

MATHEMATICAL MODELLING AND COMPUTER
SIMULATION OF HUMAN LOCOMOTION

By

THOMAS GEORGE PAL, B.Math., M.Eng.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Doctor of Philosophy

McMaster University

September 1978

MATHEMATICAL MODELLING AND COMPUTER
SIMULATION OF HUMAN LOCOMOTION

DOCTOR OF PHILOSOPHY (1978)
(Mechanical Engineering)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: Mathematical Modelling and Computer Simulation
of Human Locomotion

AUTHOR: Thomas George Pal, B. Math.
(University of Waterloo)
M. Eng.
(McMaster University)

SUPERVISOR: Professor J.N. Siddall

NUMBER OF PAGES: ix, 216

ABSTRACT

This study is a theoretical investigation of the process of human locomotion. It is restricted to the lower limbs and includes the major muscles of the legs.

A mathematical model is presented incorporating some of the findings of earlier investigators of locomotion dynamics and of muscular control. The underlying hypothesis is that locomotion is an optimal control process governed by a minimum energy condition. Pontryagin's Maximum Principle is used to implement this optimality criterion. The model is then programmed for evaluation on a high speed digital computer.

The results of the computer simulation are presented along with experimental verification of the findings. The close agreement between the two suggests that the model is an analytical tool that may be used as a foundation for programs of Functional Electro-Stimulation as aids for the physically handicapped.

Included are further suggestions regarding the application of the model to other biomedical problems as well as recommendations for extensions of the work to broaden the scope of utility.

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to his supervisor, Professor J.N. Siddall, for his continuous encouragement and invaluable advice during the course of this work.

The financial support of the National Research Council of Canada is greatly appreciated.

Finally, the author shall forever be indebted to his family, friends and colleagues without whom this would have been infinitely more difficult.

TABLE OF CONTENTS

	PAGE
LIST OF FIGURES	vii
LIST OF TABLES	ix
CHAPTER I INTRODUCTION	1
1.1 Introductory Comments	1
1.2 Overview of Previous Work	1
1.3 Method of Approach	4
CHAPTER II DÉVELOPMENT OF MATHÉMATICAL MODEL	7
CHAPTER III DÉVELOPMENT OF DYNAMICAL MODEL EXPLICIT FORM	30
3.1 Introduction	30
3.2 Mathematical Development	31
3.3 Constraint Equations	41
3.4 Ankle Moments and Ground Reactions	46
CHAPTER IV DÉVELOPMENT OF MUSCULO-MECHANICAL MODEL EXPLICIT FORM	51
4.1 Mathematical Development	51
4.2 Constraints on M. Gastrocnemius	62
4.3 Further Simplification of Musculo- mechanical Equations	64

TABLE OF CONTENTS (cont'd)

	PAGE
CHAPTER V ENERGETICS OF MUSCLES	65
5.1 Introduction	65
5.2 Derivation of Muscular Energy Expenditure	66
CHAPTER VI COMPUTER SIMULATION	73
6.1 Introduction	73
6.2 Pontryagin's Maximum Principle	73
6.3 Experimentally Determined Parameters and Initial Conditions	79
CHAPTER VII COMPUTATION RESULTS	98
7.1 Introduction	98
7.2 Results	99
7.3 Comparison of Muscle Controls with the Observable	121
CHAPTER VIII CONCLUDING DISCUSSION	132
8.1 Introduction	132
8.2 Summary	132
8.3 Suggestions for Further Work	136
REFERENCES	139
APPENDIX A SUMMARY OF EQUATIONS OF MOTION AND ENERGY	144
APPENDIX B PROGRAM LISTING	150
APPENDIX C LISTING OF OPTISEP SUBROUTINES	186
APPENDIX D LISTING OF GASP IV SUBROUTINES	197
APPENDIX E INPUT ECHO CHECK	208
APPENDIX F TYPICAL OUTPUT	215

LIST OF FIGURES

	PAGE
FIGURE 2.1 Diagram of model showing significant variables	9
FIGURE 2.2 Schematic representation of the significant muscle groups acting on the leg	14
FIGURE 3.1 Ankle variables	42
FIGURE 3.2 Free-body diagram of the foot	47
FIGURE 4.1 Diagrammatic representation of skeletal muscle components	52
FIGURE 6.1 Flow chart of numerical procedure	76
FIGURE 6.2 Hip and ankle positions	84
FIGURE 6.3 Forces acting on foot and centre of pressure	86
FIGURE 7.1 "Stroboscopic" view of stance and deploy phase of walking cycle	100
FIGURE 7.2 Experimental results for comparison with trajectories in Figure 7.1	101
FIGURE 7.3 Horizontal positions of hip, knee and ankle	103
FIGURE 7.4 Vertical positions of hip, knee and ankle	104
FIGURE 7.5 Experimentally determined horizontal and vertical positions of the hip, knee and ankle	105
FIGURE 7.6 Angular variables	106
FIGURE 7.7 Angular velocities	107

LIST OF FIGURES (cont'd)

	PAGE
FIGURE 7.8 Angular accelerations	108
FIGURE 7.9 Joint torques	109
FIGURE 7.10 Normalized force in M. iliopsoas	110
FIGURE 7.11 Normalized force in M. vastus	111
FIGURE 7.12 Normalized force in M. rectus femoris	112
FIGURE 7.13 Normalized force in hamstring muscle	113
FIGURE 7.14 Normalized force in M. gastrocnemius	114
FIGURE 7.15 Normalized stimulation frequency and motor unit recruitment in M. iliopsoas	116
FIGURE 7.16 Normalized stimulation frequency and motor unit recruitment in M. vastus	117
FIGURE 7.17 Normalized stimulation frequency and motor unit recruitment in M. rectus femoris	118
FIGURE 7.18 Normalized stimulation frequency and motor unit recruitment in hamstring muscle	119
FIGURE 7.19 Normalized stimulation frequency and motor unit recruitment in M. gastrocnemius	120
FIGURE 7.20 $(\phi\theta)^{1/2}$ for M. iliopsoas	124
FIGURE 7.21 $(\phi\theta)^{1/2}$ for M. vastus	125
FIGURE 7.22 $(\phi\theta)^{1/2}$ for rectus femoris	126
FIGURE 7.23 $(\phi\theta)^{1/2}$ for hamstring muscle	127
FIGURE 7.24 $(\phi\theta)^{1/2}$ for M. gastrocnemius	128
FIGURE 7.25 Typical EMG envelopes	129
FIGURE 7.26 Evidence of typical scatter in EMG records	130

LIST OF TABLES

TABLE 6.1	Muscle Specific Constants (Dynamics)	90
TABLE 6.2	Muscle Specific Constants (Energetics)	92
TABLE 6.3	Initial Values of Simulation Variables	97

CHAPTER I

INTRODUCTION

1.1 Introductory Comments

Human biped locomotion is a subject with which we are all familiar and of which we are probably least conscious. If we wish to go somewhere we simply get up and walk, without even thinking about it. There are, however, those individuals who are not so fortunate; as a result of some trauma, their ability to control the muscles of their legs has been lost.

It is the purpose of this research to advance knowledge in a direction that is aimed at helping these people. Since a great deal of information exists in the literature on the results of experimental programs of research on human gait, it may be considered advantageous to embark upon an investigation that is theoretical in nature, and subsequently compare the theoretically derived results with the experimental ones.

1.2 Overview of Previous Work

The study of human locomotion dates back to the early 1930's. However, by far the largest portion of the work done since then has been experimental in nature. Clearly

this is a necessary step in the understanding of any phenomenon but recent advances in the rehabilitation of paralyzed limbs through functional electro-stimulation [1, 2]* indicate that in-depth study at a fundamental theoretical level is greatly warranted. Experimental study cannot in itself answer the fundamental question, "What laws govern human locomotion?" or, "Why do we walk the way we do?". It is for this reason that a theory that attempts to incorporate the known facts about the human gait would be a major advance in this field.

To begin to understand the fundamentals of human gait we must address ourselves to problems of a more fundamental nature than the analysis of the musculo-skeletal link system which constitutes the lower limbs.

According to Hatze [3], Nubar and Contini [4] were among the first authors to suggest an optimality principle for muscle-driven systems. However, their hypothesis of "muscular effort" as the quantity to be minimized in an individual's motion or posture is not likely to be accurate since this "muscular effort" (defined as the product of a constant, the square of the joint moment and the time interval over which the minimization is to be carried out) is unrelated to any biological performance criterion.

* Numbers in brackets [] denote references.

Beckett and Chang [5] postulated a "minimum energy" hypothesis and studied the kinematics of the swing phase motion but, interestingly, obtained joint moment profiles without invoking any optimization procedure whatsoever.

In 1971 Chow and Jacobson [6] published results of a theoretical analysis on gait studies using optimal programming. This paper deserves special mention as the results are significant indeed. They were the first to apply optimal control to a model of a muscle-driven bio-structure. Using the performance index of mechanical energy (this may be somewhat questionably appropriate) they predicted the time course of the hip and knee angles (in the sagittal plane) during level walking. Unfortunately, they do not show the corresponding experimental curves for comparison, but in general they are seen to be of approximately the correct shape.

Perhaps the most serious criticism of the work relates to the choice of control parameters, being the joint torques. Since these cannot be controlled directly in the human body but rather the stimulation of muscles (i.e. stimulation frequency and motor unit recruitment), using the latter as the control variable(s) in the optimization would permit the possible realization of programs of artificial electro-stimulation.

It is well known, for example, that the muscles on the

affected side of a hemiplegic are very much alive and able to function. The problem in this case is a matter of communication. There is no signal to "turn on" the muscle, or at least is so weak that the muscles are for the most part inactive. It is also commonly acknowledged that the inactive muscles can be made to function by electrically stimulating them either with surface electrodes or indwelling wire electrodes. This is referred to as Functional Electro-Stimulation (FES). In order that this method may be used clinically as an aid for the hemi- or paraplegic it is necessary to ascertain which muscles to stimulate, when and to what extent. Being able to determine how the intact normal muscle behaves during walking is a necessary step for the realization of FES, that is, how the brain switches the muscles "on" and "off".

At this point then it is appropriate to discuss the works of Hatze. Using optimal control theory and a muscular control model Hatze [3] was able to optimize a particular motion (kicking at a target). Although the choice of motions may or may not be of direct interest, it was one which could be easily tested experimentally. The model Hatze used for the link-mechanical system and the very involved myocybernetic control model of skeletal muscle [7] are of major significance. These were shown to be reasonably accurate models as the computer simulation/

optimization were closely reproduceable in the laboratory.

1.3 Method of Approach

In view of the above comments, it appears that a reasonable foundation has been laid for the complete optimization of level walking. The five degree of freedom link-mechanical system of Chow and Jacobson [6] will be used as the starting point of the model. To be combined with this is Hatze's [7] control model of human skeletal muscles. It is this model that will be used to relate the control parameters (i.e., stimulation frequency and motor unit recruitment) to the tension force in each muscle. Then having related the location of the muscles with respect to the link-mechanical system, the control parameters may then be related to the dynamical system. The method of solution of the equations is then carried out via digital simulation and simultaneous numerical optimization.

There are, however, two aspects that require a fresh start. Firstly, since no satisfactory optimality criterion has been implemented to date, a portion of the work will be devoted to the development of such a criterion with contributions from not only the mechanical energy, but also chemical energy and heat output of the musculature. Clearly this is of major importance if a successful solution to the problem is to be found.

Secondly, and possibly most important, the application of optimization procedures must be done in such a way that inordinate amounts of computer time are not required. Hatze, for example, used twenty-one hours of computer time on an IBM 360 computer to solve his time-optimality problem. The ultimate aim of this research is to facilitate programs of electro-stimulation. If use is to be made of feedback systems to accommodate changes in boundary conditions (e.g., uphill walking), it is clear that "real time" solutions are needed. Although at this stage it is a bit much to aim for, reduction of computation time by at least one order of magnitude is required, especially if one considers that the problem to be attacked here is of higher complexity than that of Hatze.

Numerical simulation and optimization are implemented using the GASP IV simulation language and a package of optimization routines (OPTISEP) which were developed at McMaster University, Hamilton, Canada. The entire program was run on a CDC 6400 digital computer. (Note that the IBM 360 computer is approximately 20 times as fast as the CDC 6400.)

CHAPTER II

DEVELOPMENT OF THE MATHEMATICAL MODEL

We will begin by using the analysis of Chow and Jacobson [6] to derive the equations of motion. This derivation is very similar to that of Hatze [3] except that in the latter's case the hip may be considered as the origin of an inertial co-ordinate system since it is fixed in space. For the present case it is not so.

To derive the equations of motion, one may define the Lagrangian L as the difference between the total system kinetic energy, T , and potential energy, V ;
i.e. $L = T - V$. (2.1)

Then if the system co-ordinates are q_i , the equations of motion are given by

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = M_i \quad (i=1, \dots, n) \quad (2.2)$$

where M_i are the generalized forces in the q_i directions of an n degree of freedom system. However; potential energy is dependent only upon position (and not velocity), giving

$$\frac{\partial V}{\partial \dot{q}_i} = 0 \quad (2.3)$$

and hence (2.2) reduces to

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = M_i \quad (i=1, \dots, n). \quad (2.4)$$

In the present case the co-ordinates q_i are the angular variables (viz. $x_1, x_2, \phi, \gamma, \omega$ in the analysis of Chow and Jacobson) and the M_i are the net effective moments about the appropriate joints.

Using notation similar to that in [6] Fig. 2.1 shows the geometrical significance of the following variables:

- x_i - are the angular displacements of the five links.
(Note: as shown all are positive except x_1 which is negative)
- m_i - are the masses of the respective links
- l_i - are the lengths of the respective links measured from joint centres of rotation
- a_i - are the locations of the centres of mass of the respective links measured from the centre of rotation of the joint on that link nearest the hip
- u_i - are the muscle generated moments for the i -th link
- X, Y - are the ankle reaction forces in the x and y directions respectively
- v, h - are the vertical and horizontal displacements of the hip joint from some arbitrary origin
- M_a^2, M_a^4 - are the moments about the ankle

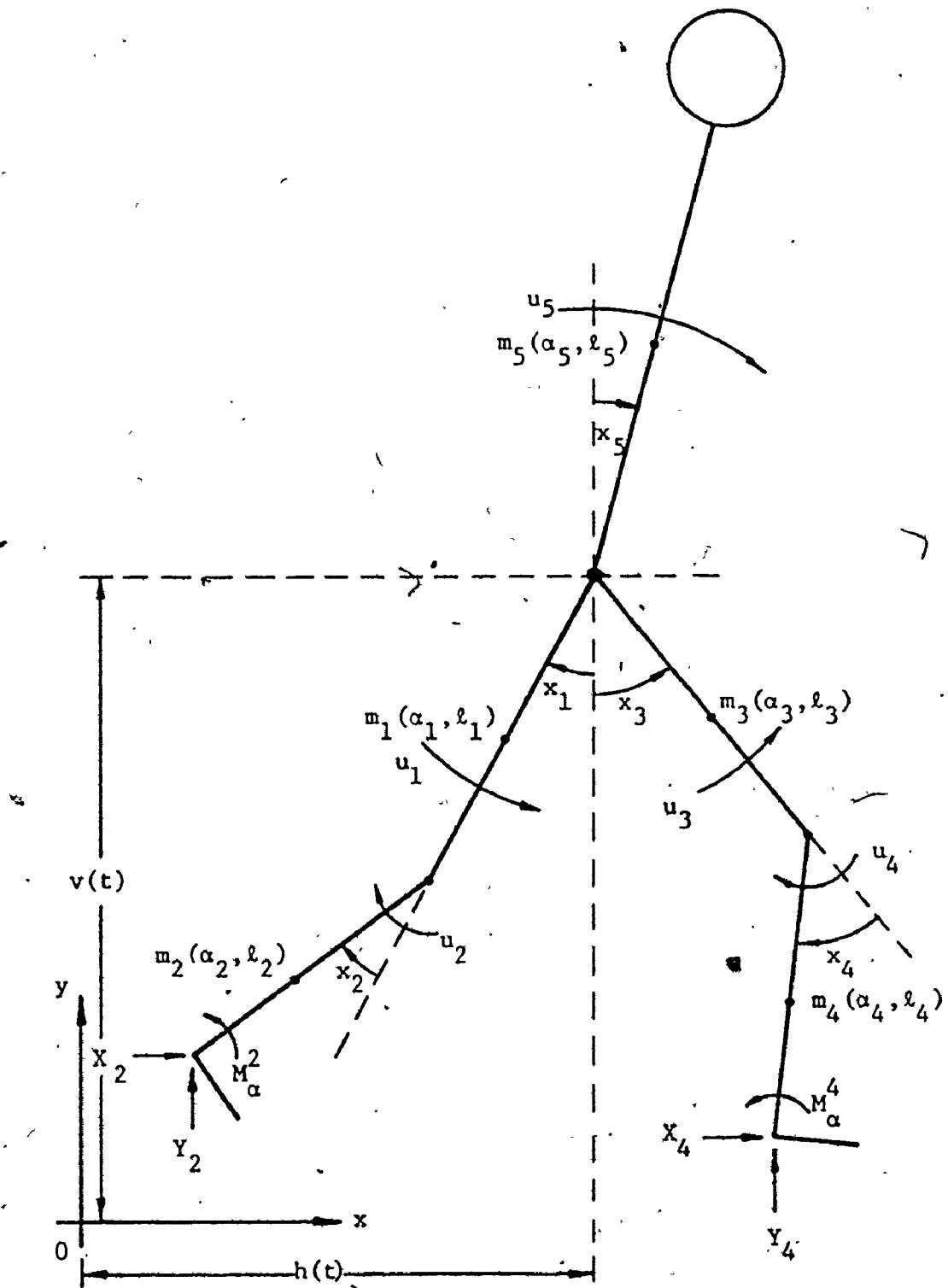


Figure 2.1 Diagram of model showing significant variables.

the center of gravity of the i -th link, one obtains

$$\begin{aligned}\hat{x}_1 &= h + \alpha_1 \sin x_1 & \hat{x}_3 &= h + \alpha_3 \sin x_3 \\ \hat{y}_1 &= v - \alpha_1 \cos x_1 & \hat{y}_3 &= v - \alpha_3 \cos x_3\end{aligned}$$

$$\hat{x}_2 = h + \ell_1 \sin x_1 - \alpha_2 \sin (x_2 - x_1)$$

$$\hat{y}_2 = v - \ell_1 \cos x_1 - \alpha_2 \cos (x_2 - x_1)$$

(2.5)

$$\hat{x}_4 = h + \ell_3 \sin x_3 - \alpha_4 \sin (x_4 - x_3)$$

$$\hat{y}_4 = v - \ell_3 \cos x_3 - \alpha_4 \cos (x_4 - x_3)$$

$$\hat{x}_5 = h + \alpha_5 \sin x_5$$

$$\hat{y}_5 = v + \alpha_5 \cos x_5$$

The velocities of the centres of gravity (cg) can be obtained by differentiating the above with respect to time and the kinetic energy of each link may be found as the sum of the kinetic energy due to translation of the cg and that due to rotation about the cg; that is,

$$T_i = (T_{cg} + T_{rot})_i \quad (2.6)$$

and the total system kinetic energy is given by

$$T = \sum_{i=1}^n T_i \quad (2.7)$$

Using equations (2.5) the total system kinetic energy may be shown to be

$$\begin{aligned}
 T = & \frac{1}{2} [A_0 (\dot{h}^2 + \dot{v}^2) + A_1 \dot{x}_1^2 + A_3 \dot{x}_3^2 + A_5 \dot{x}_5^2] \\
 & + \frac{1}{2} [A_2 (\dot{x}_2 - \dot{x}_1)^2 + A_4 (\dot{x}_4 - \dot{x}_3)^2] \\
 & + C_1 \dot{x}_1 (\dot{h} \cos x_1 + \dot{v} \sin x_1) + C_4 \dot{x}_3 (\dot{h} \cos x_3 + \dot{v} \sin x_3) \\
 & + C_2 (\dot{x}_2 - \dot{x}_1) [-\dot{h} \cos (x_2 - x_1) + \dot{v} \sin (x_2 - x_1)] \\
 & + C_5 (\dot{x}_4 - \dot{x}_3) [-\dot{h} \cos (x_4 - x_3) + \dot{v} \sin (x_4 - x_3)] \\
 & - C_3 \dot{x}_1 (\dot{x}_2 - \dot{x}_1) \cos x_2 - C_6 \dot{x}_3 (\dot{x}_4 - \dot{x}_3) \cos x_4 \\
 & + C_7 \dot{x}_5 (\dot{h} \cos x_5 - \dot{v} \sin x_5)
 \end{aligned} \quad (2.8)$$

where

$$A_0 = \sum_{i=1}^5 m_i$$

$$A_1 = I_1 + m_1 \alpha_1^2 + m_2 \ell_1^2$$

$$A_2 = I_2 + m_2 \alpha_2^2$$

$$A_3 = I_3 + m_3 \alpha_3^2 + m_4 \ell_3^2$$

$$A_4 = I_4 + m_4 \alpha_4^2$$

$$A_5 = I_5 + m_5 \alpha_5^2$$

I_i = moment of inertia of the i -th link about its

centre of gravity

and

$$C_1 = m_1 \alpha_1 + m_2 \ell_1 \quad C_4 = m_3 \alpha_3 + m_4 \ell_3$$

$$C_2 = m_2 \alpha_2 \quad C_5 = m_4 \alpha_4$$

$$C_3 = C_2 \ell_1 \quad C_6 = C_5 \ell_3$$

$$C_7 = m_5 \alpha_5 \quad \text{and} \quad h = \frac{dh}{dt}$$

The total system potential energy is

$$\frac{V}{g} = A_0 v - C_1 \cos x_1 - C_4 \cos x_3 - C_2 \cos (x_2 - x_1) \\ - C_5 \cos (x_4 - x_3) + C_7 \cos x_5 \quad (2.9)$$

It is then equations (2.8) and (2.9) that are substituted into (2.4) to yield the equations of motion. Before doing so it is appropriate to define more explicitly the functions $h(t)$ and $v(t)$ and also to discuss the origins of the functions M_i of equation (2.4).

Firstly, the position of the hip may be defined in terms of the angle variables and the position of the ankle of the foot in contact with the ground: that is,

$$h(t) = x_{\delta+1}^A - \ell_{\delta} \sin x_{\delta} + \ell_{\delta+1} \sin (x_{\delta+1} - x_{\delta}) \quad (2.10)$$

$$v(t) = y_{\delta+1}^A - \ell_{\delta} \cos x_{\delta} + \ell_{\delta+1} \cos (x_{\delta+1} - x_{\delta})$$

where

$$\delta = \begin{cases} 1, & \text{for link 2 in contact with ground} \\ & \dots \\ 3, & \text{for link 4 in contact with ground} \end{cases}$$

and $x_{\delta+1}^A, y_{\delta+1}^A$ denote the horizontal and vertical location of the ankle on link $\delta+1$. It can be seen that when both legs are in contact with the ground, two sets of equations (2.10) hold and must clearly yield the same hip position.

It is equations (2.10) that are used to decouple the equations of motion by imposing certain conditions on the functions $h(t)$ and $v(t)$. This will be demonstrated in the due course of the development.

Returning now to the functions M_i of equations (2.4), it is necessary to introduce the musculo-mechanical model of the lower limb. Hatze [3] had considerable success in his time-optimality problem using the model of the lower limb muscles depicted in Figure 2.2. Justification for the inclusion of these muscles and only these muscles may be found in references [3], [8] and [9]. Group (2) consists of the vastus lateralis, vastus medialis and vastus intermedius and together with group (3) (rectus femoris), they comprise the commonly referred to quadriceps femoris. The Hamstring group is composed of the following muscles: semi-membranosus, semitendinosus and biceps femoris (long head).

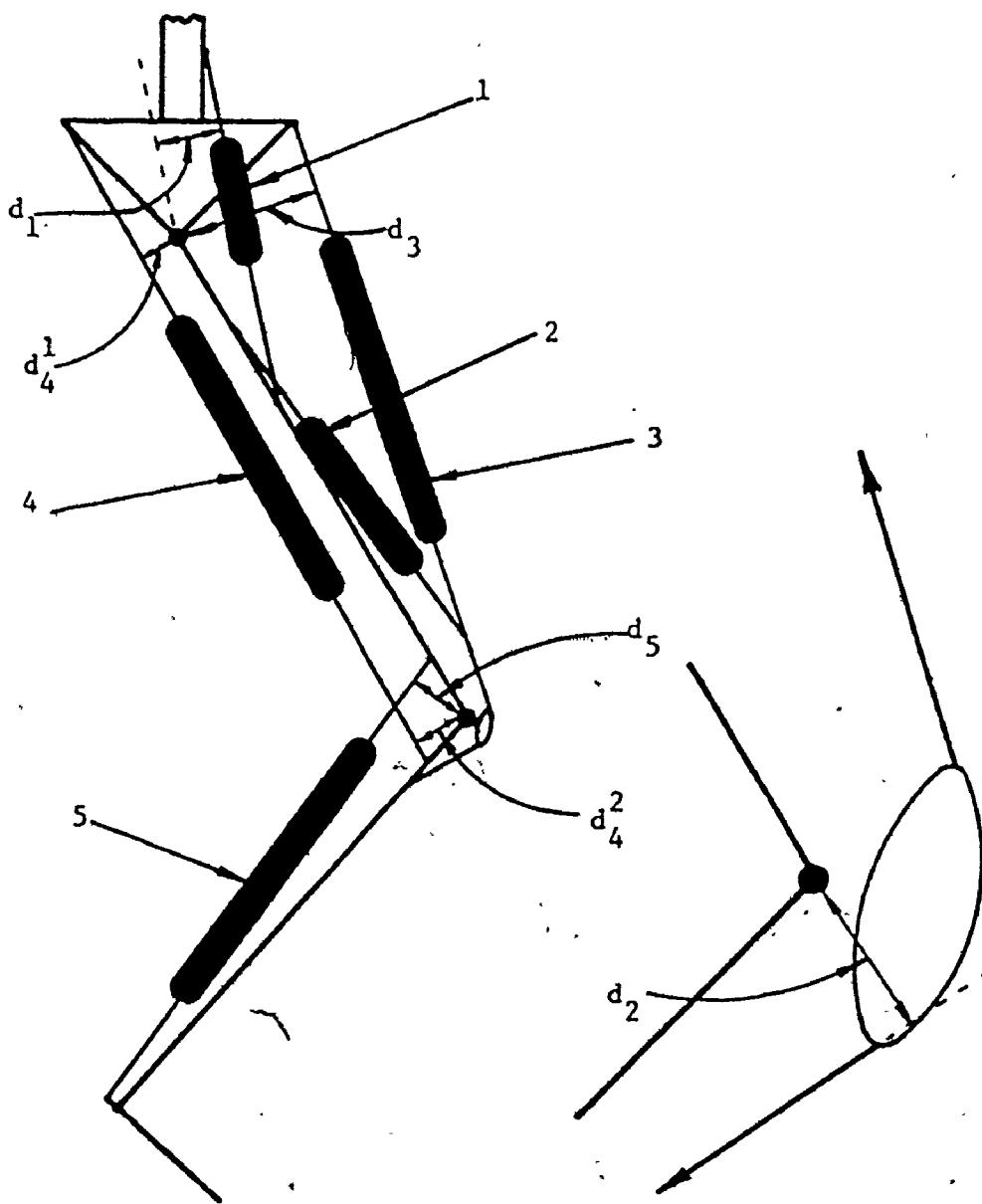


Figure 2.2 Schematic representation of the significant muscle groups acting on the leg: 1) M. iliopsoas, 2) M. vastus, 3) M. rectus femoris, 4) Hamstring group, 5) M. gastrocnemius.

gastrocnemius are the popliteus and short head of the biceps femoris.

Having established the significant muscle groups which act on the leg, the expression^s for the muscular torques generated by these groups may be written:

$$\begin{aligned} u_1 &= F_1 d_1 + F_3 d_3 - F_4 d_4^1 \\ -u_2 &= F_2 d_2 + F_3 d_2 - F_4 d_4^2 - F_5 d_5 \\ u_3 &= F_6 d_6 + F_8 d_8 - F_9 d_9^3 \\ -u_4 &= F_7 d_7 + F_8 d_7 - F_9 d_9^4 - F_{10} d_{10} \\ u_5 &= F_1 d_1 + F_6 d_6 + F_3 d_3 + F_8 d_8 - F_4 d_4^1 - F_9 d_9^3 \end{aligned} \quad (2.11)$$

where F_i ($i = 1, 2, \dots, 10$) are the tension forces generated by the i -th muscle group and d_j ($j = 1, 2, \dots, 10$), shown in Fig. 2.2, are the normal distances of the respective instantaneous joint centers to the lines of action of the corresponding muscle force vectors. (Note: Since there is similar musculature on the two legs the following notation has been adopted: $F_1 + F_5$ and $d_1 + d_5$ refer to the leg consisting of links 1 and 2; $F_6 + F_{10}$ and $d_6 + d_{10}$ refer to corresponding functions of the leg comprised of links 3 and 4; i.e., F_6 refers to M. iliopsoas etc.).

It must be noted that the functions u_j and M_j are not the same. The u_j account only for the active portion of the joint forces but it is clear from the

absence of muscular activity, there is a resistance to joint flexion (or extension). These resistances may be referred to as passive joint torques and will be denoted by the functions u_j^P . These are in general non-linear functions of the angular variables and must be experimentally determined for each individual subject.

From Fig. 1 then it can be seen that

$$\begin{aligned} M_1 = u_1 + u_1^P + M_\alpha^2 + Y_2 (\ell_1 \sin x_1 - \ell_2 \sin (x_2 - x_1)) \\ + X_2 (\ell_1 \cos x_1 + \ell_2 \cos (x_2 - x_1)) \end{aligned}$$

$$M_2 = u_2 + u_2^P - M_\alpha^2 + Y_2 \ell_2 \sin (x_2 - x_1) - X_2 \ell_2 \cos (x_2 - x_1)$$
(2.12)

$$\begin{aligned} M_3 = u_3 + u_3^P + M_\alpha^4 + Y_4 (\ell_3 \sin x_3 - \ell_4 \sin (x_4 - x_3)) \\ + X_4 (\ell_3 \cos x_3 + \ell_4 \cos (x_4 - x_3)) \end{aligned}$$

$$M_4 = u_4 + u_4^P - M_\alpha^4 + Y_4 \ell_4 \sin (x_4 - x_3) - X_4 \ell_4 \cos (x_4 - x_3)$$

$$M_5 = u_5 + u_5^P$$

The reaction forces and ankle moments will be dealt with in greater detail later.

It is now appropriate to discuss the control aspects of the active muscular torques; that is, how the functions $F_i (i = 1, \dots, 10)$ (cf. eq'n (2.11)), which are in general

dependent upon the angular variables x_i and their derivatives \dot{x}_i , are controlled. Only a very cursory discussion of the control model of skeletal muscle is presented at this time with further treatment at the appropriate point of the development of the present model.

The general nature of skeletal muscles (striated muscles) is such that each of the muscle fibres may be considered as independent force generators which are controlled by electrical pulses from indwelling nerve endings. The resultant force between the ends of one of these fibres is a function of the length λ of the fibre, its velocity of shortening (or lengthening) $\dot{\lambda}$ and stimulation frequency v . The functions λ and $\dot{\lambda}$ are in turn functions of the angular variables x_i and \dot{x}_i ; that is

$$\begin{aligned}\lambda &= \lambda(x_i) \\ \dot{\lambda} &= \dot{\lambda}(x_i, \dot{x}_i).\end{aligned}\tag{2.13}$$

If we define v_{opt} as the frequency at which the muscle fibre develops maximum isometric force, and the variable ϕ' as

$$\phi' = \frac{v}{v_{opt}}\tag{2.14}$$

the force f^j produced by the j -th such fibre is given by

$$f^j = f^j(\lambda(x_i), \dot{\lambda}(x_i, \dot{x}_i), \phi').\tag{2.15}$$

Assuming that all the fibres in a muscle are approximately parallel and of the same length, it follows that the total force F produced by a muscle containing $\bar{\rho}$ fibres is given by

$$F = \sum_{j=1}^{\bar{\rho}} f^j(\lambda(x_i), \dot{\lambda}(x_i, \dot{x}_i), \phi_j^i) \quad (2.16)$$

where ϕ_j^i refers to the frequency of stimulation of the j -th fibre. With the number ρ of the total number $\bar{\rho}$ fibres stimulated, equation (2.16) becomes

$$\begin{aligned} F &= \sum_{j=1}^{\rho} f^j(\lambda(x_i), \dot{\lambda}(x_i, \dot{x}_i), \phi_j^i) \\ &\quad + (\bar{\rho} - \rho) f_0(\lambda(x_i), \dot{\lambda}(x_i, \dot{x}_i)) \end{aligned} \quad (2.17)$$

where the function f_0 indicates the force across an unstimulated fibre (which is in general not zero). We may now introduce the second control variable θ defined as

$$\theta = \frac{\rho}{\bar{\rho}} \quad (2.18)$$

which expresses the proportion of fibres active at any given moment. In reality, only groups of fibres assembled to motor units can be stimulated selectively by the nervous system. However, the number of motor units is usually very large (of the order of several hundreds) so that θ may be regarded as a continuous variable.

It is also convenient to define an "average" stimulation frequency as

$$\phi = \frac{1}{\rho} \sum_{j=1}^{\rho} \phi_j \quad (2.19)$$

so that for the k-th muscle in the musculo-mechanical model the implicit relation

$$F_k = F_k(x_i, \dot{x}_i; \phi_k, \theta_k) \quad (k=1, \dots, 10 \ i=1, \dots, n) \quad (2.20)$$

holds. It is these (as yet undetermined) functions which must be substituted into equations (2.11), which in turn couple with the equations of motion (2.4) via relations (2.12).

For the time being, suffice it to write these equations implicitly. First the notation will be modified to reduce the system of equations to first order (non-linear) differential equations (of course, at the expense of increasing the number of simultaneous equations).

Implicitly, the equations of motion may be written as

$$G_j(x_i, \dot{x}_i, \ddot{x}_i; h(t), v(t)) = M_j \quad (i, j=1, 2, \dots, n) \quad (2.21)$$

where the G_j 's are as yet undetermined functions and n is the number of generalized co-ordinates (in the present case $n = 5$). Letting

$$x_{n+i} = \dot{x}_i \quad (i=1, \dots, n) \quad (2.22)$$

and solving equations (2.21) for \ddot{x}_i (assuming that this is in fact possible) one is left with a system of $2n$ simultaneous differential equations of the form

$$\dot{x}_i = g^i(x_j, \phi_k, \theta_k) \quad (i, j=1, \dots, 10; k=1, \dots, 10). \quad (2.23)$$

(Note that the functions h and v have been omitted for the sake of brevity and that the functions ϕ_k and θ_k have been included since the functions M_j of (2.21) are dependent upon these control variables as per equations (2.11), (2.12) and (2.20)).

The concepts of the calculus of variations together with Pontryagin's maximum principle will be utilized to demonstrate how a "forward marching" optimization process can be used to minimize the energy expended during the walking cycle.

Let us denote the instantaneous power consumption of the present bio-system by the function $\xi(x_i, \phi_k, \theta_k)$ ($i, k = 1, \dots, 10$) so that the functional

$$J(\phi_k, \theta_k) = \int_{t_0}^{t_1=t_0+\tau} \xi(x_i, \phi_k, \theta_k) dt \quad (2.24)$$

represents the total energy consumed during the time interval $[t_0, t_1 = t_0 + \tau]$, τ being the period of one double step. The functional $J(\phi_k, \theta_k)$ is to be minimized with respect to the functions ϕ_k and θ_k subject to the dynamic

equations (2.23) and the relevant boundary conditions and constraint relations.

Simplifying the notation somewhat by letting the vector functions x and γ be defined as

$$x = (x_1, x_2, \dots, x_{10})$$

and

(2.25)

$$\gamma = (\phi_1, \dots, \phi_{10}, \theta_1, \dots, \theta_{10}),$$

the dynamical equations (2.23) may be written as the following system of differential equations representing the combination of the link-mechanical system and the musculo-mechanical system:

$$\frac{dx_i}{dt} = g^i(x, \gamma) \quad (i=1, \dots, 10). \quad (2.26)$$

Define the aggregate

$$\Gamma = \{\gamma(t), t_0, t_1, x^0\} \quad (2.27)$$

consisting of a control function $\gamma(t)$, an interval $[t_0, t_1]$ and an initial value $x^0 = x(t_0)$, as a "control process". Thus to every control process there corresponds a trajectory, i.e., a solution of (2.23).

Next, let $g^0(x, \gamma)$ be a function which is defined, together with its partial derivatives $\frac{\partial g^0}{\partial x_i}$ ($i = 1, \dots, 10$) for all $x \in X$ [X is the 10-dimensional phase-space of the system] and $\gamma \in \Omega$ [Ω is the 20-dimensional control region].

Then to every control process Γ , there corresponds the number

$$J(\Gamma) = \int_{t_0}^{t_1} g^o(x, \gamma) dt \quad (2.28)$$

Here $g^o(x, \gamma)$ is simply the same as the function $\xi(x_i, \phi_k, \theta_k)$ of equation (2.24).

To find necessary conditions for a given control process and the corresponding trajectory to be optimal, we use the system of equations (2.26) with the extra condition

$$\frac{d}{dt} x_o = g^o(x, \gamma) \quad (2.29)$$

where $g^o(x, \gamma)$ is the integrand of the functional (2.28), which is to be minimized. At the same time the initial conditions

$$x_i(t_0) = x_i^o \quad (i=1, \dots, 10)$$

may be supplemented by the extra condition $x_o(t_0) = 0$ and for convenience introduce the 11-dimensional vector function

$$\bar{x}(t) = (x_o(t), x(t))$$

$$= (x_o(t), x_1(t), \dots, x_{10}(t))$$

It is clear that if Γ is an admissible control process and if $\bar{x} = \bar{x}(t)$ is the solution of the system

$$\frac{d}{dt} \bar{x}_i = g^i(x, \gamma) \quad (i=0, 1, \dots, 10) \quad (2.30)$$

corresponding to Γ and the initial conditions

$$\bar{x}(t_0) = \bar{x}^0 \quad (2.31)$$

then

$$J(\Gamma) = \int_{t_0}^{t_1} g^0(x, \gamma) dt = x_0(t_1). \quad (2.32)$$

This optimal control problem may now be stated as follows: Find the admissible control process Γ for which the solution $\bar{x}(t)$ of the system (2.30), satisfying the initial conditions (2.31) has the smallest possible value of $x_0(t_1)$.

In addition to the variables x_0, x_1, \dots, x_{10} , the new variables $\psi_0, \psi_1, \dots, \psi_{10}$ are introduced, which satisfy the following system of differential equations (known as the conjugate of the system (2.30)):

$$\frac{d\psi_i}{dt} = - \sum_{\alpha=0}^{10} \frac{\partial g^\alpha(x, \gamma)}{\partial x_i} \psi_\alpha \quad (i=0, 1, \dots, 10). \quad (2.33)$$

Let

$$\Psi(t) = (\psi_0(t), \psi_1(t), \dots, \psi_{10}(t))$$

and consider the following function of the variables

$$x_1, \dots, x_{10}, \psi_0, \psi_1, \dots, \psi_{10}, \gamma_1, \dots, \gamma_{20};$$

$$\Pi(\Psi, x, \gamma) = \sum_{\alpha=0}^{10} \psi_\alpha q^\alpha(x, \gamma) \quad (2.34)$$

In terms of Π , the equations (2.30) and (2.33) may be written in the form

$$\begin{aligned} \frac{dx_i}{dt} &= \frac{\partial \Pi}{\partial \psi_i} \\ &\quad (i=0, 1, \dots, 10) \\ \frac{d\psi_i}{dt} &= -\frac{\partial \Pi}{\partial x_i}. \end{aligned} \quad (2.35)$$

It must be noted at this point that this is as yet not a closed system for there are 22 equations in 42 unknowns. It becomes closed only after the functions γ become specified. In order to do this the following results (Pontryagin's maximum principle) will be stated without proof. However, the interested reader is referred to Pontryagin [11, 12] for proof.

The maximum principle states that if

$$\Gamma = \{\gamma(t), t_0, t_1, x(t_0)\}$$

is an admissible control process and $\bar{x}(t)$ is the corresponding integral curve of the system (2.30) passing

through the point \bar{x}^0 for $t = t_0$ and satisfies the conditions $\bar{x}(t_1) = \bar{x}^1$ for $t = t_1$, then if the control process Γ is optimal, there exists a continuous vector function $\Psi(t) = (\psi_0(t), \psi_1(t), \dots, \psi_{10}(t))$ such that:

- 1) the function $\Psi(t)$ satisfies the system (2.33) for $x = x(t)$ and $\gamma = \gamma(t)$;
- 2) for all t in $[t_0, t_1]$, the function (2.34) achieves its maximum for $\gamma = \gamma(t)$ i.e., the equations (2.35) may be written in closed form as

$$\frac{d}{dt} x_i = \frac{\partial H}{\partial \psi_i} \quad (i=0, 1, \dots, 10) \quad (2.36)$$

$$\frac{d}{dt} \psi_i = - \frac{\partial H}{\partial x_i}$$

where the function H is defined as:

$$H(\Psi, x) = \sup_{\gamma \in \Omega} \Pi(\Psi, x, \gamma); \quad (2.37)$$

- 3) the relations

$$\psi_0(t_1) \leq 0, \quad H(\Psi(t_1), x(t_1)) = 0 \quad (2.38)$$

hold at time $t = t_1$. (Actually, if $\Psi(t)$, $\bar{x}(t)$ and $\gamma(t)$ satisfy the systems (2.30) and (2.33) and the conditions (2.37), the functions $\psi_0(t)$ and $H(\Psi(t), x(t))$ turn out to be constants, and hence in (2.38) t_1 can be replaced by any t in $[t_0, t_1]$.)

A few comments will now be presented to indicate how the maximum principle can be used to render a solution to the present problem.

Equations (2.8), (2.9) and (2.12) may be substituted into equation (2.4) and after the change of variables (2.22) the resulting equations of motion (together with the controlling functions) may be written in the form of equation (2.26); i.e.

$$\frac{d}{dt} x_i = g^i(x, \gamma) \quad (i=1, \dots, 10). \quad (2.26)$$

We supplement these with the additional equation

$$\frac{d}{dt} x_0 = g^0(x, \gamma) \equiv \xi(x, \gamma). \quad (2.29)$$

Because of the nature of the problem it is easily verifiable that the functions g_i ($i = 0, 1, \dots, 10$) have continuous partial derivatives

$$\frac{\partial g^a(x, \gamma)}{\partial x_i}$$

so that the new variables $\psi_0, \psi_1, \dots, \psi_{10}$ may be introduced to satisfy

$$\frac{d}{dt} \psi_i = - \sum_{\alpha=0}^{10} \psi_\alpha \frac{\partial g^a(x, \gamma)}{\partial x_i} \quad (i=0, 1, \dots, 10). \quad (2.33)$$

If now the function Π is defined as

$$\Pi(\Psi, x, \gamma) = \sum_{\alpha=0}^{10} \psi_\alpha g^\alpha(x, \gamma) \quad (2.34)$$

equations (2.26), (2.29) and (2.33) may be written in the form of the following system.

$$\frac{dx_i}{dt} = \frac{\partial \Pi}{\partial \psi_i} \quad (i=0, 1, \dots, 10) \quad (2.35)$$

$$\frac{d\psi_i}{dt} = - \frac{\partial \Pi}{\partial x_i}$$

This is then a system of first order ordinary differential equations (since the right hand sides are known functions of the variables x, γ, ψ). The maximum principle states that in order for the system to be an optimal trajectory, the control functions γ must be chosen in such a way that the function (2.34) achieves its maximum value at any time t in $[t_0, t_1]$. In fact we have the result that the function

$$H(\Psi, x) = \sup_{\gamma \in \Omega} \Pi(\Psi, x, \gamma) \quad (2.37)$$

must vanish for all time in $[t_0, t_1]$.

Equations (2.36) contain 22 equations in 42 unknown functions x_i, ψ_i , and γ_i . Relation (2.37) provides another 20 equations although this is not immediately obvious (explanation may be sought from the footnote on page 21 of

[12]). Hence we have a closed system of relations for determining all of the variables.

Furthermore, since equation (2.37) is finite (not differential), and the number of differential equations is 22, the solutions of the system of equations (2.36) and (2.37) in general depend on 22 parameters (the initial conditions). However, one of these parameters is redundant, inasmuch as the functions $\psi_\alpha(t)$ are defined only up to a common multiple (since the function Π in (2.34) is homogeneous with respect to Ψ). In addition one of the parameters is determined by the condition that

$$\sup_{\gamma \in \Omega} \Pi(\Psi(t_0), x(t_0), \gamma)$$

vanishes.

Thus we have 20 parameters, on which the solutions of (2.36) and (2.37) depend. It follows that these 20 parameters must be chosen in such a way that the trajectory $x(t)$ passes through the point \bar{x}^0 at the given time $t = t_0$, and through a point on the "line" $(x_0(t_1), x(t_1))$ in the phase-space which is parallel to the x_0 axis, at some time $t_1 > t_0$ (That is, $x_i(t_0) = x_i^0$ and $x_i(t_1) = x_i^1 = x_i^0$ since we know that the double step is periodic).

It must be noted that the maximum principle provides only for necessary conditions. However, if only one optimal trajectory can be found, and if from physical arguments

(from which the optimal problem arose in the first place) it is "clear" that an optimal trajectory must exist, then one can trust that the just-found trajectory is indeed optimal. However, the mathematical question about the existence of an optimal trajectory is very difficult indeed.

CHAPTER III
DEVELOPMENT OF DYNAMICAL MODEL
EXPLICIT FORM

3.1 Introduction

Having indicated the general procedure to be used in the synthesis of gait, it is now appropriate to begin to derive the explicit relations which heretofore were used implicitly. To do this we begin by partitioning the gait pattern into distinct phases in a manner similar to that of Chow and Jacobson [6].

Characteristic of human gait are the two basic configurations into which the double step may be decomposed. First, there is the configuration in which both legs are in contact with the ground. Just as one leg ends its swing motion and comes into contact with the ground to restrain the falling tendency, the other leg begins its deploy action through rotation of the foot about the ball of the foot; this phase of activity is referred to as restraint/deploy. This mode is followed by the support/swing configuration in which one leg assumes complete support of the body while the other leg, which previously was in deploy, proceeds through its swing motion. The double step then is simply an alternating sequence of these two configurations with the two legs interchanging roles.

3.2 Mathematical Development

The primary purpose in resolving the biped gait into the two modes restraint/deploy and support/swing is to eliminate the possibility of singular (impossible) behaviour, such as both legs in swing motion or both in deploy. (In running, however, the "dual swing" possibility does exist but this is not of concern here). In the case of non-pathological gait, the two mode description also introduces symmetry into the problem in that the activity of one leg is identical (or very nearly so) to that of the other except that there is a 180 degree phase lag between the two legs. Since there is basically no difference between the restraint and support functions of a leg, these two phases will collectively be referred to as "stance".

Having described the phasic activity of the legs, we are now in a position to write the equations of motion (i.e. eq'ns (2.21)) explicitly.

i) For the leg consisting of links 1 and 2 in contact with the ground:

$$\begin{aligned}
 & A_0(\ddot{v} \cdot \dot{v}_{x_1} + \dot{h} \cdot \dot{h}_{x_1}) + A_1 \ddot{x}_1 - A_2(\ddot{x}_2 - \ddot{x}_1) \\
 & + C_1(h \cos x_1 + \dot{v} \sin x_1) + C_1 \ddot{x}_1(h_{x_1} \cos x_1 + \dot{v}_{x_1} \sin x_1) \\
 & + C_1 \dot{x}_1^2(\dot{v}_{x_1} \sin x_1 - \dot{h}_{x_1} \cos x_1) \\
 & - C_2[\dot{v} \sin(x_2 - x_1) - \dot{h} \cos(x_2 - x_1)] \\
 & + C_2(\ddot{x}_2 - \ddot{x}_1)[\dot{v}_{x_1} \sin(x_2 - x_1) - \dot{h}_{x_1} \cos(x_2 - x_1)] \\
 & + C_2(\dot{x}_2 - \dot{x}_1)^2[\dot{v}_{x_1} \cos(x_2 - x_1) + \dot{h}_{x_1} \sin(x_2 - x_1)]
 \end{aligned}$$

$$\begin{aligned}
 & + C_4 \ddot{x}_3 (\dot{h}_{x_1} \cos x_3 + \dot{v}_{x_1} \sin x_3) \\
 & + C_4 \dot{x}_3^2 (\dot{v}_{x_1} \cos x_3 - \dot{h}_{x_1} \sin x_3) \\
 & + C_5 (\ddot{x}_4 - \ddot{x}_3) [\dot{v}_{x_1} \sin(x_4 - x_3) - \dot{h}_{x_1} \cos(x_4 - x_3)] \\
 & + C_5 (\dot{x}_4 - \dot{x}_3)^2 [\dot{h}_{x_1} \sin(x_4 - x_3) + \dot{v}_{x_1} \cos(x_4 - x_3)] \\
 & + C_7 \ddot{x}_5 (\dot{h}_{x_1} \cos x_5 - \dot{v}_{x_1} \sin x_5) \\
 & - C_7 \dot{x}_5^2 (\dot{h}_{x_1} \sin x_5 + \dot{v}_{x_1} \sin x_5) \\
 & + [A_0 \frac{\partial v}{\partial x_1} + C_1 \sin x_1 - C_2 \sin(x_2 - x_1)] g = M_1
 \end{aligned} \tag{3.1}$$

and

$$\begin{aligned}
 & A_0 (\ddot{v} \cdot \dot{v}_{x_2} + \ddot{h} \cdot \dot{h}_{x_2}) + A_2 (\ddot{x}_2 - \ddot{x}_1) + C_1 \dot{x}_1 (\dot{h}_{x_2} \cos x_1 + \dot{v}_{x_2} \sin x_1) \\
 & + C_1 \dot{x}_1^2 (\dot{v}_{x_2} \cos x_1 - \dot{h}_{x_2} \sin x_1) - C_2 [\ddot{v} \sin(x_2 - x_1) - \ddot{h} \cos(x_2 - x_1)] \\
 & + C_2 (\ddot{x}_2 - \ddot{x}_1) [\dot{v}_{x_2} \sin(x_2 - x_1) - \dot{h}_{x_2} \cos(x_2 - x_1)] \\
 & + C_2 (\dot{x}_2 - \dot{x}_1)^2 [\dot{v}_{x_2} \cos(x_2 - x_1) + \dot{h}_{x_2} \sin(x_2 - x_1)] - C_3 (\cos x_2) \ddot{x}_1 \\
 & + C_3 (\sin x_2) \dot{x}_1^2 + C_4 \ddot{x}_3 (\dot{h}_{x_2} \cos x_3 + \dot{v}_{x_2} \sin x_3) + C_4 \dot{x}_3^2 (\dot{v}_{x_2} \cos x_3 \\
 & - \dot{h}_{x_2} \sin x_3) + C_5 (\ddot{x}_4 - \ddot{x}_3)^2 [\dot{v}_{x_2} \cos(x_4 - x_3) + \dot{h}_{x_2} \sin(x_4 - x_3)] \\
 & + [A_0 \frac{\partial v}{\partial x_2} + C_2 \sin(x_2 - x_1)] g = M_2
 \end{aligned} \tag{3.2}$$

where,

$$\dot{h} = \frac{dh}{dt}, \quad \ddot{h} = \frac{d^2 h}{dt^2}, \quad \dot{h}_{x_i} = \frac{\partial h}{\partial x_i}$$

and similar expressions hold for

\dot{v} , \ddot{v} , \dot{v}_{x_1} , etc.

