delta (1,2)) \neq 0) \text{ then} \)
    \( u_1\dot{d}(3) = 0 \)
    \( \text{else} \)
    \( u_1\dot{d}(3) = \frac{(\text{fxn}(3) + (\delta (1,4) - q_3) \cdot \text{fy}(g_4)) + (g_4 - (g_2 \cdot g_1)) \cdot \text{fy}(q_6 + \delta (2,3) + \delta (1,2)) + \text{fxn}(3) \cdot ((g_3 \cdot \text{fxn}(1) - g_1 \cdot \text{fxn}(2)) \cdot (q_6 + \delta (2,3) + \delta (1,2)) + \text{fxn}(3) \cdot ((g_3 \cdot \text{fxn}(1) - g_1 \cdot \text{fxn}(2)) \cdot (q_6 + \delta (2,3) + \delta (1,2))))}{(g_2 \cdot g_3 - g_1 \cdot g_4)} \)
    \( \text{end if} \)
end if

return
end

Subroutine INTEG

SUBROUTINE INTEG(THETA,U1)
C THIS SUBROUTINE COMPUTES YDOT(I) AS
C A FUNCTION OF Y(I) AND T.
C
REAL
THETA(3),U1(3),YDOT(3),Y1(3),AN(3),BN(3),CN(3),DN(3)
C THETA(1)=ALPHA, THETA(2)=BETA, THETA(3)=GAMMA, Y(4)=U11,
C Y(5)=U12,Y(6)=U13
C
C USE RUNGE-KUTTA METHOD FOR STEPWISE INTEGRATION: STEP
C SIZE=.001SEC
C (REFERENCE: KREYSZIG, E. ADVANCED ENGINEERING MATHEMATICS)

C
do i=1,3
  y1(i)=theta(i)
  enddo
  call f(y,u1,ydot)
do i=1,3
  an(i)=.001*ydot(i)
  Theta(i)=theta(i)+.5*an(i)
  enddo
  call f(y,u1,ydot)
do i=1,3
  bn(i)=.001*ydot(i)
  Theta(i)=y1(i)+.5*bn(i)
  enddo
  call f(y,u1,ydot)
do i=1,3
  cn(i)=.001*ydot(i)
  Theta(i)=y1(i)+cn(i)
  enddo
  call f(y,u1,ydot)
do i=1,3
  dn(i)=.001*ydot(i)
  enddo
do i=1,3
  Theta(i)=y1(i)+(an(i)+2.*bn(i)+2.*cn(i)+dn(i))/6.
  enddo
  return
  end

Subroutine F

SUBROUTINE F(THETA,U1,YDOT)
REAL THETA(3),U1(3),YDOT(3)
if (cos(theta(2)).Eq.0) Then
  theta(2)=theta(2)+.001
Endif
  ydot(1)=(u1(3)-u1(1)*sin(theta(3)))/cos(theta(2))
  ydot(2)=u1(1)*cos(theta(3))+u1(3)*sin(theta(3))
  ydot(3)=u1(2)-ydot(1)*sin(theta(2))
  return
end

Subroutine GETZ

C THIS SUBROUTINE SETS UP THE ROTATION MATRICES BETWEEN
C SEGMENTS FOR THE SIMULATION PROGRAM. THE INPUT IS JOINT
C ANGLES. THE OUTPUT IS THE ROTATION MATRICES.
C
SUBROUTINE GETZ(THETA,PLH,PRH,ZH)
C
JOINTS ARE NUMBERED AS FOLLOWS: 1-NECK, 2-R SHOULDER,
C 3-L SHOULDER, 4-R ELBOW, 5-L ELBOW, 6-UPPER BACK, 7-LOWER BACK, 8-RIGHT HIP, 9-LEFT HIP, 10-RIGHT KNEE, 11-LEFT KNEE