(ii) For the leg consisting of links 3 and 4 in contact with the ground:

$$\begin{aligned}
 & A_0 (\ddot{v} \cdot \dot{v}_{x_3} + \ddot{h} \cdot \dot{h}_{x_3}) + A_3 \ddot{x}_3 - A_4 (\ddot{x}_4 - \ddot{x}_3) + C_4 (\dot{h} \cos x_3 + \dot{v} \sin x_3) \\
 & + C_4 \ddot{x}_3 (\dot{h}_{x_3} \cos x_3 + \dot{v}_{x_3} \sin x_3) + C_4 \dot{x}_3^2 (\dot{v}_{x_3} \sin x_3 - \dot{h}_{x_3} \cos x_3)
 \end{aligned}$$

$$\begin{aligned}
& - C_5 [\ddot{v} \sin(x_4 - x_3) - \dot{h} \cos(x_4 - x_3)] \\
& + C_5 (\ddot{x}_4 - \ddot{x}_3) [\dot{v}_{x_3} \sin(x_4 - x_3) - \dot{h}_{x_3} \cos(x_4 - x_3)] \\
& + C_5 (\dot{x}_4 - \dot{x}_3)^2 [\dot{v}_{x_3} \cos(x_4 - x_3) + \dot{h}_{x_3} \sin(x_4 - x_3)] \\
& - C_6 (\cos x_4) (\ddot{x}_4 - 2\ddot{x}_3) + C_6 (\sin x_4) \dot{x}_4 (\dot{x}_4 - 2\dot{x}_3) \quad (3.1a) \\
& + C_1 \ddot{x}_1 (\dot{h}_{x_3} \cos x_1 + \dot{v}_{x_3} \sin x_1) + C_1 \dot{x}_1^2 (\dot{v}_{x_3} \cos x_1 - \dot{h}_{x_3} \sin x_1) \\
& + C_2 (\ddot{x}_2 - \ddot{x}_1) [\dot{v}_{x_3} \sin(x_2 - x_1) - \dot{h}_{x_3} \cos(x_2 - x_1)] \\
& + C_2 (\dot{x}_2 - \dot{x}_1)^2 [\dot{h}_{x_3} \sin(x_2 - x_1) + \dot{v}_{x_3} \cos(x_2 - x_1)] + C_7 \ddot{x}_5 (\dot{h}_{x_3} \cos x_5 - \dot{v}_{x_3} \sin x_5) \\
& - C_7 \dot{x}_5^2 (\dot{h}_{x_3} \sin x_5 + \dot{v}_{x_3} \cos x_5) + [A_O \frac{\partial v}{\partial x_3} + C_4 \sin x_3 - C_5 \sin(x_4 - x_3)] g = M_3
\end{aligned}$$

and

$$\begin{aligned}
& A_O (\ddot{v} \dot{v}_{x_4} + \ddot{h} \dot{h}_{x_4}) + A_4 (\ddot{x}_4 - \ddot{x}_3) + C_4 \ddot{x}_3 (\dot{h}_{x_4} \cos x_3 + \dot{v}_{x_4} \sin x_4) \\
& + C_4 \dot{x}_3^2 (\dot{v}_{x_4} \cos x_3 - \dot{h}_{x_4} \sin x_4) - C_5 [\ddot{v} \sin(x_4 - x_3) - \dot{h} \cos(x_4 - x_3)] \\
& + C_5 (\ddot{x}_4 - \ddot{x}_3) [\dot{v}_{x_4} \sin(x_4 - x_3) - \dot{h}_{x_4} \cos(x_4 - x_3)] \\
& + C_5 (\dot{x}_4 - \dot{x}_3)^2 [\dot{v}_{x_4} \cos(x_4 - x_3) + \dot{h}_{x_4} \sin(x_4 - x_3)] \quad (3.2a) \\
& - C_6 (\cos x_4) \ddot{x}_3 + C_6 (\sin x_4) \dot{x}_3^2 + C_1 \ddot{x}_1 (\dot{h}_{x_4} \cos x_1 + \dot{v}_{x_4} \sin x_1) \\
& + C_1 \dot{x}_1^2 (\dot{v}_{x_4} \cos x_1 - \dot{h}_{x_4} \sin x_1) \\
& + C_2 (\ddot{x}_2 - \ddot{x}_1) [\dot{v}_{x_4} \sin(x_2 - x_1) - \dot{h}_{x_4} \cos(x_2 - x_1)] \\
& + C_2 (\dot{x}_2 - \dot{x}_1)^2 [\dot{v}_{x_4} \cos(x_2 - x_1) + \dot{h}_{x_4} \sin(x_2 - x_1)] \\
& + [A_O \frac{\partial v}{\partial x_4} + C_5 \sin(x_4 - x_3)] g = M_4
\end{aligned}$$

iii) For the leg consisting of links 1 and 2 in swing motion:

$$A_1 \ddot{x}_1 - A_2 (\ddot{x}_2 - \ddot{x}_1) + C_1 (\ddot{h} \cos x_1 + \ddot{v} \sin x_1) - C_2 [\ddot{v} \sin(x_2 - x_1) - \ddot{h} \cos(x_2 - x_1)] - C_3 (\cos x_2) (\ddot{x}_2 - 2\dot{x}_1) + C_3 (\sin x_2) \dot{x}_2 (\dot{x}_2 - 2\dot{x}_1) + [C_1 \sin x_1 - C_2 (x_2 - x_1)] g = M_1 \quad (3.3)$$

and

$$A_2 (\ddot{x}_2 - \ddot{x}_1) + C_1 [\ddot{v} \sin(x_2 - x_1) - \ddot{h} \cos(x_2 - x_1)] - C_3 (\cos x_2) \ddot{x}_1 + C_3 (\sin x_2) \dot{x}_1^2 + C_2 g \sin(x_2 - x_1) = M_2 \quad (3.4)$$

(iv) For the leg consisting of links 3 and 4 in swing motion:

$$A_3 \ddot{x}_3 - A_4 (\ddot{x}_4 - \ddot{x}_3) + C_4 (\ddot{h} \cos x_3 + \ddot{v} \sin x_3) - C_5 [\ddot{v} \sin(x_4 - x_3) - \ddot{h} \cos(x_4 - x_3)] - C_6 (\cos x_4) (\ddot{x}_4 - 2\dot{x}_3) + C_6 (\sin x_4) \dot{x}_4 (\dot{x}_4 - 2\dot{x}_3) + [C_3 \sin x_3 - C_4 \sin(x_4 - x_3)] g = M_3 \quad (3.3a)$$

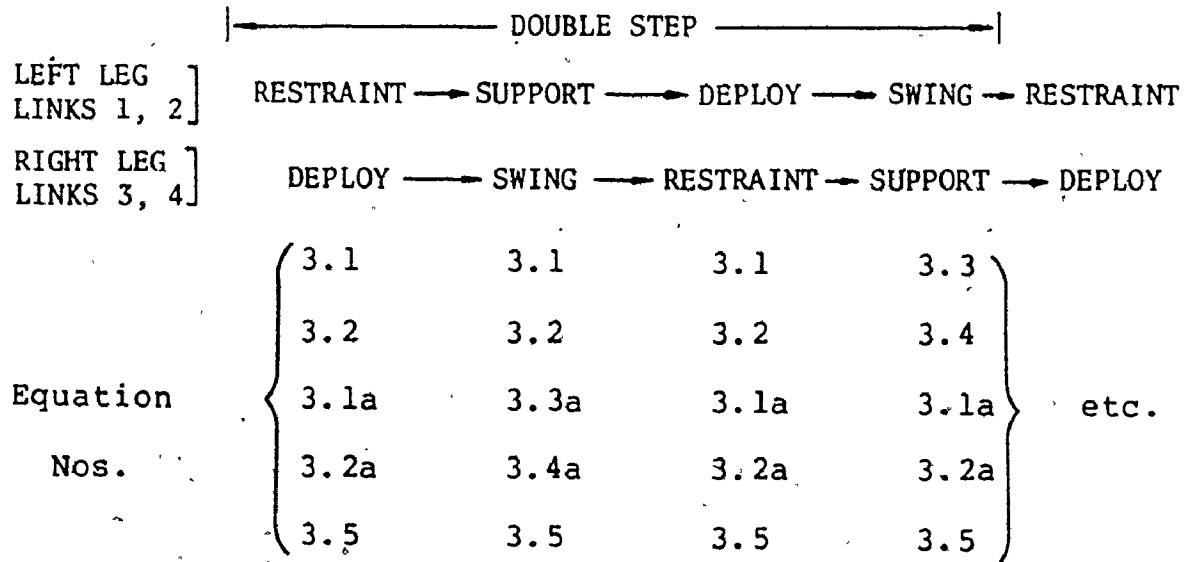
and

$$A_4 (\ddot{x}_4 - \ddot{x}_3) + C_4 [\ddot{v} \sin(x_4 - x_3) - \ddot{h} \cos(x_4 - x_3)] - C_6 (\cos x_4) \ddot{x}_3 + C_6 (\sin x_4) \dot{x}_3^2 + C_5 g \sin(x_4 - x_3) = M_4 \quad (3.4a)$$

(v) For the motion of the trunk:

$$A_5 \ddot{x}_5 + C_7 (\ddot{h} \cos x_5 - \ddot{v} \sin x_5) - C_7 g \sin x_5 = M_5 \quad (3.5)$$

It is clear that not all of the above equations apply simultaneously. The relevant equations may be chosen in accordance with the following:



The above equations completely describe the behaviour of the dynamical system in the sagittal plane and no approximations have been introduced. If motion in the other planes (i.e. frontal and horizontal) is to be considered, the equations would be supplemented by additional ones but no modification of the present ones would be required. For the present, however, only the sagittal behaviour will be considered, mainly due to the extreme complexity of higher order models.

It can easily be seen from equations (3.1) to (3.5) that the system of differential equations is always coupled. This arises from the coupling of motion between the two extremities via the hip. As indicated earlier, if the optimal programming problem is to be tractable it is necessary to write the equations of motion in the form of equation (2.23). This would imply that the five equations (3.1) to (3.5) whose form is now

$$[A] \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \vdots \\ \ddot{x}_5 \end{bmatrix} = [B],$$

or simply

$$[A] [\ddot{x}] = [B], \quad (3.6)$$

would have to be written in the form

$$[\ddot{x}] = [A]^{-1} [B] \quad (3.7)$$

where the matrices $[A]$ and $[B]$ are of dimensions 5×5 and 1×5 and have elements which are expressions. The inversion of matrix $[A]$ is far too complex to handle.

A device which has been found useful for simplifying the dynamical equations without the loss of a significant degree of accuracy was used by Beckett [5] and Chow [6] among others. This consists of prescribing the hip trajectory and at the same time considering it to be the origin of a moving co-ordinate system. In level walking over the normal range of speed, pace frequency and step length, the hip position $(h; v)$ has the following characteristic motion: the horizontal component $h(t)$ is uniform motion while the vertical component $v(t)$ describes a double period sinusoid about some mean value. Thus equations (2.10) may be re-written to incorporate the above as:

$$\begin{aligned} h(t) &= x_{\delta+1}^A - \ell_\delta \sin x_\delta + \ell_{\delta+1} \sin (x_{\delta+1} - x_\delta) \equiv \tilde{h}(t) \\ v(t) &= y_{\delta+1}^A + \ell_\delta \cos x_\delta + \ell_{\delta+1} \cos (x_{\delta+1} - x_\delta) \equiv \tilde{v}_o - \tilde{v}(t) \quad (3.8) \end{aligned}$$

where δ is defined as before, and the functions $\tilde{h}(t)$ and $\tilde{v}(t)$ are prescribed functions of time. Useful expressions for the hip position are

$$\tilde{h}(t) = h_o (t + t_o) \quad (3.9)$$

and

$$\tilde{v}(t) = \beta_1 \sin \frac{4\pi}{\tau} (t + \beta_2 \tau) \quad (3.10)$$

where h_o = velocity of level walking

t_o = initial time

β_1, β_2 = constants based on initial conditions of motion

and τ = period of double step

This simplification alone does not reduce the high-order complexity of the equations due to the presence of functions \dot{x}_i and \ddot{x}_i , which are in general non-zero. As discussed earlier, elimination of these may be achieved by considering the hip as the origin of a moving co-ordinate system. With this simplification the dynamical equations reduce to:

- (i) For the leg consisting of links 1 and 2 in contact with the ground or in swing:

$$\begin{aligned}
 & (A_1 + 2C_3 \cos x_2) \ddot{x}_1 - (A_2 + C_3 \cos x_2) \ddot{x}_2 \\
 & + C_3 \dot{x}_2 \sin x_2 (\dot{x}_2 - 2 \dot{x}_1) \\
 & + (C_1 \sin x_1 - C_2 \cos x_1) (\ddot{v} + g) = M_1
 \end{aligned} \tag{3.11}$$

and

$$\begin{aligned}
 & - (A_2 + C_3 \cos x_2) \ddot{x}_1 + A_2 \ddot{x}_2 + C_3 \dot{x}_1^2 \sin x_2 \\
 & + C_3 \cos x_2 (\ddot{v} + g) = M_2
 \end{aligned} \tag{3.12}$$

(ii) For the leg consists of links 3 and 4 in contact with the ground or in swing:

$$\begin{aligned}
 & (A_3 + 2C_6 \cos x_4) \ddot{x}_3 - (A_4 + C_6 \cos x_4) \ddot{x}_4 \\
 & + C_6 \dot{x}_4 \sin x_4 (\dot{x}_4 - 2 \dot{x}_3)
 \end{aligned} \tag{3.13}$$

and

$$+ (C_4 \sin x_3 - C_5 \cos x_3) (\ddot{v} + g) = M_3$$

$$\begin{aligned}
 & - (A_4 + C_6 \cos x_4) \ddot{x}_3 + A_4 \ddot{x}_4 + C_6 \dot{x}_3^2 \sin x_4 \\
 & + C_6 \cos x_4 (\ddot{v} + g) = M_4
 \end{aligned} \tag{3.14}$$

iii) For the motion of the trunk:

$$A_5 \ddot{x} - C_7 \sin x_5 (\ddot{v} + g) = M_5 \tag{3.15}$$

The similarity between equations (3.11) and (3.13), and between (3.12) and (3.14) can easily be seen, the only difference being the interchange of the angular variables and the corresponding constants and moment functions. The decoupling of the equations can also be easily verified as

can the independence of the equation of trunk motion from the limb angular variables (and vice versa). It turns out that the motion of the trunk need not be included in the model insofar as the energy consumption is concerned but must be considered in terms of stability of walking. This will be examined in greater detail later.

For the time being, then, only equations (3.11) to (3.14) will be examined. Because of the similarity, a common notation will be adopted. (Henceforth links 1 and 2 will be referred to as the left leg and links 3 and 4 as the right leg). Using the two indices δ and ϵ , defined as

$$\delta = \begin{cases} 1, & \text{for left leg} \\ 3, & \text{for right leg} \end{cases} \quad (3.16)$$

as before, and

$$\epsilon = \begin{cases} 1, & \text{for left leg} \\ 4, & \text{for right leg} \end{cases} \quad (3.17)$$

equations (3.11) and (3.13) may be written collectively as

$$(A_{\delta} + 2C_{\epsilon+2} \cos x_{\delta+1}) \ddot{x}_{\delta} - (A_{\delta+1} + C_{\epsilon+2} \cos x_{\delta+1}) \ddot{x}_{\delta+1} + C_{\epsilon+2} \dot{x}_{\delta+1} \sin x_{\delta+1} (\dot{x}_{\delta+1} - 2 \dot{x}_{\delta}) + (C_{\epsilon} \sin x_{\delta} - C_{\epsilon+1} \cos x_{\delta}) (\ddot{\theta} + g) = M_{\delta} \quad (3.18)$$

$$+ (C_{\epsilon} \sin x_{\delta} - C_{\epsilon+1} \cos x_{\delta}) (\ddot{\theta} + g) = M_{\delta}$$

and equations (3.12) and (3.14) as

$$-(A_{\delta+1} + C_{\epsilon+2} \cos x_{\delta+1}) \ddot{x}_{\delta} + A_{\delta+1} \ddot{x}_{\delta+1} + C_{\epsilon+2} \dot{x}_{\delta}^2 \sin x_{\delta+1} + C_{\epsilon+2} \cos x_{\delta+1} (\ddot{\theta} + g) = M_{\delta+1} \quad (3.19)$$

where now the moment functions (eq'n's 2.12) may be written as

$$\begin{aligned} M_\delta &= u_\delta + u_\delta^P + M_\alpha^{\delta+1} + Y_{\delta+1} (\ell_\delta \sin x_\delta - \ell_{\delta+1} \sin(x_{\delta+1} - x_\delta)) \quad (3.20) \\ &\quad + X_{\delta+1} (\ell_\delta \cos x_\delta + \ell_{\delta+1} \cos(x_{\delta+1} - x_\delta)) \end{aligned}$$

and

$$\begin{aligned} M_{\delta+1} &= u_{\delta+1} + u_{\delta+1}^P + M_\alpha^{\delta+1} + Y_{\delta+1} \ell_{\delta+1} \sin(x_{\delta+1} - x_\delta) \quad (3.21) \\ &\quad - X_{\delta+1} \ell_{\delta+1} \cos(x_{\delta+1} - x_\delta) \end{aligned}$$

It is now clear that from the point of view of optimal programming, the solution to the present gait problem is resolved into two separate problems, one for each of the lower limbs. Furthermore, for the normal, non-pathological gait, the two lower extremities will be, for practical purposes, identical, in which case the solution of the dynamical equations will be identical for the two limbs and hence only one set of equations need be solved. The duality of the equations will, however, be carried through the development so that the case of dissimilar legs (e.g., the unilateral amputee, or the hemiplegic patient) may be automatically accounted for.

Equations (3.18) to (3.21) are now in such a form that the dynamical equations may be written in canonical form (cf. equation (2.23)). Making substitution (2.22) and letting

$$z_1 = x_\delta$$

$$z_2 = x_{\delta+1}$$

$$z_3 = x_{\delta+5} \equiv \dot{x}_{\delta} \quad (3.22)$$

$$z_4 = x_{\delta+6} \equiv \dot{x}_{\delta+1}$$

the resulting equations are:

$$\dot{z}_1 = z_3$$

$$\dot{z}_2 = z_4$$

$$\dot{z}_3 = \Delta^{-1} [R_1 B_1 + R_2 B_2] \quad (3.23)$$

$$\dot{z}_4 = \Delta^{-1} [R_1 B_2 + R_2 B_3]$$

where

$$R_1 = -C_{\varepsilon+2} z_4 \sin z_2 (z_4 - 2z_3) - (C_\varepsilon \sin z_1 - C_{\varepsilon+1} \cos z_1) (\tilde{v} + g) + M_\delta$$

$$R_2 = -C_{\varepsilon+2} z_3^2 \sin z_2 - C_{\varepsilon+2} \cos z_2 (\tilde{v} + g) + M_{\delta+1}$$

$$B_1 = A_{\delta+1}$$

$$B_2 = A_{\delta+1} + C_{\varepsilon+2} \cos z_2$$

$$B_3 = A_\delta + 2 C_{\varepsilon+2} \cos z_2$$

$$\Delta = B_1 B_3 - B_2^2$$

Equations (3.23) now are the complete dynamical equations which describe the motion of the lower limbs. All that remains to complete the dynamic picture is the derivation of the pertinent constraints (including boundary conditions).

3.3 Constraint Equations

The constraint relations will be treated separately for the three portions of the gait cycle - deploy, swing, and stance. The deploy phase will be treated first.

Since the hip trajectory has been prescribed, the angular variables are no longer independent. They are, however, constrained so that the foot remains in contact with the ground. During deploy, the ankle of the foot describes a circular arc about the ball of the foot. Figure 3.1 shows the foot and the new variables introduced to describe its motion.

It is important to note that the foot length $d_\alpha^{1\delta}$ is somewhat less than the actual length of the foot since foot rotation does not occur about the toes but rather about the ball of the foot. The overall foot length is important, however, during the swing phase when the toe's must clear the ground.

Let the beginning and end of the deploy phase be denoted by t' and t'' respectively. Then at $t = t'$, summation of vertical displacements yields

$$\begin{aligned} v_o &= \tilde{v}(t') + l_\delta \cos z_1(t') + l_{\delta+1} \cos [z_2(t') - z_1(t')] \\ &\quad + d_\alpha^{1\delta} \sin \alpha(t'). \end{aligned} \tag{3.24}$$

Similarly, for $t' < t \leq t''$

$$\begin{aligned} v_o &= \tilde{v}(t) + l_\delta \cos z_1(t) + l_{\delta+1} \cos [z_2(t) - z_1(t)] \\ &\quad + d_\alpha^{1\delta} \sin \alpha(t). \end{aligned} \tag{3.25}$$

Combining equations (3.24) and (3.25) to eliminate v_o results in

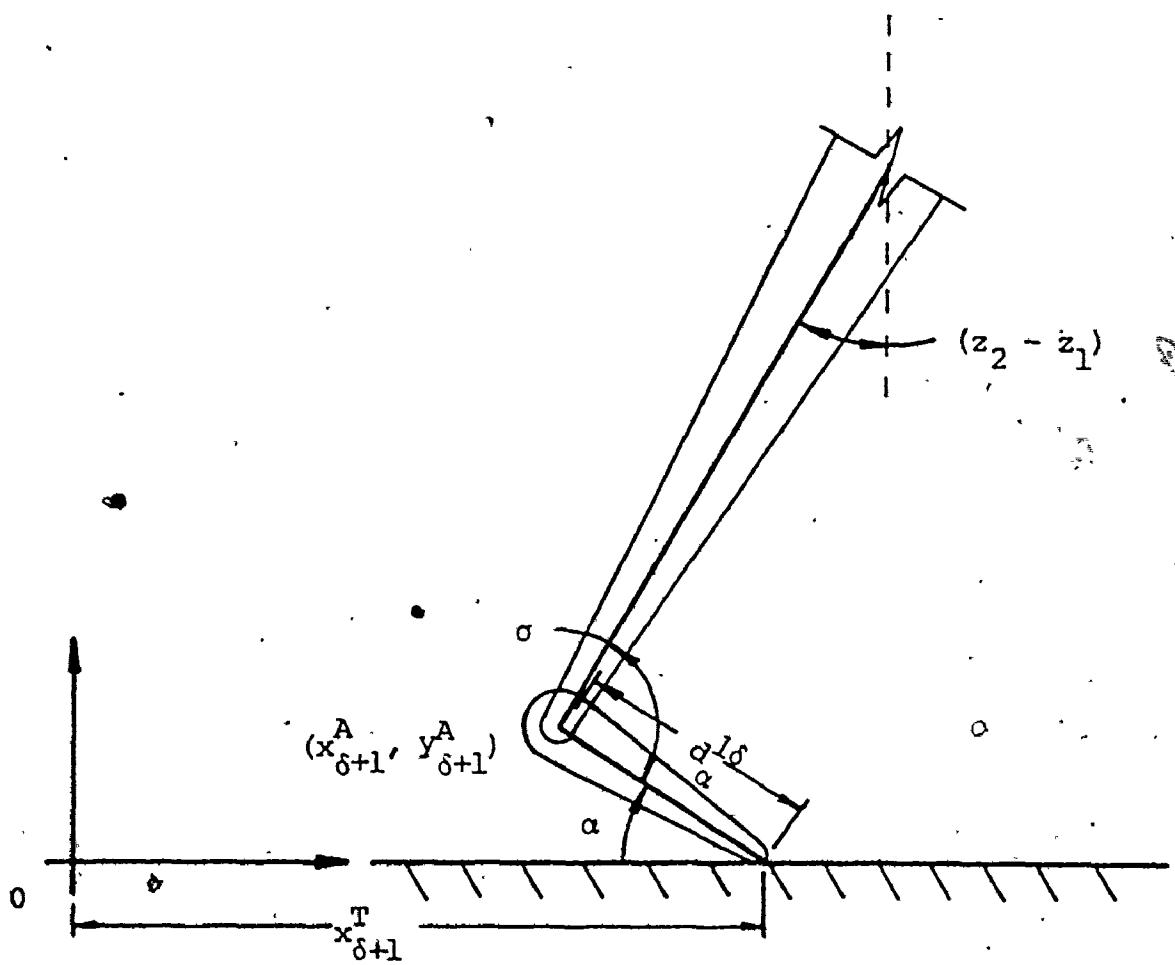


Figure 3.1. Ankle Variables.

$$\begin{aligned} d_{\alpha}^{1\delta} \sin \alpha(t) &= v_1 - \tilde{v}(t) - l_{\delta} \cos z_1 \\ -l_{\delta+1} \cos(z_2 - z_1) &\equiv y_{\delta+1}^A \end{aligned} \quad (3.26)$$

where

$$\begin{aligned} v_1 &= \tilde{v}(t') + l_{\delta} \cos z_1(t') + l_{\delta+1} \cos [z_2(t') - z_1(t')] \\ &\quad + d_{\alpha}^{1\delta} \sin \alpha(t'). \end{aligned}$$

During the deploy phase v_1 is a constant and is determined from the previous (stance) phase of activity (i.e. the functions z_1 , z_2 and α have been calculated as a result of the stance phase computation).

A similar procedure, if carried out in the horizontal direction produces

$$d_{\alpha}^{1\delta} \cos \alpha(t) = h_1 - h_o(t) - l_{\delta} \sin z_1 \quad (3.27)$$

$$+ l_{\delta+1} \sin(z_2 - z_1) \equiv x_{\delta+1}^T - x_{\delta+1}^A$$

wherein

$$\begin{aligned} h_1 &= h_o t' + l_{\delta} \sin z_1(t') - l_{\delta+1} \sin [z_2(t') - z_1(t')] \\ &\quad + d_{\alpha}^{1\delta} \cos \alpha(t'). \end{aligned}$$

The function $\alpha(t)$ can be eliminated using equations (3.26) and (3.27) to produce the first constraint equation (i.e. for $t' < t \leq t''$):

$$\begin{aligned} l_{\delta}^2 + l_{\delta+1}^2 - (d_{\alpha}^{1\delta})^2 + (y_1 - \tilde{v})^2 + (h_1 - h_o t)^2 + 2 l_{\delta} l_{\delta+1} \cos z_2 \\ - 2(v_1 - \tilde{v})(l_{\delta} \cos z_1 + l_{\delta+1} \cos(z_2 - z_1)) \quad (3.28) \end{aligned}$$

$$- 2(h_1 - h_o t)(l_{\delta} \sin z_1 - l_{\delta+1} \sin(z_2 - z_1)) = 0$$

Upon careful examination it is clear that this is simply saying that

$$(x_{\delta+1}^T - x_{\delta+1}^A)^2 + (y_{\delta+1}^A)^2 = (d_\alpha^{1\delta})^2.$$

Replacing the left hand side of equation (3.28) by the function $S_3(z; t)$, the first constraint condition becomes

$$S_3(z; t) = 0, \quad t' < t \leq t'' \quad (3.29)$$

At the end of the deploy phase, the angle σ between the shank and the foot approaches its limiting maximum value. Then, during the swing phase of activity, the foot remains "locked" at $\sigma = \sigma_{\max}$ while the toes are kept off the ground. This condition results in the following constraint relation.

$$\tilde{v}(t) + l_\delta \cos z_1 + l_{\delta+1} \cos (z_2 - z_1) \quad (3.30)$$

$$+ d_\alpha^{2\delta} \sin \zeta(t) - v_o \leq 0$$

where

$$\zeta(t) = \sigma_{\max} + z_2(t) - z_1(t) - \frac{\pi}{2}$$

being the acute angle between the foot and the horizontal.

Again, defining $S_4(z; t)$ to be the left hand side of inequality (3.30) we have

$$S_4(z; t) \leq 0, \quad t'' < t \leq t_1 \quad (3.31)$$

with the note that the constant $d_\alpha^{2\delta}$ is the actual length of the foot including the toes.

A further condition, which applies not only during swing but also during the deploy and stance phases, is that the knee angle $z_2(t)$ must remain positive. That is

$$z_2(t) > 0 \quad \text{for} \quad t'' \leq t \leq t_1. \quad (3.32)$$

In the stance portion, i.e. $t_0 \leq t \leq t'$, the foot remains flat on the ground, except for the brief moment immediately following heel strike. This can effectively be accounted for if the ankle co-ordinates $(x_{\delta+1}^A, y_{\delta+1}^A)$ are constrained to follow a prescribed trajectory, say $(q(t), p(t))$ which must be determined experimentally. The functions $q(t)$ and $p(t)$ can be treated as constant except during the restraint portion, and even then the variation with time is not large.

In light of this, two additional constraint equations may be derived, and, in keeping with earlier notation, these are:

$$\begin{aligned} S_1(z; t) &= \dot{v}(t) + l_\delta \cos z_1 + l_{\delta+1} \cos(z_2 - z_1) - v_0 + p(t) = 0 \\ S_2(z; t) &= h_0(t_0 + t) + l_\delta \sin z_1 - l_{\delta+1} \sin(z_2 - z_1) - q(t) = 0 \end{aligned} \quad (3.33)$$

3.4 Ankle Moments and Ground Reactions

What remains at this stage to complete the dynamical picture is the development of the ankle moments and reaction forces. Since the ankle itself is not in contact with the ground, the forces and moments acting on the ankle must of necessity be transferred via the foot. The following is a derivation of these ankle loads. Figure 3.2 shows a free

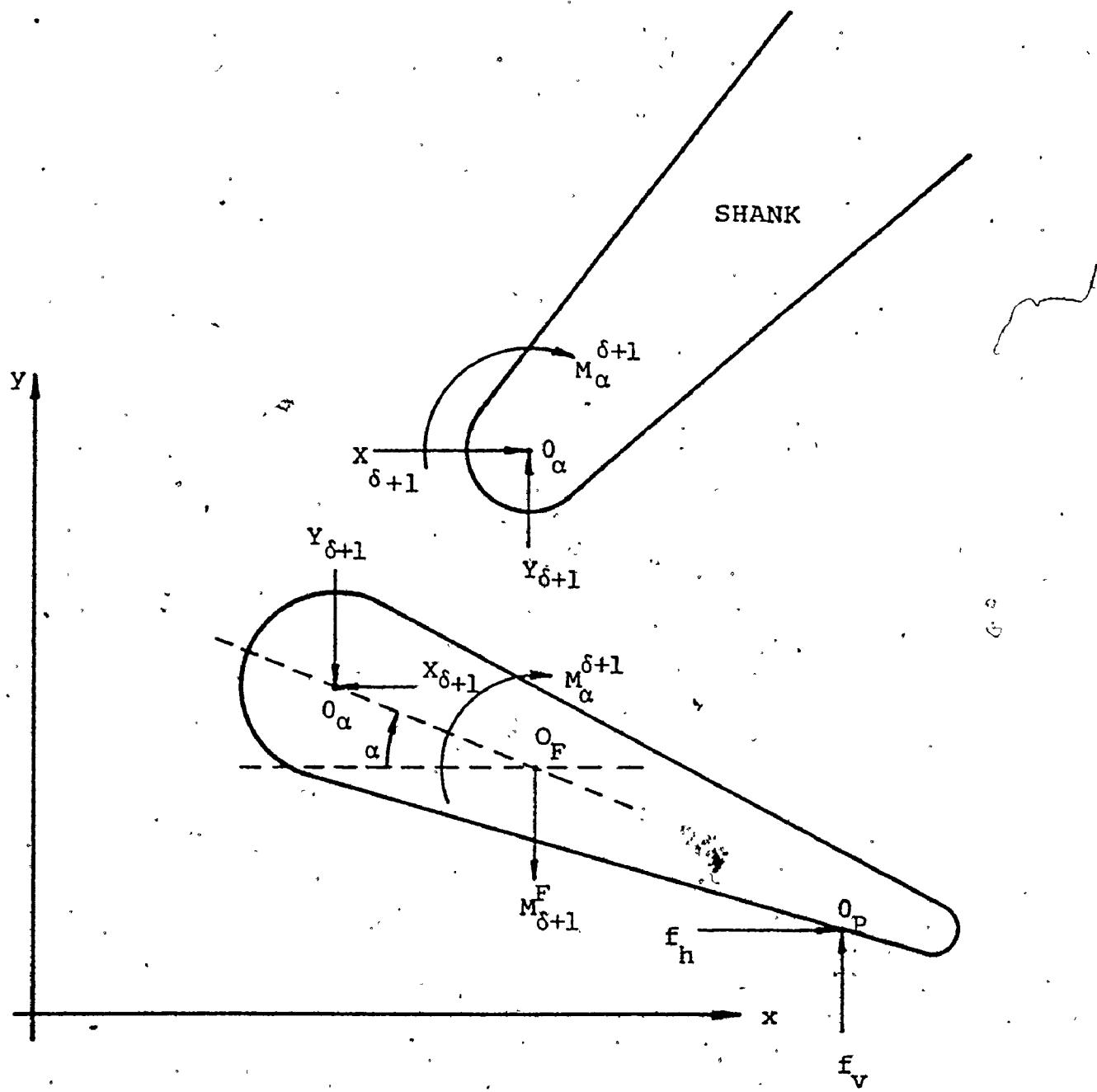


Figure 3.2. Free-body diagram of the foot. Notation: O_P , centre of pressure ($x_{\delta+1}^P, Y_{\delta+1}^P$); O_F c.g. of the foot ($x_{\delta+1}^F, Y_{\delta+1}^F$); O_A ankle ($x_{\delta+1}^A, Y_{\delta+1}^A$); $X_{\delta+1}, Y_{\delta+1}$ internal reaction forces at ankle; f_h, f_v ground reaction forces; $M_{\delta+1}^F$ mass of foot.

body diagram of the foot and ankle (due to [6]).

Referring to Figure 3.2, the motion of the foot may be described by consideration of the translation of the centre of gravity and rotation about it (O_F). Summing forces in the vertical direction produces the equation

$$M_{\delta+1}^F \ddot{y}_{\delta+1}^F = f_v - M_{\delta+1}^F g - Y_{\delta+1} \quad (3.34)$$

and in the horizontal direction

$$M_{\delta+1}^F \ddot{x}_{\delta+1}^F = f_h - X_{\delta+1} \quad (3.35)$$

Rearranging equations (3.34) and (3.35), the ankle reaction forces may be written as

$$Y_{\delta+1} = f_v - M_{\delta+1}^F g - (M_{\delta+1}^F \ddot{y}_{\delta+1}^F) \quad (3.36)$$

$$X_{\delta+1} = f_h - (M_{\delta+1}^F \ddot{x}_{\delta+1}^F) \quad (3.37)$$

Taking moments about the point O_F , the ankle moment may be written (after some rearrangement) as:

$$\begin{aligned} M_{\alpha}^{\delta+1} &= f_v(x_{\delta+1}^P - x_{\delta+1}^A) + f_h(y_{\delta+1}^A - y_{\delta+1}^P) \\ &\quad - M_{\delta+1}^F g(x_{\delta+1}^F - x_{\delta+1}^A) + M_{\delta+1}^F g(y_{\delta+1}^F - y_{\delta+1}^A) \\ &\quad - (M_{\delta+1}^F y_{\delta+1}^F)(y_{\delta+1}^F - y_{\delta+1}^A) + I_{\delta+1}^F \ddot{\alpha} \end{aligned} \quad (3.38)$$

where $I_{\delta+1}^F$ is the moment of inertia of the foot about its centre of gravity.

Equations (3.36), (3.37) and (3.38) may be simplified

with the observation that, while the foot is in contact with the ground,

$$f_v >> M_{\delta+1}^F \ddot{y}_{\delta+1}^F$$

$$f_h >> M_{\delta+1}^F \ddot{x}_{\delta+1}^F$$

and $I_{\delta+1}^F \ddot{\alpha} \approx 0$

since $I_{\delta+1}^F = 10^{-2} \text{ kg m}^2$.

Furthermore, the quantity $(M_{\delta+1}^F g)$ may be taken into account in the dynamical sense by inclusion of the foot in determining the mass, centre of gravity and moment of inertia of the shank of each leg. Also since $(M_{\delta+1}^F g)(x_{\delta+1}^F - x_{\delta+1}^A)$ is relatively small as compared to say $f_v(x_{\delta+1}^P - x_{\delta+1}^A)$, a resulting simplified estimate of the foot dynamics results in

$$y_{\delta+1} \approx f_v \quad (3.39a)$$

$$x_{\delta+1} \approx f_h \quad (3.39b)$$

and

$$M_{\alpha}^{\delta+1} \approx f_v(x_{\delta+1}^P - x_{\delta+1}^A) + f_h(y_{\delta+1}^A - y_{\delta+1}^P) \quad (3.39c)$$

The functions $f_v(t)$, $f_h(t)$, $x_{\delta+1}^P(t)$ and $y_{\delta+1}^P(t)$ must be determined experimentally. (Note: for level walking $y_{\delta+1}^P \equiv 0$ throughout stance and deploy phases; during swing phase, $x_{\delta+1} \equiv 0$, $y_{\delta+1} \equiv 0$ and $M_{\alpha}^{\delta+1} \equiv 0$).

This completes the description of the dynamical system. The equations have been derived, the constraint relations determined and ankle forces and moments evaluated. The

with the observation that, while the foot is in contact with the ground,

$$f_v >> M_{\delta+1}^F \dot{y}_{\delta+1}^F$$

$$f_h >> M_{\delta+1}^F \dot{x}_{\delta+1}^F$$

and $I_{\delta+1}^F \ddot{\alpha} \approx 0$

since $I_{\delta+1}^F \approx 10^{-2} \text{ kg.m}^2$.

Furthermore, the quantity $(M_{\delta+1}^F g)$ maybe taken into account in the dynamical sense by inclusion of the foot in determining the mass, centre of gravity and moment of inertial of the shank of each leg. Also since $(M_{\delta+1}^F g)(x_{\delta+1}^F - x_{\delta+1}^A)$ is relatively small as compared to say $f_v(x_{\delta+1}^P - x_{\delta+1}^A)$, a resulting simplified estimate of the foot dynamics results in

$$y_{\delta+1} \approx f_v \quad (3.39a)$$

$$x_{\delta+1} \approx f_h \quad (3.39b)$$

and

$$M_{\alpha}^{\delta+1} \approx f_v(x_{\delta+1}^P - x_{\delta+1}^A) + f_h(y_{\delta+1}^A - y_{\delta+1}^P) \quad (3.39c)$$

The functions $f_v(t)$, $f_h(t)$, $x_{\delta+1}^P(t)$ and $y_{\delta+1}^P(t)$ must be determined experimentally. (Note: for level walking $y_{\delta+1}^P \equiv 0$ throughout stance and deploy phases; during swing phase, $x_{\delta+1} \equiv 0$, $y_{\delta+1} \equiv 0$ and $M_{\alpha}^{\delta+1} \equiv 0$).

This completes the description of the dynamical system. The equations have been derived, the constraint relations determined and ankle forces and moments evaluated. The

trunk motion has not been included at this stage since the problem of stability can be treated separately (in parallel with the above). Chow and Jacobson [25] have modelled the stability aspects of the torso essentially as an inverted pendulum and the resulting differential equations result from a perturbation analysis. Solution of these equations is in itself a difficult task and has not been included in the present model. Because of the imposition of a specified trajectory for the hip joint in decoupling the equations of motion, the stability problem need not concern us here. For the walking individual stability is of importance but mathematically, its inclusion is not necessary.

CHAPTER IV
DEVELOPMENT OF MUSCULO-MECHANICAL MODEL
EXPLICIT FORM

4.1 Mathematical Development

With the dynamical picture complete, the next step in the development of the full optimal control model is the derivation of the explicit form of equation (2.20)

$$F_k = F_k(x_i, \dot{x}_i, \phi_k, \theta_k) \quad (k=1, \dots, 10 \ i=1, \dots, 5) \quad (2.20)$$

which represents the force generated in the k -th muscle as a function of the angular and control variables. The fine details of the control model of muscle (due to Hatze) can be found in references [3], [7] and [13]. The points of discussion here will dwell upon only those aspects of the model relevant to the particular problem at hand and necessary to arrive at the final mathematical form.

It has been generally accepted [14, 15, 16, 17] that muscle fibres (and hence the total skeletal muscle) may be functionally viewed as being comprised of three basic structural elements [7]: the active and controllable contractile element (CE), the passive and negligibly damped [14] series elastic element (SE), and the passive, damped parallel elastic element (PE). Figure 4.1 is a schematic representa-

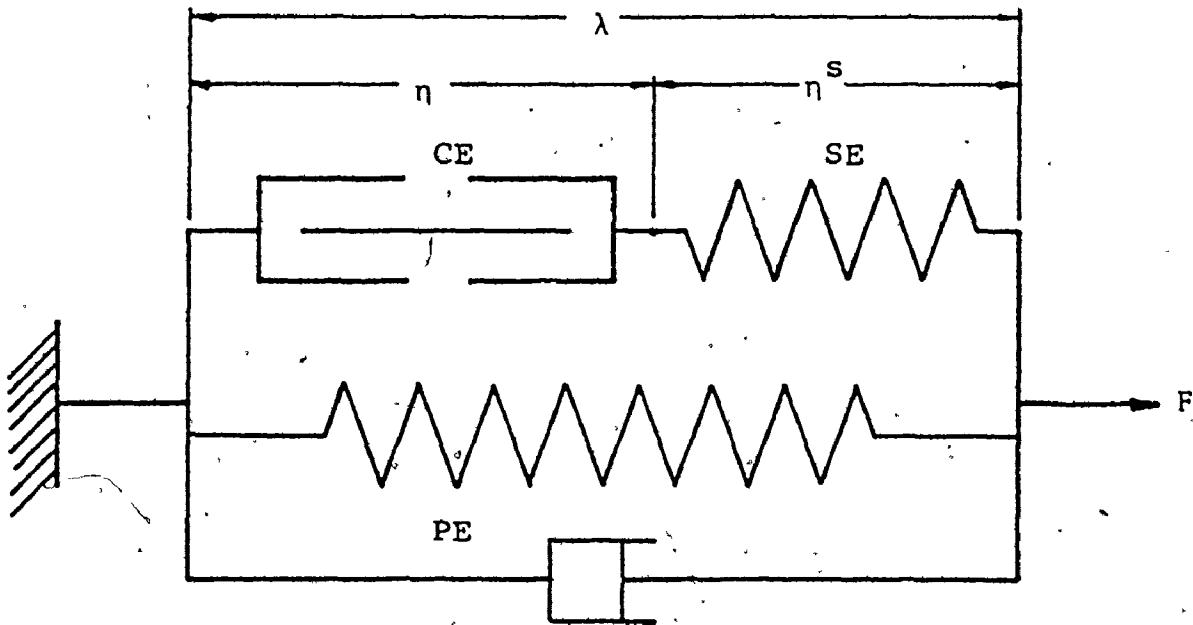


Figure 4.1 Diagrammatic representation of skeletal muscle components. Notation is as follows:

CE - contractile element of length n ;

SE - series elastic element of length n^s ;

PE - parallel elastic element of length $\lambda (= n + n^s)$;

F - total force across muscle.

tion of the arrangement of the three basic elements in the lumped model of skeletal muscle.

It must be emphasized that the exact anatomical location of the series and parallel elastic elements is not known and only their functional characteristics are being modelled. Also, the representation in Figure 4.1 is not unique in that a configuration with the PE in parallel with the CE would be equally as acceptable.

Referring to Figure 4.1, then, the length λ of the muscle depends (in general) upon the angular variables x_i and the velocity of shortening (or lengthening) $\dot{\lambda}$ upon the angles x_i and angular velocities \dot{x}_i ; i.e.

$$\begin{aligned}\lambda &= \lambda(x_i) \\ \dot{\lambda} &= \dot{\lambda}(x_i, \dot{x}_i).\end{aligned}\tag{2.13}$$

The relationships between the muscle length λ and the angular positions (i.e. first equation of (2.13)) must be experimentally determined. The second of (2.13) may then be easily found by differentiation of $\dot{\lambda}_k = \dot{\lambda}_k(x_i)$.

It is then clear that the following force relationships hold for the k -th muscle

$$F_k = F_k^{PE}(\lambda_k(x_i), \dot{\lambda}_k(x_i, \dot{x}_i)) + F_k^{SE}(\lambda_k(x_i), n_k)\tag{4.1}$$

and

$$F_k^{SE} (\lambda_k(x_i), n_k) = F_k^{CE} (n_k, \dot{n}_k, a_k, q_k) \quad (4.2)$$

where the functions $a_k(t)$ and $q_k(t)$ denote (in rather broad terms for the moment, with further discussion to follow) the "mode of activity" and "active state" of the muscle respectively.

In light of equation (4.1) then, equations (2.11), expressing the active joint torques as functions of the muscle forces may be written as follows:

$$\begin{aligned} u_1 &= F_1^{SE} d_1 + (F_3^{SE} + F_3^{PE}) d_3 - (F_4^{SE} + F_4^{PE}) d_4^1 \\ -u_2 &= F_2^{SE} d_2 + (F_3^{SE} + F_3^{PE}) d_2 - (F_4^{SE} + F_4^{PE}) d_4^2 - F_5^{SE} d_5 \\ u_3 &= F_6^{SE} d_6 + (F_8^{SE} + F_8^{PE}) d_8 - (F_9^{SE} + F_9^{PE}) d_9^3 \quad (4.3) \\ -u_4 &= F_7^{SE} d_7 + (F_8^{SE} + F_8^{PE}) d_7 - (F_9^{SE} + F_9^{PE}) d_9^4 - F_{10}^{SE} d_{10} \\ u_5 &= u_1 + u_3 . \end{aligned}$$

It is important to note that the terms F_1^{PE} , F_2^{PE} , F_5^{PE} , F_6^{PE} , F_7^{PE} and F_{10}^{PE} do not appear in equations (4.3). An explanation of this follows. The muscle groups 1, 2, and 5 (and similarly groups 6, 7 and 10) of Figure 2.2 each span only one joint. It is, therefore, not possible to distinguish between passive muscle torques and passive torques due to other structures (e.g. ligaments and cartilage) for the joints spanned by these muscles. Hence, terms such as $F_2^{PE} d_2$ must be included in the functions u_i^P which appear in equations (2.12).

[Note: The gastrocnemius muscle of course spans also the ankle joint, and thus will be considered in the determination of muscle lengths, but the joints of interest are primarily the hip and knee, and no error is introduced by the inclusion of F_5^{PE} , F_{10}^{PE} in u_2^P and u_4^P . Important also is the fact that the M. gastrocnemius is responsible for the generation of the ankle moment during plantar flexion. Hence the force level in this muscle must be suitably adjusted in the deploy portion of the gait cycle in the form of a constraint. See section 4.2.]

All passive torques and the properties of the PE for the rectus femoris and hamstring groups must, however, be found experimentally for the individual subject. The techniques for these measurements will be discussed in a later chapter.

Attention will now be focused on the function $F_k^{SE}(\lambda_k(x_i), n_k)$ of equation (4.1). As indicated in Figure 4.1, F_k^{SE} should be a function of n_k^S . Clearly, however, n_k^S may be eliminated so that the variable n_k is used; i.e.

$$n_k^S = \lambda_k - n_k, \quad k=1, \dots, 10. \quad (4.4)$$

The function $F_k^{SE}(\lambda_k(x_i), n_k)$ has been shown [13] to take the explicit form

$$F_k^{SE} = b_{7k} [\exp(b_{8k}\{\lambda_k - [(x_{10+k} + b_{9k})^2 - b_{10k}]^{1/2} - b_{11k}\}) - 1] \quad (4.5)$$

where $k = 1, \dots, 10$, b_{7k} to b_{11k} are muscle-specific constants for the k -th muscle, and

$$x_{10+k} = n_k - \bar{n}_k \quad (4.6)$$

with \bar{n}_k being the optimum length of the contractile element of the k -th muscle (i.e. the length at which the isometric-force generation is maximum).

Having discussed the characteristics of the series elastic element, what remains to be considered is the force producing contractile element. The force generated in the CE of the k -th muscle has been written implicitly as $F_k^{CE}(n_k, \bar{n}_k, a_k, q_k)$ (cf. equation (4.2)). In order that the derivation of the explicit form of this function F_k^{CE} be somewhat less confusing, the subscript k will be dropped for the discussion of the general model.

From reference [7] it can be shown that the force produced by the contractile element may be written as

$$F^{CE} = a_k(n) b_1 [1 - \frac{b_2^2}{(n - \bar{n} + b_3)^2}]^{1/2} \quad (4.7)$$

$$\times \frac{\tanh(\bar{C}_n - CD) - \tanh[\bar{C}_{n_0}(n, q) - CD]}{\tanh(-CD) - \tanh[\bar{C}_{n_0}(n, q) - CD]}$$

where

$$a(n) = b_{12} + b_{13} (\exp\{-b_{14}[b_{15} + b_{16}(n - \bar{n})]^n\})$$

$$\times \sin [b_{17}(n - \bar{n}) + b_{18}] \quad (4.8)$$

$$\dot{n}_o(n, q) = n_o(\bar{n}, 1) \exp[-b_5(n - \bar{n} + b_6)^2 - 1.43(q-1)^2] \quad (4.9)$$

with the quantities $b_1, b_2, b_3, b_5, b_6, b_{12} \dots b_{18}$, n , \bar{n} , D and $n_o(\bar{n}, 1)$ denoting muscle-specific constants. The square-root term in equation (4.7) has that form due to the fact that some skeletal muscles are of the bipennate type. For a complete discussion of this type of muscle, the reader is referred to Appendix A of [13]. The functions $a(t)$ and $q(t)$ are each defined by differential equations which depend upon the previously discussed control parameters θ and ϕ . It will be recalled that the function θ represents the relative number of active muscle fibres (cf. equation (2.18)) and must satisfy the constraint

$$0 \leq \theta \leq 1. \quad (4.10)$$

The function ϕ represents the "average" relative muscle stimulation frequency. Considering equations (2.14) and (2.19), it is clear that ϕ must satisfy a condition similar to (4.10); i.e.,

$$0 \leq \phi \leq 1. \quad (4.11)$$

Parenthetically, the active state function $q(t)$ is a quantity related to the relative number of Calcium ions (Ca^{++}) bound to the troponin molecules of the muscle cells, and is a direct measure of the number of cross-bridges attached to the actin filament [18]. The function $a(t)$

(mode-of-activity) relates the activity of the total muscle to the activity of the individual fibres. Hatze [3] has shown that the equations to describe the functions $q(t)$ and $a(t)$ may be written as

$$\dot{q} = m\{\phi(\phi, n) - q\} \quad (4.12)$$

and

$$\dot{a} = m\{\theta[\phi(\phi, n) - 0.005] + 0.005 - a\} \quad (4.13)$$

where

$$\phi(\phi, n) = 0.5025 + 0.4975 \tanh [19.3 \{\exp(-b_{19}[n-\bar{n}-b_{20}]^2) \\ (4.14)$$

$$x\{\phi-b_{21}[\exp(-b_{22}(n-\bar{n})] - b_{23}\}]$$

and the quantities m , b_{19} , ..., b_{23} are again muscle specific constants.

Having described the important relations in the model, one is now in a position to write the equations that comprise the muscle-mechanical equations of motion.

Re-inserting the subscripts k into the equations, defining a new set of variables by

$$\left. \begin{array}{l} x_{20+k} = q_k \\ x_{30+k} = a_k \end{array} \right\} k=1, \dots, 10 \quad (4.15)$$

and noting that the constants D_k of equation (4.7) are all approximately zero for this specific problem [3], the following equation is obtained (considering equation (4.2)) by solving equation (4.7) for x_{10+k} [= η_k (cf. eq'n (4.6))]:

$$\dot{x}_{10+k} = \frac{1}{C_k} \tanh^{-1} \left(\left[\frac{F_k^{SE}(x_i, x_{10+k})}{b_{1k} k(x_{10+k}) [1 - b_{2k}^2 / (x_{10+k} + b_{3k})^2]^{1/2}} x_{30+k} \right. \right. \\ \left. \left. \times \tanh(b_{4k} \exp[-b_{5k} (x_{10+k} - b_{6k})^2] - 1.43 (x_{20+k} - 1)^2] \right] \right) \quad (4.16)$$

for $k = 1, \dots, 10$, $F_k^{SE}(x_i, x_{10+k}) = F_k^{SE}(\lambda_k(x_i), n_k)$ being given by (4.5), the constant b_{4k} being equal to $-(n_0(\bar{n}, 1))_k$ (cf. (4.9)) and of course the function $k(x_{10+k})$ being given by (cf. (4.8))

$$k(x_{10+k}) = b_{12k} + b_{13k} \left(\exp \{-b_{14k} [b_{15k} + b_{16k} (x_{10+k})^{n_k}] \} \right. \\ \left. \times \sin [b_{17k} x_{10+k} + b_{18k}] \right). \quad (4.17)$$

Equations (4.12) and (4.13) become

$$\dot{x}_{20+k} = m_k \{\phi_k - x_{20+k}\} \quad (4.18)$$

and

$$\dot{x}_{30+k} = m_k \{\theta_k [\phi_k - 0.005] + 0.005 - x_{30+k}\} \quad (4.19)$$

wherein

$$\phi_k = 0.5025 + 0.4975 \tanh \left[19.3 \{\exp(-b_{19k} [x_{10+k} - b_{20k}]^2) \right. \\ \left. \times \{\phi_k - b_{21k} [\exp(-b_{22k} x_{10+k})] - b_{23k}\} \} \right]. \quad (4.20)$$

Summarizing, the muscle forces can be determined from the solution of a system of 30 (in general) first order differential equations of the form

$$x_{10+k} = f^{10+k} (x_i, x_{10+k}, x_{20+k}, x_{30+k}) \quad \left. \begin{array}{l} \\ \\ \end{array} \right\} \quad (4.21)$$

$$x_{20+k} = f^{20+k} (x_{10+k}, x_{20+k}, \theta_k) \quad \left. \begin{array}{l} \\ \\ \end{array} \right\} i, k=1, \dots, 10 \quad (4.22)$$

$$x_{30+k} = f^{30+k} (x_{10+k}, x_{30+k}, \theta_k, \phi_k) \quad \left. \begin{array}{l} \\ \\ \end{array} \right\} \quad (4.23)$$

together with the dynamical equations of the form

$$x_j = f^j (x_i, x_{10+k}, \theta_k, \phi_k) \quad i, j, k=1, \dots, 10 \quad (4.24)$$

which have been dealt with in an early chapter. Implicit in the assumption of a solution existing for equations (4.21) to (4.24) is that the initial conditions

$$x_\ell(t_0) = x_\ell^0 \quad (\ell=1, \dots, 40)$$

be given.

At first glance, it appears that there are a horrendously large number of equations to deal with. This can be placed into perspective in the following way: In the general model there are five angular variables, and together with their derivatives, produce ten equations in (4.24). It was shown earlier that this can be reduced to four equations by treating the two legs separately. In separating the two legs, the number of muscles is reduced from ten to five. For each of the five muscles there corresponds three equations of the form (4.21) to (4.23), making 15 differential equations to be solved simultaneously.

with the four dynamical equations.

As in the case of the dynamics, where the variables x_i were transformed to the variables z_j (cf. (3.22)), so can a similar transformation reduce the apparent complexity of equations (4.21) to (4.23). In a manner similar to that for the introduction of the variables δ and ϵ (equations (3.16) and (3.17)), the new variable ρ is defined by

$$\rho = \begin{cases} 0, & \text{for left leg} \\ 5, & \text{for right leg.} \end{cases} \quad (4.25)$$

Then, the following transformation is performed. Define the variables

$$\left. \begin{array}{l} z_{4+j} = x_{10+\rho+j} \\ z_{9+j} = x_{20+\rho+j} \\ z_{14+j} = x_{30+\rho+j} \end{array} \right\} \quad j=1, \dots, 5 \quad (4.26)$$

and the subscripts of the constants of the foregoing equations is such that $k = \rho + j$. Re-ordering and rewriting equations (4.21) to (4.24), one is left with a system of equations which take the form:

$$\begin{aligned} \dot{z}_j &= g^j (z_k, z_{4+i}, \theta_i, \phi_i) \\ z_{4+i} &= g^{4+i} (z_k, z_{4+i}, z_{9+i}, z_{14+i}) \\ z_{9+i} &= g^{9+i} (z_{4+i}, z_{9+i}, \theta_i, \phi_i) \\ z_{14+i} &= g^{14+i} (z_{4+i}, z_{14+i}, \theta_i, \phi_i) \end{aligned} \quad (4.27)$$

where $j, k = 1, \dots, 4$ and $i = 1, \dots, 5$. Clearly the functions $\phi(\cdot, \dots, \cdot)$ are simply shorthand notations for the right-hand sides of equations (3.23), (4.16), (4.18) and (4.19).

Again, a summary of the results obtained so far may be found in Appendix A. The reader is also reminded that the muscle specific constants, the functions $F_{3,4}^{PE}$ and $F_{8,9}^{PE}$, and the starting values $z_r(t_0)$ ($r = 1, \dots, 19$) have yet to be determined and must be done experimentally. The treatment of experimentally determined quantities will be left for another chapter.

4.2 Constraints on M. Gastrocnemius

In section 3.4 of Chapter III, consideration was given to the moments about the ankle, but this was done without regard to the origin of these moments. Some discussion will now be presented on this matter.

During the restraint portion of the walking cycle, and specifically during the short time immediately after heel strike, the foot is restrained in its motion by the tibialis anterior muscle. Since this muscle has not been included in the present model, no account for its action need be considered.

However, during the support and deploy portions of the cycle, the ankle moment $M_a^{\delta+1}$ given by equations (3.39) is positive (in the sense shown in Figure 3.2). This moment

must result from forces within the shank of the leg. As indicated by Figure 2.2, although it was not discussed at the time, the ankle moment is generated by the gastrocnemius muscle, and of course the 'passive' joint torques at the ankle. The moment generated by the gastrocnemius on the ankle joint can be evaluated in the following way:

Let d_5^α and d_{10}^α denote the normal distance from the line of action of muscle groups (5) and (10) to the respective centers of rotation of the ankle joint. These, of course, must be experimentally determined along with the other d_j 's ($j = 1, \dots, 10$) as functions of the angular variables and, in this case, the ankle angle σ (the latter being shown in Figure 3.1).

Clearly then, the ankle moment $M_\alpha^{\delta+1}$ may be written as

$$M_\alpha^{\delta+1} = u_{\alpha, \delta+1}^P + d_{\rho+5}^\alpha F_{\rho+5}^{SE} (\lambda_{\rho+5}(z_1, z_2), z_9) + M_{ta}^{\delta+1} \quad (4.28)$$

where $u_{\alpha, \delta+1}^P$ is the passive ankle joint torque (to be experimentally found) and $M_{ta}^{\delta+1} \leq 0$ is some negative torque (which may or may not be present) generated by the tibialis anterior. Rearranging terms, equation (4.28) may be rewritten to reflect a constraint on the force level in the calf muscle; i.e.,

$$F_{\rho+5}^{SE} (\lambda_{\rho+5}, z_9) \geq (M_\alpha^{\delta+1} - u_{\alpha, \delta+1}^P) / d_{\rho+5}^\alpha \quad (4.29)$$

noting that the actual value of $M_{ta}^{\delta+1}$ is of no particular interest at present. By constraining the level of $F_{\rho+5}^{SE} = F_{\rho+5}^{CE}$, the rest of the force levels in the leg musculature will automatically be taken care of by the dynamics of the system.

4.3 Further Simplification of Musculo-mechanical Equations

A further simplification of the model can be made to reduce the number of differential equations.

Letting

$$z_{14+j} = \theta_{\rho+j} z_{9+j} + 0.005 \quad (4.30)$$

equation (4.19) becomes redundant. That this is justifiable has been demonstrated by Hatze [13,26]. This simplification reduces the number of differential equations needed for solution and hence reflects as a saving in computation costs.

CHAPTER V

ENERGETICS OF MUSCLES

5.1 Introduction

It was mentioned in Chapter I that before a satisfactory simulation of walking can be achieved, it is important that a justifiable optimality criterion be found. Since it has been generally accepted that an energy optimal situation is sought, the minimization depends upon finding an expression which allows evaluation of the instantaneous energy consumption rate or power consumption.

It can easily be seen that, in general, the energy rate is simply the sum of the power expenditures of each of the muscles. For this reason, then, the derivations will proceed in a general fashion to demonstrate the power consumption of a muscle and this will then be used to evaluate the energetics of the present problem.

The study of muscle energetics is probably one of the most widely covered topics in the area of muscle biology. Recently Mommaerts [19] and Woledge [20] have published extensive reviews of the energy considerations of muscular contractions. Based on these reviews and the works found

therein, Hatze and Buys [21] have been able to determine the explicit nature of the energy consumption of skeletal muscle.

5.2 Derivation of Muscular Energy Expenditure

Let the energy consumption rate of the k -th muscle be denoted by $\xi^k(x_i, \phi_k, \theta_k)$ so that, in keeping with previous notation,

$$\xi(x, \phi, \theta) = \sum_{k=1}^m \xi^k(x_i, \phi_k, \theta_k) \quad (5.1)$$

for a system of m muscles. Mommaerts demonstrated that the power output of the contractile element of the muscle could be represented as [19]

$$\xi^{CE} = A + H + s + w \quad (5.2)$$

where A is the activation heat rate, H is the maintenance (tension-time) heat rate, s is the shortening (or lengthening) heat rate and w is the rate of work done by (or on) the muscle. If the entire muscle is to be considered, account must be taken for the dissipation rate of the damping component of the parallel elastic element. If this is denoted by r , then for the k -th muscle one immediately sees that

$$\xi^k = A_k + H_k + s_k + w_k + r_k \quad (5.3)$$

and since ξ^k is a function of the co-ordinates x_i and the control variables ϕ_k and θ_k , the dependence of the terms on the right-hand side of (5.3) upon x_i , ϕ_k and θ_k must be shown.

It will be noted that the biochemical details discussed in [19] will not be presented here, and only the pertinent results will be shown. The motivation for each will be given very briefly.

The activation heat rate A_k will be discussed first. The heat of activation is related to the excitation process and therefore should be a function of the frequency of stimulation and the mass of muscle activated. In general then

$$\dot{A}_k = A'_k(v_k) v_k \theta_k G_k \quad (5.4)$$

where $A'_k(v_k)$ is the activation heat per stimulus per unit muscle mass, v_k is the stimulus frequency, θ_k is the control parameter defined in (2.18) and G_k is the mass of the k -th muscle. From Figure 6 of [22], the function $A'_k(v_k)$ can be found to satisfy

$$\dot{A}'_k(v_k) = \bar{A}'_k \{1 - \exp(-k_{ak}(v_k^{-1} + c_{ak}))\}. \quad (5.5)$$

The values of the constants appearing in (5.5) are (for the rat soleus muscle at 27°C): $\bar{A}' = 0.5923$ joule/kg, $k_a = 19.2$ and $c_a = 0.0125$. Gibbs and Gibson [22] also give a Q_{10} value (which is the ratio by which the magnitude of a

process change for each 10°C rise or fall in temperature) of 3 for \bar{A}' . Hence at 37°C $\bar{A}' = 1.777 \text{ joule/kg}$. Assuming approximately the same activation heat rate for all mammalian muscles at 37°C [The author recognizes that this is a bit risky] and substituting for v_k from (2.14) one obtains for slow human muscle

$$(A_{k_s}) = 106.8 \{1 - \exp(-0.24 - 0.32/\phi_k)\} \phi_k \theta_k G_k \quad (5.6a)$$

and for fast human muscle

$$(A_{k_f}) = 177.7 \{1 - \exp(-0.24 - 0.19/\phi_k)\} \phi_k \theta_k G_k \quad (5.6b)$$

by noting that for slow muscle $v_{opt} = 60 \text{ Hz}$ and for fast muscle $v_{opt} = 100 \text{ Hz}$ (p.85 of [13]).

Next, the maintenance heat rate H_k will be considered. This energy consumption results from the cyclic interaction between the actin and myosin and should thus be proportional to the active state, the length of the CE and the mass of muscle active. It has been found that H is time-independent [23]. Hence

$$H_k = \bar{h}_k f(n_k - \bar{n}_k) q_k \theta_k G_k \quad (5.7)$$

where \bar{h}_k is a constant, $f(n_k - \bar{n}_k)$ is a (as yet unknown) length-dependent function, q_k is the active state, and θ_k and G_k are defined as before. Equation (5.7) may be rewritten using the state variables introduced in Chapter IV

as

$$\dot{H}_k = \bar{h}_k f(x_{10+k}) x_{20+k} \theta_k G_k. \quad (5.8)$$

According to Hatze [13], at maximum stimulation rate and at optimum length $\bar{h}_k = 0.82 \bar{A}'_k$ (approximately) where

$$\bar{A}'_k = \bar{A}'_k v_{opt} \quad (5.9)$$

and is equal to 177.7 W/kg for fast muscles and 106.8 W/kg for slow ones. Hence

$$\bar{h}_k = \begin{cases} 145.7, & \text{for fast muscles} \\ 87.6, & \text{for slow muscles} \end{cases} \quad (5.10)$$

The function $f(x_{10+k})$ has been constructed [13] from an analysis of the experimental data of Gibbs and Gibson [22] and after substitution, equation (5.8) becomes

$$\dot{H}_k = \bar{h}_k (1 - 5.86 (x_{10+k}/\bar{n}_k)^2) x_{20+k} \theta_k G_k \quad (5.11)$$

for $-0.3 \leq x_{10+k}/\bar{n}_k \leq 0.3$

An analysis performed by Hatze and Buys [21] based on the results of A.V. Hill [23] demonstrates that the heat rate of shortening (or lengthening) (the biochemical cause of this type of heat production is unclear and is itself a controversial topic (cf. [24])) may be expressed as

$$\dot{s}_k = -c_k \theta_k (x_{20+k})^{0.72} k(x_{10+k}) x_{10+k} \quad (5.12)$$

where $k(x_{10+k})$ is the length tension relationship given by

equation (4.17), c_k is a muscle specific constant and the remaining factors are as defined earlier (\dot{x}_{10+k} is given by (4.16)). It may be of interest to note that in lengthening $\dot{x}_{10+k} > 0$, $k(x_{10+k}) > 0$ so that for $\theta_k(x_{20+k})^{0.72} \neq 0$, $s_k < 0$ and hence the muscle absorbs heat. This has in fact been demonstrated by Hill [23].

The work rate can be simply stated as

$$\dot{w}_k = -\dot{x}_{10+k} F_k^{SE}(x_i, x_{10+k}) \quad (5.13)$$

where $F_k^{SE}(x_i, x_{10+k})$ is given by (5.5) and the heat dissipation rate in the parallel elastic element as

$$\dot{r}_k = |\lambda_k(x_i, \dot{x}_i)| F_k^{PE}(x_i). \quad (5.14)$$

This last equation requires some comments since it was noted earlier that $F_k^{PE}(x_i)$ cannot be evaluated for all the muscles. It is anticipated, however, that neglecting the terms \dot{r}_j for $j = 1, 2, 5, 6, 7, 10$ will not severely hinder the optimization, especially when one considers that Chow and Jacobson achieved reasonable results by considering only the component \dot{w} of the energy consumption.

One further point is of importance. Since we are dealing with an optimization using $\xi(x, \phi, \theta)$ as an objective function, and the absolute value of

$$\int_{t_0}^{t_1} \xi(x, \phi, \theta) dt$$

is of no major consequence, the constant factors in the terms of ξ^k are important only in that they give the relative contributions of the different components of energy consumption.

The total energy consumption of the musculature for the present problem can now be easily written by the substitution of equations (5.6), (5.11), (5.12), (5.13) and (5.14) into (5.3) and subsequently into (5.1) to produce the integrand of the objective functional (2.24); i.e.

$$\begin{aligned} \xi^k(x_i, \phi_k, \theta_k) = & \bar{a}_{1k} \left[(1 - e^{-0.24 - \bar{a}_{2k}/\phi_k}) \phi_k + (0.82 - 4.81 \left(\frac{x_{10+k}}{\bar{n}_k} \right)^2 \right. \\ & \left. \cdot x_{20+k} \right] \theta_k G_k - \left[c_k \theta_k (x_{20+k})^{0.72} k(x_{10+k}) \right. \\ & \left. + F_k^{SE}(x_i, x_{10+k}) \right] \dot{x}_{10+k} \\ & + |\lambda_k(x_i, \dot{x}_i)| F_k^{PE}(x_i) \end{aligned} \quad (5.15)$$

$$\text{where } (\bar{a}_{1k}, \bar{a}_{2k}) = \begin{cases} (177.7, 0.19), \text{ for fast muscle} \\ (106.8, 0.32), \text{ for slow muscle} \end{cases} \quad (5.16)$$

Making the variable transformations (3.22) and (4.26) as before, (5.15) becomes

$$\begin{aligned} \xi^{p+j}(z_i, \phi_{p+j}, \theta_{p+j}) = & \bar{a}_{1, p+j} \left[(1 - \exp(-0.24 - \bar{a}_{2, p+j}/\phi_{p+j})) \phi_{p+j} \right. \\ & \left. + (0.82 - 4.81(z_{4+j}/\bar{n}_{p+j})^2) z_{9+j} \right] \theta_{p+j} G_{p+j} \end{aligned}$$

$$\begin{aligned}
 & -[c_{\rho+j} \theta_{\rho+j} (z_{9+j})^{0.72} k(z_{4+j}) \\
 & + F_{\rho+j}^{\text{SE}}(z_i, z_{4+j})] \dot{z}_{4+j} \\
 & + |\lambda_{\rho+j}(z_i, \dot{z}_i)| F_{\rho+j}^{\text{PE}}(z_i).
 \end{aligned} \tag{5.17}$$

This then completes the derivation of the explicit forms of all of the equations necessary for the simulation up to the point of application of Pontryagin's maximum principle. The latter will be the subject of the next chapter.

CHAPTER VI

COMPUTER SIMULATION

6.1 Introduction

As discussed earlier, the differential equations derived thus far, and summarized in Appendix A, were implemented on a CDC 6400 digital computer using the high level continuous simulation language GASP IV. The latter uses a fourth order Runge-Kutta numerical integration routine to solve the differential equations. This of course requires starting values for the variables. Determination of these starting values will be discussed later in this chapter. Implementation of Pontryagin's maximum principle is tedious but straight-forward and is discussed first.

6.2 Pontryagin's Maximum Principle

6.2.1 Introduction

The equations, whose functional form was shown in Chapter II, can now be written explicitly since the set of differential equations has been derived. Because of the length and complexity of the equations it is deemed unnecessary to rewrite the equations of the previous chapters. It is, however, necessary to define the conjugate

6.2.2 Specification of Equations

Define the function Π (cf. equation (2.34)) as

$$\Pi(\psi, z, \gamma) = \sum_{\alpha=0}^{14} \psi_\alpha z_\alpha(z, \gamma) \quad (6.1)$$

where $\psi = (\psi_0, \psi_1, \dots, \psi_{14})$

$$z = (z_1, \dots, z_{14})$$

$$\gamma = (\phi_1, \dots, \phi_5, \theta_1, \dots, \theta_5)$$

and the functions $z_\alpha(z, \gamma)$ are given by the first three equations of (4.27). The function $z_0(z, \gamma)$ is simply (cf. equation (2.29))

$$\dot{z}_0 = \xi(z, \gamma) \quad (6.2)$$

Then the conjugate variables can be defined by the following equations:

$$\frac{d}{dt} \psi_i = - \sum_{\alpha=0}^{14} \frac{\partial z_\alpha(z, \gamma)}{\partial z_i} \psi_\alpha \quad (6.3)$$

for $i = 0, 1, \dots, 14$. It is interesting to note that although the form of equation (6.3) is relatively simple, each of the individual equations is exceptionally long. For example, to write out the equation for $(d\psi_1/dt)$ would require approximately ten single spaced type-written pages to complete. These equations have therefore not been included in this thesis. Clearly, they had, however, to be coded for the computation. This was not as difficult as may

appear in view of the above. Through extensive use of functional blocks, the amount of coding could greatly be reduced. The equations contain large numbers of repeated groups and use could be made of this fact to reduce repetitive calculations.

The equations may now be written in closed form as (cf. equations (2.36))

$$\frac{d}{dt} z_i = \frac{\partial H}{\partial \psi_i} \quad (i = 0, 1, \dots, 14) \quad (6.4)$$

$$\frac{d}{dt} \psi_i = - \frac{\partial H}{\partial z_i}$$

where the function H is defined as:

$$H(\psi, z) = \sup_{\gamma \in \Omega} \pi(\psi, z, \gamma) \quad (6.5)$$

Ω being the hyperplane defined by $0 \leq \phi_i, \theta_i \leq 1$.

Having defined the function π and the differential equations defining the variables ψ_i , one is in a position to discuss the philosophy of the numerical implementation of equations (6.4) and (6.5). This will be treated next.

6.2.3 Numerical Implementation

Assuming for the moment that the initial values of the variables have been determined (this will be dealt with in Section 6.3.3) the procedure for determining the variables z , ψ , ϕ , θ is outlined in the flow chart of Figure 6.1.

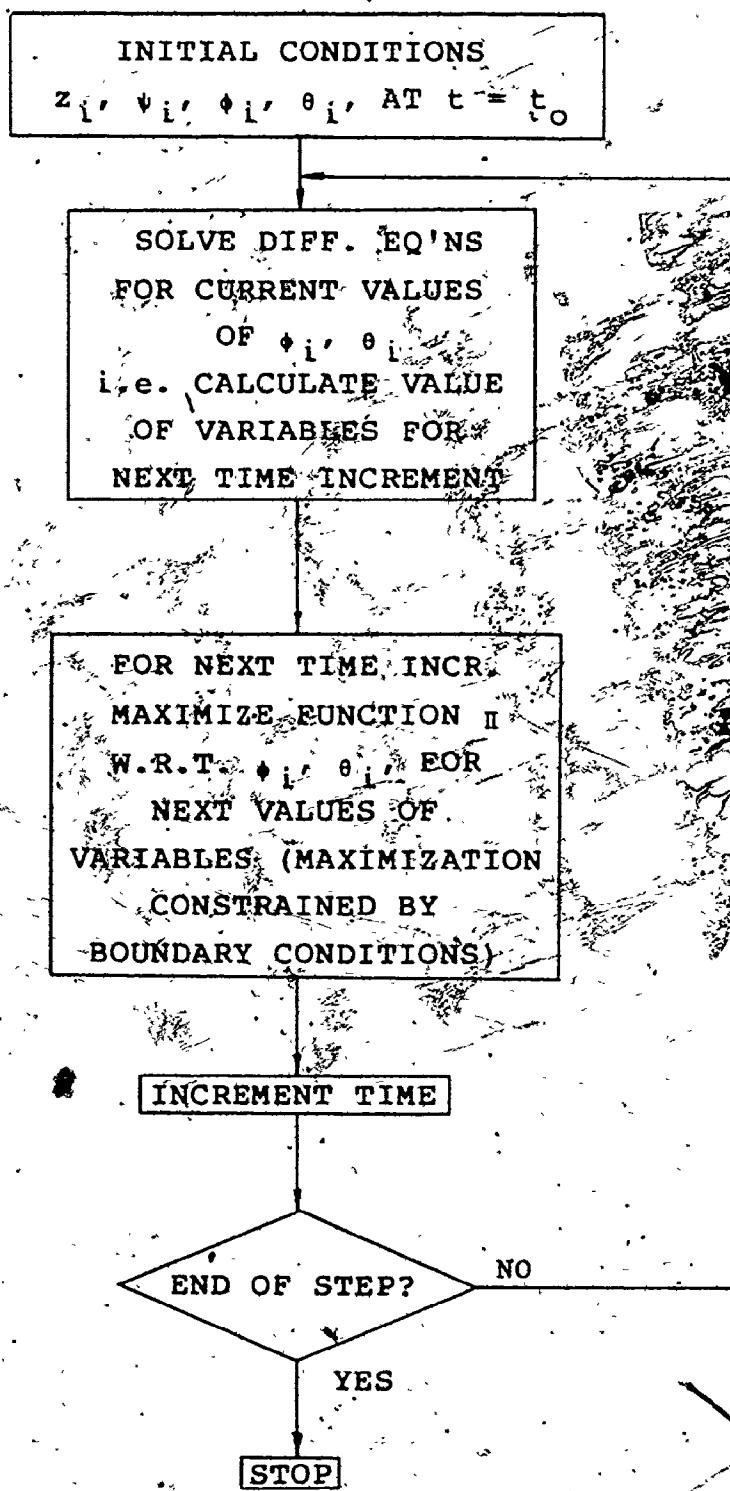


FIGURE 6.1 FLOW CHART OF NUMERICAL PROCEDURE

Although the problem was initially stated to find the minimum for the functional

$$J = \int_{t_0}^{t_1} \xi dt$$

which implies a minimization after the integration, with the application of Pontryagin's maximum principle the problem could be solved in such a way that an optimization could be carried out at each incremental point in time.

The integration of the differential equations was carried out using a fourth order Runge-Kutta numerical integration procedure which formed part of the GASP IV simulation language.

6.2.4 Optimization

The optimization was implemented using the OPTISEP optimization subroutine package developed at McMaster University. This package allows inclusion of the equality and inequality constraints dictated by the present problem.

A direct search technique* was used to find the optima. This was a relatively fast procedure since the values of the independent variables (in this case the control parameters ϕ_i and θ_i) do not change quickly between time increments.

*Davidon-Fletcher-Powell method

Hence it was necessary only to search in the immediate neighbourhood of the values found to be optimum at the last increment in time.

6.2.5. Comment on Numerical Procedure

Upon careful examination, it can be seen that the chosen procedure is not precisely correct due to discretization. This becomes evident when one considers that the numerical integration and optimization are not actually being carried out simultaneously. The problem arises from the fact that at each point in time the dynamical variables depend upon the control parameters and likewise the objective function of the optimization depends upon the dynamical variables. The procedure used is somewhat different in that the dynamical variables for the next increment in time are calculated on the basis of the current values of ϕ_i , θ_i . Then the values of ϕ_i and θ_i are updated via the optimization. Provided the time interval is kept small the error introduced is minimal. Clearly, in the limit, as the time interval becomes infinitesimal, the procedure becomes exact.

With regard to the advancement of time it is worthwhile to note that the GASP IV language allows for variable time incrementation. This is accomplished through the use of artificial "state events". That is, GASP IV

allows for decisions to be made when state variables, or combinations thereof, cross certain limiting values. The time increment is continually adjusted so that the crossing does not occur in the middle of a time increment. This facility was of value in the present problem due to the presence of the arctanh function in equation (4.16). The argument of \tanh^{-1} must remain in the open interval (-1,+1). If the time increment is too large, the nonlinearity of the equations can force the argument infeasible, and hence the equations become unstable. Therefore, it is necessary to check the argument before the time increment is accepted. In this way the time step is allowed to remain as large as possible (up to a limit of 0.01 sec.) while ensuring the existence of \dot{z}_{4+j} ($j = 1, \dots, 5$).

Appendix B shows a listing of the Fortran IV program used to implement the numerical procedures outlined here. Appendices C and D contain listings of those portions of GASP IV and OPTISEP called by the various routines of the original program.

6.3 Experimentally Determined Parameters and Initial Conditions

6.3.1 Introduction

At this point it must be emphasized that the model is highly dependent upon experimentally determined data as

evidenced in the preceding Chapters. In order for the model to describe the gait of a specific individual, a considerable number of measurements must be made to obtain physical data about such quantities as limb masses, lengths, c.g.'s and the muscle^s constants discussed in Chapter IV. Since the primary concern here is not to model one individual specifically, but rather to ascertain the feasibility of such a simulation, it was decided that the required experimental data would be gathered from existing literature and the model would be run to simulate the locomotion of a "hypothetical man". All such data was checked to validate their compatibility, primarily using a scaling by body weight. The sources for the data are given during the course of the discussion.

6.3.2 Experimentally Determined Parameters

The constants and functions necessary for the model which require experimental evaluation will be presented in the order discussed previously. All units are in the MKS system with angles in radians. Our "hypothetical man" is based on the subject used by Hatze [3,13] in his time optimality problem. The reason for this is that the muscle specific constants, $b_1, b_2, \dots, b_{23}, m, n$ and C have been tabulated for the muscles of concern. The procedure for deriving these values is very involved, indeed, and beyond

the scope of this work. The interested reader is, however, referred to Appendix A of [13] for a discussion of the methods involved.

(i) Body-Specific Constants [3,6,13]

The notation follows precisely the same format as that used in Chapter II.

$$\text{Body Weight } (W) \text{ } 87.0 \text{ kg} = 853.2 \text{ N}$$

$$m_1 = 8.375 \text{ kg}$$

$$m_2 = 5.587 \text{ kg}$$

$$l_1 = 0.510 \text{ m}$$

$$l_2 = 0.475 \text{ m}$$

$$a_1 = 0.2208 \text{ m}$$

$$a_2 = 0.2061 \text{ m}$$

$$I_1 = 0.113 \text{ kg.m}^2$$

$$I_2 = 0.153 \text{ kg.m}^2$$

(6.6)

These values may now be used to determine the values of the constants found in the dynamical equations (3.23)

$$A_1 = I_1 + m_1 a_1^2 + m_2 l_1^2 = 1.902 \text{ kg.m}^2$$

$$A_2 = I_2 + m_2 a_2^2 = 0.327 \text{ kg.m}^2$$

$$C_1 = m_1 a_1 + m_2 l_1 = 4.698 \text{ kg.m}$$

$$C_2 = m_2 a_2 = 1.152 \text{ kg.m}$$

$$C_3 = C_2 l_1 = 0.587 \text{ kg.m}^2$$

$$d_1^\delta = 0.207 \text{ m} \quad (6.7)$$

$$d_2^\delta = 0.245 \text{ m}$$

(ii) Hip and Ankle Trajectories

It was necessary to scale these values from the corresponding ones of [6] since the subject was shorter than the present subject, i.e. the sum of the lengths of the lower limb segments differed.

The horizontal location of the hip at some time t is given by: (recall that a prescribed trajectory has been applied to the hip so that the position may be described by a constant forward velocity and a vertical velocity described by a double period sinusoid; see page 36)

$$h(t) = h_0 t + h(t_0) \quad (6.8)$$

where $h'_0 = 0.857$ m/sec

and $h(t_0) = -0.396$ m being the hip position at the beginning of the restraint phase (i.e. heel strike).

The vertical position of the hip at time t has been chosen as

$$v(t) = v_0 - \beta_1 \sin \frac{4\pi}{\tau} (t + \beta_2 \tau) \quad (6.9)$$

where $\beta_1 = 0.0370$ m

$\beta_2 = 0.525$

$\tau = 2.0$ sec (period of double step)

and $v_0 = 0.995$ m, being the mean hip height above the ground. The general form of Chow and Jacobson's data has been confirmed by several authors. A good account is given in [27].

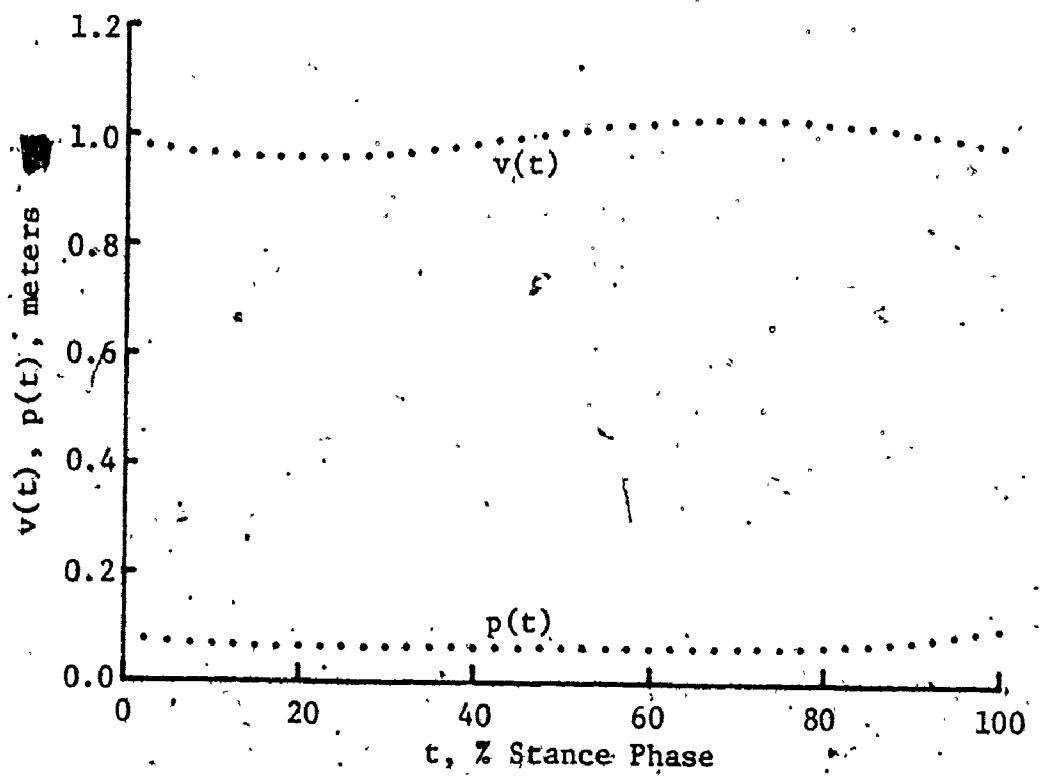
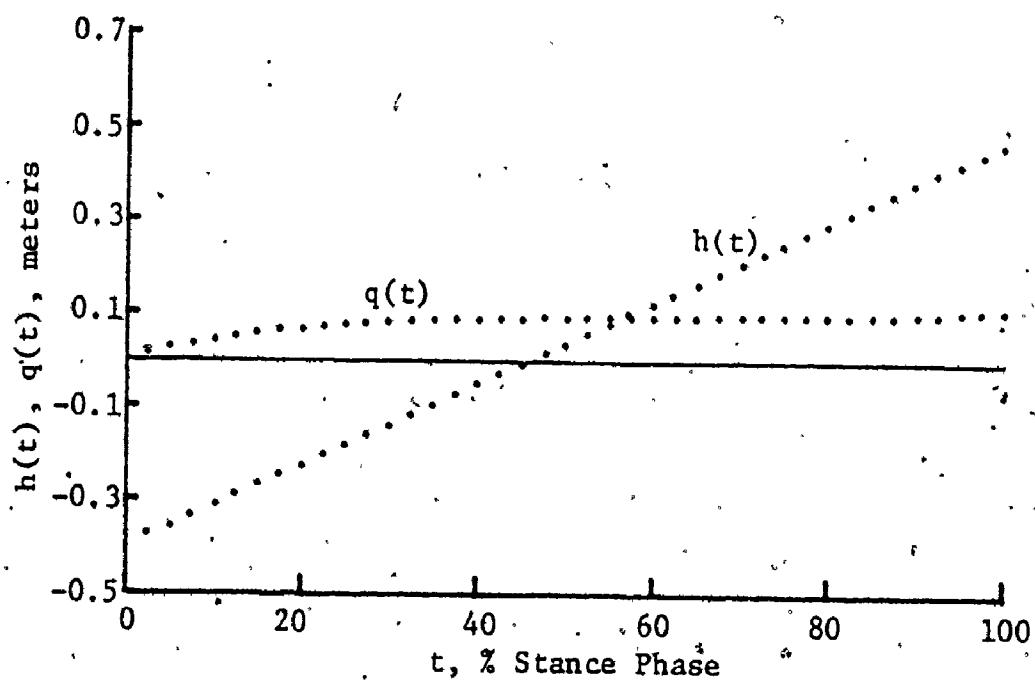
The equations given by Chow and Jacobson for the trajectory of the ankle joint have been found to be in error. These equations lack the basic requirement of continuity. Hence curves of similar construction have been fitted to graphical data presented in Bresler and Frankel [28]. During the stance phase (ankle trajectory need not be of concern during the swing phase as this is determined by the dynamics) the horizontal ankle location is given by (in meters)

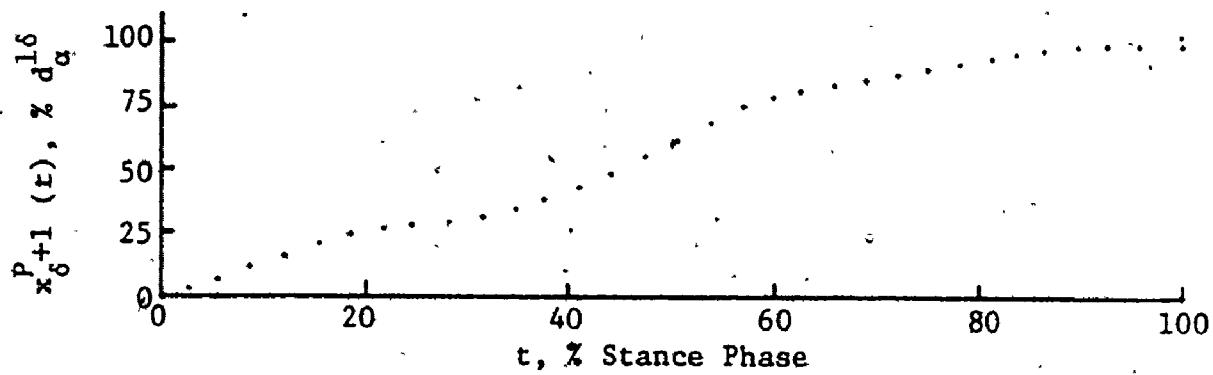
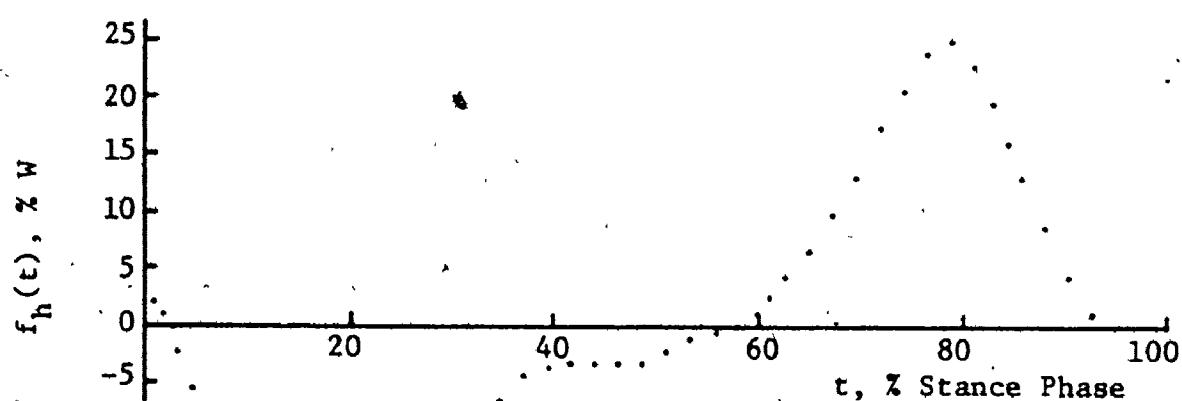
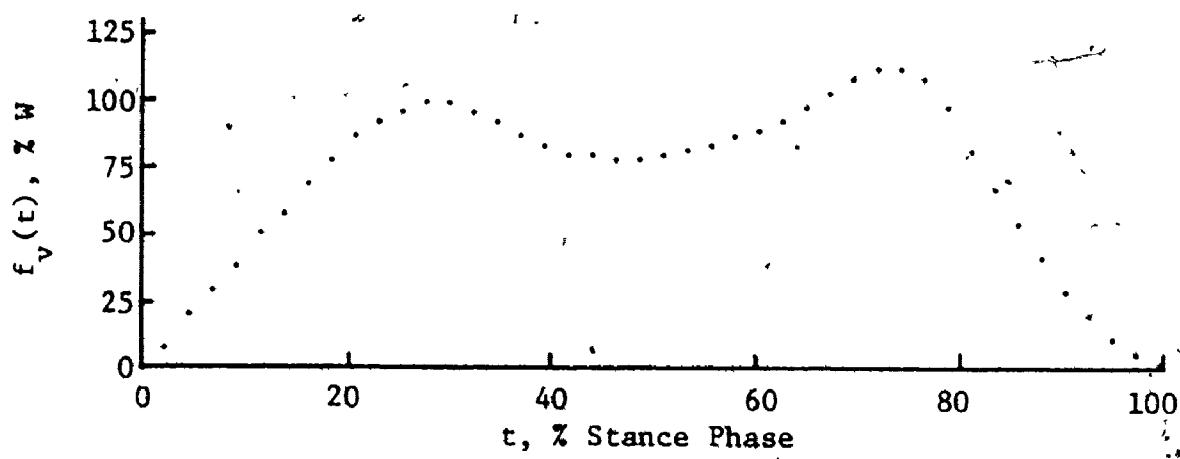
$$x_{\delta+1}^A = q(t) = q_0 + 0.095(1 - e^{-6.5t}), \quad 0.0 \leq t \leq 1.0 \quad (6.10)$$

where $q_0 = 1.71(n-1)$ for the n -th step and represents the ankle position at heel strike. For the vertical position we have

$$y_{\delta+1}^A = p(t) = \begin{cases} 0.064 + 2.852(t-0.3)^4 & , \quad 0.0 \leq t \leq 0.3 \\ 0.064 & , \quad 0.3 \leq t \leq 0.55 \\ 0.207 \sin(0.315 + 5.109(t-0.55)^4) & , \quad 0.55 \leq t \leq 1.0 \end{cases} \quad (6.11)$$

Equations (6.10) and (6.11) are shown graphically in Figure 6.2 together with the hip trajectory (6.8) and (6.9).





the lengths and velocities of shortening of the muscles, and the forces generated across the parallel elastic elements, and the moment arm functions, as functions of the angular variables and their derivatives, except F_3^{PE} and F_4^{PE} which are functions of muscle lengths (recall that $\dot{z}_1 = z_3$ and $\dot{z}_2 = z_4$).

$$u_1^P = 0.800 e^{-3.41z_1} + 0.084 e^{-15.0z_1} - 0.753 e^{2.55z_1} \\ - (7.9 e^{-272z_1} + 0.09 e^{1.8z_1}) \dot{z}_3 \quad (6.12)$$

$$u_2^P = -1.25 \times 10^{-7} e^{8.5z_2} + 6.30 e^{-2.9z_2} + 20.1 e^{-16.1z_2} \\ - 2.10 - (0.300 e^{1.02z_2} + 1.85 e^{-3.43z_2}) \dot{z}_4$$

$$\lambda_1 = 0.287 - 0.0497 z_1$$

$$\lambda_2 = 0.300 + 0.033 z_2$$

$$\lambda_3 = 0.517 + 0.045 \cos(1.128z_1 + 0.748) + 0.033z_2$$

$$\lambda_4 = 0.483 - 0.062 \cos(1.047z_1 + 0.838) \quad (6.13) \\ + 0.07 \cos(1.076z_2 + 0.280)$$

$$\lambda_5 = 0.088 + 0.019 \cos(1.160z_2 + 0.464)$$

$$\begin{aligned}
 \lambda_1 &= -0.0497z_3 \\
 \lambda_2 &= 0.033z_4 \\
 \lambda_3 &= -0.051z_3 \sin(1.128z_1 + 0.748) + 0.033z_4 \\
 \lambda_4 &= 0.065z_3 \sin(1.047z_1 + 0.838) \\
 &\quad - 0.075z_4 \sin(1.076z_2 + 0.280) \\
 \lambda_5 &= -0.022z_4 \sin(1.160z_2 + 0.464)
 \end{aligned} \tag{6.14}$$

$$\begin{aligned}
 F_3^{\text{PE}} &= 5.393 (e^{90.4(\lambda_3 - 0.58)} - 1) + 257.1 \lambda_3 \\
 F_4^{\text{PE}} &= 64.7 (e^{23.95(\lambda_4 - 0.48)} - 1) \\
 &\quad + 0.0068 (e^{239.8(\lambda_4 - 0.53)} - 1) + 378.0 \lambda_4
 \end{aligned} \tag{6.15}$$

$$\begin{aligned}
 d_1 &= 0.024 + 0.0188z_1 \\
 d_2 &= 0.03 e^{-4.33(0.17-z)^2} + 0.036 \\
 d_3 &= 0.052 \cos(z_1 - 0.63) - 0.002 \\
 d_4^1 &= 0.037 \cos(1.309z_1 - 0.916) + 0.026 \\
 d_4^2 &= 0.058(z_2 + 0.685)^2 e^{-1.187z_2} \\
 d_5 &= 0.055 \\
 d_5^a &= 0.0534
 \end{aligned} \tag{6.16}$$

(v) Muscle-Specific Constants (Dynamics)

The muscle-specific constants of equations (4.5), (4.16), (4.17) and (4.18) which were taken from [3] are tabulated in Table 6.1. All values are in the MKS system of units and hence compatible with other experimental data. As indicated earlier, an account of the procedure for determining these values is given in [13].

(vi) Muscle-Specific Constants (Energetics)

In Chapter V it was shown that the constants $\bar{a}_{1,j}$ and $\bar{a}_{2,j}$ have the respective values 177.7 and 0.19 for fast muscles and 106.8 and 0.32 for slow muscles. In the present model, muscle groups 2, 4 and 5 (vastus, hamstring and gastrocnemius) are comprised (primarily) of fast twitch fibres and groups 1 and 3 (iliopsoas and rectus femoris) of slow twitch fibres. It remains to indicate the values for the constants c_j , \bar{n}_j , and G_j for each of the muscles.

It can easily be shown that the constant c_k of equation (5.12) is equivalent to the constant a in equation (15) of [21]. In the same paper it was shown that

$$-(\bar{a}/\bar{F})^2 \bar{\lambda}_o = (\bar{g} + \bar{h})G \quad (6.17)$$

using their notation, where \bar{F} is the maximum tetanic

Table 6.1 Muscle Specific Constants (Dynamics)

j=	1	2	3	4	5
$b_{1,j}$	12250	13683	2780.7	3900	1880
$b_{2,j}$	0	0.07	0.02394	0	0
$b_{3,j}$	0	0.14	0.07	0	0
$b_{4,j}$	3.6327	2.37	3.843	2.31	2.352
$b_{5,j}$	670.3	3711.1	875.5	47.04	2970.9
$b_{6,j}$	0	-0.00204	0.0035	0	0
$b_{7,j}$	3381.4	3172.7	717.7	1076.4	168.12
$b_{8,j}$	170.1	53.23	27.72	31.9	250
$b_{9,j}$	0	0.14	0.07	0	0
$b_{10,j}$	0	0.0049	0.0005731	0	0
$b_{11,j}$	0.275	0.19	0.429	0.487	0.087
$b_{12,j}$	0	0.5	0.5	0.32	0
$b_{13,j}$	1	0.914	0.544	0.71	1
$b_{14,j}$	846.9	0.3267	33.143	3.682	5000.0
$b_{15,j}$	0	0.7484	0	0	0
$b_{16,j}$	1	123.32	1	1	1
$b_{17,j}$	0	93.824	47.14	12.325	0
$b_{18,j}$	$\pi/2$	0.67	1.122	1.28	$\pi/2$
$b_{19,j}$	437.5	2422.1	571.4	30.7	1939.1
$b_{20,j}$	0.024	0.01224	0.0175	0.0906	0.0114
$b_{21,j}$	0.1632	0.1938	0.1585	0.1884	0.1884
$b_{22,j}$	7.25	13.9	8.29	1.566	12.45
$b_{23,j}$	0.1088	0.1256	0.1088	0.1256	0.1256
m_j	32	125	32	125	125
n_j	2	2	1	1	2
\bar{C}_j	5.93	15.8	1.8477	3.3	14.7

force produced by the CE at length $\bar{\eta}$, the constant $-\lambda_0$ is equivalent to the present $n_0(\bar{\eta}, l) = b_4$, G is as used in Chapter V and denotes muscle mass. The quantity $(\bar{g} + \bar{h})$ is a muscle-specific quantity, and has been found to take the values 150 W/kg and 24.4 W/kg for slow and fast human muscle fibres respectively. Since \bar{F} has been represented here by the constant $b_{7,j}$, one may easily deduce that

$$c_j^2 = \frac{(\bar{g} + \bar{h})_j}{b_{4,j}} b_{7,j} G_j \quad (6.18)$$

The calculated values of c_j are shown in Table 6.2. The constant $\bar{\eta}_j$ is the optimum length of the contractile element (CE) (c.f. equation (4.6)). Using equation (204) of [13] combined with equation (4.9) of Chapter IV, $\bar{\eta}_j$ may be written as

$$\bar{\eta}_j = (4.29/b_{5,j})^{1/2} \quad (6.19)$$

Values for $\bar{\eta}_j$ along with values of G_j (muscle masses) are given also in Table 6.2.