C

REAL RELRM(3,3),THETA(3),PLH(3),PRH(3),ZH(2),V1(3),V2(3)
relrm(1,1)=cos(theta(1))*cos(theta(3))-sin(theta(1))
& *sin(theta(2))*sin(theta(3))
relrm(1,2)=sin(theta(1))*cos(theta(2))
& *sin(theta(2))*cos(theta(3))
relrm(1,3)=cos(theta(1))*sin(theta(3))+sin(theta(1))
& *sin(theta(2))*sin(theta(3))
relrm(2,1)=sin(theta(1))*cos(theta(3))+cos(theta(1))
& *sin(theta(2))*sin(theta(3))
relrm(2,2)=cos(theta(1))*cos(theta(2))
relrm(2,3)=sin(theta(1))*sin(theta(3))-cos(theta(1))
& *sin(theta(2))*cos(theta(3))
relrm(3,1)=-sin(theta(3))*cos(theta(2))
relrm(3,2)=sin(theta(2))
relrm(3,3)=cos(theta(2))*cos(theta(3))
call matvec(relrm,plh,v1)
call matvec(relrm,prh,v2)
zh(1)=v2(3)
zh(2)=v1(3)
return
end

Subroutine FINDANG

SUBROUTINE FINDANG(NFRAMES,N,ANGLE,FN10,FN9)
C
C THIS SUBROUTINE USES THE ANGLE OF THE CG AT THE END OF THE
C FLIGHT PHASE, USES THE "MEASURED" ANGLE FROM THE ORIGINAL
C DATA TAKES THE DERIVATIVE, THEN USES NUMERICAL INTEGRATION
C TO DETERMINE THE CG ANGLE THROUGHOUT THE RECATCH PHASE
C
C REAL ANGLE, CGA(270),X(270),D(270),A(270),WK(340),T(270)
CHARACTER*50 FN9,FN10
INTEGER N,IERR
C
C READ IN THE CG DATA
C
open (25,file=fn10)
do i=1,nframes
   read(25,*)cga(i)
endo
close(25)
do i=n,nframes
   x(i-n+1)=cga(i)
t(n+1) = .02*(i-n+1)

enddo

ntot = nframes - n + 1

lwk = 340

call pchez(ntot, t, x, d, true, wk, lwk, ierr)

if (ierr ne 0) then
print *, ierr
read *, go
endif

C USING TRAPEZOIDAL RULE, CALCULATE CG LOCATIONS
C

a(l) = angle
dt = .02

do i = 2, ntot
a(i) = a(i-1) + (d(i) + d(i-1)) * .5 * dt
enddo

do i = 1, ntot
write (25, *) a(i)
enddo

close(25)

return

eend

Subroutine INTT2

SUBROUTINE INTT2(FN1,NF)
REAL A(500), B(500), G(500)
REAL AD(500), BD(500), GD(500), ADD(500), BDD(500), GDD(500)
REAL T(1000), WK(1000)
CHARACTER*50, FN1
INTEGER NF

open (15, file=fn1)
do k = 1, nf
   t(k) = .001*(k-1)
   read(15, *) A(k), B(k), G(k)
enddo

close(15)
call pchez(nf, t, a, ad, true, wk, 1000, ierr)
call pchez(nf, t, ad, add, true, wk, 1000, ierr)
call pchez(nf, t, b, bd, true, wk, 1000, ierr)
call pchez(nf, t, bd, bdd, true, wk, 1000, ierr)
call pchez(nf, t, g, gd, true, wk, 1000, ierr)
call pchez(nf, t, gd, gdd, true, wk, 1000, ierr)

open(15, file=fn1)
do i = 1, nf
   write(15, *) a(I), ad(I), add(I)
   write(15, *) b(I), bd(I), bdd(I)
   write(15, *) g(I), gd(I), gdd(I)
enddo
close(15)
return
end