This then is a complete delineation of all the necessary experimentally determined constants and functions and their respective values.

6.3.3 Initial Conditions

It was decided to begin the simulation at 15% of

Table 6.2 Muscle Specific Constants (Energetics)*

$j =$	1	2	3	4	5
$\bar{a}_{1,j}$	106.8	177.7	106.8	177.7	177.7
$\bar{a}_{2,j}$	0.32	0.19	0.32	0.19	0.19
c_j	574.0	980.3	126.8	410.7	292.4
\bar{n}_j	0.08	0.034	0.07	0.302	0.038
G_j	0.25	0.90	1.10	1.50	1.40

* All values are in the MKS system of units.

double step period measured from heel strike. At this point a great deal of useful information exists in the literature not only about the dynamics of the leg but also about the force levels in the muscles.

The starting values for the dynamical variables were taken from [6] and [27] combined with geometrical considerations. The values are as follows: (recall that double step period was chosen as 2.0 sec).

$$z_1(0.3) = 0.5711 \text{ radians}$$

$$z_2(0.3) = 0.6886 \text{ radians}$$

$$z_3(0.3) = \dot{z}_1(0.3) = -0.69 \text{ radians/sec} \quad (6.20)$$

$$z_4(0.3) = \dot{z}_2(0.3) = 0.29 \text{ radians/sec}$$

$$\dot{z}_3(0.3) = -5.89 \text{ radians/sec}^2$$

$$\dot{z}_4(0.3) = -10.19 \text{ radians/sec}^2$$

Having values for \dot{z}_3 and \dot{z}_4 at the starting point enables one to determine the values of z_5, \dots, z_9 provided some further information about the muscular forces is available. This additional information will be shown in the course of the development.

From the last two equations of (3.23) it is possible to solve for the functions R_1 and R_2 (see page 41) since B_1 , B_2 , B_3 and δ are constants once z_1 and z_2 are determined. Having determined R_1 and R_2 , the next two equations permit solution for the joint torques M_1 and M_2 ($\delta = \epsilon = 1$). These values may then be used in equations (3.20) and (3.21) to

determine u_1 and u_2 . Seireg and Arvikar [29] were able to determine the muscular loads in the leg during level walking. They publish absolute values (i.e. forces in kg.). In order, however, to enable use of the information for any weight of individual, the data was transformed into a form which provides the ratios of the muscular loads. The results are as follows (the subscripts refer to the muscle groups indicated in Figure 2.2)

$$\begin{aligned} F_3 &\doteq 0.50 F_4 \\ F_1 &\doteq 0 \\ F_5 &\doteq 0 \end{aligned} \quad (6.21)$$

where

$$\begin{aligned} F_1 &= F_1^{SE} \\ F_2 &= F_2^{SE} \\ F_3 &= F_3^{SE} + F_3^{PE} \\ F_4 &= F_4^{SE} + F_4^{PE} \\ F_5 &= F_5^{SE} \end{aligned} \quad (6.22)$$

and are the force levels in the respective muscles. The three equations of (6.21) together with the first two equations of (4.3) produce a closed system of five equations in five unknowns allowing solution for F_i^{SE} ($i = 1, \dots, 5$) provided values for u_1 , u_2 , F_3^{PE} , F_4^{PE} and d_i are available. This has in fact been shown to be the case. It is then an

easy matter to solve equation (4.5) for z_{4+j} using the values for λ_j shown in equations (6.11).

It remains then to determine starting values for the variables z_{9+j} , ϕ_j and θ_j ($j = 1, \dots, 5$) and the conjugate variables. Considering the minimum energy hypothesis presented earlier, it is reasonable to assume that once the starting values for the variables other than z_{9+j} , ϕ_j and θ_j have been determined, the latter variables must be chosen in such a way as to minimize the instantaneous value of power expenditure.

As an approximation, the assumption is made that the time derivative of the variables z_{9+j} is small (i.e. $z'_{9+j}=0$). Solving equation (4.18) provides a relationship between z_{9+j} and ϕ_j :

$$z_{9+j} = \phi_j \quad (6.23)$$

for $j = 1, \dots, 5$ where ϕ_j is given by (4.20) and is a function of ϕ_j . It can also be shown [30] that under these conditions the following holds:

$$\frac{F_j^{\text{SE}}}{F} = \theta_j(z_{9+j} - 0.005) + 0.005 \quad (6.24)$$

where F is the maximum isometric tetanic force produced by the contractile element.

With these approximations the function $\xi(z, \phi, \theta)$ may be

written as

$$\sum_{j=1}^5 \xi_j(z_{9+j}(\phi_j), \phi_j, \theta_j(\phi_j)) = \sum_{j=1}^5 \xi_j(\phi_j) \quad (6.25)$$

bearing in mind that the remaining variables have been evaluated earlier. Also since the i -th component of ξ is dependent only upon ϕ_i , minimizing ξ amounts to five independent minimizations in one variable (ϕ_i). A direct search method was implemented on a Hewlett-Packard model 9830A programmable calculator to produce the required starting values of z_{9+j} , ϕ_j and θ_j .

To find starting values for the conjugate variables ψ_i ($i = 0, 1, \dots, 14$) use is made of the cyclic nature of the present problem. At $t = t_1$, the conjugate variables are related to the objective functional by

$$\psi_i(t_1) = \frac{\partial}{\partial z_i} \int_{t_0}^{t_1} \xi(z, \phi, \theta; t) dt. \quad (6.26)$$

Also, since the physical situation is entirely cyclic, $\psi_i(t_0) = \psi_i(t_1)$. It is then a simple matter to show that at $t = t_0$

$$\psi_i(t_0) = \xi(z, \phi, \theta; t_0) / z_i \quad (6.27)$$

This then completes all the necessary derivation and evaluation of variables needed to implement the simulation. Initial values for all of the variables are presented in Table 6.3.

Table 6.3 Initial Values of Simulation Variables*

i	z_i
0	0.0
1	0.5711
2	0.6886
3	-0.6984
4	0.2971
5	-0.0165
6	-0.0016
7	0.0497
8	-0.0161
9	-0.0061
10	0.9996
11	0.8136
12	0.9944
13	0.2469
14	0.9999

j	ϕ_j	θ_j
1	0.3417	0.2806
2	0.4456	0.2882
3	0.1213	0.0629
4	0.3667	0.1415
5	0.0060	0.0001

*Values for $\psi_i(t_0)$ are calculated from $z_i(t_0)$ by the program.

Appendix E shows the echo check of the input data for GASP IV and the main program.

CHAPTER VII

COMPUTATION RESULTS

7.1 Introduction

The derivations of the previous chapters were shown to include all phases of the walking cycle - stance, deploy and swing; however, the computation was carried out only to the end of the deploy phase. The reason for this is primarily as a result of the high computation costs of a program as long as the present. The simulation of the stance and deploy phases required approximately 3.5 hours of execution time on the CDC 6400 machine. It is anticipated that the swing phase alone would require considerably more time due to the necessity of meeting terminal conditions.

In spite of the lack of results for the swing phase, the computations of the stance and deploy phases are quite useful on their own. The level of muscular activity is far greater during the foot contact portion of the walking cycle (cf. [27,28,29]).

It was shown in Chapter VI that the starting time of the simulation was chosen as 0.3 sec. (i.e. 15% of the period after heel strike). To obtain results in the portion 0.0 - 0.3 sec., the equations were integrated backwards.

conditions at time $t_0 = 0.0$ sec. were available.

7.2 Results

The computation results are presented here in graphical form for ease of interpretation. The output from the program was listed on the line printer (examples of typical output are shown in Appendix F) as well as on punched cards. The plotting was done off line from the data on the punched cards. The cards were read on an HP 9830A calculator based system and stored in matrix form on disk. The plots could then be produced in a suitable form.

7.2.1 Trajectories of Limb Segments

Figure 7.1 shows the calculated trajectories of the shank and thigh segments as one would expect to view them using interrupted-light photography. (Note: the foot segment is shown drawn from the ankle to the toe and not the sole of the foot). For comparison Figure 7.2 shows the experimental results of Eberhart et. al. [27]. The portion of the latter corresponding to the calculated trajectories is that section between the arrows on Figure 7.2.

The experimental and analytical results show very good agreement in general form. The two could not be expected to agree exactly due to the unlikelihood of the individuals having the same lengths of limb segments or of the

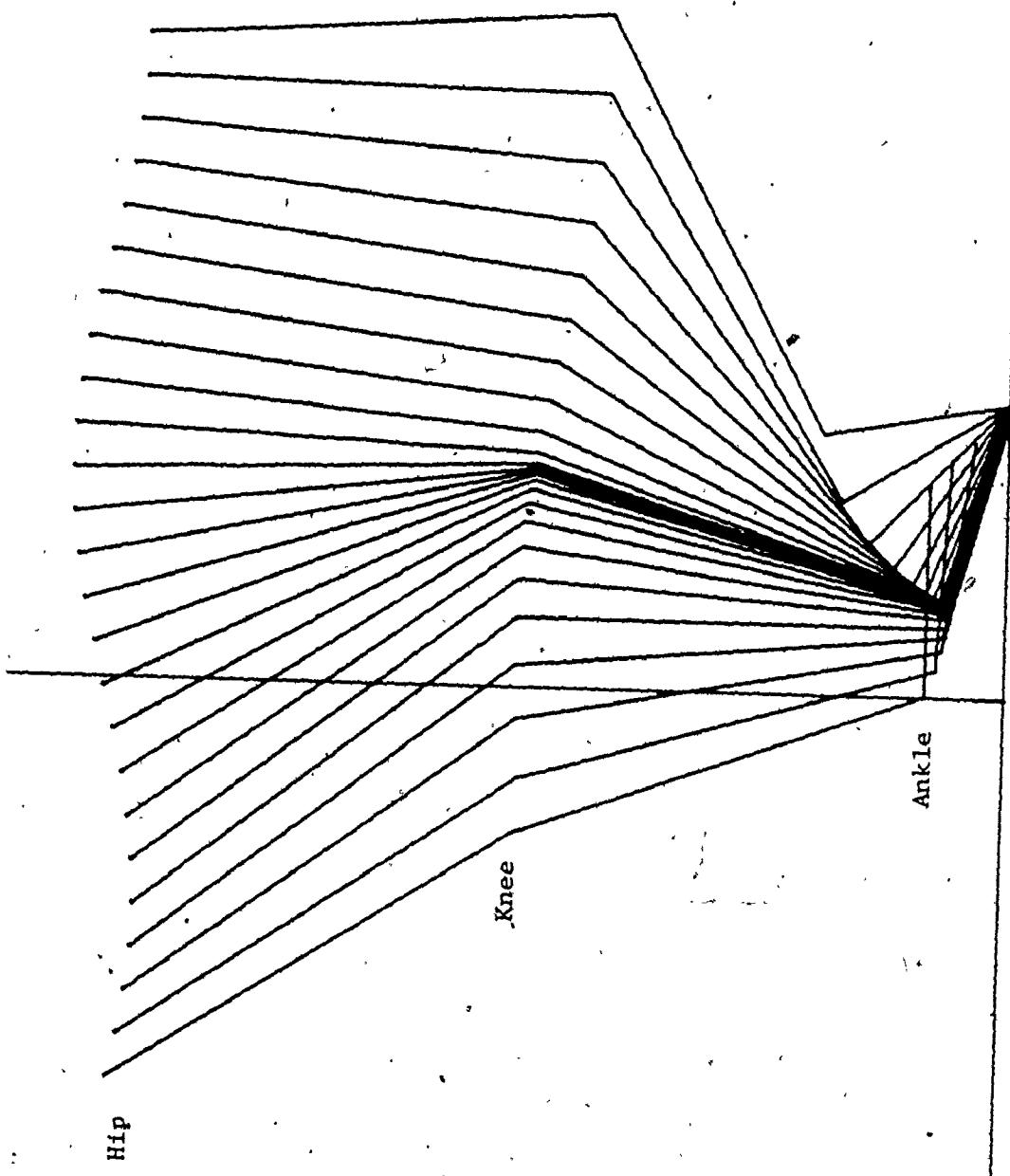


Figure 7.1 "Stroboscopic" View of Stance and Deploy Phase of Walking Cycle

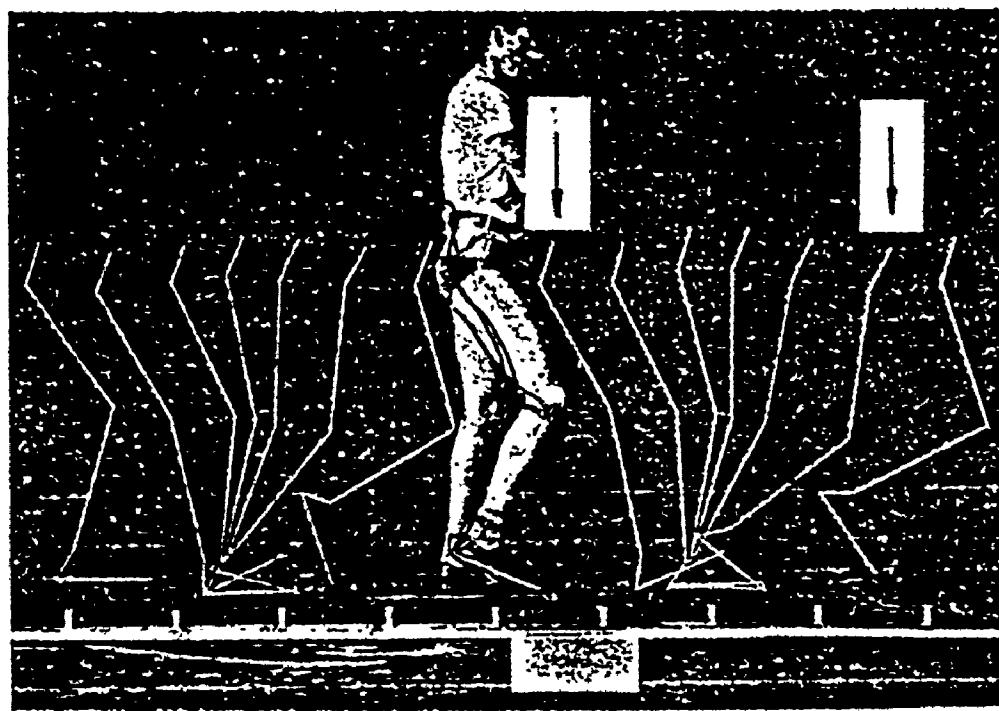


Fig. 15.5 Typical record from interrupted-light photography.

Figure 7.2 Experimental Results for Comparison with Trajectories shown in Figure 7.1. (Portion between arrows corresponds to foot contact phase.) From [27]

"experiments" being conducted under the same conditions (i.e. cadence, step length, etc.).

Corresponding to the leg positions shown in Figure 7.1, the horizontal and vertical positions of the hip, knee and ankle are shown as functions of time in Figures 7.3 and 7.4. The reference point for the horizontal position is the location of the ankle at heel strike, and for the vertical position the ground was chosen as the reference. Figure 7.5 shows the same functions determined experimentally [27]. In the latter, the foot contact portion (up to the dashed line) was approximately 65% of the cycle whereas in the computer analysis 60% cycle was the toe-off time, thus accounting for some of the differences between the experimental and computational results.

Figures 7.6, 7.7 and 7.8 show the angular variables z_1 and z_2 , and their first and second time derivatives respectively.

7.2.2 Joint Torques and Muscle Forces

The active joint torques u_1 and u_2 are shown in Figure 7.9. It is interesting to note that the major peaks in the curves coincide very closely to the peaks in the $f_h(t)$ and $f_v(t)$ curves of Figure 6.3 as one would expect. These moments are generated by the force levels in the muscles, shown in Figures 7.10 to 7.14. These graphs indicate

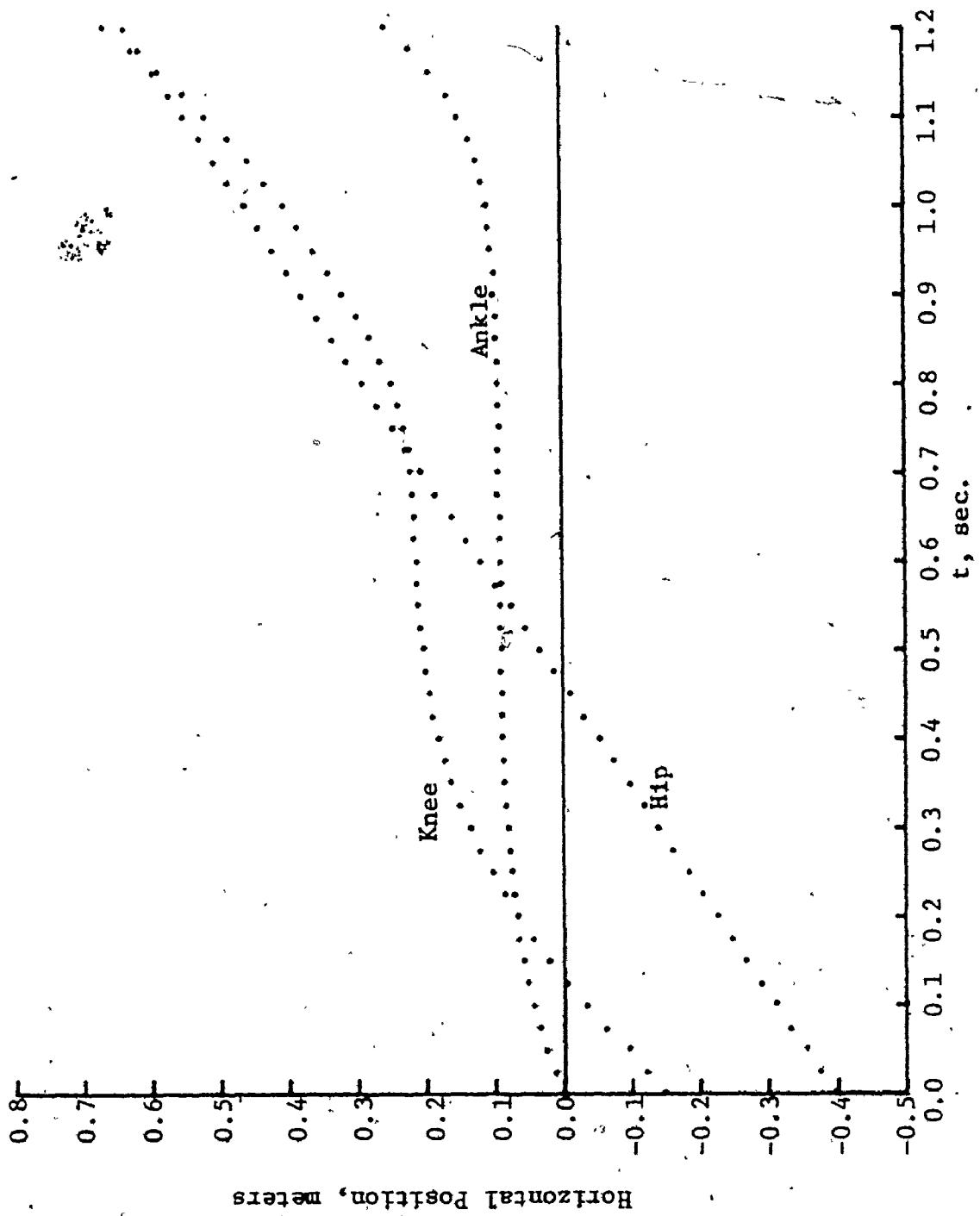


Figure 7.3 Horizontal Positions of Hip, Knee and Ankle

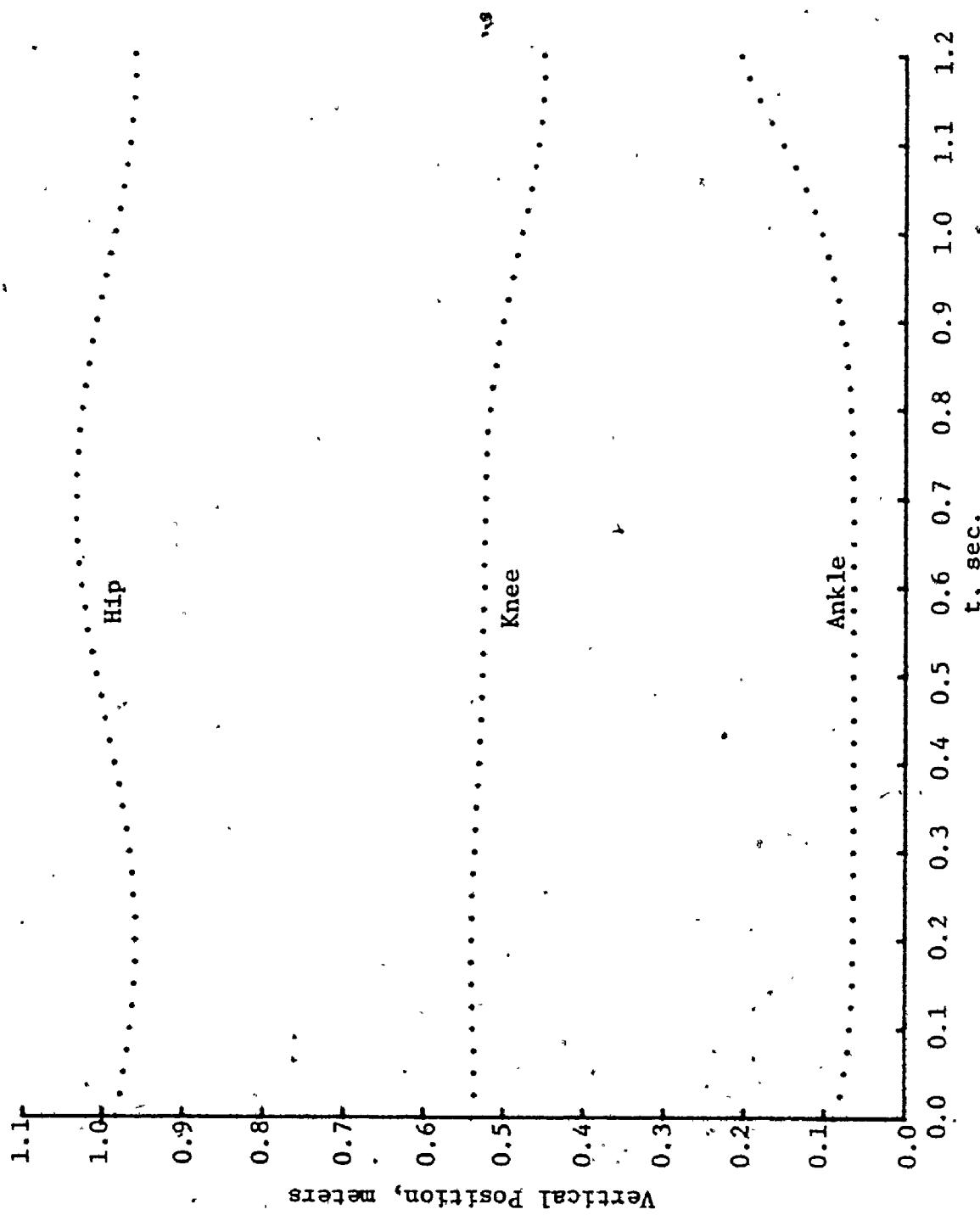


Figure 7.4 Vertical Positions of Hip, Knee and Ankle

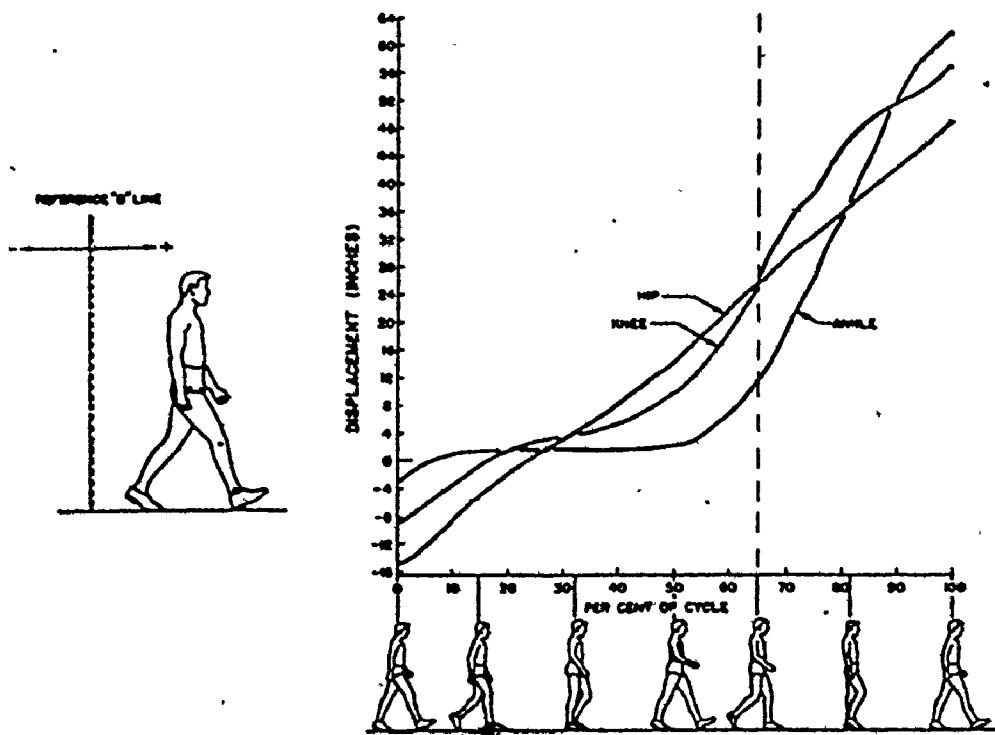


Fig. 15.15 Fore-and-aft displacements of leg joints. Force studies, typical subject.

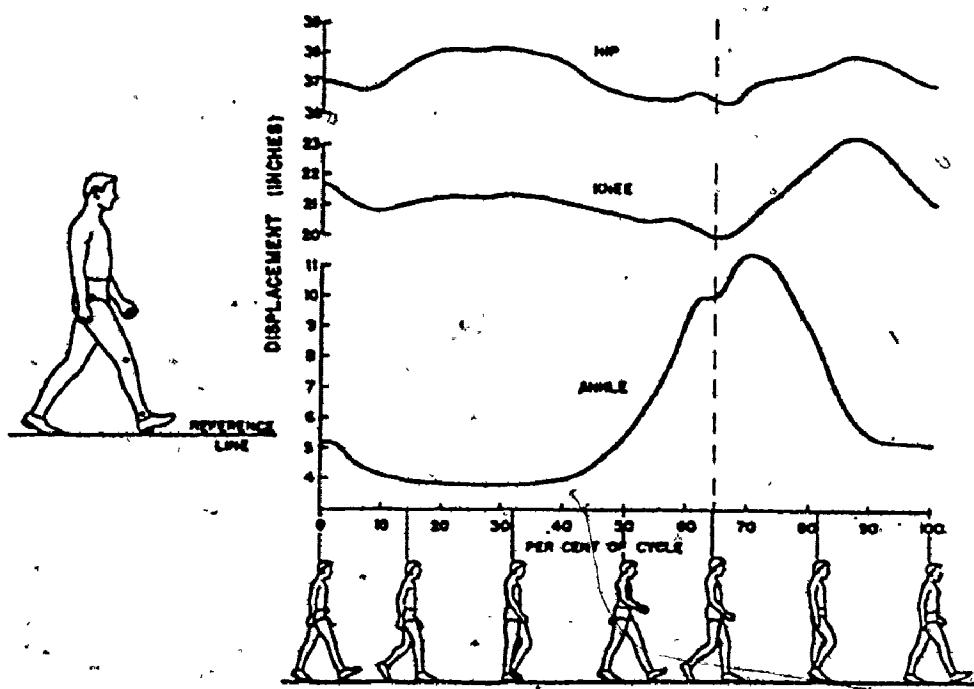


Fig. 15.14 Vertical displacements of leg joints. Force studies, typical subject.

Figure 7.5 Experimentally Determined Horizontal and Vertical.
of the "— and

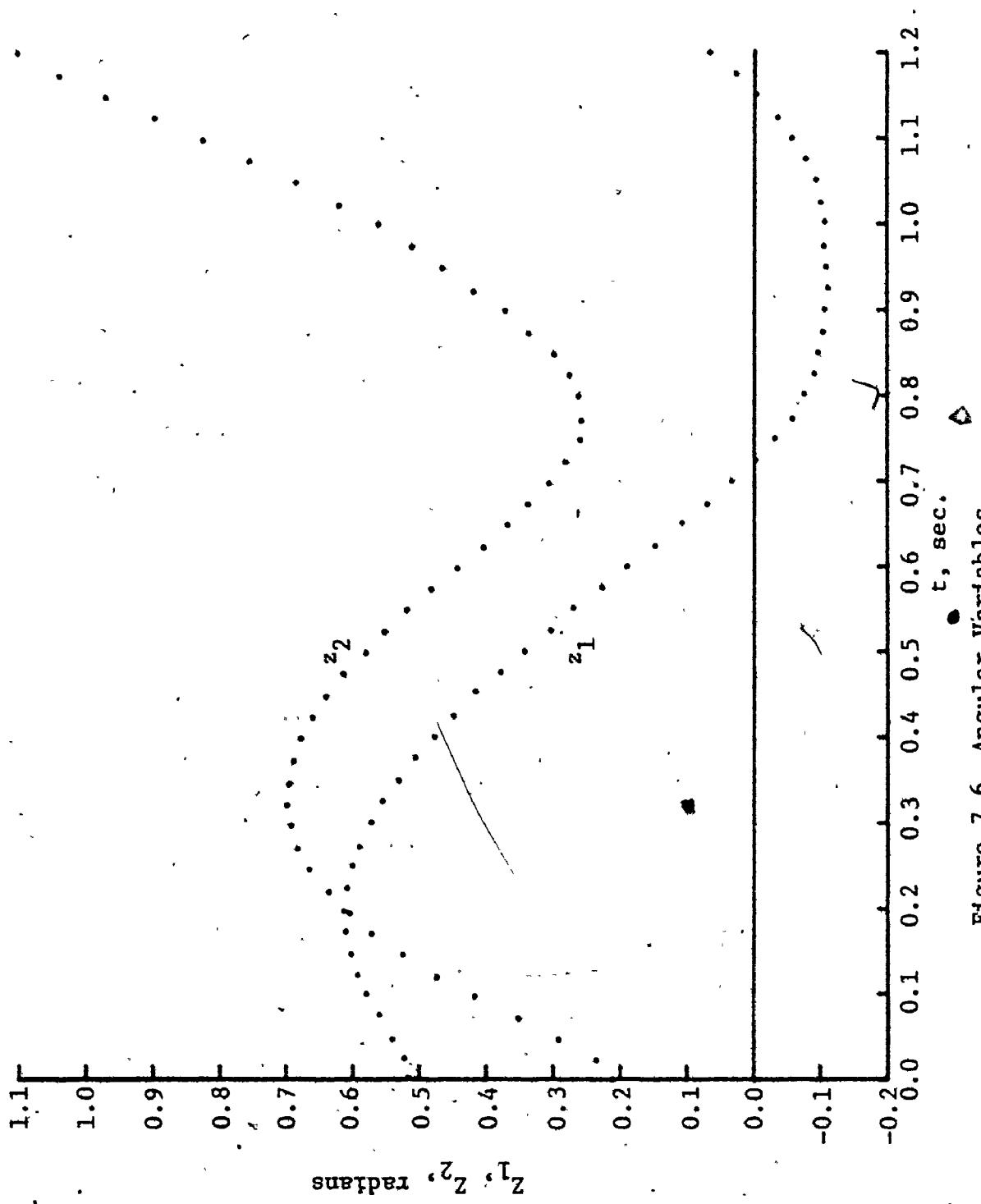


Figure 7.6 Angular Variables

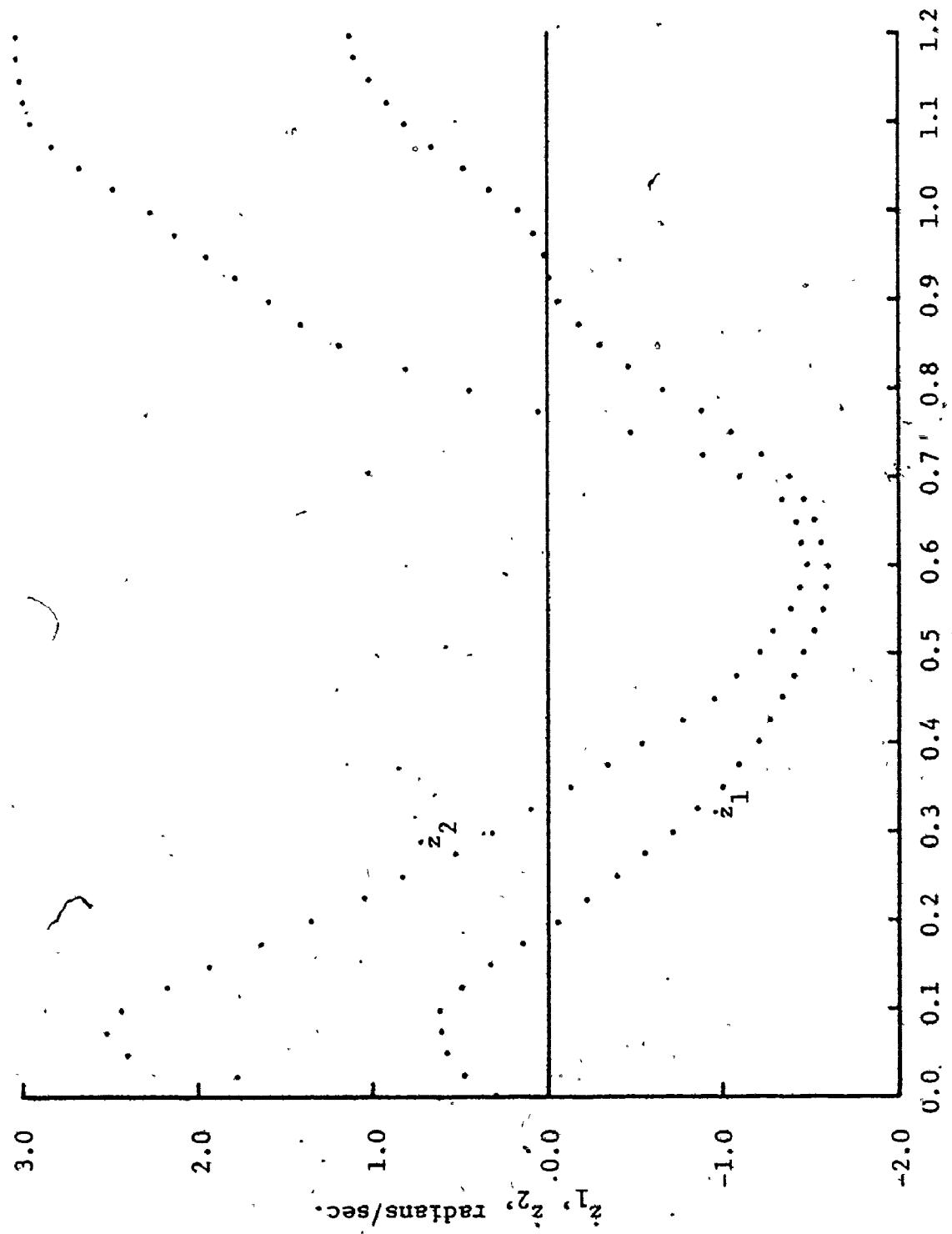


Figure 7.7 Angular Velocities

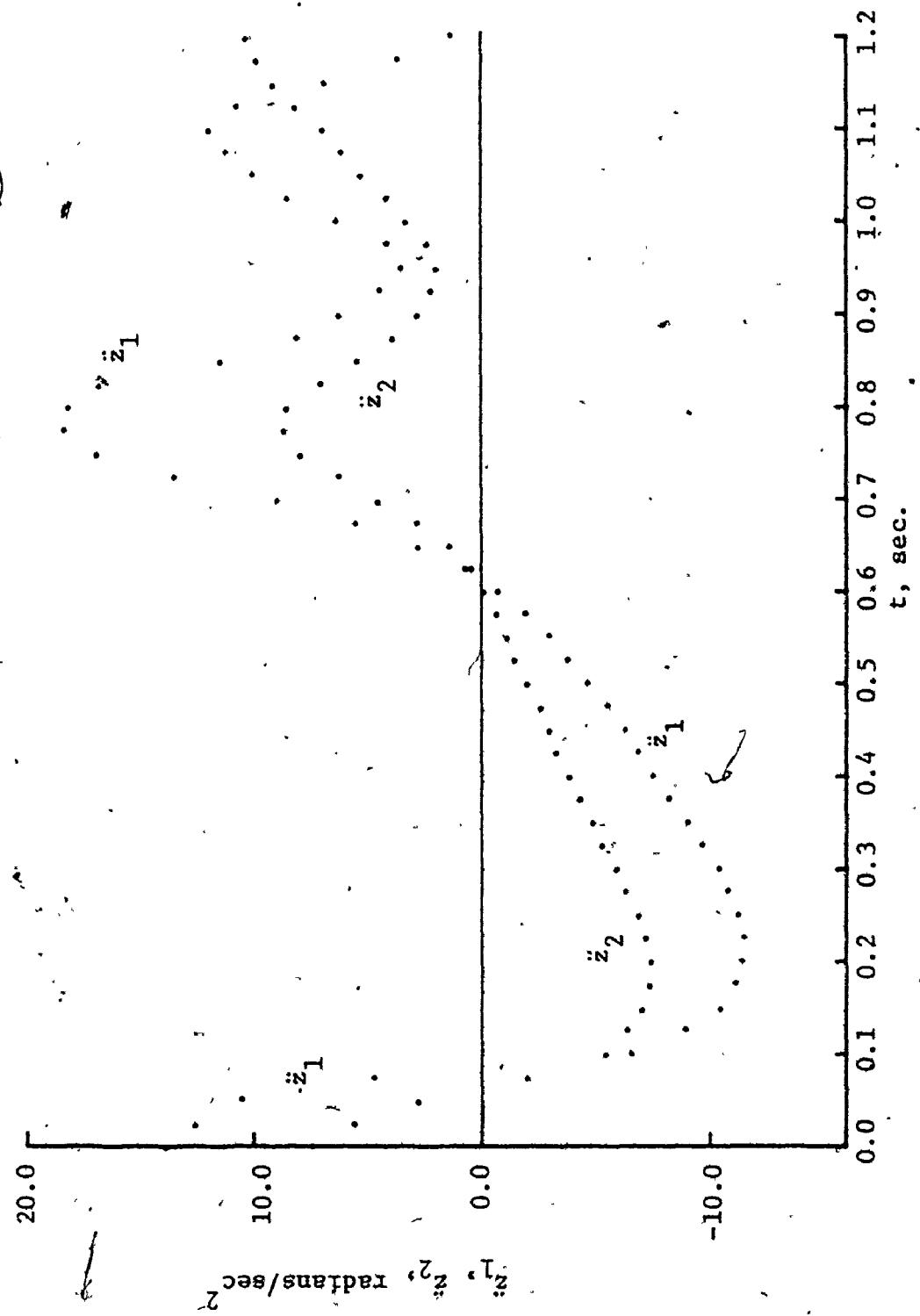


Figure 7.8 Angular Accelerations

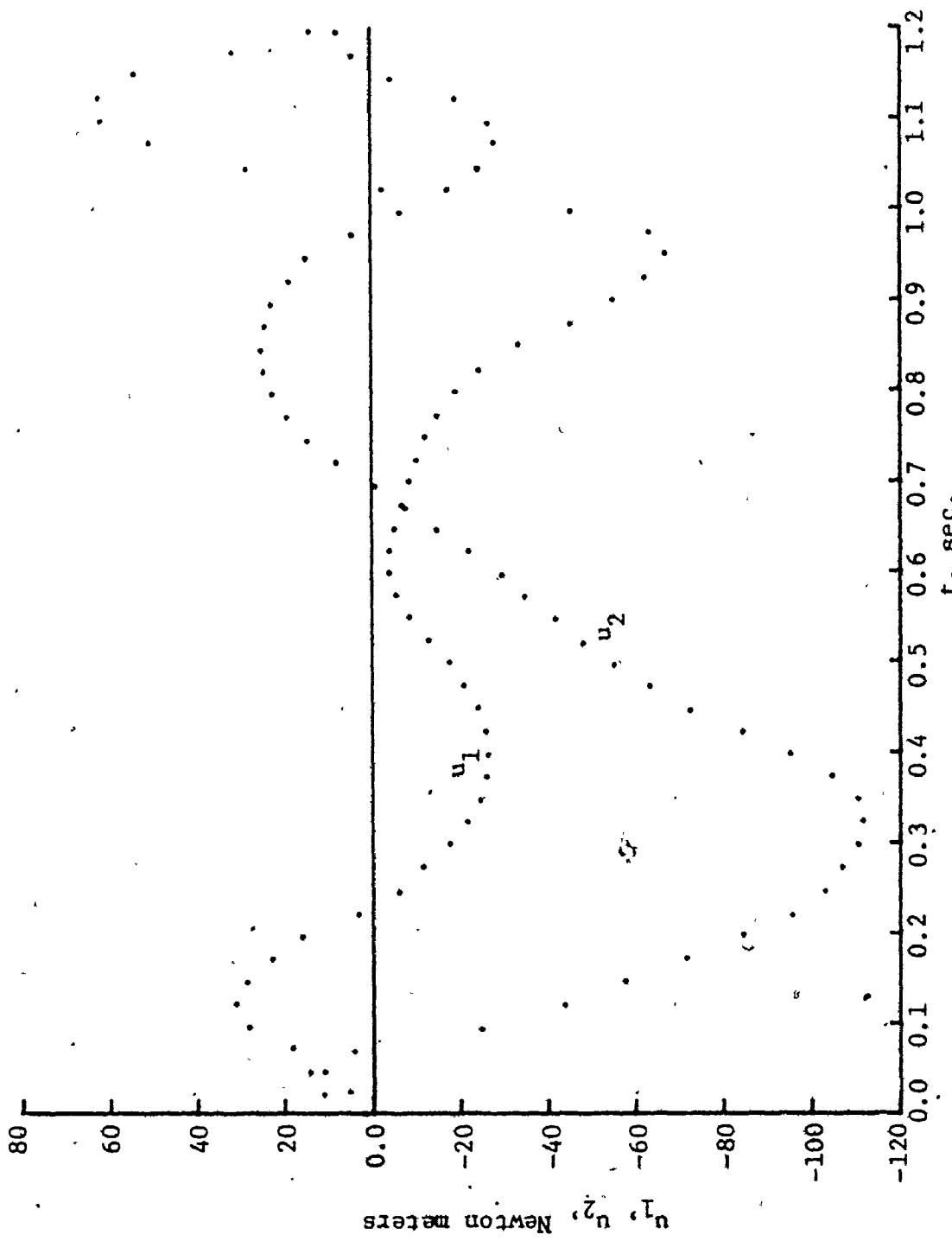


Figure 7.9 Joint Torques

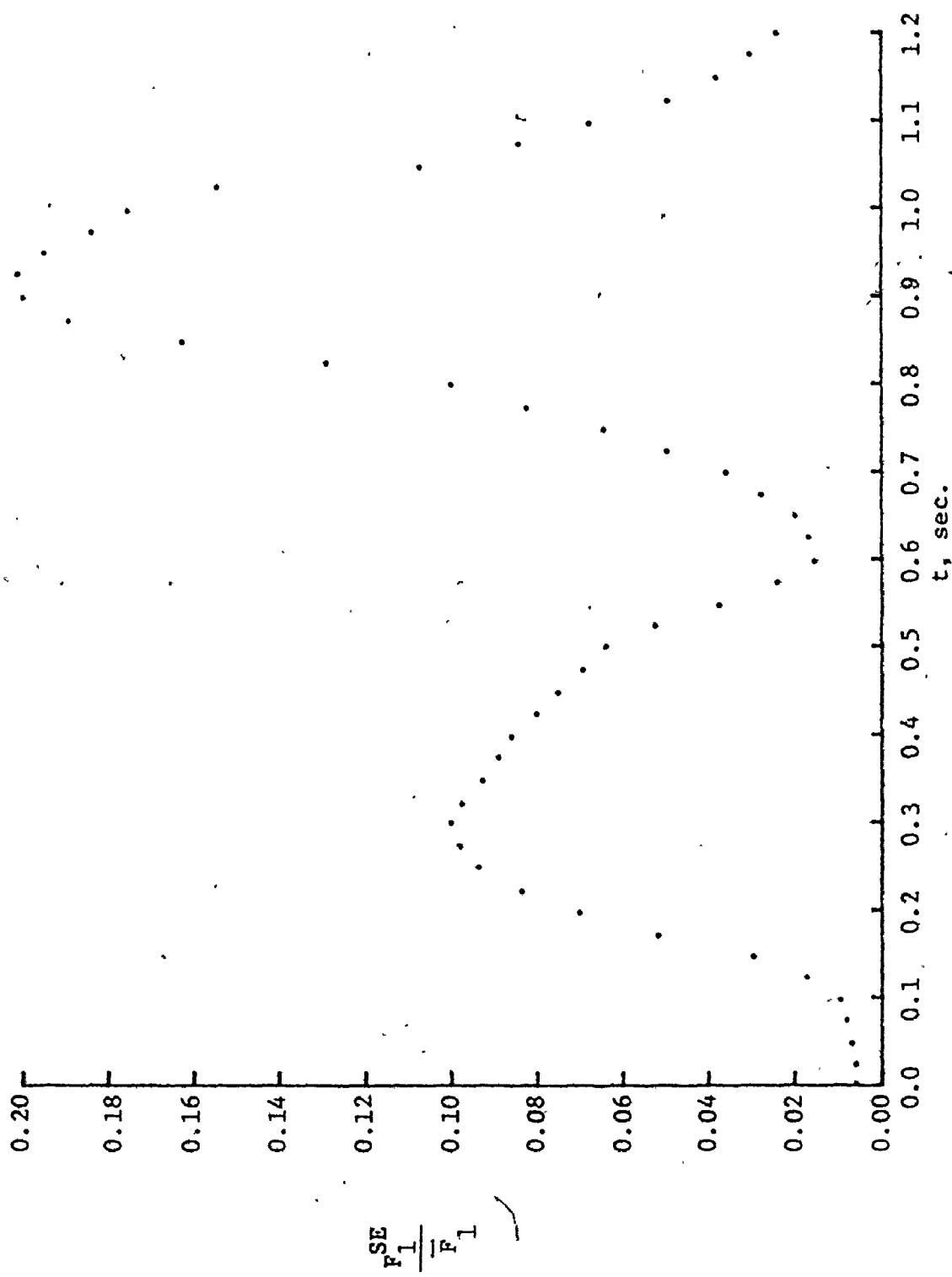


Figure 7.10 Normalized Force in *M. illopoas*

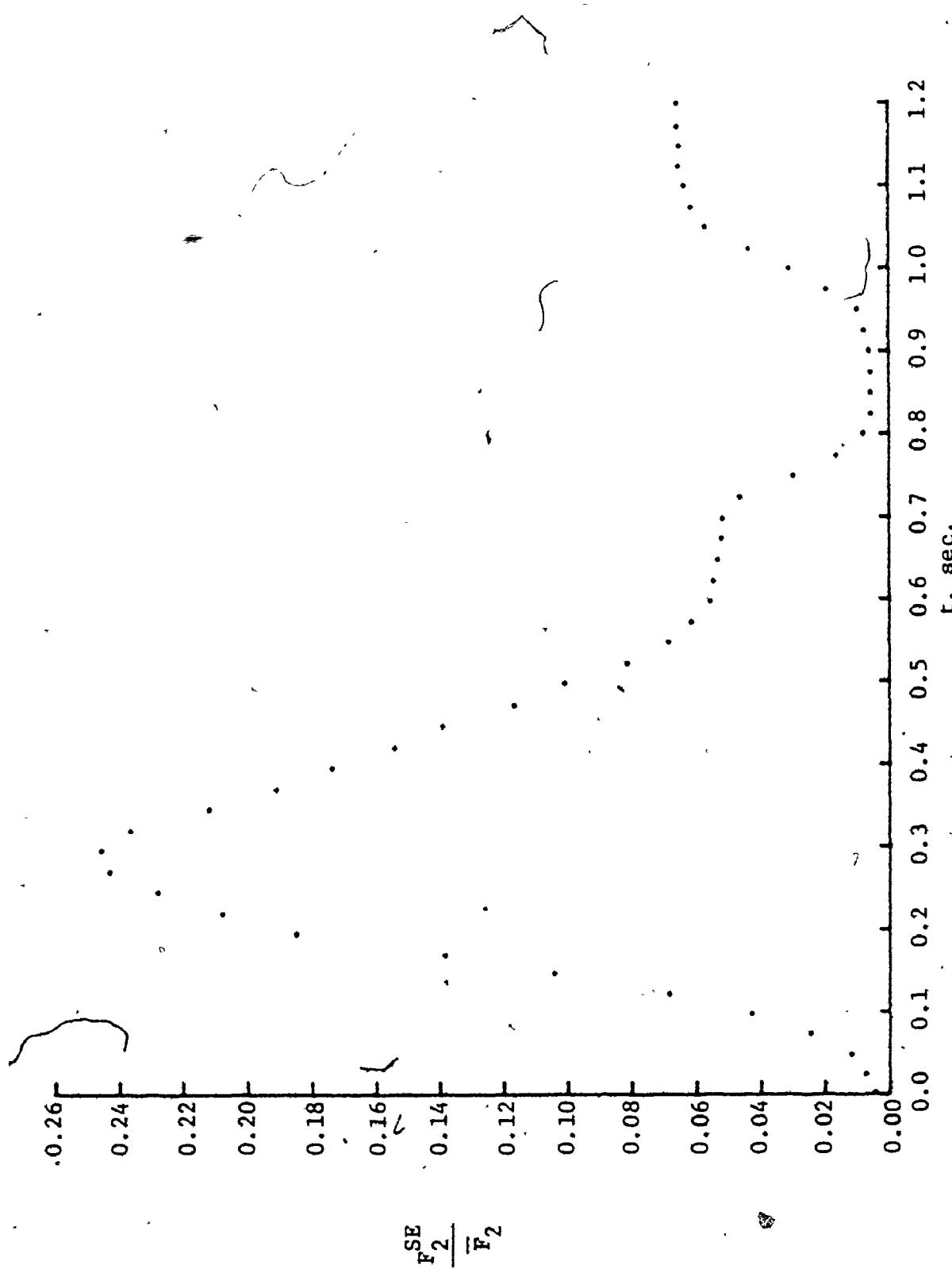


Figure 7.11 Normalized Force in *M.vastus*

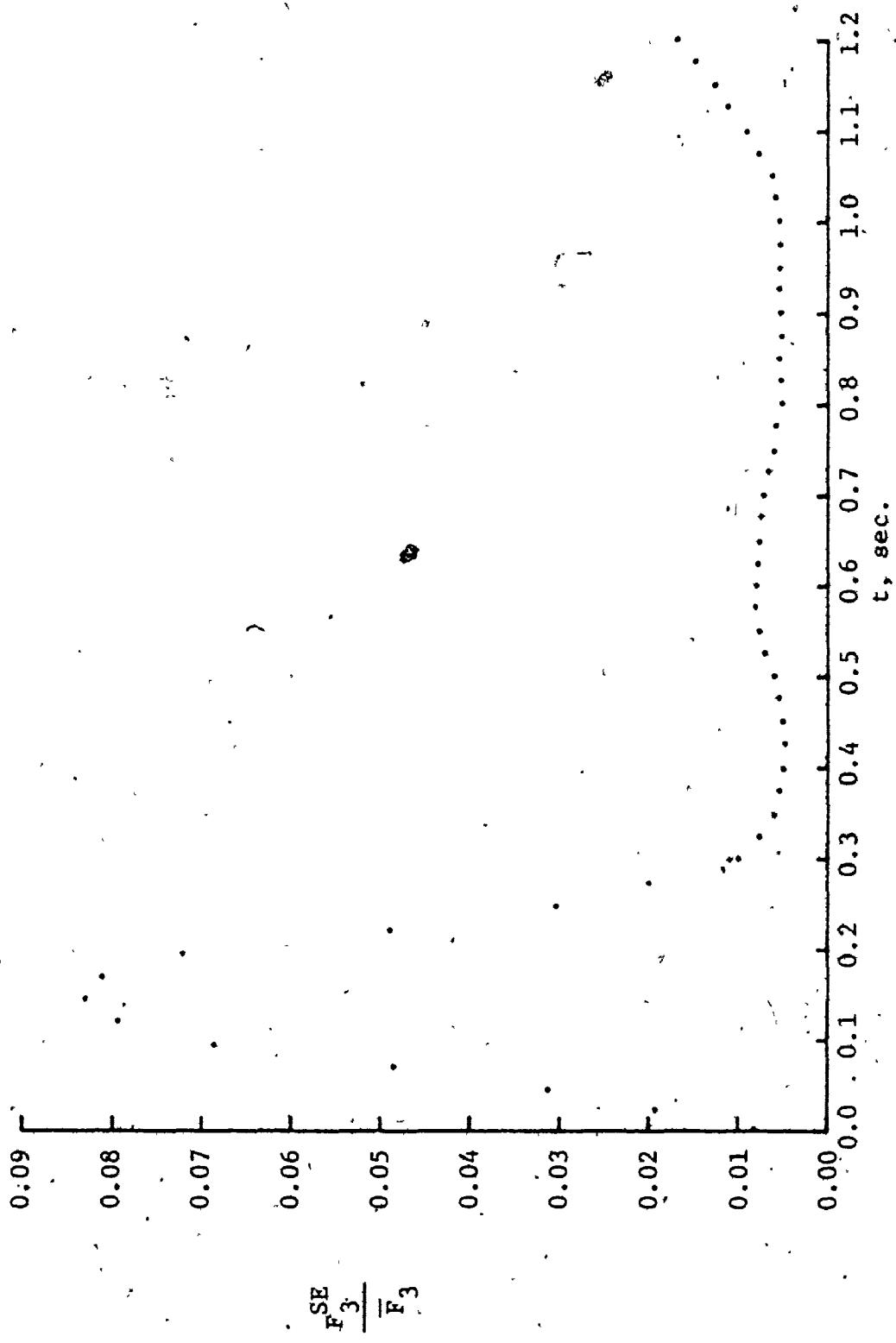


Figure 7.12 Normalized Force in *M. rectus femoris*

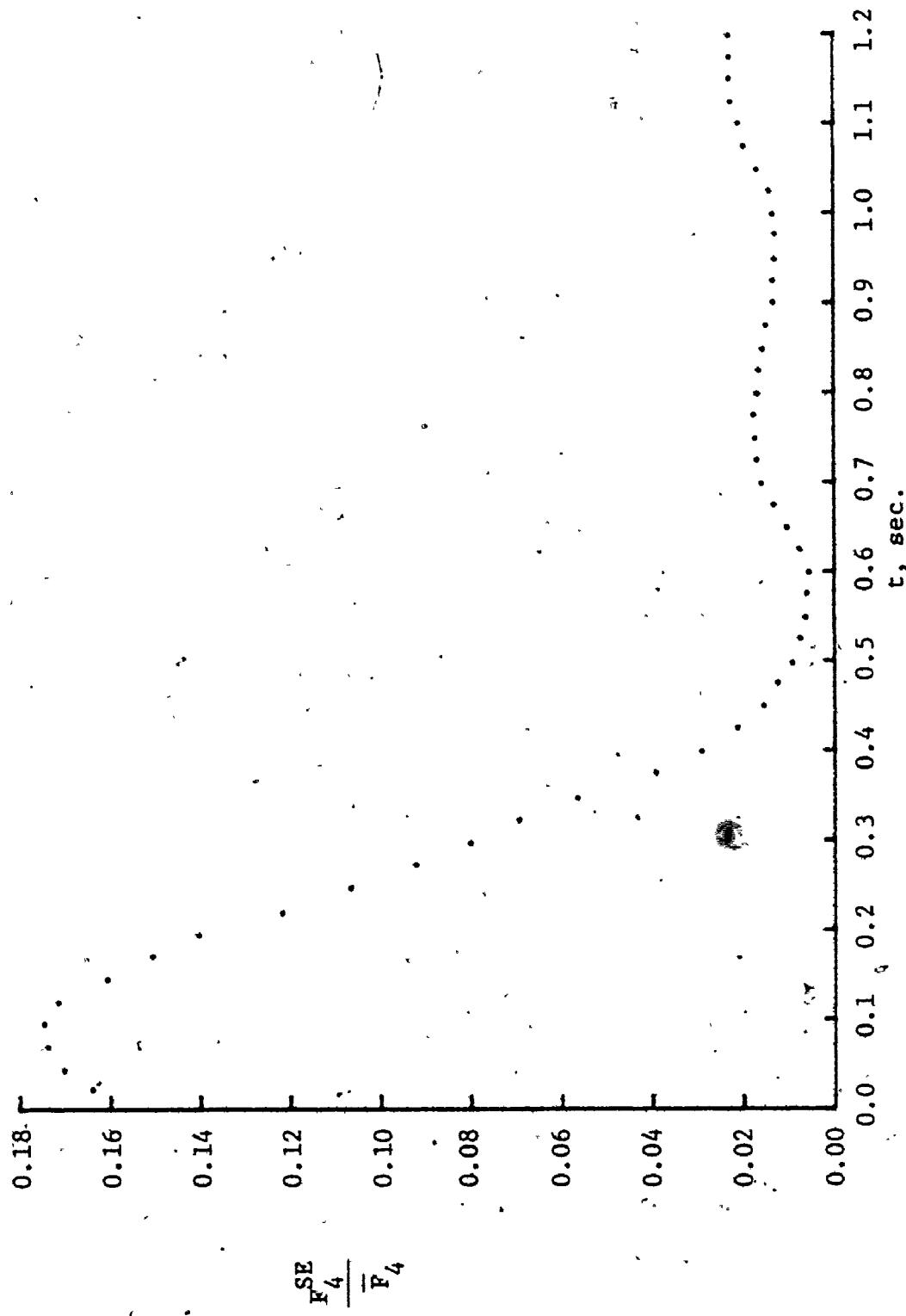


Figure 7.13 Normalized Force in Hamstring Muscle

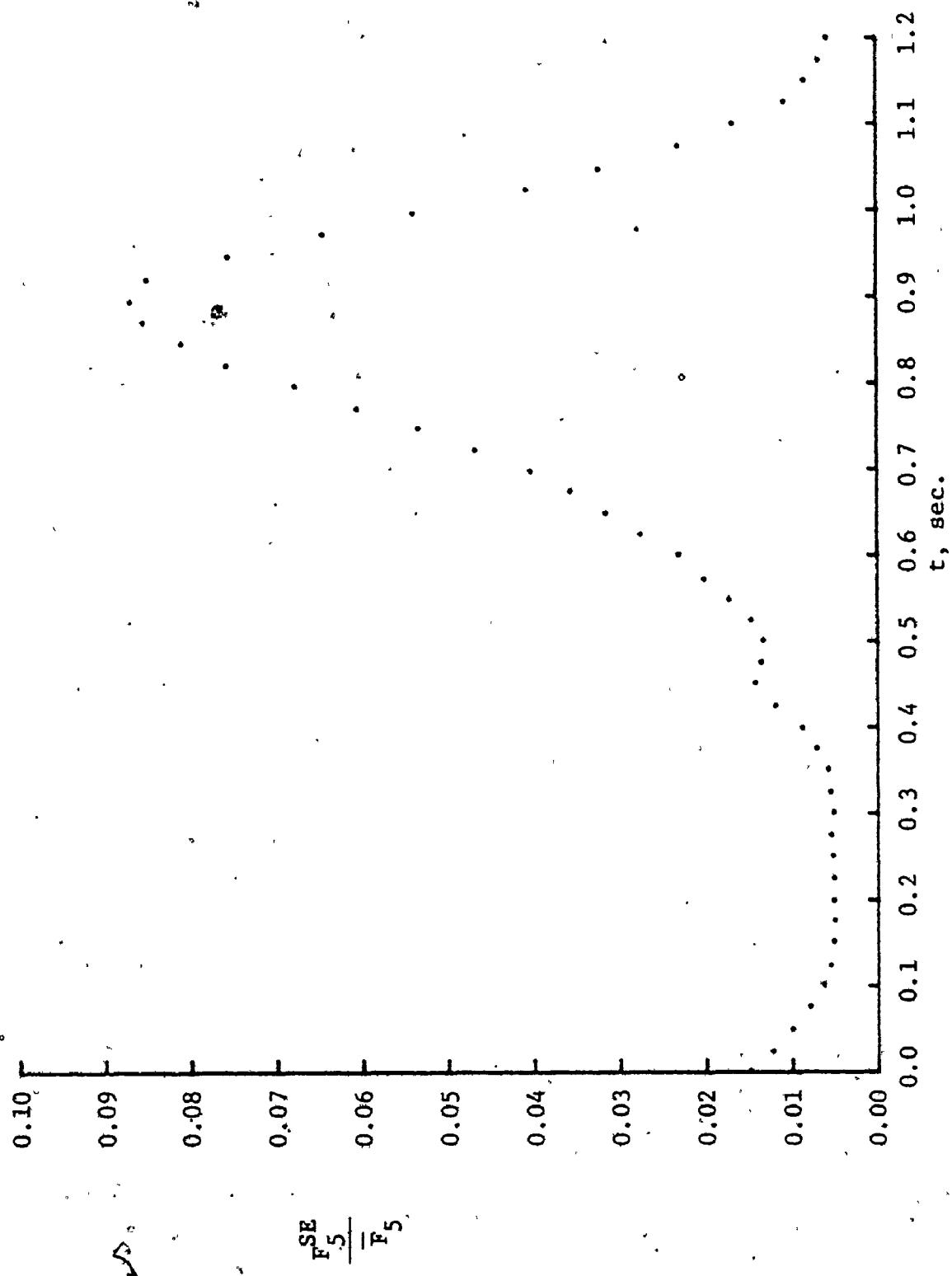


Figure 7.14 Normalized Force in *M. gastrocnemius*

force) as functions of time. Since the forces cannot be directly measured in vivo, discussion of the validity of the calculated forces is left to a later section after the calculated muscle controls have been presented.

7.2.3 Muscular Controls

One of the primary purposes of the present work was to demonstrate the feasibility of analytically determining the time course of the muscular controls giving rise to locomotion. It was shown earlier that two controls, namely stimulation frequency, ϕ , and motor unit recruitment, θ , were necessary to describe the control of each of the muscles. Figures 7.15 to 7.19 show the trajectories of these normalized controls.

It will be noted that in all cases the value of ϕ at any time is higher than the corresponding value of θ . Moreover, the trajectory of ϕ peaks before that of θ . Both of these phenomena can be explained as follows. At very low stimulation rates, the time interval between successive nerve impulses to each of the motor units is very long, relatively speaking. If simultaneously, the proportion θ of the muscle being stimulated is small, the net effect on the tendons of the muscles would be one of rather "jerky" forces. Hence the frequency is increased substantially with the appropriate reduction in motor unit recruitment, noting that the force in a motor unit does not vary linearly with

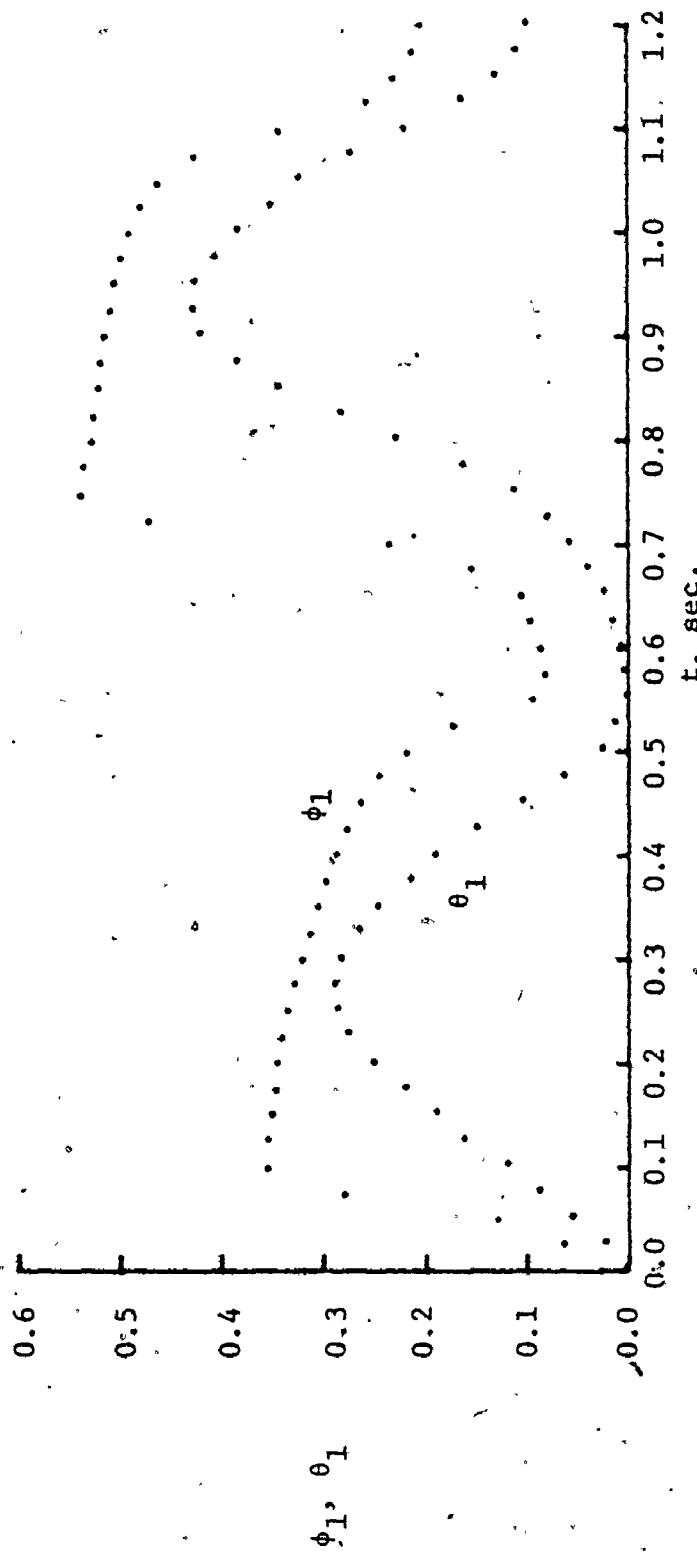


Figure 7.15 Normalized Stimulation Frequency and Motor Unit Recruitment in *M. ilioptosas*

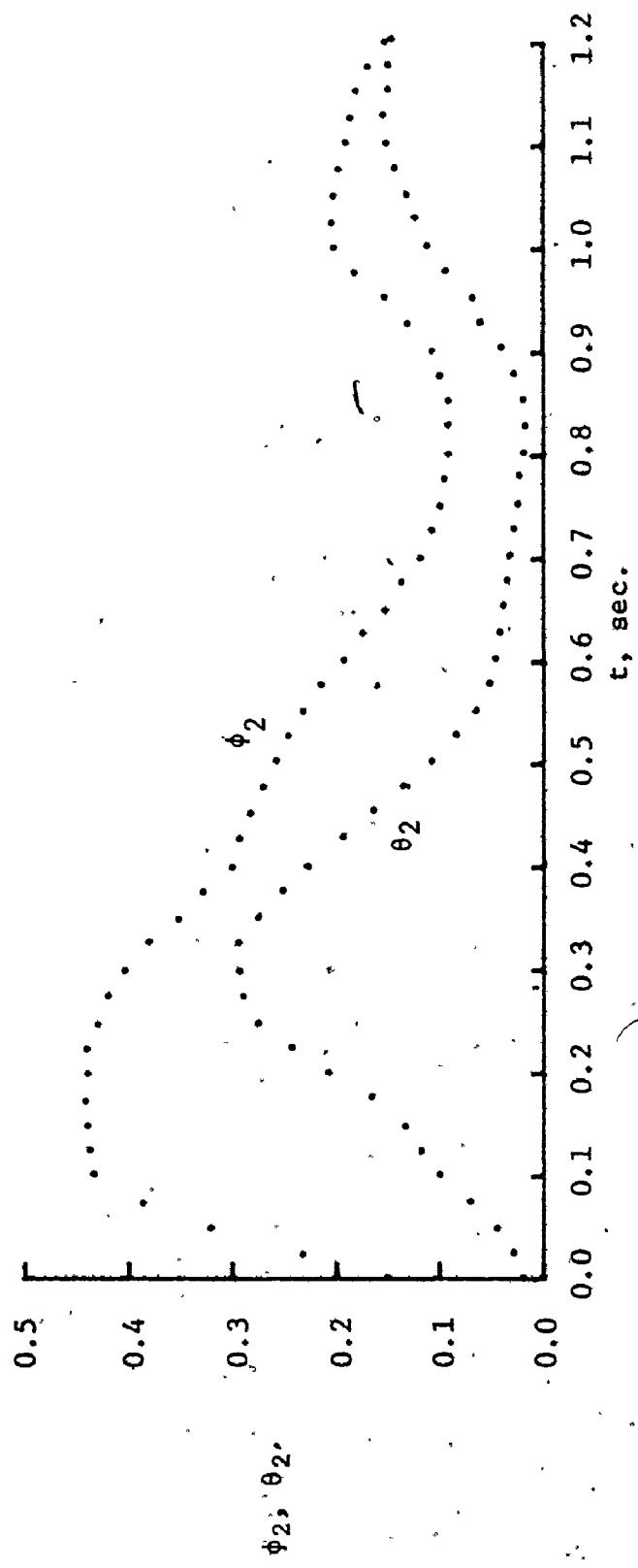


Figure 7.16 Normalized Stimulation Frequency and Motor Unit Recruitment in M.vastus

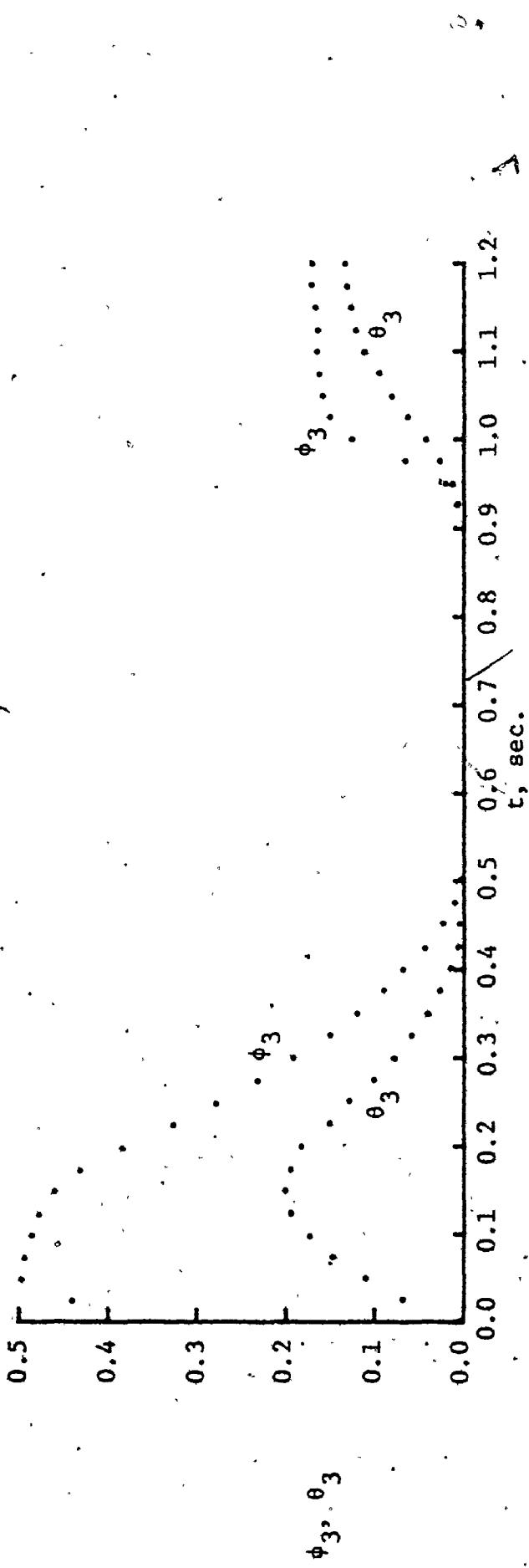


Figure 7.17 Normalized Stimulation Frequency and Motor Unit Recruitment in *M. rectus femoris*

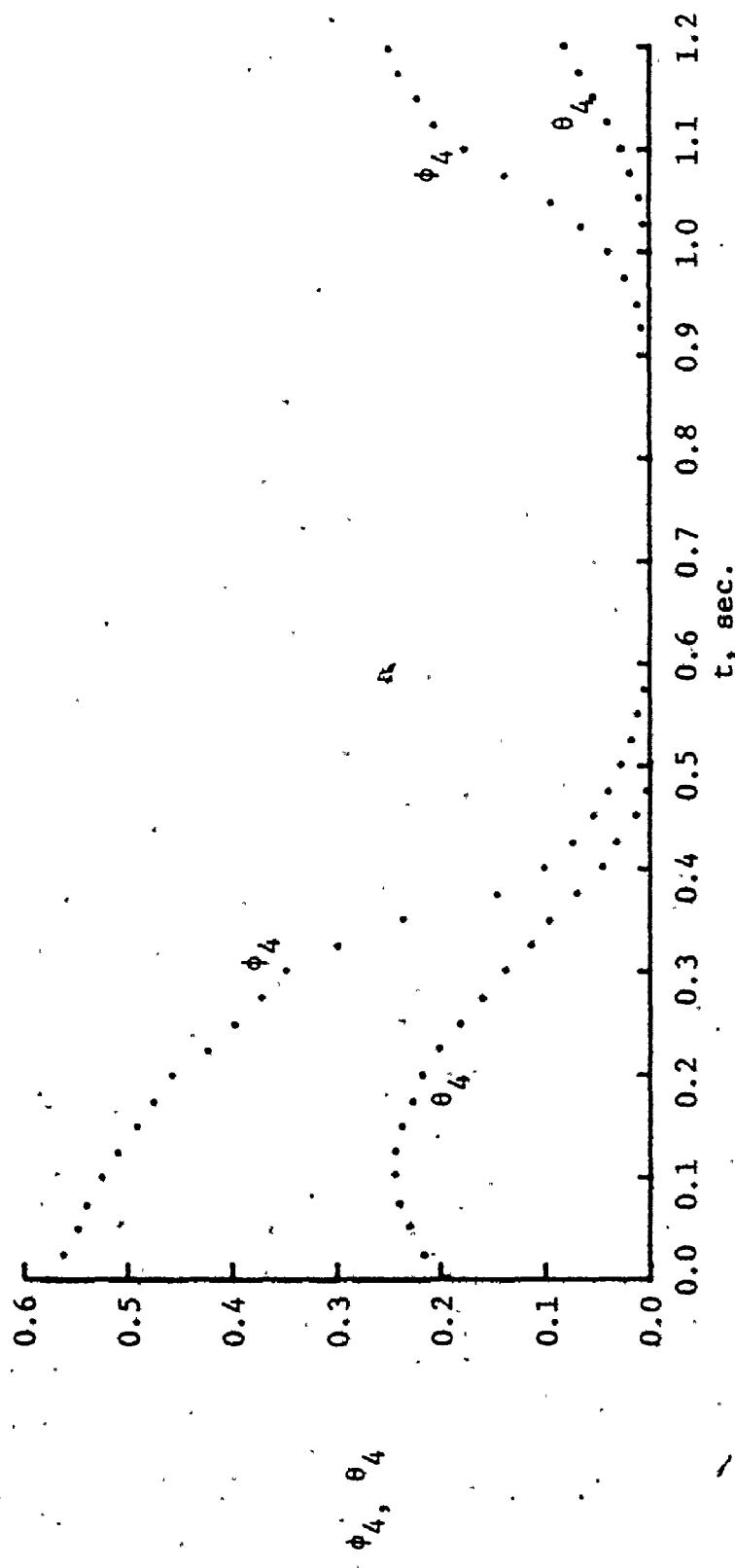


Figure 7.18 Normalized Stimulation Frequency and Motor Unit Recruitment in Hamstring Muscle

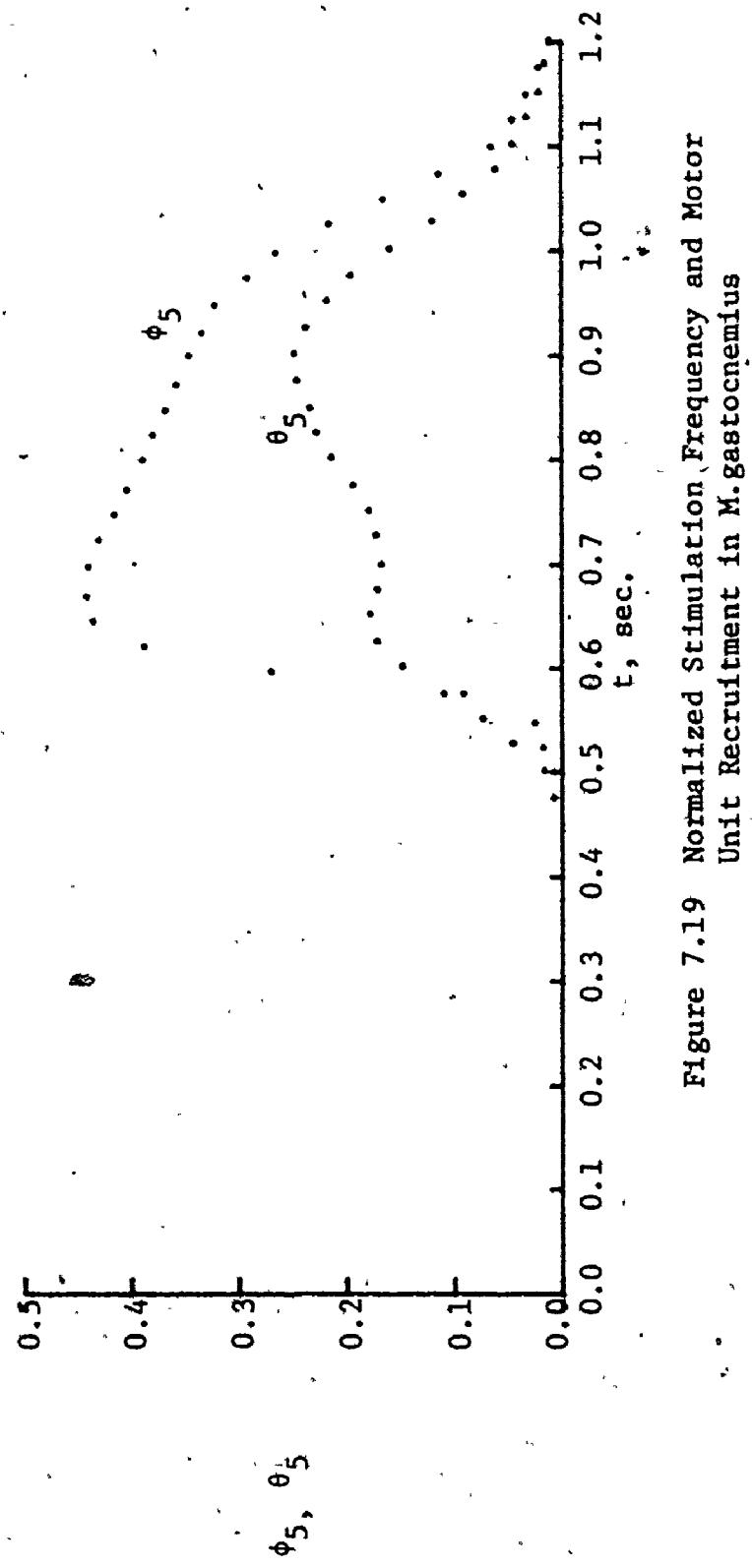


Figure 7.19 Normalized Stimulation Frequency and Motor Unit Recruitment in *M. gastrocnemius*

frequency.

Considering also equations (5.17), it can easily be shown that

$$\frac{\partial \xi_j}{\partial \theta} \geq \frac{\partial \xi_j}{\partial \phi} \quad (7.1)$$

so that the energy consumption dictates that preference be given to the highest levels of θ possible.

These arguments and the results of the analysis are in complete agreement with the suggestions by Hatze [30] that (at least under isometric conditions) at force levels below approximately 28% of maximum, most of the "controlling" is accomplished via the variable θ ; i.e. motor unit recruitment.

7.3 Comparison of Muscle Controls with the Observable

It would be desirable to be able to determine a relationship between the controls $\theta(t)$ and $\phi(t)$ and some measureable signal emanating from the muscle and having a direct connection to the controls. This is in fact possible. Such a signal is the electromyogram (EMG). This can be shown in the following way.

Assume that the muscle of concern consists of fast and slow twitch fibres in the proportions a_f and a_s respectively, and let the total cross-sectional area of the muscle be A. The excitation, $\epsilon(t)$, in that muscle may then

be defined by (cf. [13])

$$\epsilon(t) = \theta_f(t) a_f \phi_f(t) v_{f,opt} + \theta_s(t) a_s \phi_s(t) v_{s,opt} . \quad (7.2)$$

Biro and Partridge [31] have shown that for an appropriate EMG voltage sample, the variance σ^2 of the sample is a linear function of $\epsilon(t)$, i.e.

$$\sigma^2(t) = k_1(a_f \theta_f(t) \phi_f(t) v_{f,opt} + a_s \theta_s(t) \phi_s(t) v_{s,opt}) + k_2 \quad (7.3)$$

wherein k_1 and k_2 are constants. The EMG signal, obtained by means of surface electrodes positioned at the motor points of the muscle, may be full-wave rectified and electronically integrated over some time interval Δt . The standard deviation σ can then be calculated to be

$$\sigma = UT/0.403\Delta t + 0.0333 \quad (7.4)$$

where U (in volts) is the integrated EMG signal and T (in seconds) is the time constant of the integrator. Equation (7.4) is due to Harding and Sen [32]. Combining equations (7.4) and (7.3), recalling that $v_{f,opt} = 100$ Hz and $v_{s,opt} = 60$ Hz, one obtains

$$(U(t)T/0.403 \Delta t + 0.0333)^2 \\ = k_1(100 a_f \theta_f(t) \phi_f(t) + 60 a_s \theta_s(t) \phi_s(t)) + k_2 \quad (7.5)$$

with the constants k_1 and k_2 depending upon the apparatus used. Most muscles can be regarded as b

predominantly fast or slow twitch fibres (cf. [30]) (i.e. one type only). This being the case, a_f , or a_s , is equal to unity and equation (7.5) simplifies accordingly.

Assume that the output of the i -th channel of the EMG is U_i^0 volts when $\theta_i = \phi_i = 0$ (i.e. muscle unstimulated). Equation 7.5 may then be written as

$$U_i(t) = k_3 \{ (\hat{k}_1 \theta_i(t) \phi_i(t) + (U_i^0/k_3 + 0.0333)^2)^{1/2} - 0.0333 \} \quad (7.6)$$

where $\hat{k}_1 = v_{opt} k_1$ and $k_3 = 0.403^4 t/T$ and are constants. Then provided the quantity $(U_i^0/k_3 + 0.0333)^2$ is small compared to $\hat{k}_1 \theta_i(t) \phi_i(t)$, and this is usually the case, it can be seen that the envelope of the EMG signal is directly proportional to the quantity $(\theta_i(t) \phi_i(t))^{1/2}$.

Figures 7.20 to 7.24 show the square roots of the products of θ_i and ϕ_i as functions of time. It is difficult and dangerous to obtain EMG records for the iliopsoas muscle due to its deep lying location, and hence data is unavailable for this muscle. Figure 7.25 shows the results obtained by Eberhart et.al. for the calf group (muscle group 5), the quadriceps group (muscles 2 and 3) and the hamstring group (muscle group 4). Comparison of Figures 7.21 to 7.24 with 7.25 shows excellent agreement between the analytically determined controls and the experimentally measured EMG records.

It is worth pointing out that there is a slight

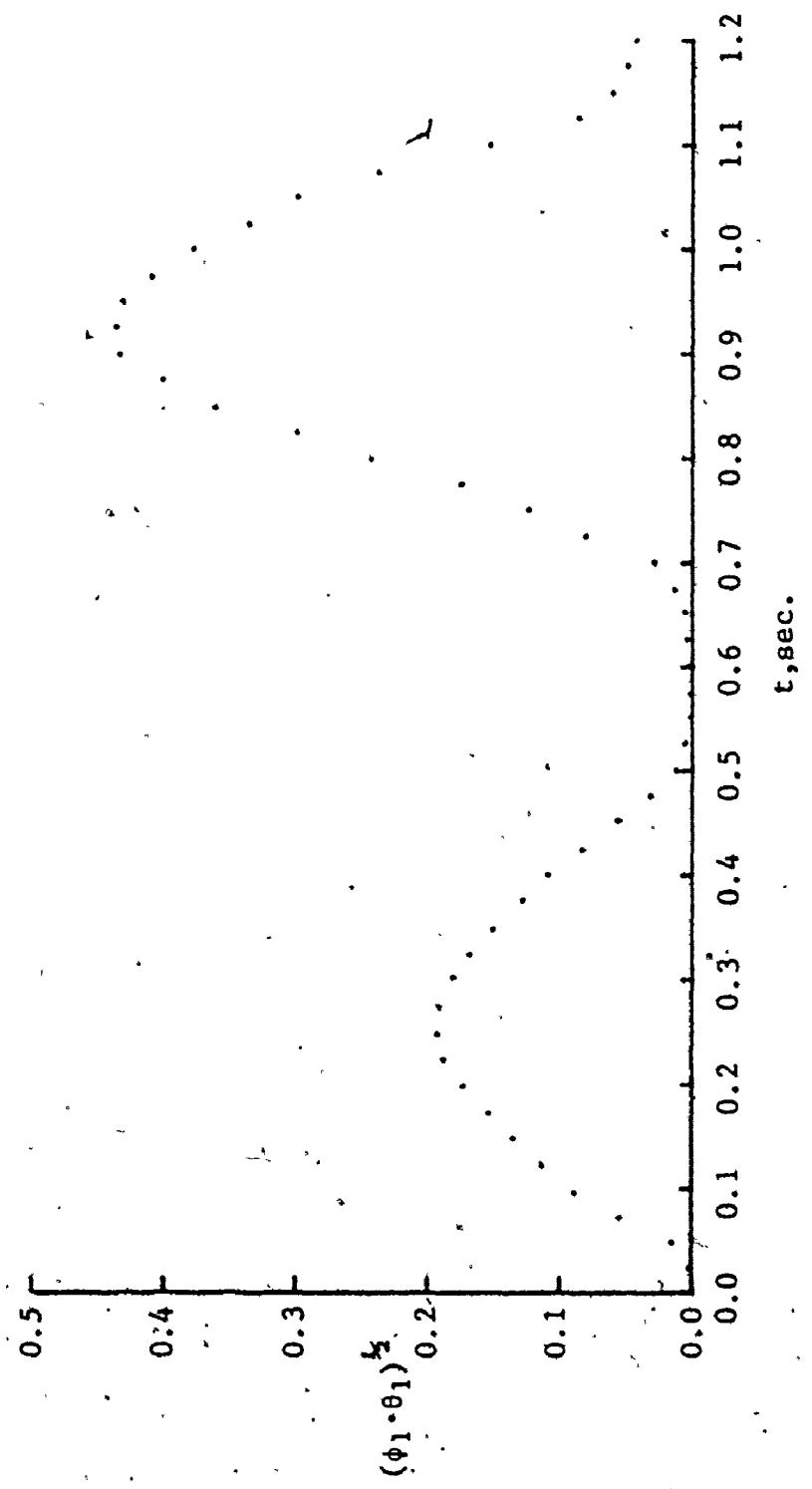


Figure 7.20 $(\phi_1 \cdot \theta_1)^{1/2}$ for *M. illopsoas*

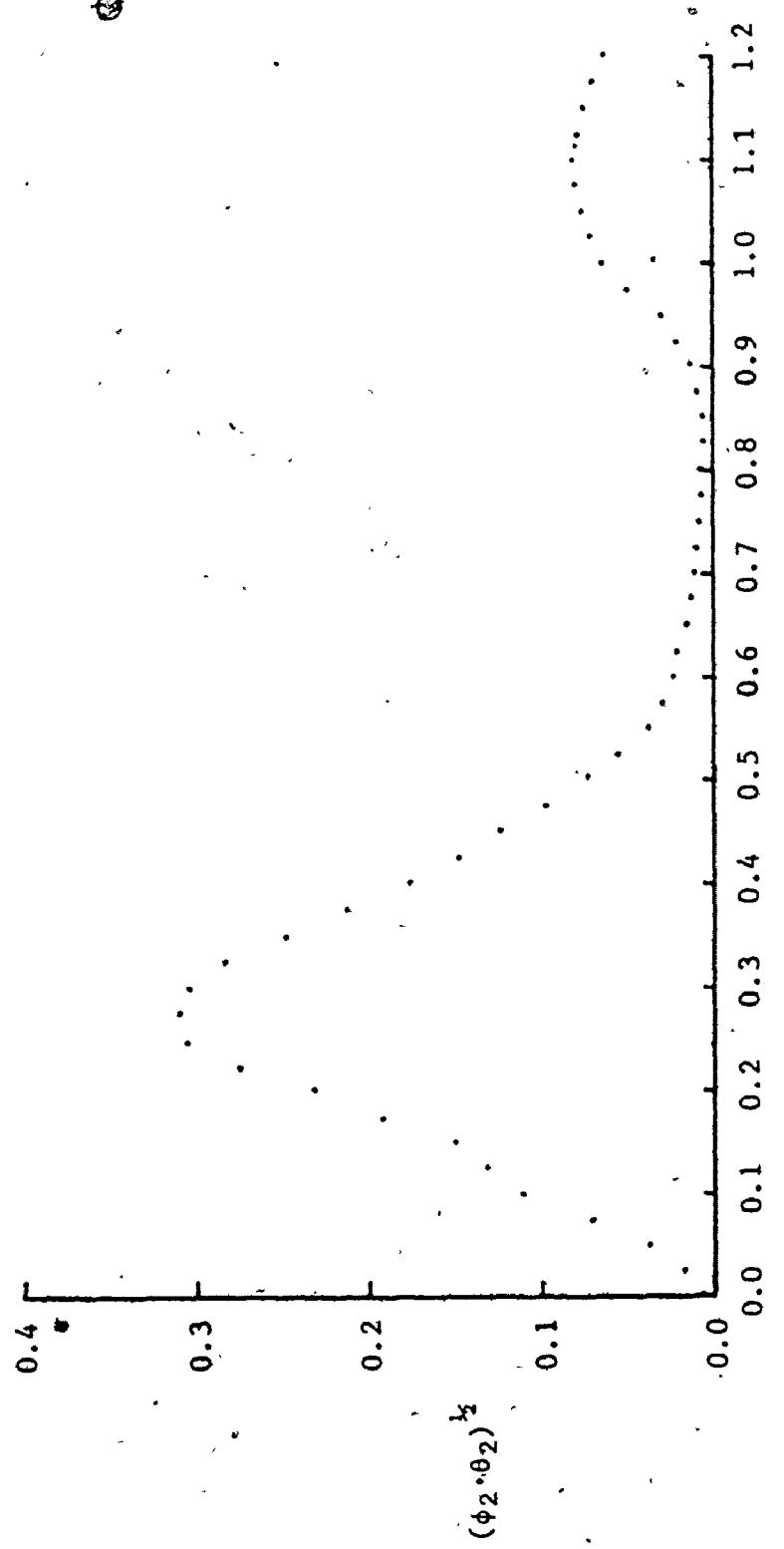


Figure 7.21 $(\phi_2 \cdot \theta_2)^{1/2}$ for *M. vastus*

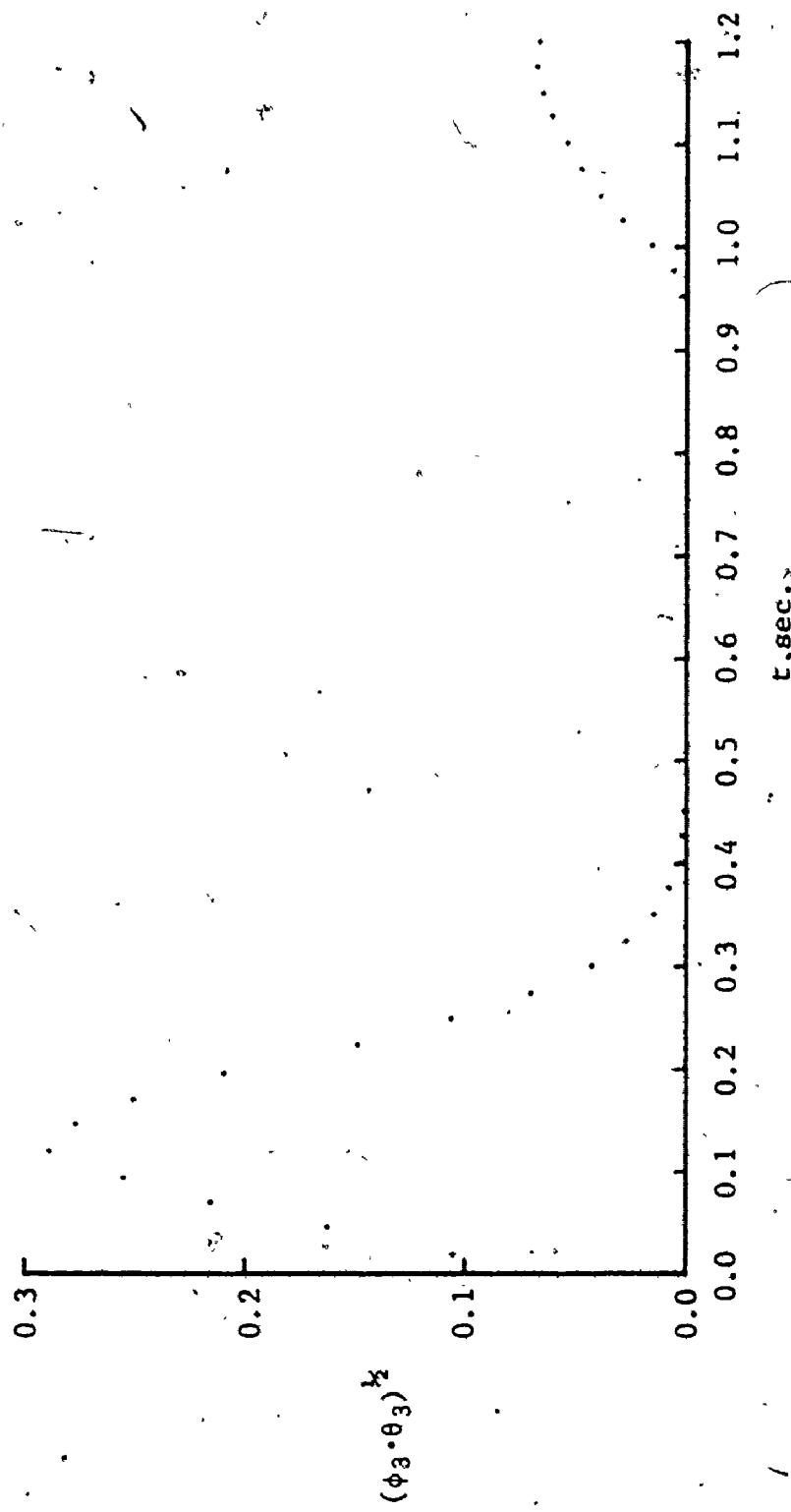


Figure 7.22 $(\phi_3 \cdot \theta_3)^{1/2}$ for M. rectus femoris

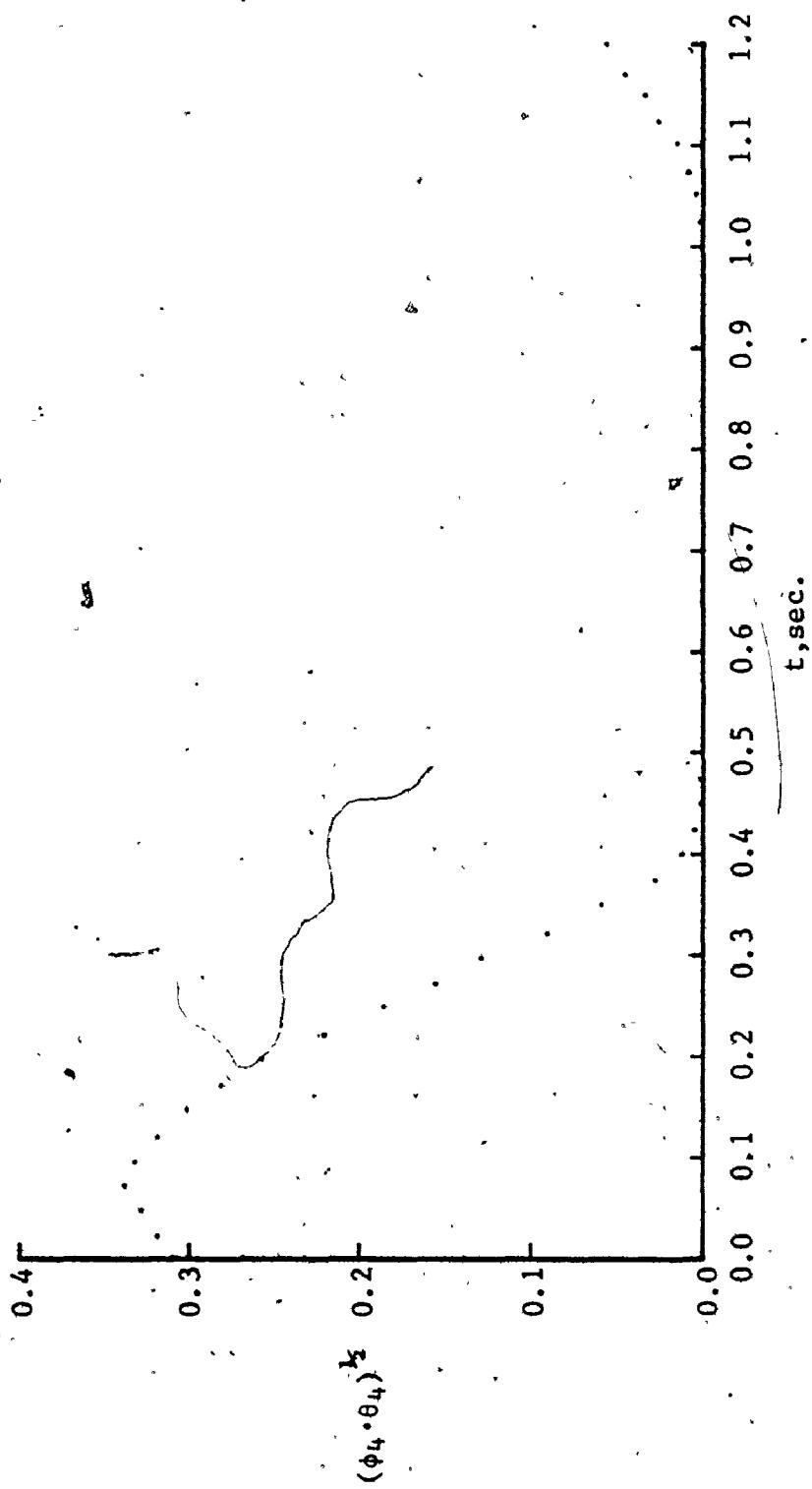


Figure 7.23 $(\phi_4 \cdot \theta_4)^{1/2}$ for Hamstring Muscle

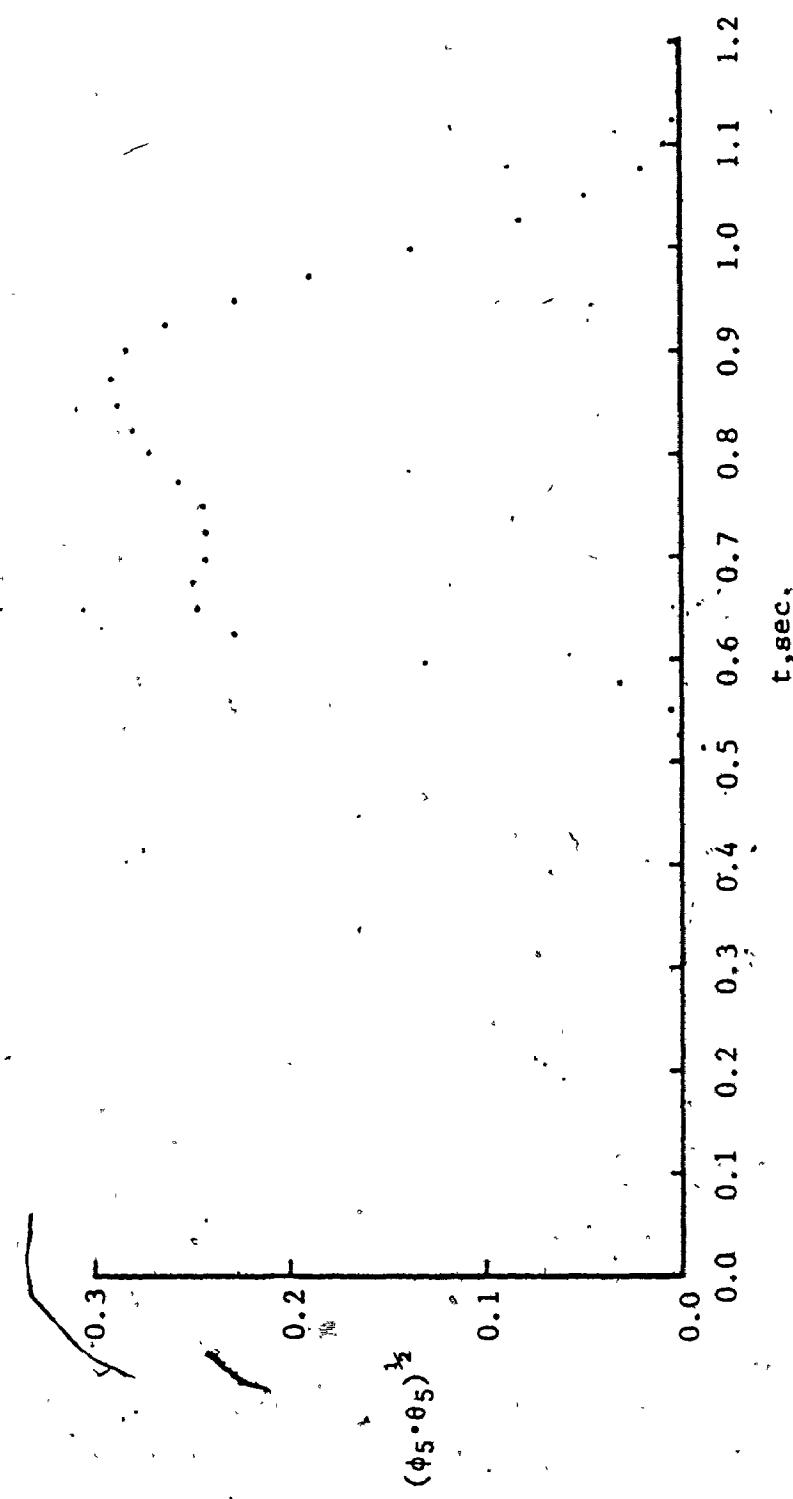


Figure 7.24 $(\phi\theta)^{1/2}$ for *M. gastrocnemius*

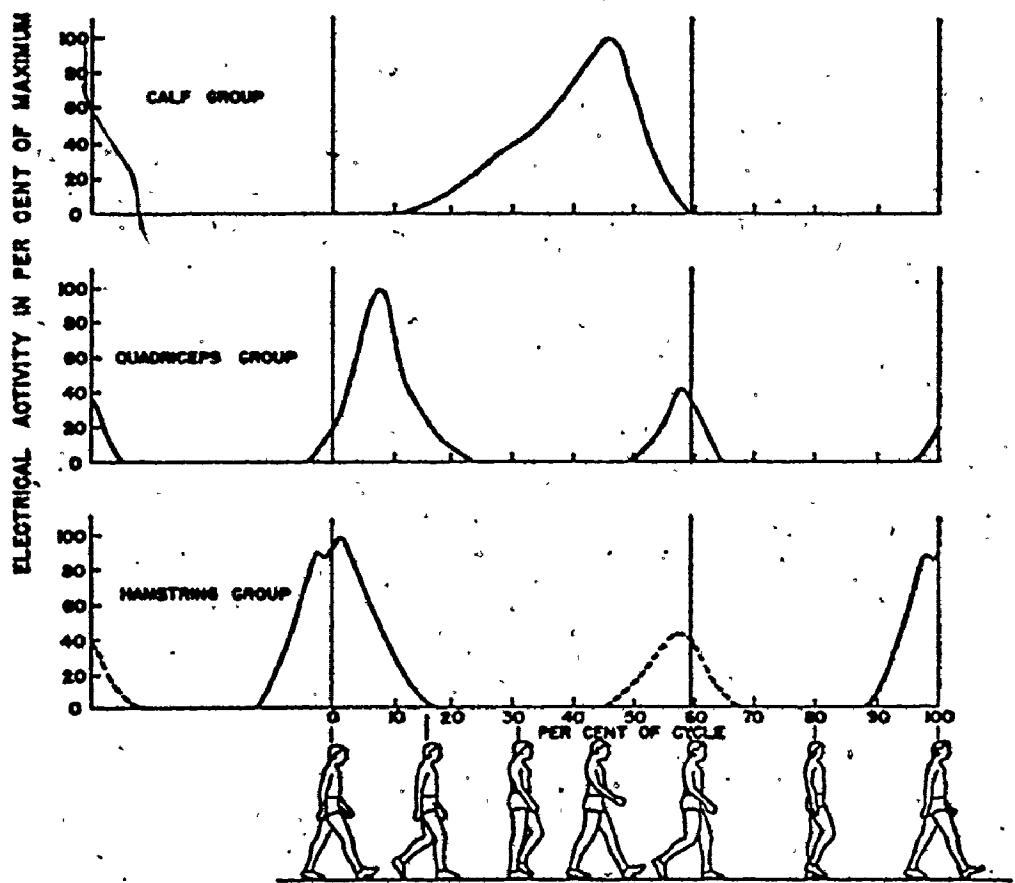


Fig. 15.21 Phasic action of major muscle groups during level

Figure 7.25 Typical EMG Envelopes. From [27]