**Subroutine INTERPOLATEC**

SUBROUTINE INTERPOLATEC(X, FN, N)
REAL X(170,3), XX1, XX2, XX3
REAL A(170)
CHARACTER*50 FN, FN1
DO I = 1, N
   A(I) = X(I, 1)
ENDDO
FN1 = 'C:\fortran\dummy1.fil'
call interpolatea(a, fn1, n, 0)
DO I = 1, N
   A(I) = X(I, 2)
ENDDO
FN1 = 'C:\fortran\dummy2.fil'
call interpolatea(a, fn1, n, 0)
DO I = 1, N
   A(I) = X(I, 3)
ENDDO
FN1 = 'C:\fortran\dummy3.fil'
call interpolatea(a, fn1, n, 0)
OPEN(20, FILE=FN)
OPEN(21, FILE='C:\fortran\dummy1.fil')
OPEN(22, FILE='C:\fortran\dummy2.fil')
OPEN(23, FILE='C:\fortran\dummy3.fil')
DO I = 1, N*20
   READ(21, *) XX1
   READ(22, *) XX2
   READ(23, *) XX3
   WRITE(20, '*') XX1, XX2, XX3
ENDDO
CLOSE(20)
CLOSE(21)
CLOSE(22)
CLOSE(23)
RETURN
END

**Program POSTPROCESSING**

C
C PART 4: PROVIDE OUTPUT FOR THE USER
C
PROGRAM POSTPROCESSING
CHARACTER*50 FN5, FN6, FN7
CHARACTER*25 DIR
CHARACTER*8 PREFIX
INTEGER NFRAMES,YN
open (1, file='c:\fortran\data\filename.dat')
read (1,*) dir
read (1,*) prefix
close(1)
print*," 
print *,"filename"
print *,dir//prefix//'***'
PRINT *," 
PRINT *,"DO YOU WANT TO VIEW THE ANIMATION AGAIN?"
122 PRINT *,"1=YES,2=NO"
read *,yn
if ((yn.Ne.1).And.(Yn.Ne.2)) Then
goto 122
endif
open (2, file='c:\fortran\ntot.dat')
Read(2,*) n
close(2)
print *,"number of frames",n
fn5=dir(1:len_trim(dir))/
print*,"K1'
fn6=dir(1:len_trim(dir))/
print *,"K2'
fn7=dir(1:len_trim(dir))//
print*,"Mks'
!OUTPUT FILE FOR KINEMATIC DATA
!OUTPUT FILE FOR KINETIC DATA
!OUTPUT FILE WITH MARKER POSITIONS
if (yn.Eq.1) Then
call reanimate(fn7,n)
endif
print *," 
PRINT *,"CALLING REORDERING PROGRAM"
call ftreorder(fn6,n)
call knreorder(fn5,n)
print *," 
PRINT *,"DO YOU WANT TO VIEW PLOTS OF THE FORCES AND & TORQUES?"
124 PRINT *,"1=YES,2=NO"
read *,yn
if ((yn.Ne.1).And.(Yn.Ne.2)) Then
goto 124
endif
if (yn.Eq.1) Then
call plotft(fn6,n)
endif
print *," 
print *," 
print *," 
print *,"
PRINT *,"DATA FROM THIS SIMULATION ARE STORED IN DOS 
& FILES"
PRINT *, "THE FILE FOR THE KINEMATIC DATA IS:"  
PRINT *, FN5  
PRINT *, " "  
PRINT *, "THE FILE FOR THE KINETIC DATA IS:"  
PRINT *, FN6  
PRINT *, " "  
PRINT *, "THESE FILES CAN BE READ DIRECTLY OR IMPORTED INTO  
& A"  
PRINT *, "SPREADSHEET PROGRAM FOR FURTHER ANALYSIS"  
PRINT *, " "  
PAUSE "PRESS RETURN TO CONTINUE"  
stop  
end
APPENDIX H

FORCES AT HANDS BASED ON SPRING MODEL OF BAR

The relationship between hand forces and bar deflections was derived in the Theory section of the main document. The final equations were:

\[ y_{rh} = \frac{F_{zby}(L-c)^2c^2}{3EIL} + \frac{F_{lby}(L-d)[c^3 - (L^2 - (L-d)^2)]}{6EIL} \]  
\[ x_{rh} = \frac{F_{zbx}(L-c)^2c^2}{3EIL} + \frac{F_{lx}(L-d)[c^3 - (L^2 - (L-d)^2)]}{6EIL} \]  
(1)

\[ y_{lh} = \frac{F_{zby}C[(L-d)^3 - (L^2 - c^2)(L-d)]}{6EIL} - \frac{F_{lby}(L-d)^2d^2}{3EIL} \]  
\[ x_{lh} = \frac{F_{zbx}C[(L-d)^3 - (L^2 - c^2)(L-d)]}{6EIL} - \frac{F_{lx}(L-d)^2d^2}{3EIL} \]  
(2)