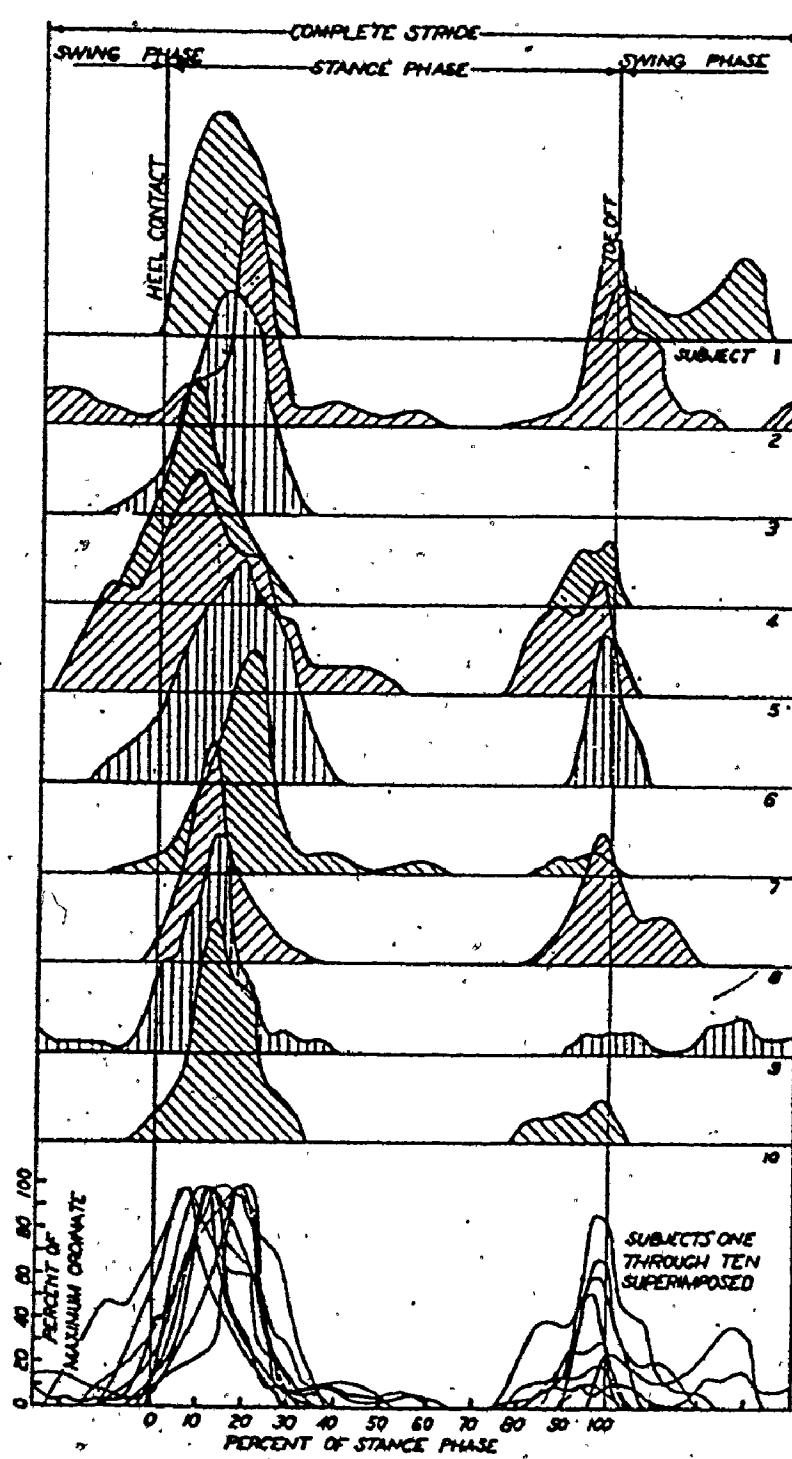


Fig. 15.9 Electromyographic summary curves for quadriceps group of 10 subjects. Cadence 95 steps per minute; level walking.

Figure 7.26 Evidence of Typical Scatter in EMG Records
From [27]

scatter in the shape of the EMG envelopes from subject to subject. Figure 7.26 shows this variability. Hence differences between the experimental and computed results are not unexpected. The general trend and phasic natures are seen to agree closely.

CHAPTER VIII

CONCLUDING DISCUSSION

8.1 Introduction

The preceding chapters have dealt with the development of a mathematical model of human locomotion, its numerical implementation and the results of the computer simulation. The results were compared to published experimental data and the agreement was found to be well within the limits one might expect purely on the basis of subject-to-subject differences.

This chapter serves to summarize the findings and significance of the work, suggesting areas in which it may be applied as well as recommending ways it may be extended.

8.2 Summary

Initially, it was set out to develop a detailed mathematical model with the capacity to accurately describe and predict the physiological phenomenon of walking. The model development, although extremely complicated and cumbersome, was relatively straight-forward. It was done in such a way as to allow for description of essentially any condition of walking, both normal and pathological. However, in order to validate the model, it had to be

implemented for numerical solution (the reader will have easily recognized the impossibility of exact analytical solution), and the specific condition of walking had also to be decided; a general solution would also have been an impossibility. It was suggested earlier that this condition be normal level walking since this would be the simplest to analyze.

It will have been noticed that the period of the cycle was chosen as two seconds. The author recognizes that this corresponds to an unnaturally slow gait. The results from a faster gait would have yielded results which would have been substantially different in magnitude but very similar in form. Nevertheless, the feasibility has been demonstrated along with the drastic reduction of computing time compared to previous attempts at similar simulations.

There are other drawbacks of this method of analysis, which although are not to be taken lightly, do not diminish the value of this research. Recommendations for means by which improvements can be made in the model follow in the next section.

Attention will now be focussed on some of the applications of the method to clinical problems. The first, one which was described earlier, is use of the present mathematical model and its numerical implementation in clinical FES programs. Provided the physical constants can

be measured or otherwise obtained (further comments regarding this follow in the next section) sufficient information can be obtained to correctly modulate the control signals to the appropriate muscles.

This can be viewed in the following way. The current practice of electrically stimulating muscles by means of surface (or indwelling) electrodes consists of only one pair of electrodes per muscle (group). The controls ϕ_i and θ_i could be considered as the frequency and amplitude, respectively, of the differential voltage applied to the electrodes. In this way all parts of the muscle would experience the same frequency of stimulation but the proportion of the muscle stimulated would depend upon the amplitude of the signal due to the attenuation of the voltage spikes as they pass through the muscle. Certainly multiple electrode sets would be advantageous, since, then, the voltage spikes could be delayed with respect to each other producing less "jerky" motion at low mean force levels.

The ultimate aim would be achieved if the model could be implemented on a small portable dedicated machine with the program running in "real time" and the boundary conditions continuously changeable. Provided a suitable digital-to-analog process has been developed, the subject could carry the computer with him and the signals allowing

him to do so would be generated by the computer itself. This "science fiction" idea is not unobtainable in light of the success of the present research.

A second application of the model in its present form might be in the testing of artificial joints. As an example, prosthetic knee joints are currently undergoing continuous development and improvement. It would be advantageous if the joints could be tested in the laboratory for longevity before clinical trials are begun. The problem in doing this is that it has been very difficult to determine the actual forces across the knee joint under normal walking conditions. Previous test equipment has relied upon determination of these forces based upon the ground reaction forces measured on force plates (cf. Figure 6.3). It has been seen, however, that during portions of the stance phase the agonist-antagonist muscle pairs are both active, particularly at the peaks of the ground forces (cf. Figures 7.11, 7.12, 7.13). This being the case the net force across the joint is significantly different from that component resulting purely from the ground reaction. Now that the correct muscle forces are known, the test equipment can simulate all of the forces on the knee, and the fatigue of the prosthetic joints more accurately predicted.

Lower limb prosthesis design could be enhanced by a slightly modified version of the present model. For the

modelling of locomotion, the limb segment masses, lengths, etc. were considered as constants. These could be considered as the variables of an "outer" optimization wherein the appropriate muscles have been removed, as in the case of the amputee. In this way the prosthesis could be designed in such a way as to minimize the energy consumed per step. Of course in this case Pontryagin's maximum principle could not be applied since the new variables are not "controls". The economy of such a design program would have to be investigated before its implementation.

These have been just some of the applications of the present model and are by no means exhaustive. Following implementation of the recommendations of the next section, the scope of utility of the model may be significantly enhanced.

8.3 Suggestions for Further Work

It was noted in Chapter IV that the pretibial muscle group was not included in the model, and the "foot-locking" assumption was made. From the energy consumption point of view, this is not expected to have a major effect on the model. However, for completeness as far as clinical application is concerned, the tibialis anterior muscle should be incorporated. This then involves determination of the muscle specific constants for a sixth muscle and

extension of the model to include two additional controls.

With regard to the muscle specific constants and functions (i.e. λ_i , $\dot{\lambda}_i$, d_i etc.) it is suggested that an experimental program be undertaken to establish mean values and variances of these quantities. It is most awkward to be forced to collect such data for each individual; especially when one considers that in many cases it might not be possible. Collection of the data on an individual who has already suffered some trauma affecting the leg musculature can be very difficult indeed, if at all possible. It would be advantageous to be able to draw upon a bank of data previously gathered.

Recently Hatze [33] has modified the control model of skeletal muscle with demonstrated success. Although the current model appears to yield satisfactory results, the new model, though more general, is somewhat simpler in form. The reduced complexity might significantly reduce the computation times involved in the simulation. Implementation of the new model would involve considerable modification to the program but the changes in the theoretical derivations would be minimal.

At this point, modifications to the solution algorithm could be incorporated to improve convergence in the swing phase (taking into account terminal conditions).

One further suggestion for modification involves the

specification of walking speed, cadence and step length. It is known that these three parameters are not independent. In fact, the cadence and step length are linearly proportional, and each is proportional to the square root of the walking speed. Moreover, the mean height of the hip above the ground and the amplitude of oscillation of the hip about this mean are also related to walking speed. Provided suitable expressions could be found, it would be advantageous to be able to specify only the walking speed, and with the physical constants known, i.e. segment lengths, foot length etc., the remaining parameters could be calculated.

With the foregoing suggestions incorporated into the model, the applications seem almost limitless, and a significant improvement in the progress of rehabilitation programs becomes imminent.

REFERENCES

- [1] Milner, M., A.O. Quanbury, and E.P. Edwards, Progress Report No. 1 - Human Locomotion by Ordered Electrostimulation of the Available Musculature, LTR-CS-6, April 1969, NRC, Ottawa, Canada.
- [2] Milner, M., A.O. Quanbury, and E.P. Edwards, Progress Report No. 2 - Human Locomotion by Ordered Electrostimulation of the Available Musculature, LTR-CS-11, Sept. 1969, NRC, Ottawa, Canada.
- [3] Hatze, H., The Complete Optimization of a Human Motion, *Math. Biosci.*, 28, 99-135 (1976).
- [4] Nubar, Y., and R. Contini, A Minimal Principle in Biomechanics, *Bull. Math. Biophys.*, 23, 377-391 (1961).
- [5] Beckett, R., and K. Chang, An Evaluation of the Kinematics of Gait by Minimum Energy, *J. Biomech.*, 1, 147-159 (1968).
- [6] Chow, C.K. and D.H. Jacobson, Studies of Human Locomotion VIA Optimal Programming, *Math. Biosci.*, 10, 239-306 (1971).
- [7] Hatze, H., A Myocybernetic Control Model of Skeletal Muscle, CSIR Special Report, Wisk 220, Pretoria, S. Africa (1976).

- [8] Basmajian, J.V., Muscles Alive, Bailliere Tindall and Cox, London (1962).
- [9] Morrison, J.B., The Mechanics of Muscle Function in Locomotion, J. Biomech., 3, 431-451, (1970).
- [10] Gelfand, I.M., and S.V. Fomin, Calculus of Variations, (Translated from Russian), Prentice-Hall, Englewood Cliffs, New Jersey (1963).
- [11] Pontryagin, L.S., Optimal Control Processes, USP. Mat. Nauk, 14, No. 1, 3 (1959).
- [12] Pontryagin, L.S., V.G. Boltyanskii, R.V. Gamkrelidze and E.F. Mishchenko, The Mathematical Theory of Optimal Processes, (Translated from Russian), Wiley-Interscience, New York (1962).
- [13] Hatze, H., A Control Model of Skeletal Muscle and its Application to a Time-Optimal Biomotion, Ph.D. Thesis, University of South Africa, (1975).
- [14] Bahler, A.S., Series Elastic Component of Mammalian Skeletal Muscle, Am. J. Physiol., 213, 1560-1564, (1967).
- [15] Hill, A.V., The Series Elastic Component of Muscle, Proc. Roy. Soc., B 137, 273-280, (1950).
- [16] Jewell, B.R. and D.R. Wilkie, An Analysis of the Mechanical Components in Frog's Striated Muscle, J. Physiol., 143, 515-540, (1958).

- [17] Yamada, H., Strength of Biological Materials, Williams and Wilkins, Baltimore, Md., (1970).
- [18] Ebashi, S., and M. Endo, Calcium and Muscle Contraction, Progr. Biophys. and Mol. Biol., 18, 123-183, (1968).
- [19] Mommaerts, W.F.H.M., Muscular Contraction, Physiol. Rev., 49, 427-508, (1969).
- [20] Woledge, R.C., Heat Production and Chemical Change in Muscle, Progr. Biophys. and Mol. Biol., 22, 39-74, (1971).
- [21] Hatze, H. and J.D. Buys, Energy-optimal Controls in the Mammalian Neuromuscular System, CSIR Special Report, WISK 243, Pretoria, S. Africa, (1977).
- [22] Gibbs, C.L. and W.R. Gibson, Energy Production of Rat Solens Muscle, Am. J. Physiol., 223, 864-871, (1972).
- [23] Hill, A.V., The Heat of Shortening and the Dynamic Constants of Muscle, Proc. Roy. Soc., 126 B, 136-195, (1938).
- [24] Davies, R.E., M.J. Kushmerick and R.E. Larson, Professor A.V. Hill's "Further Challenge to Biochemists": ATP, Activation and Heat of Shortening of Muscle, Nature, 214, 148-151, (1967).
- [25] Chow, C.K. and D.H. Jacobson, Further Studies of Human Locomotion; Postural Stability and Control, Math. Biosci., 15, 93-108 (1972).

- [26] Hatze, H., Private Communications, (1977).
- [27] Eberhart, H.D., V.T. Inman, and B. Bresler, The Principal Elements in Human Locomotion, 437-471, in Human Limbs and their Substitutes, Klopsteg and Wilson Editors, McGraw-Hill Book Co., New York (1954).
- [28] Bresler, B. and J.P. Frankel, The Forces and Moments in the Leg During Level Walking, University of California (Berkeley), Prosthetic Devices Research Project, Report Series 11, Issue 6 (1948).
- [29] Seireg, A. and R.J. Arvikar, The Prediction of Muscular Load Sharing and Joint Forces in the Lower Extremities During Walking, J. Biomechanics, 8, 89-102 (1975).
- [30] Hatze, H., The Relative Contribution of Motor Unit Recruitment and Rate Coding to the Production of Static Isometric Muscle Force, CSIR Special Report, WISK 244, Pretoria, S. Africa (1977).
- [31] Biro, G. and L.D. Partridge, Analysis of Multiunit Spike Records, J. Appl. Physiol., 30, 521-526 (1971).
- [32] Harding, R.H. and R.N. Sen, Evaluation of Total Muscular Activity by Quantification of Electromyograms through a Summing Amplifier, Med. and Biol. Eng., 3, 343-356 (1970).

[33] Hatze, H., A General Myocybernetic Control Model of
Skeletal Muscle, CSIR Special Report, WISK 280,
Pretoria, S. Africa (1977).

APPENDIX A

SUMMARY OF EQUATIONS OF MOTION AND ENERGY

(i) Dynamical Equations:

$$\left. \begin{array}{l} \dot{z}_1 = z_3 \\ \dot{z}_2 = z_4 \\ \dot{z}_3 = \Delta^{-1} [R_1 B_1 + R_2 B_2] \\ \dot{z}_4 = \Delta^{-1} [R_1 B_2 + R_2 B_3] \end{array} \right\} \quad (3.23)$$

$$R_1 = -C_{\epsilon+2} z_4 \sin z_2 (z_4 - 2z_3) - (C_\epsilon \sin z_1 - C_{\epsilon+1} \cos z_1) (v + g) + M_1$$

$$R_2 = -C_{\epsilon+2} z_3^2 \sin z_2 - C_{\epsilon+2} \cos z_2 (v + g) + M_2$$

$$B_1 = A_{\delta+1} \quad B_2 = A_{\delta+1} + C_{\epsilon+2} \cos z_2$$

$$B_3 = A_\delta + 2 C_{\epsilon+2} \cos z_2 \quad \Delta = B_1 B_3 - B_2^2$$

$$A_\delta = I_\delta + m_\delta \alpha_\delta^2 + m_{\delta+1} \lambda_\delta^2$$

$$A_{\delta+1} = I_{\delta+1} + m_{\delta+1} \alpha_{\delta+1}^2$$

$$C_\epsilon = m_\delta \alpha_\delta + m_{\delta+1} \lambda_\delta$$

$$C_{\epsilon+1} = m_{\delta+1} \alpha_{\delta+1}$$

$$C_{\epsilon+2} = C_{\epsilon+1} \lambda_\delta$$

body-specific constants

$$\tilde{v}(t) = \beta_1 \sin \frac{4\pi}{\tau} (t + \beta_2 \tau) \quad (3.10)$$

$$\begin{aligned} M_1 &= u_\delta + u_\delta^P + M_\alpha^{\delta+1} + Y_{\delta+1} (\ell_\delta \sin z_1 - \ell_{\delta+1} \sin(z_2 - z_1)) \\ &\quad + X_{\delta+1} (\ell_\delta \cos z_1 + \ell_{\delta+1} \cos(z_2 - z_1)) \end{aligned} \quad (3.20)$$

$$\begin{aligned} M_2 &= u_{\delta+1} + u_{\delta+1}^P - M_\alpha^{\delta+1} + Y_{\delta+1} \ell_{\delta+1} \sin(z_2 - z_1) \\ &\quad - X_{\delta+1} \ell_{\delta+1} \cos(z_2 - z_1) \end{aligned} \quad (3.21)$$

$$Y_{\delta+1} = f_v \quad X_{\delta+1} = f_h \quad (3.39)$$

$$M_\alpha^{\delta+1} = f_v (x_{\delta+1}^P - x_{\delta+1}^A) + f_h (y_{\delta+1}^A - y_{\delta+1}^P)$$

$$\left. \begin{array}{l} u_\delta^P(z_i), u_{\delta+1}^P(z_i) \\ f_h(t), f_v(t) \\ x_{\delta+1}^P(t), y_{\delta+1}^P(t) \end{array} \right\} \text{to be experimentally determined}$$

iii) Constraint Relations

a) Stance Portion ($t_o \leq t \leq t'$)

$$S_1(z; t) = \tilde{v}(t) + \ell_\delta \cos z_1 + \ell_{\delta+1} \cos(z_2 - z_1) - v_o + p(t)$$

$$S_2(z; t) = h_o(t_o + t) + \ell_\delta \sin z_1 + \ell_{\delta+1} \sin(z_2 - z_1) - q(t)$$

$p(t), q(t)$ to be experimentally determined

$$\left. \begin{array}{l} S_1(z; t) = 0 \\ S_2(z; t) = 0 \\ z_2(t) > 0 \end{array} \right\} \quad (t_o \leq t \leq t') \quad (3.33)$$

b) Deploy Portion ($t' < t \leq t''$)

$$\begin{aligned}
 S_3(z; t) = & \ell_\delta^2 + \ell_{\delta+1}^2 - (d_\alpha^{1\delta})^2 + (v_1 - \tilde{v}(t))^2 \\
 & + (h_1 - h_0 t)^2 + 2\ell_\delta \ell_{\delta+1} \cos z_2 \\
 & - 2(v_1 - \tilde{v}(t)) [\ell_\delta \cos z_1 + \ell_{\delta+1} \cos(z_2 - z_1)] \\
 & - 2(h_1 - h_0 t) [\ell_\delta \sin z_1 - \ell_{\delta+1} \sin(z_2 - z_1)] \quad (3.28)
 \end{aligned}$$

$$\begin{aligned}
 v_1 = & \tilde{v}(t') + \ell_\delta \cos z_1(t') + \ell_{\delta+1} \cos[z_2(t') - z_1(t')] \\
 & + d_\alpha^{1\delta} \sin \alpha(t') \quad (3.26)
 \end{aligned}$$

$$\begin{aligned}
 h_1 = & h_0 t' + \ell_\delta \sin z_1(t') - \ell_{\delta+1} \sin[z_2(t') - z_1(t')] \\
 & + d_\alpha^{1\delta} \cos \alpha(t') \quad (3.27)
 \end{aligned}$$

$$\alpha(t') = \sin^{-1} \frac{p(t')}{d_\alpha^{1\delta}}$$

$$\begin{aligned}
 \alpha(t) = & \frac{1}{d_\alpha^{1\delta}} \cos^{-1} [h_1 - h_0(t) - \ell_\delta \sin z_1 \\
 & + \ell_{\delta+1} \sin(z_2 - z_1)] \quad (3.27)
 \end{aligned}$$

$$\left. \begin{array}{l} S_3(z; t) = 0 \\ z_2(t) > 0 \end{array} \right\} \quad (t' < t \leq t'') \quad (3.29)$$

c) Swing Portion ($t'' < t \leq t_1$)

$$\begin{aligned}
 S_4(z; t) = & \tilde{v}(t) + \ell_\delta \cos z_1 + \ell_{\delta+1} \cos(z_2 - z_1) \\
 & + d_\alpha^{2\delta} \sin \zeta(t) - v_0 \quad (3.30)
 \end{aligned}$$

$$\zeta(t) = \sigma_{\max} + z_2(t) - z_1(t) - \frac{\pi}{2}$$

$$\zeta(t'') = \alpha(t''); \quad \sigma_{\max} = \alpha(t'') + z_1(t'') - z_2(t'') + \frac{\pi}{2}$$

$$\left. \begin{array}{l} S_4(z; t) \leq 0 \\ z_2(t) > 0 \end{array} \right\} \quad (t'' < t \leq t_1) \quad (3.30)$$

$$\left. \begin{array}{l} S_4(z; t) \leq 0 \\ z_2(t) > 0 \end{array} \right\} \quad (3.32)$$

iii) Dynamical to Muscular Coupling Equations

$$u_\delta = F_{\rho+1}^{\text{SE}} d_{\rho+1} + (F_{\rho+3}^{\text{SE}} + F_{\rho+3}^{\text{PE}}) d_{\rho+3} - (F_{\rho+4}^{\text{SE}} + F_{\rho+4}^{\text{PE}}) d_{\rho+4} \quad (4.3)$$

$$u_{\delta+1} = -F_{\rho+2}^{\text{SE}} d_{\rho+2} - (F_{\rho+3}^{\text{SE}} + F_{\rho+3}^{\text{PE}}) d_{\rho+2} + (F_{\rho+4}^{\text{SE}} + F_{\rho+4}^{\text{PE}}) d_{\rho+4} + F_{\rho+5}^{\text{SE}} d_{\rho+5}$$

$$\left. \begin{array}{l} d_j(z_i) \\ F_k^{\text{PE}}(\lambda_k, \dot{\lambda}_k) \end{array} \right\} \quad \text{to be experimentally determined}$$

iv) Musculo-Mechanical Equations

$$\dot{z}_{4+j} = \frac{1}{C_{\rho+j}} \tanh^{-1} \left\{ \left[\frac{F_{\rho+j}^{\text{SE}}(\lambda_{\rho+j}, z_{4+j})}{b_{1,\rho+j} k(z_{4+j}) [1 - b_{2,\rho+j}^2 / (z_{4+j} + b_{3,\rho+j})^2]^{1/2} z_{14+j}} - 1 \right] \right. \\ \left. \times \tanh(b_{4,\rho+j} \exp[-b_{5,\rho+j} (z_{4+j} - b_{6,\rho+j})^2 - 1.43(z_{9+j} - 1)^2]) \right\} \quad (4.16)$$

$$\dot{z}_{9+j} = m_{\rho+j} \{ \Phi_j - z_{9+j} \} \quad (4.18)$$

$$z_{14+j} = \theta_{\rho+j} z_{9+j} + 0.005 \quad (4.30)$$

for $j = 1, \dots, 5$

where:

$$F_{\rho+j}^{SE} = b_{7,\rho+j} [\exp(b_{8,\rho+j} \{\lambda_{\rho+j} - ([z_{4+j} + b_{9,\rho+j}]^2 - b_{10,\rho+j})^{1/2} - b_{11,\rho+j}\}) - 1] \quad (4.5)$$

$$k(z_{4+j}) = b_{12,\rho+j} + b_{13,\rho+j} (\exp(-b_{14,\rho+j} [b_{15,\rho+j} + b_{16,\rho+j} (z_{4+j})^{n_{\rho+j}}])) \\ \times \sin[b_{17,\rho+j} z_{4+j} + b_{18,\rho+j}] \quad (4.17)$$

$$\phi_j = 0.5025 + 0.4975 \tanh[19.3 \{ \exp(-b_{19,\rho+j} [z_{4+j} - b_{20,\rho+j}]^2) \} \\ \times \{ \phi_{\rho+j} - b_{21,\rho+j} [\exp(-b_{22,\rho+j} z_{4+j})] - b_{23,\rho+j} \}] \quad (4.20)$$

and $b_{1,\rho+j} \dots b_{23,\rho+j}$, $m_{\rho+j}$, $n_{\rho+j}$, $\bar{C}_{\rho+j}$ are muscle-specific constants which along with $\lambda_{\rho+j}(z_1, z_2)$ must be determined experimentally.

$$F_{\rho+5}^{SE} (\lambda_{\rho+5}, z_9) \geq (M_{\alpha}^{\delta+1} - u_{\alpha, \delta+1}^P) / d_{\rho+5}^{\alpha} \quad (4.28)$$

v) System Energetics

$$\int_{t_0}^{t_1} \xi(z, \phi; \theta) dt = \int_{t_0}^{t_1} \left[\sum_{j=1}^5 \xi_{\rho+j}^j (z_i, \phi_{\rho+j}, \theta_{\rho+j}) \right] dt$$

where

$$\xi_{\rho+j}^j = \bar{a}_{1,\rho+j} [(1 - \exp(-0.24 - \bar{a}_{2,\rho+j}/\phi_{\rho+j})) \phi_{\rho+j} \\ + (0.82 - 4.81 (z_{4+j}/\bar{n}_{\rho+j})^2) z_{9+j}] \theta_{\rho+j} G_{\rho+j} \quad (5.17)$$

$$- [c_{\rho+j} \theta_{\rho+j} (z_{9+j})^{0.72} k(z_{4+j}) + F_{\rho+j}^{SE} (z_i, z_{4+j})] \dot{z}_{4+j} \\ + |\dot{\lambda}_{\rho+j}(z_i, \dot{z}_i)| F_{\rho+j}^{PE}(z_i)$$

$$(\bar{a}_{1,\rho+j}, \bar{a}_{2,\rho+j}) = \begin{cases} (177.7, 0.19), & \text{for fast muscle} \\ (106.8, 0.32), & \text{for slow muscle} \end{cases} \quad (5.16)$$

$c_{\rho+j}$, $\bar{\eta}_{\rho+j}$, $G_{\rho+j}$ to be experimentally determined

$\dot{\lambda}_{\rho+j}(z_i, \dot{z}_i) = \dot{\lambda}_{\rho+j}(z_1, z_2, z_3, z_4)$ found by differentiating the experimentally determined functions $\lambda_{\rho+j}(z_1, z_2)$.

Note: In all of sections (i) to (v) the values of δ , ϵ , and ρ are chosen so that

$$(\delta, \epsilon, \rho) = \begin{cases} (1, 1, 0), & \text{for left leg} \\ (3, 4, 5), & \text{for right leg} \end{cases}$$

and the simulation is carried out for each leg independently if there are significant differences between them. Otherwise, one simulation is sufficient with the knowledge that the walking pattern differs by a 180° phase shift between legs.

It is of course imperative that starting values of the functions $z_1(t), \dots, z_{14}(t)$ be known at time $t = t_0$.

APPENDIX B
PROGRAM LISTING

PROGRAM MAIN(INPUT, OUTPUT, DATA1, PUNCH, TAPE3= INPUT, TAPE6= OUTPUT, TAPMAI
1E7= DATA1, TAPE8= PUNCHED)

C

C*****THIS PROGRAM INITIALIZES THE COMMON
C*****BLOCKS USED BY CASP IV AND PASSES
C*****ON THE LOGICAL UNIT NO'S OF READER
C*****AND PRINTER. CONTROL THEN PASSES TO
C*****CASP IV.

C

```
DIMENSION NSET(5000)
COMMON QSET(5000)
COMMON /CCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MLE(100), MSTOP, NCRDR, NMAI
1NAPO, NNAPT, NNATR, NNFILE, NNQ(100), NNTRY, NPRINT, PPARM(50,4), TNOW, TTBEGLMAI
2, TTCLR, TTFIN, TTRIB(25), TTSET
COMMON /CCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, MAI
1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX
COMMON /CCOM3/ AAERR, DTMAX, DTMIN, DTSAV, IITES, LLERR, LLSAV, LLSEV, RHEMAI
1RR, TTLAS, TTSAV
COMMON /CCOM4/ DTPLT(10), HHLOW(25), HHWID(25), IIICRD, IIITAP(10), JJCELMAI
1(500), LLABC(25,2), LLABH(25,2), LLABP(11,2), LLABT(25,2), LLPHI(10), LLMAI
2PLO(10), LLPLT, LLSUP(15), LLSYM(10), MMPTS, NNCEL(25), NNCLT, NNHIS, NNPLMAI
3T, NNPTS(10), NNSTA, NNVAR(10), PPHI(10), PPLOC(10)
COMMON /CCOM5/ IIevt, IISED(6), JJBEG, JJCLR, MMNIT, MMON, NNAME(3), NNGFMAI
1I, NNDAY, NNPT, NNSET, NNPRJ, NNPRM, NNRNS, NNRUN, NNSTR, NNYR, SSEED(6)
COMMON /CCOM6/ EENQ(100), IINN(100), KKRNK(100), MMAXQ(100), QQTIM(100)MAI
1), SSOBV(25,5), STTPV(25,6), VVNQ(100)
EQUIVALENCE (NSET(1), QSET(1))
```

C

```
NCRDR=7
NPRINT=6
CALL GASP
STOP
END
```

```

SUBROUTINE INTLC          INT    10
C
C*****THIS SUBROUTINE IS CALLED BY GASP IV      INT    20
C*****TO SET UP TIME EVENTS SO THAT TIME      INT    30
C*****INCREMENTS COINCIDE WITH SPECIFIED      INT    40
C*****TIMES. VIZ. TPRIM, T2PRIM.      INT    50
C*****INITIALIZATION SUBROUTINE IS ALSO CALLED      INT    60
C
COMMON /GCOM1/ ATRIB(25),JEVNT,MFA,MFE(100),MLE(100),MSTOP,NCRDR,NINT   90
1NAPO,NNAPT,NNATR,NNFIL,NNQ(100),NNTRY,NPRNT,PPARM(50,4),TNOW,TTBEGINT 100
2,TTCLR,TIFIN,TTRIB(25),TTSET      INT    110
COMMON /TIME/ TINIT,TPRIM,T2PRIM,TEND      INT    120
C
CALL INPUT      INT    130
CALL SET      INT    140
C
C*****SET UP TIME EVENTS      INT    150
C
ATRIB(1)=TPRIM      INT    160
ATRIB(2)=1.      INT    170
CALL FILEM (1)      INT    180
ATRIB(1)=T2PRIM      INT    190
ATRIB(2)=2.      INT    200
CALL FILEM (1)      INT    210
ATRIB(1)=TEND      INT    220
ATRIB(2)=3.      INT    230
CALL FILEM (1)      INT    240
RETURN      INT    250
END      INT    260
INT    270
INT    280
INT    290

```

SUBROUTINE INPUT

C*****THIS SUBROUTINE IS USED TO INITIALIZE
C*****ALL VARIABLES AND CONSTANTS. CALLED
C*****BY INTLC

```

COMMON /GCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MLE(100), MSTOP, NCRDR, NINP    10
1NAPO, NNAPT, NNATH, NNFIL, NNQ(100), NNTRY, NPRINT, PPARM(50,4), TNOW, TTBEGINP   20
2, TTCLR, TIFIN, TIR, B(25), TISET                                         INP    30
COMMON /GCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, INP    40
1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX                           INP    50
COMMON /BODCON/ B(26,5), A1,A2,C1,C2,C3,GRAV                                INP    60
COMMON /PHYS/ ALL1, ALL2                                         INP    70
COMMON /ENERGY/ ABAR(2,5), C(5), ETA(5), GG(5)                            INP    80
COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND                                     INP    90
COMMON /WALK/ BETA1, BETA2, TAV, XHOR, VMEAN, XNOT                         INP
COMMON /FOOT/ DALPH1, DALPH2                                         INP    100
COMMON /OPT/ RMAX(10), RMIN(10), N, NCONS, NEQUS, IPRINT, IDATA               INP    110
COMMON /MUSCLE/ PPHI(5), PTHETA(5), PHILAS(5), THELAS(5)                   INP    120
COMMON /GAST/ FSE, NFSE                                         INP    130
COMMON /VARIAB/ ALAMB(10), D(23), DLDDZ(4,10)                            INP    140
COMMON /EXPT/ FPE3, FPE4, DFPE3(9), DFPE4(9), UP1, UP2, DUP1(4), DUP2(4), INP    150
1SD(6), DSD(6,9)                                         INP    160
COMMON /FORCE/ TSCALE, VSCALE, HSCALE, XSCALE, TT1, TT2, TT3, TV, TH, TX, W   INP    170
COMMON /MODE/ MODE, IMODE                                         INP    180
COMMON /SAVE/ SAVTH, OPTTIM                                         INP    190
LOGICAL IMODE, LFLAG                                         INP    200
DIMENSION X(6), FSE(5), TT1(45), TT2(47), TT3(18), TV(45), TH(47), INP    210
1 TX(18)
DATA RMAX, RMIN/10*1.0, 10*0.0/                                         INP    220
INP    230
INP    240
INP    250
INP    260
INP    270
INP    280
INP    290
INP    300
INP    310
INP    320
INP    330

```

C*****READ INPUT DATA

```

C
READ (7,60) ((B(I,J),J=1,5), I=1,26)
READ (7,70) A1,A2,C1,C2,C3,ALL1,ALL2,GRAV,W
READ (7,80) (ABAR(1,I),ABAR(2,I),C(I),ETA(I),GG(I), I=1,5)
READ (7,90) TINIT, TPRIM, T2PRIM, TEND
READ (7,100) BETA1, BETA2, TAV, XHOR, VMEAN, XNOT
READ (7,110) DALPH1, DALPH2
READ (7,120) N, NCONS, NEQUS, IPRINT, IDATA, INDEX, MAXM
READ (7,130) F, FG, R, REDUCE
READ (7,140) (X(I), I=1,6)
READ (7,150) TSCALE, VSCALE, HSCALE, XSCALE
READ (7,160) (TT1(I), TV(I), I=1,45)
READ (7,160) (TT2(I), TH(I), I=1,47)
READ (7,160) (TT3(I), TX(I), I=1,18)
DO 10 I=1,5
  READ (7,170) PTHETA(I), PPHI(I)
CONTINUE
10

```

C*****PRINT OUT INPUT DATA

```

C
WRITE (6,180) (I,(B(I,J),J=1,5), I=1,26)
WRITE (6,190) A1,A2,C1,C2,C3,ALL1,ALL2,GRAV,W
WRITE (6,200) (I,ABAR(1,I),ABAR(2,I),C(I),ETA(I),GG(I), I=1,5)
WRITE (6,210) TINIT, TPRIM, T2PRIM, TEND
WRITE (6,220) BETA1, BETA2, TAV, VMEAN, XHOR, XNOT
WRITE (6,230) DALPH1, DALPH2
WRITE (6,240) X
WRITE (6,250) TSCALE, VSCALE, (TT1(I), TV(I), I=1,45)
WRITE (6,260) TSCALE, HSCALE, (TT2(I), TH(I), I=1,47)
WRITE (6,270) TSCALE, XSCALE, (TT3(I), TX(I), I=1,18)
WRITE (6,300)

```

MODE=0

DO 20 I=1,16

LFLAG(I)=.FALSE.

CONTINUE

SS(1)=X(1)

SS(2)=X(2)

SS(3)=X(3)

SS(4)=X(4)

DD(3)=X(5)

DD(4)=X(6)

INP	10
INP	20
INP	30
INP	40
INP	50
INP	60
INP	70
INP	80
INP	90
INP	100
INP	110
INP	120
INP	130
INP	140
INP	150
INP	160
INP	170
INP	180
INP	190
INP	200
INP	210
INP	220
INP	230
INP	240
INP	250
INP	260
INP	270
INP	280
INP	290
INP	300
INP	310
INP	320
INP	330
INP	340
INP	350
INP	360
INP	370
INP	380
INP	390
INP	400
INP	410
INP	420
INP	430
INP	440
INP	450
INP	460
INP	470
INP	480
INP	490
INP	500
INP	510
INP	520
INP	530
INP	540
INP	550
INP	560
INP	570
INP	580
INP	590
INP	600
INP	610
INP	620
INP	630
INP	640
INP	650
INP	660
INP	670
INP	680
INP	690
INP	700
INP	710
INP	720
INP	730
INP	740

```

*****EVALUATE EXPERIMENTAL VALUES
C
    CALL DLVAL
    CALL DLAMB
    CALL EXPT
C
C*****INVERT DYNAMICAL EQUATIONS.
C
    XX=X(2)-X(1)
    TC1=COS(X(1))
    TS1=SIN(X(1))
    TC2=COS(X(2))
    TS2=SIN(X(2))
    TCX=COS(XX)
    TSX=SIN(XX)
    BB1=A2
    BB2=A2+C3*TC2
    BB3=A1+2.*C3*TC2
    VG=VDDOT(TTBEG)+GRAV
    R1=X(5)*BB3-X(6)*BB2
    R2=-X(5)*BB2+X(6)*BB1
    AM1=R1+C3*X(4)*TS2*(X(4)-2.*X(3))
    AM1=AM1+(C1*TS1-C2*TC1)*VG
    AM2=R2+C3*(X(3)**2)*TS2+C3*TC2*VG
    YD1=FVERT(TTBEG)
    XD1=FRORZ(TTBEG)
    AM21=YD1*ALL2*TSX
    AM22=XD1*ALL2*TCX
    AM11=YD1*ALL1*TS1-AM21
    AM12=XD1*ALL1*TC1+AM22
    AMALPH=YD1*(XP(TTBEG)-PP(TTBEG))+XD1*(QQ(TTBEG)-YP(TTBEG))
    U1=AM1-UP1-AMALPH-AM11-AM12
    U2=AM2-UP2+AMALPH-AM21+AM22
    FBAR=3.62279731

C
C*****CALCULATE FORCE LEVELS
C
    FSE(5)=0.005*FBAR*B(7,5)
    F4=(U2-FSE(5)*SD(6))/(SD(5)-9.00*SD(2))
    F3=(U1+F4*(SD(4)-3.969*SD(1))/(SD(3)-0.441*SD(1))
    FSE(2)=9.00*F4-F3
    FSE(1)=0.441*FSE(2)
    FSE(3)=F3-FPE3
    FSE(4)=F4-FPE4
    DO 30 I=1,5
    J=4+I
    K=20+I
    T1=ALOC(FSE(I)/B(7,I)+1)
    T1=T1/B(8,I)-ALAMB(I)+B(11,I)
    SS(J)=SQRT(T1**2+B(10,I))-B(9,I)
    IF (B(9,I).EQ.0.0) SS(J)=-T1
    TEMP=EXP(-B(19,I)*(SS(J)-B(20,I))**2)
    TEMP1=PPHI(I)-B(21,I)*(EXP(-B(22,I)*SS(J))-B(23,I))
    SS(K)=0.5023+0.4975*TANH(19.3*TEMP*TEMP1)
    WRITE (6,280) I,PTHETA(I),PPHI(I),SS(K)
30  CONTINUE
    WRITE (6,310)
    DO 40 I=1,5
    L=I+4
    PHILAS(I)=PPHI(I)
    THELAS(I)=PTHETA(I)
    F1=FSE(I)/(FBAR*B(7,I))
    WRITE (6,290) FSE(I),F1,SS(L),I
40  CONTINUE
    IMODE=.TRUE.

C
C*****CALCULATE REMAINING,VARIABLES
C
    CALL STATE1(IMODE)
    DO 50 J=1,9
    L=J+10
    AR=DD(J)
    IF (AR.EQ.0.0) AR=1.E-6
    SS(L)=DD(10)/(9.0*AR)
50  CONTINUE

```

```

50 CONTINUE          INP 1490
SS(20)=-1.0          INP 1500
C
C***RECORD INITIAL VALUES
C
SAVTIM=TTBEG          INP 1510
OPTTIM=TTBEG          INP 1520
DTNOW=0.0              INP 1530
CALL ORDER             INP 1540
CALL SSAVE             INP 1550
IMODE=.FALSE.           INP 1560
LFLAG(17)=.TRUE.        INP 1570
RETURN                 INP 1580
C
C
60 FORMAT ((5F12.5))    INP 1590
70 FORMAT (4E11.4/4E11.4/E11.4) INP 1600
80 FORMAT ((5E11.4))    INP 1610
90 FORMAT (4F5.3)        INP 1620
100 FORMAT (6E12.5)       INP 1630
110 FORMAT (2F5.3)        INP 1640
120 FORMAT (7I3)          INP 1650
130 FORMAT (4F5.3)        INP 1660
140 FORMAT (6F7.4)        INP 1670
150 FORMAT (4E11.4)        INP 1680
160 FORMAT ((2E11.4))    INP 1690
170 FORMAT (3F11.8)        INP 1700
180 FORMAT (1H1,6X,1H1,9X,6HB(I,1),6X,6HB(I,2),6X,6HB(I,3),6X,6HB(I,4)) INP 1710
1,6X,6HB(I,5)//,1X,26(5X,I2,5X,5F12.5//,1X)          INP 1720
190 FORMAT (1H1,2X,3HA1=,F9.4,5X,3HA2=,F9.4,5X,3HC1=,F9.4,5X,3HC2=,F9.4,5X,3HC3=,F9.4,//1X,5HALL1=,F9.4,3X,5HALL2=,F9.4,//,1X,5HGRAV=,F9.4//,2X,2HW=,F9.4//) INP 1730
* 14,5X,3HC3=,F9.4,//1X,5HALL1=,F9.4,3X,5HALL2=,F9.4,//,1X,5HGRAV=,F9.4//,2.4,6X,2HW=,F9.4//)          INP 1740
200 FORMAT (1H0,5X,1H1,5X,9HABAR(1,1),2X,9HABAR(2,1),6X,4HB(I),7X,6HETIMP(1A(I),5X,3HGG(I)//,1X,5(5X,I1,5X,F8.4,3X,F6.4,6X,F8.4,5X,F6.4,4X,F6.4//,1X))          INP 1750
210 FORMAT (1H0,6HTINIT=,F7.3,2X,6HTPRIM=,F7.3,2X,7HT2PRIM=,F7.3,2X,5HIMP(1TEND=,#7.3//))          INP 1760
220 FORMAT (1H0,6HBETA1=,F8.5,2X,6HBETA2=,F8.5,2X,4HTAU=,F8.5,2X,6HVME(F8.5//,2X,5HXBON=,F8.5,3X,5HXNOT=,F8.5//))          INP 1770
230 FORMAT (1H0,7HDALPH1=,F5.3,7X,7HDALPH2=,F5.3//)          INP 1780
240 FORMAT (1H0,6HSS(1)=,F8.4,2X,6HSS(2)=,F8.4,2X,6HSS(3)=,F8.4,2X,6HSS(4)=,F8.4)          INP 1790
1S(4)=,F8.4,2X,6HDD(3)=,F8.4,2X,6HDD(4)=,F8.4          INP 1800
250 FORMAT (1H1,7HTSCALE=,F7.3,2X,7HVSSCALE=,F7.3,//,7X,3HTT1,9X,2HTV//,1X,45(2F11.4/,1X))          INP 1810
260 FORMAT (1H1,7HTSCALE=,F7.3,2X,7HESCALE=,F7.3,//,7X,3HTT2,9X,2HTH//,1X,47(2F11.4/,1X))          INP 1820
270 FORMAT (1H1,7HTSCALE=,F7.5,2X,7HNSCALE=,F7.5,//,7X,3HTT3,9X,2HTX//,1X,18(2F11.4/1X))          INP 1830
280 FORMAT (1H0,5(4X,I1,F13.8,F13.8,F13.8/))          INP 1840
290 FORMAT (1H0,3F13.6,19)          INP 1850
300 FORMAT (1H1,4X,1H1,4X,9HPTTHETA(I),7X,7HPPHI(I),6X,7HSS(I+9)/)          INP 1860
310 FORMAT (1H-,6X,6HFSE(I),5X,1.1HFSE(I)/FBAR,4X,7HSS(4+I),8X,1H1/)          INP 1870
END