The constant terms can be combined and these equations can be rearranged to solve for the forces in terms of the deflections:

\[ c_1 = \frac{-(L-c)^2c^2}{3EIL} \]  
(3)
\[ C_2 = \frac{(L-d) \left( C^3 - (L^2 - (L-d)^2) \right)}{6EIL} \]  
\[ C_3 = \frac{C \left( (L-d)^3 - (L^2 - c^2) (L-d) \right)}{6EIL} \]  
\[ C_4 = \frac{-(L-d)^2 d^2}{3EIL} \]

then:

\[ F_{lxh} = \frac{C_3 X_{xh} + C_1 X_{1h}}{C_2 C_3 - C_4 C_1} \]  
\[ F_{rxh} = \frac{C_4 X_{xh} + C_2 X_{1h}}{C_2 C_3 - C_4 C_1} \]  
\[ F_{lxh} = \frac{C_3 Y_{xh} + C_1 Y_{1h}}{C_2 C_3 - C_4 C_1} \]  
\[ F_{rxh} = \frac{C_4 Y_{xh} + C_2 Y_{1h}}{C_2 C_3 - C_4 C_1} \]  

These equations can be written in the form:

\[ F_{rxh} = K_1 X_{1h} + K_2 X_{xh} \]  
\[ F_{lxh} = K_3 X_{1h} + K_4 X_{xh} \]  
\[ F_{rxh} = K_1 Y_{1h} + K_2 Y_{xh} \]  
\[ F_{lxh} = K_3 Y_{1h} + K_4 Y_{xh} \]
where the constants $K_1$ to $K_4$, like $C_1$ to $C_4$ are functions of the distances $c$ and $d$, which are the distances from the end of the bar to the right and left hands, and the elastic modulus of the bar $E$, the area moment of inertia of the bar cross section $I$, and $L$, the length of the bar.

In order to use the bar forces in differential equations to solve for the body orientation and trajectory, they must be written in terms of the unknowns, $\alpha$, $\beta$, $\gamma$, $X_1$, $X_2$ and $X_3$. In Appendix A, the location vectors from the center of gravity to the hands were found in segment 1 coordinates. In order to determine these in global coordinates, the transformation matrix from the global coordinates to the segment 1 coordinates must be used. If

$$\overrightarrow{P_{th}} = A_1 \overrightarrow{D_1} + A_2 \overrightarrow{D_2} + A_3 \overrightarrow{D_3}$$

where the $A$'s are known values, then the vectors can be written in fixed global coordinates using the rotation matrix between segment 1 and the fixed reference frame.

$$\overrightarrow{P_{th}} = (A_4 (c \alpha \gamma - s \alpha s \beta \gamma) - A_5 s \alpha c \beta + A_6 (c \alpha \gamma + s \alpha s \beta \gamma)) \hat{i} + (A_4 (s \alpha \gamma + c \alpha s \beta \gamma) + A_5 c \alpha c \beta + A_6 (s \alpha \gamma - c \alpha s \beta \gamma)) \hat{j} + (-A_4 s \alpha \gamma c \beta + A_5 s \beta + A_6 c \beta \gamma) \hat{k}$$

(10)
\[ \vec{F}_{lh} = (A_1 (c_1 - s_1 s_2) - A_2 s_1 c_2 + A_3 (c_2 + s_2 c_2) \hat{i} \\
+ (A_1 (s_2 c_2 + c_1 s_2) + A_2 c_2 c_3 + A_3 (s_3 s_1 c_2) \hat{j} \\
+ (-A_1 s_1 c_2 + A_2 s_2 + A_3 c_2 c_2) \hat{k} \] (11)

Then \( x_{lh}, x_{rh}, y_{rh}, y_{rh} \) can be written in terms of the unknowns using the \( P \)'s. Since the \( P \) vector is the location of the hand minus the location of the center of gravity of the entire body, then the location of the hand is equal to the location of the center of gravity plus the \( P \) vector (See Figure A-3). Using this relationship, the forces at the hands can be rewritten in terms of the unknowns:

\[ \vec{F}_{rhx} = K_1 (X_{gh} + A_1 (c_1 - s_1 s_2) - A_2 s_1 c_2 + A_3 (c_2 + s_2 c_2)) \\
+ K_2 (X_{gh} + A_4 (c_2 + s_2 c_2) - A_5 s_1 c_2 + A_6 (c_2 + s_2 c_2)) \] (12)

\[ \vec{F}_{lhx} = K_2 (X_{gh} + A_1 (c_1 - s_1 s_2) - A_2 s_1 c_2 + A_3 (c_2 + s_2 c_2)) \\
+ K_4 (X_{gh} + A_4 (c_2 + s_2 c_2) - A_5 s_1 c_2 + A_6 (c_2 + s_2 c_2)) \]
This allows the writing of three second order differential equations from the force vector equation:

\[\Sigma F_x = M_{tot} \dot{x}_{G1}\]
\[= (K_1 + K_2) (x_{G1} + a_1 (c_1 c_2 - s_1 s_2) - a_2 s_2 c_1 + a_3 (c_1 s_2 + s_2 c_1)) + (K_2 + K_4) (x_{G1} + a_4 (c_1 c_2 - s_1 s_2) - a_5 s_2 c_1 + a_6 (c_1 s_2 + s_2 c_1))\]

\[\Sigma F_y = M_{tot} \dot{y}_{G2}\]
\[= (K_1 + K_2) (x_{G2} + a_1 (s_1 c_2 + c_1 s_2) + a_2 c_2 c_1 + a_3 (s_1 s_2 - c_1 c_2)) + (K_2 + K_4) (x_{G2} + a_4 (s_1 c_2 + c_1 s_2) + a_5 c_2 c_1 + a_6 (s_1 s_2 - c_1 c_2)) - M_{tot} g\]

\[\Sigma F_z = M_{tot} \dot{z}_{G3} = 0\]

The constant terms can be consolidated further, so that the equations can be written in a simpler form:

\[\Sigma F_x : M_{tot} \dot{x}_{G1} - B_1 x_{G1} - B_2 (c_1 c_2 - s_1 s_2) + B_3 s_2 c_1 - B_4 (c_1 s_2 + s_2 c_1) = 0.\]

\[\Sigma F_y : M_{tot} \dot{y}_{G2} - B_1 x_{G2} - B_2 (s_1 c_2 + c_1 s_2) - B_3 c_2 c_1 - B_4 (s_1 s_2 - c_1 c_2) + M_{tot} g = 0.\]

\[\Sigma F_z : M_{tot} \dot{z}_{G3} = 0.\]

During the flight phase, the forces at the hands are zero and
equation (16) reduces to:

\[ \sum F_x: M_{\text{tot}} \ddot{x}_{c1} = 0. \]
\[ \sum F_y: M_{\text{tot}} \ddot{x}_{c2} + M_{\text{tot}} g = 0. \quad (17) \]
\[ \sum F_z: M_{\text{tot}} \ddot{x}_{c3} = 0. \]
APPENDIX I

SAMPLE RUN OF SIMULATION PROGRAM

Note: BOLD type indicates the user is responding to a computer prompt or entering information.

C:\FORTRAN\SOURCE\SIMULATE>SIMULATE

C:\FORTRAN\SOURCE\SIMULATE>DATA.EXE

Welcome to the RELEASE MOVE simulation program. This program can be run using default values OR the input data can be customized to perform a number of experiments.

Do you want to run the default program (enter 1) or do you want to run an experiment (enter 2)?

2

The default directory for simulation data files is C:\FORTRAN\DATA\

You may store the data files for your experiment in the default directory or choose another directory.