```

SUBROUTINE SSAVE

C
 *****THIS SUBROUTINE IS USED TO RECORD
 *****INTERMEDIATE RESULTS FOR PLOTTING.
 *****QUANTITIES RECORDED ARE:
 *****FORCE LEVELS FSE(I) I=1,...,5
 *****HIP POSITION X(1), Y(1)
 *****KNEE POSITION X(2), Y(2)
 *****ANKLE POSITION X(3), Y(3)

C
 COMMON /GCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MLE(100), MSTOP, NCNDR, NSSA
 110
 1NAPO, NNAPT, NNATR, NNFL, NNQ(100), NNTRY, NPRINT, PPARM(50,4), TNOW, TTBEGLSSA
 120
 2, TTCLR, TTFIN, TTRIB(25), TTSET
 COMMON /GCOM2/ DD(100), DDL(100), DTFL, DINOW, ISEES, LFLAG(50), NFLAG, SSA
 130
 1NNEQD, NNEQS, NNEGAT, SS(100), SSL(100), TTNEK
 COMMON /MUSCLE/ PPHI(5), PTHETA(5), PHILAS(5), THELAS(5)
 COMMON /PHYS/ ALL1, ALL2
 COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
 COMMON /PSAVE/ XV1, XH1, SIGMAX
 COMMON /WALK/ BETA1, BETA2, TAV, XBOR, VMEAN, XNOT
 COMMON /FOOT/ DALPH1, DALPH2
 COMMON /MODE/ MODE, IMODE
 COMMON /CAST/ FSE(5), NFSE
 COMMON /EXPT/ FPE3, FPE4
 COMMON /SAVE/ SAVTIM, OPTTIM
 LOGICAL LFLAG, PUNCH, IMODE
 DIMENSION X(3), Y(3), Z(7)

C
 CALL ORDER
 IF (TNOW.EQ.TTBEGLSSA AND .NOT. IMODE) GO TO 30
 IF ((SAVTIM-TNOW).LT.0.025 AND .NOT. IMODE) GO TO 30
 SAVTIM=TNOW
 DO 10 I=1,5
 PHILAS(I)=PPHI(I)
 THELAS(I)=PTHETA(I)
 CONTINUE

10
 C
 *****PUNCHED OUTPUT FOR FUTURE USE.

C
 PUNCH=.TRUE.
 DO 20 I=1,5
 Z(I)=FSE(I)
 IF (I.EQ.3) Z(I)=Z(I)+FPE3
 IF (I.EQ.4) Z(I)=Z(I)+FPE4
 CONTINUE

20
 X(1)=XBOR+TNOW+XNOT
 Y(1)=VMEAN-VBAR(TNOW)
 X(2)=X(1)+ALL1*SIN(Z(1))
 Y(2)=Y(1)-ALL1*COS(Z(1))
 ZZ=Z(2)-Z(1)
 X(3)=X(2)+ALL2*SIN(ZZ)
 Y(3)=Y(2)-ALL2*COS(ZZ)
 WRITE (6,60) TNOW, MODE
 IF (PUNCH) WRITE (8,110) TNOW, MODE
 WRITE (6,70) (X(I), Y(I), I=1,3)
 IF (PUNCH) WRITE (8,120) (X(I), Y(I), I=1,3)
 WRITE (6,80) (PPHI(I), I=1,5)
 IF (PUNCH) WRITE (8,130) (PPHI(I), I=1,5)
 WRITE (6,90) (PTHETA(I), I=1,5)
 IF (PUNCH) WRITE (8,140) (PTHETA(I), I=1,5)
 WRITE (6,100) (Z(I), I=1,5)
 IF (PUNCH) WRITE (8,150) (Z(I), I=1,5)
 WRITE (6,130) (FSE(I), I=1,5)
 IF (PUNCH) WRITE (8,160) (FSE(I), I=1,5)
 WRITE (6,140) (SS(I), I=1,4)
 IF (PUNCH) WRITE (8,150) (SS(I), I=1,4), DD(3), DD(4)
 30
 IF (DTNOW.NE.0.0.OR.TNOW.NE.OPTTIM) GO TO 50
 DO 40 I=1,9
 LFLAG(I)=.FALSE.
 CONTINUE

40
 50
 CALL ORDER
 RETURN

C
 SSA 10
 SSA 20
 SSA 30
 SSA 40
 SSA 50
 SSA 60
 SSA 70
 SSA 80
 SSA 90
 SSA 100
 SSA 110
 SSA 120
 SSA 130
 SSA 140
 SSA 150
 SSA 160
 SSA 170
 SSA 180
 SSA 190
 SSA 200
 SSA 210
 SSA 220
 SSA 230
 SSA 240
 SSA 250
 SSA 260
 SSA 270
 SSA 280
 SSA 290
 SSA 300
 SSA 310
 SSA 320
 SSA 330
 SSA 340
 SSA 350
 SSA 360
 SSA 370
 SSA 380
 SSA 390
 SSA 400
 SSA 410
 SSA 420
 SSA 430
 SSA 440
 SSA 450
 SSA 460
 SSA 470
 SSA 480
 SSA 490
 SSA 500
 SSA 510
 SSA 520
 SSA 530
 SSA 540
 SSA 550
 SSA 560
 SSA 570
 SSA 580
 SSA 590
 SSA 600
 SSA 610
 SSA 620
 SSA 630
 SSA 640
 SSA 650
 SSA 660
 SSA 670
 SSA 680
 SSA 690
 SSA 700
 SSA 710
 SSA 720
 SSA 730
 SSA 740

C
60 FORMAT (1H1,///,8X,2H'T= ,2X,F5.3,2X,5H'MODE= ,2X,I1) SSA 750
70 FORMAT (1H0,3X,6H(X,Y)=,3(2X,1H(,F6.4,1H,,F6.4,1H))) SSA 760
80 FORMAT (1H0,9H PHI(I)=,3(2X,F8.4)) SSA 770
90 FORMAT (1H0,9HTHETA(I)=,3(2X,F8.4)) SSA 780
100 FORMAT (1H0,9HFORCE(I)=,3F10.4) SSA 790
110 FORMAT (2X,F3.2X,I1) SSA 800
120 FORMAT (6(2X,F6.4)) SSA 810
130 FORMAT (1H0,9H FSE(I)=,3F10.4) SSA 820
140 FORMAT (1H0,9H Z(I)=,4(2X,F8.4)) SSA 830
150 FORMAT (6(2X,F6.4)) SSA 840
160 FORMAT (3F10.4)
END

SUBROUTINE STATE

C
 ****THIS SUBROUTINE IS THE MAIN PROGRAM
 ****SEGMENT USED BY GASP IV. IT IS THIS
 ****PROGRAM THAT EVALUATES THE STATE
 ****VARIABLES AND DERIVATIVES AS FUNCTIONS
 ****OF TIME. NOTE THAT CONTROL IS PASSED ON
 ****DIRECTLY TO STATE1 FOR EVALUATION OF
 ****Z(I), PSI(I), D Z(I)/DT, AND D PSI(I)/DT.
 ****OPTIMIZATION IS PROCESSED IN THIS
 ****SEGMENT.

C
 COMMON /CCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MLE(100), MSTOP, NCRDR, NSTA,
 1NAP0, NNAPT, NNATR, NNFIL, NNQ(100), NTRY, NPRINT, PPARM(50,4), TNOW, TTREGSTA
 2, TTCLR, TTIFIN, TTRIB(25), TTSET
 COMMON /CCOM2/ DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, STA
 1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX
 COMMON /CCOM3/ AAERR, DTMAX, DTMIN, DTSAV, IITES, LLERR, LLSAV, LLSEV, RRESTA
 1RR, TTLAS, TTSAV
 COMMON /MUSCLE/ PPHI(5), PTHETA(5), PHILAS(5), THELAS(5)
 COMMON /OPT/ RMAX, RMIN, N, NCNS, NEQDS, IPRINT, IDATA
 COMMON /CAST/ FSE(5), NFSE
 COMMON /BODCON/ B(26,5)
 COMMON /SAVE/ SAVTIM, OPTTIM
 COMMON /MODE/ MODE, IMODE
 DIMENSION XA(10,11), XC(10), XE(10)
 DIMENSION DTV(10,10) C(10, 10), YT(10, 10)
 DIMENSION PEX(10,11), PSX(10,11), PART(11), PAST(1)
 DIMENSION FGAB(10), UX(10), CH(10), XNEW(10), XTRIAL(10)
 LOGICAL IMODE, LFLAG

C
 ****STATE1 IS BEING CALLED BY GASP IV AND
 ****NOT BY SUBROUTINE INPUT

C
 CALL ORDER
 LFLAG(16)=.FALSE.
 IMODE=.FALSE.
 CALL STATE1 (IMODE)

C
 ****EVALUATE CONSTRAINT EQUATIONS

C
 CALL SEVNT
 CALL SCOND
 C
 ****OPTIMIZATION NECESSARY ONLY FOR
 ****NEW TIME INCREMENT AT .01 SEC INTERVALS

C
 IF (.NOT. LFLAG(17)) GO TO 100
 CALL ORDER
 CALL SSAVE
 CALL ORDER
 OPTTIM=TNOW
 FBAR=3.62279731
 NFSE=0

C
 ****FOR VERY SMALL FORCE LEVELS IN MUSCLE,
 ****IT CAN BE EXEMPTED FROM OPTIMIZATION

C
 DO 10 I=1,5
 II=9+I
 LFLAG(II)=TNOW.EQ.TTREG.OR.I.EQ.5.OR.FSE(I)/(FBAR*B(7,I)),GT.0.001STA
 10 IF (LFLAG(II)) NFSE=NFSE+1
 CONTINUE
 N=2*NFSE
 NCNS=N+1
 J=0

C
 ****FOR MUSCLE EXEMPTED, RESET PHI AND
 ****THETA TO VALUE AT LAST TIME INCREMENT

C
 DO 30 I=1,5
 II=9+I
 IF (.NOT. LFLAG(II)) GO TO 20
 J=J+1

STA	10
STA	20
STA	30
STA	40
STA	50
STA	60
STA	70
STA	80
STA	90
STA	100
STA	110
STA	120
STA	130
STA	140
STA	150
STA	160
STA	170
STA	180
STA	190
STA	200
STA	210
STA	220
STA	230
STA	240
STA	250
STA	260
STA	270
STA	280
STA	290
STA	300
STA	310
STA	320
STA	330
STA	340
STA	350
STA	360
STA	370
STA	380
STA	390
STA	400
STA	410
STA	420
STA	430
STA	440
STA	450
STA	460
STA	470
STA	480
STA	490
STA	500
STA	510
STA	520
STA	530
STA	540
STA	550
STA	560
STA	570
STA	580
STA	590
STA	600
STA	610
STA	620
STA	630
STA	640
STA	650
STA	660
STA	670
STA	680
STA	690
STA	700
STA	710
STA	720
STA	730
STA	740

```

JJ=J+NFSE
XSTRT(J)=PHILAS(I)
XSTRT(JJ)=THELAS(I)
GO TO 30
20 PPHI(I)=PHILAS(I)
PTHETA(I)=THELAS(I)
CONTINUE
R*Q
WRITE(6,110)
DO 40 I=1,5
F1=FSE(I)/(FBAR*B(7,I))
II=I+9
WRITE(6,120) I,LFLAG(II),F1
40 CONTINUE
TNOW=TNOW+DTMIN*0.5
DO 50 I=1,20
J=21-I
SSL(J)=SS(I)
50 CONTINUE
C
*****PERFORM OPTIMIZATION
C
CALL DAVID(N,RMAX,RMIN,NCONS,NEQUS,XSTRT,1.E-4,1.E-6,50,IPRINT,1DSTA
1ATA,1...005,U,X,PHI,PSI,XA,WORK1,WORK2,WORK3,WORK4,X0,DT,C,YT,PHX,STA
2PSX,PART,PAST,XR,XE)
DO 60 I=1,N
X(I)=ABS(X(I))
60 CONTINUE
CALL ANSWER(U,X,PHI,PSI,N,NCONS,NEQUS)
J=0
DO 80 I=1,5
II=9+I
IF (.NOT.LFLAG(II)) GO TO 70
J=J+1
JJ=J+NFSE
PPHI(I)=X(J)
PHILAS(I)=PPHI(I)
PTHETA(I)=X(JJ)
THELAS(I)=PTHETA(I)
GO TO 90
70 PPHI(I)=PHILAS(I)
PTHETA(I)=THELAS(I)
80 CONTINUE
IMODE=.FALSE.
LFLAG(16)=.FALSE.
TNOW=TNOW-DTMIN*0.5
DO 90 I=1,20
J=21-I
SS(I)=SSL(J)
90 CONTINUE
C
*****RE-EVALUATE STATE VARIABLES AND DERIVATIVES
C
CALL STATE1(IMODE)
IMODE=.TRUE.
CALL ORDER
CALL SSAVE
CALL ORDER
IMODE=.FALSE.
LFLAG(17)=.FALSE.
ISEES=0
100 CALL ORDER
RETURN
C
C
110 FORMAT(1H1)
120 FORMAT(1H0,I5,L5,F13.6)
END

```

	STA	750
	STA	760
	STA	770
	STA	780
	STA	790
	STA	800
	STA	810
	STA	820
	STA	830
	STA	840
	STA	850
	STA	860
	STA	870
	STA	880
	STA	890
	STA	900
	STA	910
	STA	920
	STA	930
	STA	940
	STA	950
	STA	960
	STA	970
	STA	980
	STA	990
	STA	1000
	STA	1010
	STA	1020
	STA	1030
	STA	1040
	STA	1050
	STA	1060
	STA	1070
	STA	1080
	STA	1090
	STA	1100
	STA	1110
	STA	1120
	STA	1130
	STA	1140
	STA	1150
	STA	1160
	STA	1170
	STA	1180
	STA	1190
	STA	1200
	STA	1210
	STA	1220
	STA	1230
	STA	1240
	STA	1250
	STA	1260
	STA	1270
	STA	1280
	STA	1290
	STA	1300
	STA	1310
	STA	1320
	STA	1330
	STA	1340
	STA	1350
	STA	1360
	STA	1370
	STA	1380
	STA	1390
	STA	1400
	STA	1410
	STA	1420

SUBROUTINE STATE1 (IMODE)

```

C
C*****THIS SUBROUTINE EVALUATES THE STATE
C*****VARIABLES AND THEIR DERIVATIVES
C
C*****Z(0)=Z(10)
C
C*****PSI(I)=Z(I+10)
C
C*****D Z(I)/DT=DD(I)
C
C*****D PSI(I)/DT=DD(I+10)
C
C
COMMON /CCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MLE(100), MSTOP, NCRDR, NST1
1NAPO, NNAPT, NNATR, NNQIL, NNQ(100), NNTRY, NPRNT, PPARM(50,4), TNOW, TTBEGST1
2, TTCLR, TIFIN, TTRIB(25), TTSET
COMMON /CCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, ST1
1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX
COMMON /VARIAB/ ALAMB(10), D(23), DLDZ(4,10)
COMMON /BODCON/ B(26,5), A1, A2, C1, C2, C3, GRAV
COMMON /MUSCLE/ QPHI(5), QTHETA(5), PHILAS(5), THELAS(5)
COMMON /EXPT/ FPE3, FPE4, DFPE3(9), DFPE4(9), UP1, UP2, DUP1(4), DUP2(4), ST1
1SD(6), DSD(6,9)
COMMON /PHYS/ ALL1, ALL2
COMMON /ENERGY/ ABAR(2,5), C(5), ETA(5), GG(5)
COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
COMMON /ANKLE/ DSALPHA, UP2ALPH, MALPHA
COMMON /WALK/ BETA1, BETA2, TAV, XHOR, VMEAN, XNOT
COMMON /GAST/ FSE, NFSE
DIMENSION FSE(5), FK(5), FPHI(5), DFSE(5,9), DFK(5)
DIMENSION DMDZ(2,4), DUDZ(2,9), DFPHI(5)
DIMENSION Z(100), G(10,10)
DIMENSION PPHI(5), PTHETA(5)
LOGICAL IMODE, ISTEP(16)
REAL MALPHA, M(2,2), MI, M2
EQUIVALENCE (SS, Z)
EQUIVALENCE (ISTEP(1), LFLAG(1))
C
C      ZERO=0.00001
C
C*****QPHI(I) AND THETA(I) MUST REMAIN
C*****MATHEMATICALLY FEASIBLE
C
DO 10 I=1,5
PPHI(I)=ABS(QPHI(I))
PTHETA(I)=ABS(QTHETA(I))
IF (PPHI(I).EQ.0.0) PPHI(I)=ZERO
10 CONTINUE
C
C*****EVALUATE EXPERIMENTALLY DETERMINED
C*****FUNCTIONS AND CONSTANTS. NECESSARY
C*****ONLY AT NEW TIME INCREMENT
C
IF (ISTEP(16)) GO TO 20
CALL DLVAL
CALL DLAMB
CALL EXPT
IF (TNOW.GT.T2PRIM .AND. TNOW.LT.TEND) GO TO 20
AC1=VMEAN-VBAR(TNOW)-PP(TNOW)
AC2=QQ(TNOW)-(XHOR-TNOW-XNOT)
AC3=AC1**2+AC2**2
AC4=SQRT(AC3)
AC5=ALL1**2+ALL2**2-AC3
AC5=AC5/(2.*ALL1*ALL2)
SS(2)=ACOS(-AC5)
AC6=AC3+ALL1**2-ALL2**2
AC6=AC6/(2.*ALL1*AC4)
SS(1)=ACOS(AC6)+ATAN(AC2/AC1)
20 ISTEP(15)=.FALSE.
C
C*****EVALUATE FORCE LEVELS
C
DO 30 I=1,5
ST1   10
ST1   20
ST1   30
ST1   40
ST1   50
ST1   60
ST1   70
ST1   80
ST1   90
ST1   100
ST1   110
ST1   120
ST1   130
ST1   140
ST1   150
ST1   160
ST1   170
ST1   180
ST1   190
ST1   200
ST1   210
ST1   220
ST1   230
ST1   240
ST1   250
ST1   260
ST1   270
ST1   280
ST1   290
ST1   300
ST1   310
ST1   320
ST1   330
ST1   340
ST1   350
ST1   360
ST1   370
ST1   380
ST1   390
ST1   400
ST1   410
ST1   420
ST1   430
ST1   440
ST1   450
ST1   460
ST1   470
ST1   480
ST1   490
ST1   500
ST1   510
ST1   520
ST1   530
ST1   540
ST1   550
ST1   560
ST1   570
ST1   580
ST1   590
ST1   600
ST1   610
ST1   620
ST1   630
ST1   640
ST1   650
ST1   660
ST1   670
ST1   680
ST1   690
ST1   700
ST1   710
ST1   720
ST1   730
ST1   740

```

```

ARG=(Z(J)+B(9,I))**2-B(10,I)
IF (ARG.LT.0.0) ARG=0.0
TEMP=ALAMB(I)-SQRT(ARG)-B(11,I)
IF (B(9,I).EQ.0.0) TEMP=ALAMB(I)-Z(J)-B(11,I)
FSE(I)=B(7,I)*(EXP(B(8,I)*TEMP)-1.)
FNOT=0.005*(EXP(1.531)-1.0)*B(7,I)
IF (FSE(I).LT.FNOT) FSE(I)=FNOT
30 CONTINUE
C
*****EVALUATE FUNCTION K(Z(4+I))
C
DO 40 I=1,5
J=4+I
KK=IFIX(B(25,I))
TEMP=-B(14,I)*(B(15,I)+B(16,I)*Z(J))**KK
FK(I)=B(12,I)+B(13,I)*EXP(TEMP)*SIN(B(17,I)*Z(J)+B(18,I))
40 CONTINUE
C
*****EVALUATE FUNCTION PHI(Z(4+I),PHI(I))
C
DO 50 I=1,5
J=4+I
TEMP=EXP(-B(19,I)*(Z(J)-B(20,I))**2)
TEMP1=PPHI(I)-B(21,I)*(EXP(-B(22,I)*Z(J))-B(23,I))
FPHI(I)=0.5025+0.4975*TANH(19.3*TEMP*TEMP1)
50 CONTINUE
C
*****EVALUATE DERIVATIVE EQUATIONS
C
DO 60 I=1,5
J=4+I
K=20+I
L=25+I
Z(K)=FPHI(I)
Z(L)=PTHETA(I)*Z(K)+0.005
TEMP1=-B(5,I)*(Z(J)+B(6,I))**2-1.43*(Z(K)-1.)**2
TEMP1=TANH(B(4,I)*EXP(TEMP1))
TEMP2=SQRT(1.-(B(2,I)/(Z(J)+B(3,I))))**2
TEMP2=B(1,I)*FK(I)*Z(L)*TEMP2
TEMP=(FSE(I)/TEMP2-1.)*TEMP1
ISTEP(I)=(TEMP**2.GE.1.0)
IF (ISTEP(I)) ISTEP(15)=.TRUE.
IF (ISTEP(I)) TEMP=TEMP/ABS(TEMP)*.999999999999
DD(J)=ATANH(TEMP)/B(26,I)
60 CONTINUE
IF (ISTEP(15).AND..NOT. IMODE) RETURN
YD1=FVERT(TNOW)
XD1=FHZRZ(TNOW)
MALPHA=YD1*(XP(TNOW)-PP(TNOW))+XD1*(QQ(TNOW)-YP(TNOW))
U1=FSE(1)*SD(1)+(FSE(3)+FPE3)*SD(3)-(FSE(4)+FPE4)*SD(4)
U2=-FSE(2)*SD(2)-(FSE(3)+FPE3)*SD(2)+(FSE(4)+FPE4)*SD(5)+FSE(5)*SDST1
1(6)
ZZ=Z(2)-Z(1)
TC1=COS(Z(1))
TS1=SIN(Z(1))
TC2=COS(Z(2))
TS2=SIN(Z(2))
TCZ=COS(ZZ)
TSZ=SIN(ZZ)
VG=VDDOT(TNOW)+GRAV
M(2,1)=YD1*ALL2*TSZ
M(2,2)=XD1*ALL2*TCZ
M(1,1)=YD1*ALL1*TS1-M(2,1)
M(1,2)=XD1*ALL1*TC1+M(2,2)
M1=U1+UP1+MALPHA+M(1,1)+M(1,2)
M2=U2+UP2-MALPHA+M(2,1)-M(2,2)
BB1=A2
BB2=A2+C3*TC2
BB3=A1+2.*C3*TC2
DEL=BB1*BB3-BB2**2
R1=C3*Z(4)*TS2*(2.*Z(3)-Z(4))+(C2*TC1-C1*TS1)*VG+M1
R2=-C3*((Z(3)**2)*TS2+TC2*VG)+M2
DD(1)=Z(3)
ST1 750
ST1 760
ST1 770
ST1 780
ST1 790
ST1 800
ST1 810
ST1 820
ST1 830
ST1 840
ST1 850
ST1 860
ST1 870
ST1 880
ST1 890
ST1 900
ST1 910
ST1 920
ST1 930
ST1 940
ST1 950
ST1 960
ST1 970
ST1 980
ST1 990
ST1 1000
ST1 1010
ST1 1020
ST1 1030
ST1 1040
ST1 1050
ST1 1060
ST1 1070
ST1 1080
ST1 1090
ST1 1100
ST1 1110
ST1 1120
ST1 1130
ST1 1140
ST1 1150
ST1 1160
ST1 1170
ST1 1180
ST1 1190
ST1 1200
ST1 1210
ST1 1220
ST1 1230
ST1 1240
ST1 1250
ST1 1260
ST1 1270
ST1 1280
ST1 1290
ST1 1300
ST1 1310
ST1 1320
ST1 1330
ST1 1340
ST1 1350
ST1 1360
ST1 1370
ST1 1380
ST1 1390
ST1 1400
ST1 1410
ST1 1420
ST1 1430
ST1 1440
ST1 1450
ST1 1460
ST1 1470
ST1 1480

```

```

DD(3)=(R1*BB1+R2*BB2)/DEL
DD(4)=(R1*BB2+R2*BB3)/DEL
C
C*****DD(10)=G(0)=DZ(0)/DT
C
    DD(10)=0
    DO 70 J=1,5
    TEMP1=(1.-EXP(-0.24-ABAR(2,J)/PPHI(J)))*PPHI(J)
    TEMP1=TEMP1+(0.82-4.81*(Z(4+J)/ETA(J))*2)*Z(20+J)
    TEMP1=ABAR(1,J)*TEMP1*PTHETA(J)*GG(J)
    TEMP2=G(J)*PTHETA(J)*FK(J)*(Z(20+J)**0.72)
    TEMP2=(TEMP2+FSE(J))*DD(4+J)
    DD(10)=DD(10)+TEMP1+TEMP2
    TEMP3=0.
    IF (J.EQ.3) TEMP3=ABS(ALAMB(8))*FPE3
    IF (J.EQ.4) TEMP3=ABS(ALAMB(9))*FPE4
    DD(10)=DD(10)+TEMP3
70    CONTINUE
    IF (IMODE) RETURN
C
C*****CALCULATE PARTIALS DG(I)/DZ(J)=G(I,J)
C
    DO 80 I=1,10
    DO 80 J=1,10
    G(I,J)=0.0
80    CONTINUE
C
C*****CALCULATE G(I,J), I=1,4, J=1,10
C
    DO 90 I=1,5
    DO 90 J=1,9
    DFSE(I,J)=0.0
90    CONTINUE
    DO 110 I=1,5
    DO 100 J=1,2
    DFSE(I,J)=B(8,I)*DUDZ(I,J)*(FSE(I)+B(7,I))
100   CONTINUE
    DO 110 J=3,9
    IF (I.NE.(J-4)) GO TO 110
    DFSE(I,J)=(FSE(I)+B(7,I))*B(8,I)*(Z(J)+B(9,I))
    DFSE(I,J)=DFSE(I,J)/SQRT((Z(J)+B(9,I))**2-B(10,I))
110   CONTINUE
    DO 120 I=1,9
    DUDZ(1,I)=SD(1)*DFSE(1,I)+DSD(1,I)*FSE(1)+SD(3)*(DFSE(3,I)+DFPE3(I))
    1(I))+DSD(3,I)*(FSE(3)+FPE3)+SD(4)*(DFSE(4,I)+DFPE4(I))-DSD(4,I)*(FSE
    1(4)+FPE4)
120   CONTINUE
    DO 130 I=1,9
    DUDZ(2,I)=SD(2)*(DFSE(2,I)+DFSE(3,I)+DFPE3(I))-DSD(2,I)*(FSE(2)+FST1
    1(I))+DSD(3,I)*(FSE(3)+FPE3)+SD(5)*(DFSE(4,I)+DFPE4(I))+DSD(5,I)*(FSE(4)+FPE4)+SD(6
    2(I))*DFSE(5,I)
130   CONTINUE
    DO 140 I=1,4
    DMDZ(1,I)=DUDZ(1,I)+DUP1(I)
    DMDZ(2,I)=DUDZ(2,I)+DUP2(I)
140   CONTINUE
    QUOT=YD1/XD1
    DMDZ(1,1)=DMDZ(1,1)+QUOT*M(1,2)-M(1,1)/QUOT
    TEMP=QUOT*M(2,2)+M(2,1)/QUOT
    DMDZ(2,1)=DMDZ(2,1)-TEMP
    DMDZ(1,2)=DMDZ(1,2)-TEMP
    DMDZ(2,2)=DMDZ(2,2)+TEMP
    G(1,3)=1.
    G(2,4)=1.
    CA1=-(C1*TC1+C2*TS1)*VG+DMDZ(1,1)
    CA2=DMDZ(2,1)
    G(3,1)=(BB1*CA1+BB2*CA1)/DEL
    G(4,1)=(BB2*CA1+BB3*CA2)/DEL
    CA1=C3*Z(4)*TC2*(Z(4)-2.*Z(3))-DMDZ(1,2)
    CA2=C3*((Z(3)**2)*TC2-TS2*VG)-DMDZ(2,2)
    CA3=C3*TS2*(2.*DD(3)*(BB1+BB2)-R2)
    G(3,2)=(CA3-BB1*CA1+BB2*CA2)/DEL
    CA3=C3*TS2*(2.*DD(4)*(BB1+BB2)-(R1+2.*R2))

```

```

CA1=2.*C3*Z(4)*TS2+DMDZ(1,3) ST1 2230
CA2=2.*C3*Z(3)*TS2+DMDZ(2,3) ST1 2240
C(3,3)=(BB1*CA1-BB2*CA2)/DEL ST1 2250
C(4,3)=(BB2*CA1-BB3*CA2)/DEL ST1 2260
CA1=C3*TS2*2.*((Z(3)-Z(4))+DMDZ(1,4)) ST1 2270
CA2=DMDZ(2,4) ST1 2280
C(3,4)=(BB1*CA1+BB2*CA2)/DEL ST1 2290
C(4,4)=(BB2*CA1+BB3*CA2)/DEL ST1 2300
DO 150 I=5,9 ST1 2310
G(3,1)=(BB1*DUDZ(1,1)+BB2*DUDZ(2,1))/DEL ST1 2320
G(4,1)=(BB2*DUDZ(1,1)+BBS*DUDZ(2,1))/DEL ST1 2330
150 CONTINUE ST1 2340
C ST1 2350
C*****THIS COMPLETES G(I,J), I=1,4 , J=1,10 ST1 2360
C ST1 2370
C*****NEXT COMPUTE G(I,J), I=5,9 , J=1,10 ST1 2380
C ST1 2390
DO 180 I=5,9 ST1 2400
K=I-4 ST1 2410
ARC1=B(4,K)*EXP(-B(5,K)*(Z(I)+B(6,K)**2-1.43*(Z(16+I)-1.)***2) ST1 2420
ARC2=TANH(ARC1) ST1 2430
DENOM=B(1,K)*Z(21+I)*FK(K)*SQRT(1.-(B(2,K)/(Z(I)+B(3,K))**2) ST1 2440
ARC3=FSE(K/DENOM) ST1 2450
ARC=(ARC3-1.)*ARC2 ST1 2460
DO 180 J=1,10 ST1 2470
IF (J.GE.3) GO TO 160 ST1 2480
G(I,J)=ARC2*DFSE(I,J)/((1.-ARC**2)*DENOM*B(26,K)) ST1 2490
GO TO 180 ST1 2500
160 IF (J.GE.5) GO TO 170 ST1 2510
GO TO 180 ST1 2520
170 L=J-4 ST1 2530
IF (K.NE.L) GO TO 180 ST1 2540
DFK(K)=(B(17,K)/TAN(B(17,K)*Z(I)+B(18,K))-B(14,K)*B(16,K)*B(25,K)*ST1 2550
1*(B(15,K)+B(16,K)*Z(I))**2*(IFIX(B(25,K)-1.))*(FK(K)-B(12,K)) ST1 2560
DARG1=-2.*B(5,K)*(Z(I)-B(6,K))*ARC1 ST1 2570
SQ=1.-(B(2,K)/(Z(I)+B(3,K))**2) ST1 2580
SQ1=SQRT(SQ) ST1 2590
SQ2=(1.-SQ)/(Z(I)+B(3,K)) ST1 2600
DARG3=DFSE(K,J)/DENOM-((Z(21+I)*B(1,K)*(FK(K)*SQ2/SQ1+SQ1*DFK(K)))ST1 2610
1*(ARC3**2)/FSE(K)) ST1 2620
G(I,J)=ARC2*DARG3+(ARC3-1.)*DARG1/(COSH(ARC1)**2) ST1 2630
G(I,J)=G(I,J)/((1.-ARC**2)*B(26,K)) ST1 2640
180 CONTINUE ST1 2650
C ST1 2660
C*****THIS COMPLETES G(I,J) , I=5,9 , J=1,10 ST1 2670
C ST1 2680
C*****NEXT COMPUTE G(10,J) , J=1,10 ST1 2690
C ST1 2700
DO 210 J=1,4 ST1 2710
DO 210 I=1,5 ST1 2720
G(10,J)=G(10,J)+DD(4+I)*DFSE(I,J)+FSE(I)*G(4+I,J) ST1 2730
G(10,J)=G(10,J)+(C(I)*PTHETA(I)*Z(I+20)**0.72*FK(I))*G(4+I,J) ST1 2740
GO TO (210,210,190,200,210), I ST1 2750
190 G(10,J)=G(10,J)+FPE3*DLDZ(J,I+5)*SIGN(1.,ALAMB(I+5))+ABS(ALAMB(I+5)*ST1 2760
1)*DFPE3(J) ST1 2770
GO TO 210 ST1 2780
200 G(10,J)=G(10,J)+FPE4*DLDZ(J,I+5)*SIGN(1.,ALAMB(I+5))+ABS(ALAMB(I+5)*ST1 2790
1)*DFPE4(J) ST1 2800
210 CONTINUE ST1 2810
DO 220 I=1,5 ST1 2820
J=4+I ST1 2830
K=20+I ST1 2840
G(10,J)=-9.62*ABAR(I,I)*Z(K)*PTHETA(I)*GG(I)*Z(J) ST1 2850
G(10,J)=G(10,J)/ETA(I)**2+(C(I)*PTHETA(I)*Z(K)**0.72*DFK(I)+DFSE(I)*ST1 2860
1,J))*DD(J)+(C(I)*PTHETA(I)*Z(K)**0.72*FK(I)+FSE(I))*G(J,J) ST1 2870
220 CONTINUE ST1 2880
C ST1 2890
C*****CALCULATE D PSI(I)/DT I=1,10 ST1 2900
C ST1 2910
C D PSI(I)/DT = DZ(I+10)/DT ST1 2920
C ST1 2930
DO 230 I=1,10 ST1 2940
K=I+10 ST1 2950

```

```
DO 230 J=1,10
L=J+10
DD(K)=DD(K)-Z(L)*G(J,I)
230  CONTINUE
C
C*****THIS COMPLETES THE DYNAMICAL EQUATION
C
      RETURN
END
```

```
ST1 2970
ST1 2980
ST1 2990
ST1 3000
ST1 3010
ST1 3020
ST1 3030
ST1 3040
ST1 3050
```

```

SUBROUTINE UREAL (X,P1)
C
C*****THIS SUBROUTINE EVALUATES THE OBJECTIVE
C*****FUNCTION PI(Z,PHI,THETA) WHICH IS TO BE
C*****MAXIMIZED
C
COMMON /MUSCLE/ PPHI(5),PTHETA(5),PHILAS(5),THELAS(5)
COMMON /GCOM1/ ATRIB(25),JEVNT,MFA,MFE(100),MLE(100),NSTOP,NCRDR,NURE
INAPO,NNAPT,NNATR,NNFIL,NNQ(100),NNTRY,NPRNT,PPARM(50,4),TNOW,TTBEGURE
2,TTCLR,TTFIN,TTRIB(25),TTSET
COMMON /GCOM2/ DD(100),DDL(100),DTFUL,DTNOW,ISEES,LFLAG(50),NFLAG,URE
1NNEQL,NNEQS,NNEQT,SS(100),SSL(100),TTINEX
COMMON /GCOM3/ AAERR,DTMAX,DTMIN,DTSAV,IITES,LLERR,LLSAV,LLSEV,RREURE
IRR,TTLAS,TISAV
COMMON /CAST/ FSE(5),NFSE
COMMON /MODE/ MODE,JMODE
DIMENSION X(1)
LOGICAL IMODE,LFLAG,FLAG

C
C*****SET ALL CONTROL FUNCTIONS
C
J=0
DO 10 I=1,5
II=9+I
IF (.NOT.LFLAG(II)) GO TO 10
J=J+1
JJ=J+NFSE
PPHI(I)=X(J)
PTHETA(I)=X(JJ)
CONTINUE
10 LFLAG(16)=.FALSE.
IMODE=.TRUE.
FLAG=TNOW.EQ.TTBEG

C
C*****EVALUATE ALL STATE VARIABLES AND DERIVATIVES
C
TNOW=TNOW-DTMIN*.5
CALL STATE1(IMODE)
IF (FLAG) GO TO 30

C
C*****EVALUATE OBJECTIVE FUNCTION
C
DO 20 I=1,20
J=21-I
SS(I)=SSL(J)+DD(I)*DTMIN*.5
CONTINUE
20 TNOW=TNOW+DTMIN*.5
30 IF (FLAG) GO TO 40
CALL STATE1(IMODE)
PI=0.0
40 DO 50 I=1,10
PI=PI+DD(I)*SS(I+10)
CONTINUE
50 PI=-PI/1.E04
RETURN
END

```

SUBROUTINE CONST (X,NCONS,PHI)

CON	10
CON	20
CON	30
CON	40
CON	50
CON	60
CON	70
CON	80
CON	90
CON	100
CON	110
CON	120
CON	130
CON	140
CON	150
CON	160
CON	170
CON	180
CON	190
CON	200
CON	210
CON	220
CON	230
CON	240
CON	250
CON	260
CON	270
CON	280
CON	290
CON	300
CON	310

C ****THIS SUBROUTINE IS USED TO EVALUATE THE
C ****CONSTRAINTS AND TO ASCERTAIN FEASIBILITY

C
COMMON /GCOM2/ DD(100),DDL(100),DTFUL,DTNOW,ISEES,LFLAG(50),NFLAG,
INNEQD,NNEQS,NNEGQ,SS(100),SSL(100),TNEX
COMMON /CAST/ FSE(5),NFSE
DIMENSION X(1), PHI(1)
LOGICAL ISTEP(16)
EQUIVALENCE (ISTEP(1),LFLAG(1))

C
C ****CONTROL PARAMETERS MUST BE IN THE
C ****RANGE (0,1). ALL EVALUATIONS ARE
C ****PERFORMED ON ABS(PHI(I)) AND
C ****ABS(THETA(I))

C
DO 10 I=1,NFSE
J= I+NFSE
PHI(I)=1.0-ABS(X(I))
PHI(J)=1.0-ABS(X(J))
10 CONTINUE
PHI(NCONS)=0.0

C
C ****IF ISTEP(15)=.TRUE. AT LEAST ONE
C ****CONSTRAINT NOT SATISFIED

C
CALL SEVNT
IF (ISTEP(15)) PHI(NCONS)=-1000.
RETURN
END

SUBROUTINE SEVNT

C **** THIS SUBROUTINE CHECKS ALL CONSTRAINTS
 C TO SIGNAL SCOND THE PASSING OF A
 C STATE EVENT

C

```

COMMON /CCOM1/ ATRIB(25), JEVNT, MFA, MFE(100), MSTOP, NCRDR, NSEV    10
1NAPO, NNAPT, NNATR, NNFIL, NNQ(100), NNTRY, NPRINT, PPARM(50,4), TNOW, TTBEGINSEV 20
2, TTCLR, TTFIN, TTRIB(25), TTSET                                     SEV 30
  COMMON /CCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, SEV 40
1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX                         SEV 50
  COMMON /PSAVE/ XV1, XH1, SIGMAX                                       SEV 60
  COMMON /WALK/ BETA1, BETA2, TAV, XHOR, VMEAN, XNOT                      SEV 70
  COMMON /PHYS/ ALL1, ALL2                                         SEV 80
  COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND                           SEV 90
  COMMON /FOOT/ DALPH1, DALPH2                                     SEV 100
  COMMON /CAST/ FSE(5)                                         SEV 110
  COMMON /ANKLE/ D5ALPHA, UP2ALPH, MALPHA                         SEV 120
DIMENSION PHI(2), PSI(2)
REAL MALPHA
LOGICAL LFLAG
```

G

```

T-TNOW
IF (T.GT.TEND) T=T-TEND
```

C ****CHECK INEQUALITY CONSTRAINTS

C

```

PHI(1)=SS(2)                                         SEV 10
IF (T-T2PRIMD 10, 10, 20                               SEV 20
10 CALL ANKLE                                         SEV 30
  PHI(2)=FSE(5)-(MALPHA-UP2ALPH)/D5ALPHA             SEV 40
  GO TO 30                                         SEV 50
20 PI=3.1415926536                                     SEV 60
  ZETA=SIGMAX+SS(2)-SS(1)-PI/2.                      SEV 70
  PHI(2)=-VBAR(TNOW)-ALL1*COS(SS(1))-ALL2*COS(SS(2)-SS(1))+DALPH2*S1SEV 80
  1N(ZETA)-VMEAN                                     SEV 90
```

C ****CHECK EQUALITY CONSTRAINTS

C

```

30 CALL SEQUAL (PSI)
```

C ****SET INDICATOR FLAGS

C ****LFLAG(I) = .TRUE. IF CONSTRAINT(I)
C ****IS NOT SATISFIED

C

```

DO 40 I=6,9
LFLAG(I)=.FALSE.
40 CONTINUE
IF (PHI(1).LT.0.0) LFLAG(6)=.TRUE.
IF (PHI(2).LT.0.0) LFLAG(7)=.TRUE.
IF ((0.01-ABS(PSI(1))).LT.0.0) LFLAG(8)=.TRUE.
IF ((0.01-ABS(PSI(2))).LT.0.0) LFLAG(9)=.TRUE.
```

C ****LFLAG(15) = .TRUE. IF ANY CONSTRAINT
C ****IS NOT SATISFIED

C

```

DO 50 I=6,9
IF (LFLAG(I)) LFLAG(15)=.TRUE.
50 CONTINUE
RETURN
END
```

	SEV	10
	SEV	20
	SEV	30
	SEV	40
	SEV	50
	SEV	60
	SEV	70
	SEV	80
	SEV	90
	SEV	100
	SEV	110
	SEV	120
	SEV	130
	SEV	140
	SEV	150
	SEV	160
	SEV	170
	SEV	180
	SEV	190
	SEV	200
	SEV	210
	SEV	220
	SEV	230
	SEV	240
	SEV	250
	SEV	260
	SEV	270
	SEV	280
	SEV	290
	SEV	300
	SEV	310
	SEV	320
	SEV	330
	SEV	340
	SEV	350
	SEV	360
	SEV	370
	SEV	380
	SEV	390
	SEV	400
	SEV	410
	SEV	420
	SEV	430
	SEV	440
	SEV	450
	SEV	460
	SEV	470
	SEV	480
	SEV	490
	SEV	500
	SEV	510
	SEV	520
	SEV	530
	SEV	540
	SEV	550
	SEV	560
	SEV	570
	SEV	580
	SEV	590
	SEV	600
	SEV	610

SUBROUTINE SEQUAL (PSI)

C

*****THIS SUBROUTINE EVALUATES THE DYNAMIC
EQUALITY CONSTRAINTS AT DIFFERENT
POINTS IN THE CYCLE

C

COMMON /PSAVE/ XV1,XH1,SIGMAX	SEQ 10
COMMON /WALK/ BETA1,BETA2,TAV,XHOR,VMEAN,XNOT	SEQ 20
COMMON /GCOM2/ DD(100),DDL(100),DTFUL,DTNOW,ISEES,LFLAG(50),NFLAG,SEQ.	SEQ 30
1NNEQD,NNEQS,NNEGQT,SS(100),SSL(100),TTNEX	SEQ 40
COMMON /PHYS/ ALL1,ALL2	SEQ 50
COMMON /TIME/ TINIT,TPRIM,T2PRIM,TEND	SEQ 60
COMMON /GCOM1/ ATRIB(25),JEVNT,MFA,MFE(100),MLE(100),MSTOP,NCRDR,NSEQ.	SEQ 70
1NAPO,NNAPT,NNATR,NNFIL,NNQ(100),NNTRY,NPRNT,PPARM(50,4),TNOW,TTBECSEQ	SEQ 80
2,TTCLR,TTFIN,TTRIB(25),TTSET	SEQ 90
COMMON /FOOT/ DALPH1,DALPH2	SEQ 100
DIMENSION PSI(1)	SEQ 110
COMMON /MODE/ MODE,IMODE	SEQ 120
C	SEQ 130
T=TNOW	SEQ 140
IF (T.GT.TEND) T=T-TEND	SEQ 150
IF (T-TPRIMD 10,30,40	SEQ 160
C	SEQ 170
*****TZERO .LT. T .LE. TPRIM	SEQ 180
C	SEQ 190
10 MODE=1	SEQ 200
20 PSI(1)=VBAR(TNOW)+ALL1*COS(SS(1))+ALL2*COS(SS(2)-SS(1))-VMEAN+PP(TSEQ.	SEQ 210
1NOW)	SEQ 220
1NOW)	SEQ 230
PSI(2)=(XHOR*TNOW+XNOT)+ALL1*SIN(SS(1))-ALL2*SIN(SS(2)-SS(1))-QQ(TSEQ.	SEQ 240
1NOW)	SEQ 250
RETURN	SEQ 260
C	SEQ 270
*****T = TPRIM	SEQ 280
C	SEQ 290
30 ALPHA=ASIN(PP(TNOW)/DALPH1)	SEQ 300
XV1=VBAR(TNOW)+ALL1*COS(SS(1))+ALL2*COS(SS(2)-SS(1))+DALPH1*SIN(ALSEQ	SEQ 310
1PHA)	SEQ 320
XH1=(XHOR*TNOW+XNOT)+ALL1*SIN(SS(1))-ALL2*SIN(SS(2)-SS(1))+DALPH1*SEQ	SEQ 330
1COS(ALPHA)	SEQ 340
MODE=2	SEQ 350
GO TO 20	SEQ 360
40 IF (T-T2PRIMD 50,70,80	SEQ 370
C	SEQ 380
*****TPRIM .LT. T .LE. T2PRIM	SEQ 390
C	SEQ 400
50 MODE=3	SEQ 410
60 PART1=XV1-VBAR(TNOW)	SEQ 420
PART2=XH1-(XHOR*TNOW+XNOT)	SEQ 430
ZZ=SS(2)-SS(1)	SEQ 440
PSI(1)=ALL1**2+ALL2**2-DALPH1**2+PART1**2+PART2**2+2.* (ALL1*ALL2*CSEQ.	SEQ 450
10*(SS(2))-2.0*PART1*(ALL1*COS(SS(1))+ALL2*COS(ZZ))-2.0*PART2*(ALL1*SEQ	SEQ 460
2*SIN(SS(1))-ALL2*SIN(ZZ)))	SEQ 470
PSI(2)=0.	SEQ 480
RETURN	SEQ 490
C	SEQ 500
*****T = T2PRIM	SEQ 510
C	SEQ 520
70 ALPHA=XH1-(XHOR*TNOW+XNOT)-ALL1*SIN(SS(1))+ALL2*SIN(SS(2)-SS(1))	SEQ 530
ALPHA=ACOS(ALPHA)/DALPH1	SEQ 540
PI=3.1415926536	SEQ 550
SIGMAX=ALPHA+SS(1)-SS(2)+PI/2.	SEQ 560
MODE=4	SEQ 570
GO TO 60	SEQ 580
C	SEQ 590
*****T .GT. T2PRIM	SEQ 600
C	SEQ 610
80 PSI(1)=0,	SEQ 620
PSI(2)=0.	SEQ 630
MODE=5	SEQ 640
RETURN	SEQ 650
END	SEQ 660
	SEQ 670
	SEQ 680
	SEQ 690
	SEQ 700
	SEQ 710

SUBROUTINE SCOND

```

C
C THIS SUBROUTINE SIGNALS GASP IV THE
C OCCURRANCE OF A STATE EVENT. IF ISEES
C LT. ZERO STATE EVENT HAS BEEN PASSED
C

COMMON /GCOM1/ ATRIB(25), JEVNT, NFA, MFE(100), MLE(100), MSTOP, NCRDR, NSCO
1NAPO, NNAPT, NNATR, NNFIL, NNQ(100), NNTRY, NPRINT, PPARM(50,4), TNOW, TTREGSCO
2, TTFIN, TTIB(25), TTSET
COMMON /GCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG, SCO
1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX
COMMON /SAVE/ SAVTIN, OPTTIM
LOGICAL ISTEP(17)
EQUIVALENCE (ISTEP(1),LFLAG(1))

C
ISEES=0
DO 10 I=1,9
IF (ISTEP(I)) ISEES=10*ISEES-I
CONTINUE
T-TNOW-OPTTIM
IF (T-0.01) 60,50,20
IF (ISEES) 30,40,40
30 ISEES=ISEES#10
RETURN
40 ISEES=-100
ISTEP(17)=.TRUE.
RETURN
50 ISEES=100
ISTEP(17)=.TRUE.
60 RETURN
END
      
```

SCO	10
SCO	20
SCO	30
SCO	40
SCO	50
SCO	60
SCO	70
SCO	80
SCO	90
SCO	100
SCO	110
SCO	120
SCO	130
SCO	140
SCO	150
SCO	160
SCO	170
SCO	180
SCO	190
SCO	200
SCO	210
SCO	220
SCO	230
SCO	240
SCO	250
SCO	260
SCO	270
SCO	280
SCO	290
SCO	300
SCO	310