Do you want to use the default directory (enter 1) or do you want to use another directory (enter 2)?

1

Enter an 8 character identifier for the data files. Enter the name enclose in single quotes 'INCHIPR'.

Which file do you want to use as the reference file?

1= 3 degrees of freedom, move not symmetric
2= Move symmetric using right side data, 1 dof in trunk
3= Move symmetric using left side data, 1 dof in trunk

2

There are several experimental variables that you may alter in this simulation. You may make changes in one or more of the following categories:

> Change the anthropometry of the gymnast
> Change the speed of the giant swing
> Change the constant of proportionality between the forces
and moments at the hand
> Eliminate some joint rotations
> Increase or decrease the range of motion of a joint
> Redistribute the force between the two hands
or
> Change the release point

Enter 1 to make changes, 0 to run as-is
0 is only valid with a name that has already been run

1

The following are the anthropometric data for the default gymnast:

1=Head circ: 58.2  2=Chest circ: 97.7
3=Axillary circ: 31.7  4=Elbow circ: 27.0
5=Forearm circ: 27.8  6=Wrist circ: 18.3
7=Thigh circ: 49.9  8=Knee circ: 35.1
9=Calf circ: 34.6  10=Forearm len: 26.3
11=Wrist breadth: 6.5  12=Hand breadth: 9.2
13=Stature: 174.5  14=Shoulder ht: 136.8
15=Substernal ht: 128.2  16=Troch. ht: 95.2
17= Sitting ht: 86.6  18=Waist ht: 111.7
19=Ankle circ: 26.3  20=Iliac fat: 6.0E-1
21=Chest br: 30.4  22=Chest depth: 26.0
23=Waist depth: 20.5  24=Hip breadth: 30.4
25=Buttcock depth: 24.5  26=Upper arm len: 31.1
27=Tibiale ht: 50.2  28=Foot len: 25.5
29=Weight: 70.3  30=Sphyrion ht: 7.6

All meas. in cm, except wt in kg

Enter the # of the parameter you want to change

0=go on to next menu

0

The average angular speed of the default giant swing is approximately 210 degrees per second (3.665 rad/sec)

Do you want to change the speed of the move?
1=Yes, 2=No

2

The default value of the constant relating the hand forces and moments in .03 Newton-meters per Newton
Do you want to change this value?
1=Yes 2=No
1

Enter new value
0

In the default model, motion is allowed at the following joints:

1=neck, 2=right shoulder, 3=left shoulder, 4=right elbow, 5=left elbow, 6=upper back, 7=lower back, 8=right hip, 9=left hip, 10=right knee, 11=left knee

Do you want to disallow motion at any of these joints?
1=Yes, 2=No
1

How many joints do you want fixed?
1

Enter joint number you want fixed
1

Would you like to modify the range of motion of one or more joints?
1=Yes, 2=No
1

The range of motion of the joints are:
Neck
1-Flexion/Extension: 0
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of the rotation (0=no changes)
0

Right shoulder
1-Flexion/Extension: 6.25
2-Lateral flexion: 4.95
3-Rotation: 6.68

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Left shoulder
1-Flexion/Extension: 6.25
2-Lateral flexion: 4.95
3-Rotation: 6.68

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Right elbow
1-Flexion/Extension: .92375
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Left elbow
1-Flexion/Extension: .92375
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Upper back
1-Flexion/Extension: .96375
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Lower back
1-Flexion/Extension: 1.07875
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of
the rotation (0=no changes)

0
Right hip
1-Flexion/Extension: 1.87625
2-Lateral flexion: 1.145
3-Rotation: .96625

If you want to change one of these ranges, enter the number of the rotation (0=no changes)

1

New range:
2.5

Left hip
1-Flexion/Extension: 1.87625
2-Lateral flexion: 1.145
3-Rotation: .96625

If you want to change one of these ranges, enter the number of the rotation (0=no changes)

1

New range:
2.5

Right knee
1-Flexion/Extension: .7925
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of the rotation (0=no changes)

0

Left knee
1-Flexion/Extension: .7925
2-Lateral flexion: 0
3-Rotation: 0

If you want to change one of these ranges, enter the number of the rotation (0=no changes)

0

In the default giant swing, the release occurs at a point 61% of the way through the move.
This will be at frame #75 in your simulation. You may change this frame number if you wish. 1=change it 2=keep same release

2

In the default simulation, the hand forces are divided equally between the hands.

Do you want to change it for this simulation?
1=Yes, 2=No

2

WRITING DATA TO FILES....
Interpolating angles for joint number 1
Interpolating angles for joint number 2
Interpolating angles for joint number 3
Interpolating angles for joint number 4
Interpolating angles for joint number 5
Interpolating angles for joint number 6
Interpolating angles for joint number 7
Interpolating angles for joint number 8
Interpolating angles for joint number 9
Interpolating angles for joint number 10
Interpolating angles for joint number 11

C:\FORTRAN\SOURCE\SIMULATE>SIMULATE.EXE
C:\FORTRAN\DATA\INCHIPR
Number of frames 119

Interpolating cg vectors, please wait....

(Computer animates the giant swing phase)

RELEASE FRAME
Please wait...computing kinematics and kinetics for giant swing phase
......

At the time of release

Angular momentum in Segment 1 coordinates (units are kg-cm2/sec2):
-.95138
.025174
-467053.1
Velocity x: 212.05  y: 317.86

Expected recatch frame: 104

Angular velocity: 0 0 -7.244476

Position: -87.6 84.469 0
Orientation: -6.011459 0 0

DO YOU WANT TO ALLOW THE REFERENCE SEGMENT TO ROTATE ABOUT ITS LONG AXIS AND ITS AP AXIS DURING FLIGHT?
1=YES, 2=NO

2

ANGULAR MOMENTUM IN GLOBAL COORDINATES:
X (horizontal axis perpendicular to bar): 0
Y (vertical axis): 0
Z (horizontal axis parallel to bar): -467053.1

Do you want to change these values?
1=yes, 2=no

1

Do you want to change the momentum around the X axis?
1=yes, 2=no

2

Do you want to change the momentum around the Y axis?
1=yes, 2=no

2

Do you want to change the momentum around the Z axis?
1=yes, 2=no

1

Enter the new angular momentum around the Z axis:
500000

CHANGING THE ANGULAR MOMENTUM CHANGED THE LINEAR VELOCITY TO:
209.38  320.629

ENTER 1 TO CONTINUE, ENTER 2 TO CHANGE THESE VALUES

2

ENTER VX

270

ENTER VY

204
Do you want to force arms to be symmetric during flight? 
1-yes, 2-no
2

(computer animates flight phase until frame 104, expected recatch frame)

****HANDS ARE NOT NEAR THE BAR!!! FLIGHT CONTINUES***
105
Enter 1 to continue
1

****HANDS ARE NOT NEAR THE BAR!!! FLIGHT CONTINUES***
106
Enter 1 to continue
1

RECATCH!!!!
Recatch frame 107
Enter 1 to continue
1

(computer animates recatch phase)

C:\FORTRAN\SOURCE\SIMULATE>POSTP.EXE

C:\FORTRAN\DATA\INCHIPR.***

DO YOU WANT TO VIEW THE ANIMATION AGAIN? 
1=YES, 2=NO
2

CALLING REORDERING PROGRAM

DO YOU WANT TO VIEW PLOTS OF THE FORCES AND TORQUES 
1=YES, 2=NO
2

DATA FROM THIS SIMULATION ARE STORED IN DOS TEXT FILES

THE FILE FOR THE KINEMATIC DATA IS:
C:\FORTRAN\DATA\INCHIPR.K1

THE FILE FOR THE KINETIC DATA IS:
C:\FORTRAN\DATA\INCHIPR.K2

THESE FILES CAN BE VIEWED DIRECTLY OR IMPORTED INTO A SPREADSHEET FILE FOR FURTHER ANALYSIS

PRESS RETURN TO CONTINUE
STOP-PROGRAM TERMINATED