SUBROUTINE DLVAL.

```

C
C*****THIS SUBROUTINE EVALUATES THE FUNCTIONS
C*****LAMBDA(I) AND LAMBDA.DOT(I)
C
COMMON /CCOM2/ DD(100), DDL(100), DTFLU, DTNOW, ISEE$,
INNEQD, NNEQS, NNEGAT, SS(100), SSL(100), TTINEX
COMMON /VARIAB/ ALAMB(10), D(23)
DIMENSION T1(5), T2(5), Z(100)
EQUIVALENCE (SS,Z)
DATA T1/-3.41,-15.8,2.55,-2.72,1.8/, T2/8.5,-2.9,-16.1,1.02,-3.43/
C
C*****VARIABLES D(I) ARE COMMONLY FOUND TERMS
C
TEMP= 1.128*Z(1)+0.748      DLV 10
D(1)=COS(TEMP)               DLV 20
D(5)=SIN(TEMP)               DLV 30
TEMP= 1.047*Z(1)+0.838      DLV 40
D(2)=COS(TEMP)               DLV 50
D(6)=SIN(TEMP)               DLV 60
TEMP= 1.076*Z(2)+0.280      DLV 70
D(3)=COS(TEMP)               DLV 80
D(7)=SIN(TEMP)               DLV 90
TEMP= 1.160*Z(2)+0.464      DLV 100
D(4)=COS(TEMP)               DLV 110
D(8)=SIN(TEMP)               DLV 120
DLV 130
DLV 140
DLV 150
DLV 160
DLV 170
DLV 180
DLV 190
DLV 200
DLV 210
DLV 220
DLV 230
DLV 240
DLV 250
DLV 260
DLV 270
DLV 280
DLV 290
DLV 300
DLV 310
DLV 320
DLV 330
DLV 340
DLV 350
DLV 360
DLV 370
ALAMB(1)=0.287-0.0497*Z(1)  DLV 380
ALAMB(2)=0.3+0.033*Z(2)     DLV 390
ALAMB(3)=0.517+0.045*D(1)+0.033*Z(2)  DLV 400
ALAMB(4)=0.483-0.062*D(2)+0.07*D(3)    DLV 410
ALAMB(5)=0.088+0.019*D(4)    DLV 420
DLV 430
DLV 440
DLV 450
DLV 460
DLV 470
DLV 480
DLV 490
DLV 500
DLV 510
DLV 520
DLV 530
DLV 540
DLV 550
C
C*****ALAMB(I) = LAMBDA(I), I=1,...,5
C
ALAMB(1)=0.287-0.0497*Z(1)  DLV 300
ALAMB(2)=0.3+0.033*Z(2)     DLV 310
ALAMB(3)=0.517+0.045*D(1)+0.033*Z(2)  DLV 320
ALAMB(4)=0.483-0.062*D(2)+0.07*D(3)    DLV 330
ALAMB(5)=0.088+0.019*D(4)    DLV 340
DLV 350
DLV 360
DLV 370
ALAMB(6)=-0.0497*Z(3)      DLV 380
ALAMB(7)=0.033*Z(4)        DLV 390
ALAMB(8)=-0.051*Z(3)*D(5)+0.033*Z(4)  DLV 400
ALAMB(9)=0.065*Z(3)*D(6)-0.075*Z(4)*D(7)  DLV 410
ALAMB(10)=-0.022*Z(4)*D(8)   DLV 420
D(9)=EXP(90.4*(ALAMB(3)-0.58))  DLV 430
D(10)=EXP(23.95*(ALAMB(4)-0.48))  DLV 440
D(11)=EXP(239.8*(ALAMB(4)-0.53))  DLV 450
D(12)=EXP(-4.33*(0.17-Z(2))*2)   DLV 460
D(13)=EXP(-1.187*Z(2))       DLV 470
DO 10 I=1,5
J=13+I
K=18+I
D(J)=EXP(T1(I)*Z(1))
D(K)=EXP(T2(I)*Z(2))
CONTINUE
RETURN
END

```

```

SUBROUTINE DLAMB
C
C*****THIS SUBROUTINE EVALUATES THE PARTIAL
C*****DERIVATIVES OF LAMBDA(I) AND LAMBDA DOT(I)
C
COMMON /CCOM2/ DD(100),DDL(100),DTIFUL,DTNOW,ISEES,LFLAG(50),NFLAG,
1NNEQD,NNEQS,NNEGQT,SS(100),SEL(100),TINEX
COMMON /VARIAB/ ALAMB(10),D(23),DLDZ(4,10)
DIMENSION Z(100)
EQUIVALENCE (SS,Z)

DO 10 J=1,4
DO 10 I=1,10
DLDZ(J,I)=0.
10 CONTINUE

C*****DLDZ(J,I) = DALAMB(I)/DZ(J)

C
DLDZ(1,1)=-0.0497
DLDZ(1,3)=0.51*D(3)
DLDZ(1,4)=-0.065*D(6)
DLDZ(1,8)=-0.058*Z(3)*D(1)
DLDZ(1,9)=0.068*Z(3)*D(2)
DLDZ(2,2)=0.033
DLDZ(2,3)=0.033
DLDZ(2,4)=0.075*D(7)
DLDZ(2,5)=-0.022*D(8)
DLDZ(2,9)=-0.081*Z(4)*D(3)
DLDZ(2,10)=-0.026*Z(4)*D(4)
DLDZ(3,6)=DLDZ(1,1)
DLDZ(3,8)=-DLDZ(1,3)
DLDZ(3,9)=-DLDZ(1,4)
DLDZ(4,7)=DLDZ(2,2)
DLDZ(4,8)=DLDZ(2,3)
DLDZ(4,9)=-DLDZ(2,4)
DLDZ(4,10)=DLDZ(2,5)
RETURN
END
      DLA   10
      DLA   20
      DLA   30
      DLA   40
      DLA   50
      DLA   60
      DLA   70
      DLA   80
      DLA   90
      DLA  100
      DLA  110
      DLA  120
      DLA  130
      DLA  140
      DLA  150
      DLA  160
      DLA  170
      DLA  180
      DLA  190
      DLA  200
      DLA  210
      DLA  220
      DLA  230
      DLA  240
      DLA  250
      DLA  260
      DLA  270
      DLA  280
      DLA  290
      DLA  300
      DLA  310
      DLA  320
      DLA  330
      DLA  340
      DLA  350
      DLA  360
      DLA  370
      DLA  380

```

SUBROUTINE EXPT

C ****THIS SUBROUTINE EVALUATES THE EXPERIMENTALLY
CALCULATED DETERMINED FUNCTIONS FPE3, FPE4, UP1, UP2, D(.)
CALCULATED AND ALL THEIR NECESSARY PARTIAL DERIVATIVES

C

```

COMMON /CCOM2/ DD(100), DDL(100), DTIFUL, DTINOW, ISEZS, LFLAG(50), NFLAG,
1 NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TINEK
COMMON /VARIAB/ ALAMB(10), D(23), DLDZ(4,10)
COMMON /EXPT/ FPE3, FPE4, DFPE3(9), DFPE4(9), UP1, UP2, DUP1(4), DUP2(4),
1 SD(6), DSD(6,9)
DIMENSION Z(100)
EQUIVALENCE (SS,Z)
```

C

```

FPE3=5.393*(D(9)-1.)+257.1*ALAMB(8)
FPE4=64.7*(D(10)-1.)+0.0068*(D(11)-1.)+378.0*ALAMB(9)
DF33=487.53*D(9)
DF38=257.1
DF44=1549.6*D(10)+1.631*D(11)
DF49=378.0
```

C

```

C****DFPE.(I) = D FPE./D Z(I)
```

C

```

DO 10 I=1,4
DFPE3(I)=DF33*DLDZ(I,3)+DF38*DLDZ(I,8)
DFPE4(I)=DF44*DLDZ(I,4)+DF49*DLDZ(I,9)
CONTINUE
DO 20 I=5,9
DFPE3(I)=0.
DFPE4(I)=0.
```

20 CONTINUE

```

UP1=0.8*D(14)+0.084*D(15)-0.753*D(16)
UP1=UP1-(7.9*D(17)+0.09*D(18))*Z(3)
UP2=1.23E-07*D(19)-6.30*D(20)-20.1*D(21)+2.10
UP2=UP2+(0.39*D(22)+1.85*D(23))*Z(4)
UP2=-UP2
```

C

```

C****DUP.(I) = D UP./D Z(I)
```

C

```

DUP1(2)=0.
DUP1(4)=0.
DUP2(1)=0.
DUP2(3)=0.
DUP1(1)=-2.73*D(14)-1.26*D(15)-1.92*D(16)
DUP1(1)=DUP1(1)-(0.162*D(18)-21.49*D(17))*Z(3)
DUP1(3)=-(7.9*D(17)+0.09*D(18))
DUP2(2)=1.06E-06*D(19)+18.27*D(20)+323.6*D(21)
DUP2(2)=DUP2(2)+(0.306*D(22)-6.35*D(23))*Z(4)
DUP2(2)=-DUP2(2)
DUP2(4)=0.30*D(22)+1.85*D(23)
DUP2(4)=-DUP2(4)
SD(1)=0.024+0.0188*Z(1)
SD(2)=0.03*D(12)+0.036
ARC3=Z(1)-0.63
ARC4=1.309*Z(1)-0.916
ARC5=Z(2)+0.685
SD(3)=0.952*COS(ARC3)-0.002
SD(4)=0.037*COS(ARC4)+0.026
SD(5)=0.058*(ARC5**2)*D(13)
SD(6)=0.055
```

C

```

C****DSD(I,J) = D SD(I)/D Z(J)
```

C

```

DO 30 I=1,9
DO 30 J=1,6
DSD(I,J)=0.0
30 CONTINUE
DSD(1,1)=0.0188
DSD(2,2)=-0.260*(Z(2)+0.17)*D(12)
DSD(3,1)=-0.052*SIN(ARC3)
DSD(4,1)=-0.048*SIN(ARC4)
DSD(5,2)=0.116*ARC5*D(13)-0.69*(ARC5**2)*D(13)
RETURN
END
```

EXP 10
EXP 20
EXP 30
EXP 40
EXP 50
EXP 60
EXP 70
EXP 80
EXP 90
EXP 100
EXP 110
EXP 120
EXP 130
EXP 140
EXP 150
EXP 160
EXP 170
EXP 180
EXP 190
EXP 200
EXP 210
EXP 220
EXP 230
EXP 240
EXP 250
EXP 260
EXP 270
EXP 280
EXP 290
EXP 300
EXP 310
EXP 320
EXP 330
EXP 340
EXP 350
EXP 360
EXP 370
EXP 380
EXP 390
EXP 400
EXP 410
EXP 420
EXP 430
EXP 440
EXP 450
EXP 460
EXP 470
EXP 480
EXP 490
EXP 500
EXP 510
EXP 520
EXP 530
EXP 540
EXP 550
EXP 560
EXP 570
EXP 580
EXP 590
EXP 600
EXP 610
EXP 620
EXP 630
EXP 640
EXP 650
EXP 660
EXP 670
EXP 680
EXP 690
EXP 700
EXP 710
EXP 720
EXP 730
EXP 740

FUNCTION VBAR (TIME)

C
*****THIS FUNCTION EVALUATES THE VERTICAL
*****POSITION OF THE HIP AS A FUNCTION OF TIME
C
COMMON /WALK/ BETA1,BETA2,TAV,XHOR,VMEAN
C
PI=3.1415926536
VBAR=BETA1*SIN(4.*PI*(TIME+BETA2*TAV)/TAV)
RETURN
END

VBA	10
VBA	20
VBA	30
VBA	40
VBA	50
VBA	60
VBA	70
VBA	80
VBA	90
VBA	100
VBA	110

FUNCTION VDDOT (TIME)

C

C*****THIS FUNCTION EVALUATES THE SECOND TIME
C*****DERIVATIVE OF THE VERTICAL HIP POSITION

C

```
COMMON /WALK/ BETA1,BETA2,TAV,XHOR,VMEAN
PI=3.1415926536
ARC=4.*PI/TAV
VDDOT=-BETA1*(ARC**2)*SIN(ARC*(TIME+BETA2*TAV))
RETURN
END
```

VDD	10
VDD	20
VDD	30
VDD	40
VDD	50
VDD	60
VDD	70
VDD	80
VDD	90
VDD	100
VDD	110

C SUBROUTINE ANKLE

C THIS SUBROUTINE EVALUATES THE ANGLE
C AND ANGLE MOMENT AT SEVEN TIME STEPS IN THE CYCLE

C
COMMON /CCOM2/ ATRIB(25), LEVNT, NFA, NFE(100), MLE(100), MSTOP, NCNRDR, NANK
11APO, NMAT, NNATR, NNFL(11), NNQ(100), NNTRY, NPRNT, PPART(50,4), TNOW, TTBCGANK

2, TICL, TTIN, TTRIB(25), TTSET

COMMON /VARIAB/ ALAMB(10)

COMMON /CCOM2/ DDI(100), DDL(100), DTIFUL, DTNOW, ISEES, LFLAG(50), NFLAG, ANK

1NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TINEX

COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND

COMMON /PHYS/ ALL1, ALL2

COMMON /PSAVE/ XH1, XH2, SIGMAX

COMMON /FOOT/ DALPH1, DALPH2

COMMON /ANKLE/ D5ALPHA, UP2ALPH, MALPHA

COMMON /WALK/ BETA1, BETA2, TAV, XBOR, VMEAN, XNOT

C
T=TNOW

IF (T.GT. TEND) T=T-TEND

IF (T-TPRIMD 10,10,20

C*****LT. TPRIM

C

10 ALPHA=0.

ALPDOT=0.

GO TO 30

C

C*****T > TPRIM

C

20 ALPHA=XH1-XBOR*TNOW-ALL1*SIN(SS(1))+ALL2*SIN(SS(2)-SS(1))

ALPHA=ALPHA-XNOT

ALPHA=A COS(ALPHA)/DALPH1

ALPDOT=XBOR+ALL1*COS(SS(1))*DD(1)-ALL2*COS(SS(2)-SS(1))*(DD(2)-DD(

11))

ALPDOT=ALPDOT/(DALPH1*SQRT(1.-ALPHA**2))

C

C*****LT. TPRIM

30 PI=3.1415926536

C

SIGMA=ALPHA+SS(1)-SS(2)+PI/2.

SIGDOT=ALPDOT*DD(1)-DD(2)

ARG=((SIGMA-PI/2.)/0.401426)

ARG1=ABS(ARG)

ARC=SQRT(SQRT(ARG1))

ARC=ARC*PI/2.

D5ALPHA=0.20*DALPH2

C

C*****UP2ALPH = ANGLE MOMENT

C

SCN=ARG1/ABS(ARG1)

UP2ALPH=-SCN*TAN(ARG)

RETURN

END

ANK 10

ANK 20

ANK 30

ANK 40

ANK 50

ANK 60

ANK 70

ANK 80

ANK 90

ANK 100

ANK 110

ANK 120

ANK 130

ANK 140

ANK 150

ANK 160

ANK 170

ANK 180

ANK 190

ANK 200

ANK 210

ANK 220

ANK 230

ANK 240

ANK 250

ANK 260

ANK 270

ANK 280

ANK 290

ANK 300

ANK 310

ANK 320

ANK 330

ANK 340

ANK 350

ANK 360

ANK 370

ANK 380

ANK 390

ANK 400

ANK 410

ANK 420

ANK 430

ANK 440

ANK 450

ANK 460

ANK 470

ANK 480

ANK 490

ANK 500

ANK 510

ANK 520

ANK 530

ANK 540

ANK 550

FUNCTION FVERT (TIME)

C
 THIS FUNCTION CALCULATES THE VERTICAL
 FORCE ON THE FOOT AS A FUNCTION OF
 TIME BY LINEAR INTERPOLATION OF DATA

C
 COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
 COMMON /FORCE/ TSCALE, VSCALE, RSCALE, XSCALE, TT1(45), TT2(47), TT3(18)
 1, TV(45), TH(47), TX(18), W

C
 T=TIME
 IF (TIME.GT. TEND) T=TEND
 TAV=T2PRIM-TINIT
 XX=(T-TINIT)/(TAV*TSCALE)
 FVERT=FTABLE(TT1, TV, XX, 45)*VSCALE*X
 RETURN
 END

FVE	10
FVE	29
FVE	39
FVE	49
FVE	59
FVE	69
FVE	79
FVE	89
FVE	99
FVE	109
FVE	119
FVE	129
FVE	139
FVE	149
FVE	159
FVE	169
FVE	179

FUNCTION FHORZ (TIME)

C
C*****THIS FUNCTION CALCULATES THE HORIZONTAL.
C*****FORCE ON THE FOOT AS A FUNCTION OF
C*****TIME BY LINEAR INTERPOLATION OF DATA

C
COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
COMMON /FORCE/ TSCALE, VSCALE, HSCALE, XSCALE, TT1(45), TT2(47), TT3(18)
1, TV(45), TH(47), TX(18), W

C
T=TIME
.IE. (TIME.GT.TEND) T=T-TEND
TAV=T2PRIM-TINIT
XX=(T-TINIT)/(TAV*TSCALE)
FHORZ=(FTABLE(TT2, TH, XX, 47))*HSCALE*W
RETURN
END

FHO	10
FHO	20
FHO	30
FHO	40
FHO	50
FHO	60
FHO	70
FHO	80
FHO	90
FHO	100
FHO	110
FHO	120
FHO	130
FHO	140
FHO	150
FHO	160
FHO	170

FUNCTION YP (TIME)

C
*****THIS FUNCTION CALCULATES THE VERTICAL
*****POSITION OF THE CENTRE OF PRESSURE ON
*****THE FOOT AS A FUNCTION OF TIME

C

YP=0.

RETURN

END

YP	10
YP	20
YP	30
YP	40
YP	50
YP	60
YP	70
YP	80
YP	90

FUNCTION XP (TIME)

C
*****THIS FUNCTION CALCULATES THE HORIZONTAL
*****POSITION OF THE CENTRE OF PRESSURE ON
*****THE FOOT AS A FUNCTION OF TIME BY
*****LINEAR INTERPOLATION.

C /
COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
COMMON /FORCE/ TSCALE, VSCALE, HSCALE, SSCALE, TT1(45), TT2(47), TT3(18)
1, TV(45), TH(47), TX(18), W
COMMON /FOOT/ DALPH1, DALPH2

C
T=TIME
IF (TIME.GT.TEND) T=T-TEND
TAV=T2PRIM-TINIT
XX=(T-TINIT)/(TAV*TSCALE)
XP=(FTABLE(TT3, TX, XX, 18))*DALPH1*XSCALE
XP= 1.15*XP-0.15*DALPH1
IF (TIME.GT.TEND) XP=XP+XBOR*TAV
RETURN
END

XP	10
XP	20
XP	30
XP	40
XP	50
XP	60
XP	70
XP	80
XP	90
XP	100
XP	110
XP	120
XP	130
XP	140
XP	150
XP	160
XP	170
XP	180
XP	190
XP	200
XP	210

```

FUNCTION FTABLE (VAR, FUNC, XX, ID
C
C*****THIS FUNCTION PERFORMS LINEAR
C***INTERPOLATION OF A TABULATED FUNCTION
C
DIMENSION VAR(1), FUNC(1)
C
NEND=M-1
DO 10 I=1,NEND
INT=I
IF (XX.GE.VAR(I).AND.XX.LE.VAR(I+1)) GO TO 20
CONTINUE
10 FTABLE=FUNC(INT)+(XX-VAR(INT))*(FUNC(INT+1)-FUNC(INT))/(VAR(INT+1)
1-VAR(INT))
RETURN
END

```

FTA	10
FTA	20
FTA	30
FTA	40
FTA	50
FTA	60
FTA	70
FTA	80
FTA	90
FTA	100
FTA	110
FTA	120
FTA	130
FTA	140
FTA	150
FTA	160

FUNCTION QQ (TIME)

```

C
*****THIS FUNCTION CALCULATES THE HORIZONTAL
*****POSITION OF THE ANKLE AS A FUNCTION
*****OF TIME
C
COMMON /TIME/ TINIT, TPRIM, T2PRIM, TEND
COMMON /WALK/ BETA1, BETA2, TAV, XHOR, VMEAN, XNOT
COMMON /PHYS/ ALL1, ALL2
COMMON /GCOM2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG,
INNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX
COMMON /FOOT/ DALPH1, DALPH2
COMMON /QFOOT/ F1

C
T=TIME
IF (TIME.GT.TEND) T=T-TEND
IF (T.GT.T2PRIM) GO TO 30
TAVI=TPRIM-TINIT
T=(T-TINIT)/TAVI
IF (T.LE.0.55) GO TO 10
I=0
IF (T.GE.1.0) I=1
ALPHA=5.1091708*(T**1)*(T-0.55)**4+0.315469
QQ=F1-DALPH1*COS(ALPHA)
GO TO 20
10 QQ=0.0948161*(1.0-EXP(-6.5*T))
F1=QQ+SQRT(DALPH1**2-(PP(TIME))**2)
20 IF (TIME.GE.TEND) QQ=QQ+XHOR*TAV
RETURN
30 QQ=XNOT*XHOR*TIME+ALL1*SIN(SS(1))+ALL2*SIN(SS(1)-SS(2))
RETURN
END

```

QQ	10
QQ	20
QQ	30
QQ	40
QQ	50
QQ	60
QQ	70
QQ	80
QQ	90
QQ	100
QQ	110
QQ	120
QQ	130
QQ	140
QQ	150
QQ	160
QQ	170
QQ	180
QQ	190
QQ	200
QQ	210
QQ	220
QQ	230
QQ	240
QQ	250
QQ	260
QQ	270
QQ	280
QQ	290
QQ	300
QQ	310
QQ	320

```

FUNCTION FTABLE (VAR, FUNC, XX, ID
C
C THIS FUNCTION PERFORMS LINEAR
C INTERPOLATION OF A TABULATED FUNCTION
C
C      DIMENSION VAR(1), FUNC(1)
C
NEND=M-1
DO 10 I=1,NEND
INT=I
IF (XX.GE.VAR(I).AND.XX.LE.VAR(I+1)) GO TO 20
CONTINUE
10 FTABLE=FUNC(INT)+(XX-VAR(INT))*(FUNC(INT+1)-FUNC(INT))/(VAR(INT+1)
1-VAR(INT))
RETURN
END

```

FTA	10
FTA	20
FTA	30
FTA	40
FTA	50
FTA	60
FTA	70
FTA	80
FTA	90
FTA	100
FTA	110
FTA	120
FTA	130
FTA	140
FTA	150
FTA	160

FUNCTION ATANH(X)
C
*****EVALUATE ARC HYPERBOLIC TANGENT
C
ATANH=ALOG((1.+X)/(1.-X)/2.
RETURN
END

ATA	10
ATA	20
ATA	30
ATA	40
ATA	50
ATA	60
ATA	70

SUBROUTINE EVNTS (IX)

C
C NUMBER OF EQUATIONS CHANGES DURING SWING PHASE

COMMON /GCOR2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG,
NNEQD, NNEQS, NNEQT, SS(100), SSL(100), TTINEX

CO TO (10, 20, 20, 20, 30), IX

10 NNEQD=18
NNEQS=12
RETURN

20 NNEQD=20
NNEQS=10
RETURN

30 END

	EVN	10
	EVN	20
	EVN	30
	EVN	40
	EVN	50
	EVN	60
	EVN	70
	EVN	80
	EVN	90
	EVN	100
	EVN	110
	EVN	120
	EVN	130
	EVN	140

SUBROUTINE ORDER

C
 C*****THIS SUBROUTINE REVERSES THE ORDER
 C*****OF THE 20 SS(I) AND DD(I) VARIABLES
 C

COMMON /GC0M2/ DD(100), DDL(100), DTFUL, DTNOW, ISEES, LFLAG(50), NFLAG,	ORD 10
INNEQD, NNEQS, NNEQT, SS(100), SSL(100), TIMEX	ORD 20
COMMON /MODE/ MODE, IMODE	ORD 30
DO 10 I= 1, 10	ORD 40
J=21-I	ORD 50
STORE=SS(I)	ORD 60
SS(I)=SS(J)	ORD 70
SS(J)=STORE	ORD 80
STORE=DD(I)	ORD 90
DD(I)=DD(J)	ORD 100
DD(J)=STORE	ORD 110
CONTINUE	ORD 120
RETURN	ORD 130
END	ORD 140
	ORD 150
	ORD 160
	ORD 170
	ORD 180
	ORD 190

10

APPENDIX C

LISTING OF
OPTISEP SUBROUTINES

```

SUBROUTINE DAVID (N, RMAX, RMIN, NCONS, NEQUS, XSTRT, G, F, MAXM, IPRINT, IDDAV
1ATA, R, REDUCE, U, X, PHI, PSI, H, GS, D, GN, GA, Y, DT, C, YT, PHX, PSX, PART, PAST, DAV
2CH, UX) 1
C DAVIDON FLETCHER AND POWELL METHOD OF OPTIMIZATION DAV 2
C DIMENSION X(1), RMAX(1), RMIN(1), XSTRT(1), H(N,N), GS(1), D(1), GDAV 3
C IN(1), GA(1), Y(1), DT(N,N), YT(N,N), C(N,N), PHI(1), PSI(1), PHX(NDAV 4
C 2,1), PSX(N,1), PART(1), PAST(1), CH(1), UX(1) DAV 5
C COMMON /OPTI/ KD, NINDEX DAV 6
C CLEARING ALL THE ARRAYS BEFORE USE DAV 7
C DO 10 I=1,N DAV 8
C GS(I)=0.0 DAV 9
C CN(I)=0.0 DAV 10
C GA(I)=0.0 DAV 11
C Y(I)=0.0 DAV 12
C UX(I)=0.0 DAV 13
C DO 10 J=1,N DAV 14
C DT(I,J)=0.0 DAV 15
C YT(I,J)=0.0 DAV 16
C 10 C(I,J)=0.0 DAV 17
C DO 20 I=1,N DAV 18
C CH(I)=F*(ABS(RMAX(I))-RMIN(I)) DAV 19
C X(I)=XSTRT(I) DAV 20
C 20 CONTINUE DAV 21
C ROLD=R DAV 22
C LK=1 DAV 23
C L=0 DAV 24
C WRITE (6,390) DAV 25
C KOUNT=0 DAV 26
C IF (IDATA.NE.1) GO TO 30 DAV 27
C WRITE (6,460) IPRINT DAV 28
C WRITE (6,470) IDATA DAV 29
C WRITE (6,480) N DAV 30
C WRITE (6,490) NCONS DAV 31
C WRITE (6,500) F DAV 32
C WRITE (6,510) MAXM DAV 33
C WRITE (6,520) R DAV 34
C WRITE (6,530) REDUCE DAV 35
C WRITE (6,520) G DAV 36
C WRITE (6,530) NEQUS DAV 37
C WRITE (6,540) (RMAX(I),I=1,N) DAV 38
C WRITE (6,550) (RMIN(I),I=1,N) DAV 39
C WRITE (6,560) (XSTRT(I),I=1,N) DAV 40
C 30 CALL OPTIME (X,FUN1,PHI,PSI,NCONS,NEQUS,NVIOL,R) DAV 41
C C SUBROUTINE PRTIAL RETURNS THE GRADIENTS REQUIRED FOR COMPUTATION DAV 42
C C TO START WITH MATRIX H IS CHOSEN AS A UNIT MATRIX DAV 43
C CALL PRTIAL (X,N,NCONS,NEQUS,PHI,PSI,GS,R,CH,UX,PSX,PHX,PART,PAST) DAV 44
C JJ=0 DAV 45
C 40 DO 50 I=1,N DAV 46
C 50 DO 50 J=1,N DAV 47
C H(I,J)=0.0 DAV 48
C DO 60 I=1,N DAV 49
C KK=I DAV 50
C H(I,KK)=1.0 DAV 51
C 60 CONTINUE DAV 52
C JJ=JJ+1 DAV 53
C 70 DO 80 I=1,N DAV 54
C D(I)=0.0 DAV 55
C 80 CONTINUE DAV 56
C DO 100 I=1,N DAV 57
C DO 90 J=1,N DAV 58
C D(I)=(H(I,J)*GS(J))+D(I) DAV 59
C 90 CONTINUE DAV 60
C IF (D(I).EQ.0) D(I)=1.E-50 DAV 61
C D(I)=-IN(I) DAV 62
C 100 CONTINUE DAV 63
C C IF D(I) DOES NOT ENSURE THAT FUNCTION WILL DECREASE THEN RESET DAV 64
C H MATRIX AS A UNIT MATRIX DAV 65
C IF (JJ.GT.1) GO TO 330 DAV 66
C DO 110 I=1,N DAV 67
C IF ((GS(I)/D(I)).GT.0.) GO TO 40 DAV 68
C 110 CONTINUE DAV 69
C JJ=0 DAV 70
C L=L+1 DAV 71
C FUNC=FUN1 DAV 72
C DAV 73
C DAV 74

```

```

C SUBROUTINE FIND RETURNS ALMDA, WHICH GIVES OPTIMUM STEP LENGTH
    CALL FIND (X, ALMDA, D, N, PHI, PSI, NCNS, NEQUS, FUNC, RD)
    DO 120 I=1,N
    X(I)=X(I)+ALMDA*D(I)
120 CONTINUE
    CALL OPTIM2 (X, FUN2, PHI, PSI, NCNS, NEQUS, NVIOL, RD)
    IF FUNCTION STARTS INCREASING PROGRAM IS RESTARTED WITH NEW R
    KOUNT=KOUNT+1
    IF (IPRINT.LE.0) GO TO 140
    IF (L.GT.1) GO TO 130
    WRITE (6,400)
    WRITE (6,410)
    WRITE (6,420)
    WRITE (6,430)
130 IF (IPRINT.NE.KOUNT) GO TO 140
    KOUNT=0
    CALL UREAL (X, UD)
    WRITE (6,440) L, U, FUN2, (X(I), I=1,N)
C CRITERION FOR OPTIMUM
140 IF (ABS(FUN1-FUN2).LE.C) GO TO 340
    IF (L.GE.MAXMD GO TO 330
    IF (FUN2.LE.FUN1) GO TO 160
    DO 150 I=1,N
    X(I)=X(I)-ALMDA*D(I)
150 CONTINUE
    FUN2=FUN1
    GO TO 340
160 CONTINUE
    CALL PRTIAL (X, N, NCNS, NEQUS, PHI, PSI, CN, R, CH, UX, PSX, PHX, PART, PAST)
    C THIS SECTION COMPUTES MATRIX H TO BE USED IN THE NEXT ITERATION
    DO 170 I=1,N
    Y(I)=CN(I)-GS(I)
170 CONTINUE
    DO 180 I=1,N
    GA(I)=0.0
180 CONTINUE
    DO 190 I=1,N
    DO 190 K=1,N
    190 GA(I)=GA(I)+(H(I,K)*GS(K))
    PROD1=0.0
    DO 200 I=1,N
    PROD1=PROD1+GA(I)*GS(I)
200 CONTINUE
    DO 210 I=1,N
    GA(I)=0.0
210 CONTINUE
    DO 220 I=1,N
    DO 220 K=1,N
    220 GA(I)=GA(I)+(H(I,K)*Y(K))
    PROD2=0.0
    DO 230 I=1,N
    PROD2=PROD2+(GA(I)*Y(I))
230 CONTINUE
    DO 240 I=1,N
    DO 240 J=1,N
240 DT(I,J)=D(I)*D(J)
    DO 250 I=1,N
    DO 250 J=1,N
    250 YT(I,J)=Y(I)*Y(J)
    DO 260 I=1,N
    DO 260 J=1,N
    260 C(I,J)=0.0
    DO 270 I=1,N
    DO 270 J=1,N
    270 C(I,J)=(H(I,K)*YT(K,J))+C(I,J)
    DO 290 I=1,N
    DO 290 J=1,N
    SUM=0.0
    DO 280 K=1,N
    SUM=SUM+(C(I,K)*H(K,J))
280 CONTINUE
    C(I,J)=SUM
290 IF (ABS(PROD1).LT.1.E-30) PROD1=1.E-30

```

IF (ABS(PROD2).LT. 1.E-30) PROD2=1.E-30	DAV	149
QU01=ALMDA/PROD1	DAV	150
QU02=1./PROD2	DAV	151
DO 300 I=1,N	DAV	152
DO 300 J=1,N	DAV	153
DT(I,J)=DT(I,J)*QU01	DAV	154
300 C(I,J)=C(I,J)*QU02	DAV	155
DO 310 I=1,N	DAV	156
DO 310 J=1,N	DAV	157
310 H(I,J)=H(I,J)+DT(I,J)-C(I,J)	DAV	158
DO 320 I=1,N	DAV	159
GS(I)=GN(I)	DAV	160
320 CONTINUE	DAV	161
FUN1=FUN2	DAV	162
GO TO 70	DAV	163
K0=1	DAV	164
WRITE (6,450) L	DAV	165
RETURN	DAV	166
340 R=R*REDUCE	DAV	167
CALL UREAL (X,UNEW)	DAV	168
IF (LK.EQ.1) GO TO 350	DAV	169
IF (ABS(UOLD-UNEW).LE.G) GO TO 360	DAV	170
350 UOLD=UNEW	DAV	171
LK=LK+1	DAV	172
GO TO 30	DAV	173
360 CALL OPTIM2 (X,FUNZ,PHI,PSI,NCONS,NEQUS,NVIOL,R)	DAV	174
IF (NVIOL.NE.0) GO TO 380	DAV	175
K0=0	DAV	176
370 U=UNEW	DAV	177
R=ROLD	DAV	178
RETURN	DAV	179
380 K0=1	DAV	180
WRITE (6,590)	DAV	181
GO TO 370	DAV	182
C	DAV	183
C	DAV	184
390 FORMAT (1H1,59HOPTIMIZATION BY DAVIDON FLETCHER AND POWELL METHOD,	DAV	185
1/)	DAV	186
400 FORMAT (1H1)	DAV	187
410 FORMAT (1H0,52H INTERMEDIATE OUTPUT FOR DAVIDON FLETCHER AND POWELL DAV	DAV	188
1L,/)	DAV	189
420 FORMAT (1H0,59H UART IS THE ARTIFICIAL UNCONSTRAINED OPTIMIZATION DAV	DAV	190
1FUNCTION,/)	/)	191
430 FORMAT (1H0,BESTEP NO.,6X,1HU,12X,4HUART,30X,26HINDEPENDENT VARIAB/)	DAV	192
ILES X(I),/)	/)	193
440 FORMAT (15,3X,6E16.8,/(40X,4E16.8))	/)	194
450 FORMAT (1H0,26HDAVIDON HAS HUNG UP AFTER ,14,10HITERATIONS,/)	/)	195
460 FORMAT (61H0INTERMEDIATE OUTPUT EVERY 1PRINT TH CYCLE. IPR/)	DAV	196
1INT =,16)	/)	197
470 FORMAT (61H0INPUT DATA IS PRINTED OUT FOR IDATA=1 ONLY. IDA/)	DAV	198
1TA =,16)	/)	199
480 FORMAT (61H0NUMBER OF INDEPENDENT VARIABLES /)	DAV	200
1 N =,16)	/)	201
490 FORMAT (61H0NUMBER OF INEQUALITY (.GE.) CONSTRAINTS NC/)	DAV	202
1ONS =,16)	/)	203
500 FORMAT (61H0FRACTION OF RANGE USED AS STEP SIZE /)	DAV	204
1 F =,E19.8)	/)	205
510 FORMAT (61H0MAXIMUM NUMBER OF MOVES PERMITTED M/)	DAV	206
1AXM =,16)	/)	207
520 FORMAT (61H0STEP SIZE FRACTION USED AS CONVERGENCE CRITERION. /)	DAV	208
1 G =,E19.8)	/)	209
530 FORMAT (61H0NUMBER OF EQUALITY CONSTRAINTS. NE/)	DAV	210
1QUS =,16)	/)	211
540 FORMAT (61H0ESTIMATED UPPER BOUND ON RANGE OF X(I). RMAX/)	DAV	212
1(I) =,/(5E16.8)	/)	213
550 FORMAT (61H0ESTIMATED LOWER BOUND ON RANGE OF X(I). RMIN/)	DAV	214
1(I) =,/(5E16.8)	/)	215
560 FORMAT (61H-STARTING VALUES OF X(I). XSTRT/)	DAV	216
1(I) =,/(5E16.8)	/)	217
570 FORMAT (61H0PENALTY MULTIPLIER USED IN DAVID. /)	DAV	218
1 R =,E19.8)	/)	219
580 FORMAT (61H0REDUCTION FACTOR FOR (R) AFTER EACH MINIMIZATION. RED/)	DAV	220
1UCE =,E19.8)	/)	221
590 FORMAT (69H0DAVID HAS HUNG UP IN AN INFEASIBLE ZONE - TRY ANOTHER /)	DAV	222

1 STARTING POINT)
END)

223
224

```

SUBROUTINE FIND (X, ALMDA, D, N, PHI, PSI, NCONS, NEQUS, FUN1, RD)
DIMENSION X(1), D(1), PHI(1), PSI(1)
COMMON /OPT1/ K0, NINDEX
L=0
A1=1.0
KK=1
10 K=1
KK=KK+1
IF (KK.GT.50) GO TO 250
S=2.0
C THIS SECTION FINDS BOUNDS ON THE VALUE OF ALMDA
20 AL=A1*((S**(K-1))/(S-1.0))
DO 30 I=1,N
X(I)=X(I)+AL*D(I)
30 CONTINUE
CALL OPTIM2 (X, FUN2, PHI, PSI, NCONS, NEQUS, NVIOL, RD)
DO 40 I=1,N
X(I)=X(I)-AL*D(I)
40 CONTINUE
IF (FUN2.GT.FUN1) GO TO 50
K=K+1
FUN1=FUN2
IF (K.GT.75) GO TO 260
GO TO 20
50 IF (K.NE.1) GO TO 60
A1=A1/2.
GO TO 10
60 IF (K.EQ.2) GO TO 70
GO TO 80
70 A=0.0
B=A1
C=AL
GO TO 90
80 A=A1*((S**(K-2))-1.)/(S-1.)
B=A1*((S**(K-1))-1.)/(S-1.)
C=AL
90 CONTINUE
C THIS SECTION FINDS THE EXACT VALUE OF ALMDA BY POLYNOMIAL SEARCH
C BEST VALUE OF ALMDA IS BRACKETED WITHIN A AND C
DO 100 I=1,N
X(I)=X(I)+A*D(I)
100 CONTINUE
CALL OPTIM2 (X, FA, PHI, PSI, NCONS, NEQUS, NVIOL, RD)
DO 110 I=1,N
X(I)=X(I)-A*D(I)
110 CONTINUE
DO 120 I=1,N
X(I)=X(I)+B*D(I)
120 CONTINUE
CALL OPTIM2 (X, FB, PHI, PSI, NCONS, NEQUS, NVIOL, RD)
DO 130 I=1,N
X(I)=X(I)-B*D(I)
130 CONTINUE
DO 140 I=1,N
X(I)=X(I)+C*D(I)
140 CONTINUE
CALL OPTIM2 (X, FC, PHI, PSI, NCONS, NEQUS, NVIOL, RD)
DO 150 I=1,N
X(I)=X(I)-C*D(I)
150 CONTINUE
160 AD1=((B*B)-(C*C))*FA+(((C*C)-(A*A))*FB)+(((A*A)-(B*B))*FC)
AD2=2.*(((B-C)*FA)+((C-A)*FB)+((A-B)*FC))
IF (ABS(AD2).LT.1.E-40) GO TO 170
AD=AD1/AD2
GO TO 180
170 AD=(A+C)/2.
180 CONTINUE
IF (AD.LT.A) AD=(A+B)/2.
IF (AD.GT.C) AD=(B+C)/2.
L=L+1
IF (L.GT.10) GO TO 240
C AD IS THE MINIMUM OF THE POLYNOMIAL PASSING THROUGH A B AND C
DO
FIN 1
FIN 2
FIN 3
FIN 4
FIN 5
FIN 6
FIN 7
FIN 8
FIN 9
FIN 10
FIN 11
FIN 12
FIN 13
FIN 14
FIN 15
FIN 16
FIN 17
FIN 18
FIN 19
FIN 20
FIN 21
FIN 22
FIN 23
FIN 24
FIN 25
FIN 26
FIN 27
FIN 28
FIN 29
FIN 30
FIN 31
FIN 32
FIN 33
FIN 34
FIN 35
FIN 36
FIN 37
FIN 38
FIN 39
FIN 40
FIN 41
FIN 42
FIN 43
FIN 44
FIN 45
FIN 46
FIN 47
FIN 48
FIN 49
FIN 50
FIN 51
FIN 52
FIN 53
FIN 54
FIN 55
FIN 56
FIN 57
FIN 58
FIN 59
FIN 60
FIN 61
FIN 62
FIN 63
FIN 64
FIN 65
FIN 66
FIN 67
FIN 68
FIN 69
FIN 70
FIN 71
FIN 72
FIN 73
FIN 74

```

190	CONTINUE	FIN	75
	CALL OPTIM2 (X,FD,PHI,PSI,NCONS,NEQUS,NVIOL,R)	FIN	76
	DO 200 I=1,N	FIN	77
	X(I)=X(I)-AD*D(I)	FIN	78
200	CONTINUE	FIN	79
	IF (B.GT.AD) GO TO 220	FIN	80
	IF (FB.GT.FD) GO TO 210	FIN	81
	C=AD	FIN	82
	FC=FD	FIN	83
	GO TO 160	FIN	84
210	A=B	FIN	85
	FA=FB	FIN	86
	B=AD	FIN	87
	FB=FD	FIN	88
	GO TO 160	FIN	89
220	IF (FB.GT.FD) GO TO 230	FIN	90
	A=AD	FIN	91
	FA=FD	FIN	92
	GO TO 160	FIN	93
230	C=B	FIN	94
	FC=FB	FIN	95
	B=AD	FIN	96
	FB=FD	FIN	97
	GO TO 160	FIN	98
240	ALMDA=B	FIN	99
	IF (FA.LT.FB) ALMDA=A	FIN	100
	GO TO 270	FIN	101
250	ALMDA=0.0	FIN	102
	GO TO 270	FIN	103
260	ALMDA=AL	FIN	104
270	RETURN	FIN	105
	END	FIN	106

```

C SUBROUTINE OPTIM2 (X,UART,PHI,PSI,NCONS,NEQUS,NVIOL,R) OPT 1
C DIMENSION X(1), PHI(1), PSI(1) OPT 2
C COMMON /OPT1/ K0,NNINDEX OPT 3
C VERY MINOR VIOLATIONS OF INEQUALITY CONSTRAINTS SHOULD NOT MAKE OPT 4
C THE ENTIRE SOLUTION INFEASIBLE. THEREFORE TEST FOR PHI(I).GE.ZERO OPT 5
C WHERE ZERO=-1.0E-10 OPT 6
C ZERO=-1.0E-10 OPT 7
C NVIOL=0 OPT 8
C SUM1=0.0 OPT 9
C SUM2=0.0 OPT 10
C CALL UREAL (X,U) OPT 11
C SEEIC3 PENALTY FUNCTIONS - OPT 12
C
C THE ARTIFICIAL OBJECTIVE FUNCTION IS OF THE FORM OPT 13
C     UART=UREAL + R*SUM( 1./PHI(I)) + SUM((PSI(J)**2)/SQRT(R)) OPT 14
C
C DIV=SQRT(R) OPT 15
C IF (NCONS.LE.0) GO TO 30 OPT 16
C CALL CONST (X,NCONS,PHI) OPT 17
C DO 20 I=1,NCONS OPT 18
C IF (PHI(I).GE.ZERO) GO TO 10 OPT 19
C NVIOL=NVIOL+1 OPT 20
C ADD A SEVERE PENALTY TO ANY PHI(I) WHICH IS VIOLATED OPT 21
C SUM1=SUM1+ABS(PHI(I))*10.0E+20 OPT 22
C GO TO 20 OPT 23
C AVOID DIVIDING BY APPROXIMATELY ZERO, THERE IS NO POINT PENALIZING OPT 24
C A VERY SMALL PHI(I) ANYWAY OPT 25
C IF (ABS(PHI(I)).LT.-ZERO) GO TO 20 OPT 26
C SUM1=SUM1+R/ABS(PHI(I)) OPT 27
C CONTINUE OPT 28
C IF (NEQUS.LE.0) GO TO 50 OPT 29
C CALL EQUAL (X,PSI,NEQUS) OPT 30
C DO 40 J=1,NEQUS OPT 31
C IF (PSI(J).GE.ZERO.AND.PSI(J).LE.-ZERO) GO TO 40 OPT 32
C SUM2=SUM2+(ABS(PSI(J))**2)/DIV OPT 33
C NVIOL=NVIOL+1 OPT 34
C CONTINUE J OPT 35
C UART=U+SUM1+SUM2 OPT 36
C RETURN OPT 37
C END OPT 38

```

```

SUBROUTINE PRTIAL (X, N, NCONS, NEQUS, PHI, PSI, C, R, CH, UX, PSX, PHX, PART, PRT
1PAST)
DIMENSION X(1), C(1), PHI(1), PSI(1), UX(1), PHX(N,1), PSX(N,1), CPRT
1H(1), PART(1), PAST(1)
DIV=SQRT(R)
ZERO=-1.0E-10
CALL SUPPLY (X, CH, PHI, PSI, PSX, PHX, UX, N, NCONS, NEQUS, PART, PAST)
NN=0
DO 10 I=1,N
G(I)=UX(I)
10 CONTINUE
IF (NCONS.EQ.0) GO TO 40
CALL CONST (X, NCONS, PHI)
DO 30 I=1,N
DO 30 J=1,NCONS
IF (PHI(J).GT.ZERO) GO TO 20
NN=NN+1
G(I)=G(I)+(10.E+20)*ABS(PHX(I,J))
GO TO 30
20 IF (PHI(J).LT.-ZERO) GO TO 30
G(I)=G(I)-(R*PHX(I,J)/(PHI(J)**2))
30 CONTINUE
40 CONTINUE
IF (NEQUS.EQ.0) GO TO 70
DO 60 I=1,N
DO 50 J=1,NEQUS
G(I)=G(I)+2.*(PAST(J))*PSX(I,J)/DIV
50 CONTINUE
60 CONTINUE
70 IF (NCONS.EQ.0) GO TO 130
IF (NN.GT.0) GO TO 130
DO 120 I=1,N
X(I)=X(I)+CH(I)
CALL CONST (X, NCONS, PHI)
X(I)=X(I)-2.*CH(I)
CALL CONST (X, NCONS, PART)
X(I)=X(I)+CH(I)
DO 90 J=1,NCONS
IF (PHI(J).GT.ZERO) GO TO 80
GO TO 100
80 IF (PART(J).GT.ZERO) GO TO 90
GO TO 110
90 CONTINUE
GO TO 120
100 G(I)=1.
GO TO 120
110 G(I)=-1.0
120 CONTINUE
130 RETURN
END

```

```

SUBROUTINE ANSWER (U,X,PHI,PSI,N,NCONS,NEQUS)          ANS   1
COMMON /OPTI/ KO,NNDEX                                ANS   2
DIMENSION X(1), PHI(1), PSI(1)                         ANS   3
THIS SUBROUTINE IS USED MERELY TO OUTPUT THE FINAL SOLUTION IN A    ANS   4
STANDARD FORM. IF AN OPTIMUM IS NOT REACHED(KO=1) THEN THE RESULTS    ANS   5
AT THE LAST ITERATION MAY BE PRINTED OUT.                 ANS   6
CALL UREAL (X,U)                                       ANS   7
IF (KO.EQ.0) GO TO 10                                 ANS   8
WRITE (6,60)                                         ANS   9
WRITE (6,70) U                                       ANS  10
GO TO 30                                              ANS  11
10 IF (NNDEX.EQ.3) GO TO 20                           ANS  12
WRITE (6,80)                                         ANS  13
WRITE (6,90) U                                       ANS  14
GO TO 30                                              ANS  15
20 WRITE (6,150)                                     ANS  16
WRITE (6,70) U                                       ANS  17
30 WRITE (6,100) (I,X(I),I=1,N)                      ANS  18
IF (NCONS.EQ.0) GO TO 40                           ANS  19
CALL CONST (X,NCONS,PHI)                            ANS  20
WRITE (6,110)                                         ANS  21
WRITE (6,120) (I,PHI(I),I=1,NCONS)                  ANS  22
40 IF (NEQUS.EQ.0) GO TO 50                           ANS  23
CALL EQUAL (X,PSI,NEQUS)                            ANS  24
WRITE (6,130)                                         ANS  25
WRITE (6,140) (I,PSI(I),I=1,NEQUS)                  ANS  26
50 RETURN                                            ANS  27
C
C
C
60 FORMAT (1H-,16X,25HRESULTS AT LAST ITERATION,/)      ANS  28
70 FORMAT (29X,3HU *,E16.8//)                          ANS  29
80 FORMAT (1H1,21X,22HOPTIMUM SOLUTION FOUND,/)       ANS  30
90 FORMAT (20X,12HMINIMUM U =,E16.8//)                ANS  31
100 FORMAT (25X,2HX(,12,3H) *,E16.8)                 ANS  32
110 FORMAT (1H-,22HINEQUALITY CONSTRAINTS)             ANS  33
120 FORMAT (23X,4HPHI(,12,3H) *,E16.8)                ANS  34
130 FORMAT (1H-,22HEQUALITY CONSTRAINTS)               ANS  35
140 FORMAT (23X,4HPSI(,12,3H) *,E16.8)                ANS  36
150 FORMAT (1H1,19X,23HFEASIBLE STARTING POINT,/)     ANS  37
END                                                 ANS  38
                                         ANS  39
                                         ANS  40
                                         ANS  41

```

APPENDIX D
LISTING OF
GASP IV SUBROUTINES


```

GSP 271
GSP 272
GSP 273
GSP 274
GSP 275
GSP 276
GSP 277
GSP 278
GSP 279
GSP 280
GSP 281
GSP 282
GSP 283
GSP 284
GSP 285
GSP 286
GSP 287
GSP 288
GSP 289
GSP 290
GSP 291
GSP 292
GSP 293
GSP 294
GSP 295
GSP 296
GSP 297
GSP 298
GSP 299
GSP 300
GSP 301
GSP 302
GSP 303
GSP 304
GSP 305
GSP 306
GSP 307
GSP 308
GSP 309
GSP 310
GSP 311
GSP 312
GSP 313
GSP 314
GSP 315
GSP 316
GSP 317
GSP 318
GSP 319
GSP 320
GSP 321
GSP 322
GSP 323
GSP 324
GSP 325
GSP 326
GSP 327
GSP 328
GSP 329
GSP 330
GSP 331
GSP 332
GSP 333
GSP 334
GSP 335
GSP 336
GSP 337
GSP 338
GSP 339
GSP 340
GSP 341
GSP 342
GSP 343
GSP 344
GSP 345
GSP 346
GSP 347
GSP 348
GSP 349
GSP 350
GSP 351
GSP 352
GSP 353
GSP 354
GSP 355
GSP 356
GSP 357
GSP 358
GSP 359
GSP 360
GSP 361
GSP 362
GSP 363
GSP 364
GSP 365
GSP 366
GSP 367
GSP 368
GSP 369
GSP 370
GSP 371
GSP 372
GSP 373
GSP 374
GSP 375
GSP 376
GSP 377
GSP 378
GSP 379
GSP 380
GSP 381
GSP 382
GSP 383
GSP 384
GSP 385
GSP 386
GSP 387
GSP 388
GSP 389
GSP 390
GSP 391
GSP 392
GSP 393
GSP 394
GSP 395
GSP 396
GSP 397
GSP 398
GSP 399
GSP 400
GSP 401
GSP 402
GSP 403
GSP 404
GSP 405
GSP 406
GSP 407
GSP 408
GSP 409
GSP 410
GSP 411
GSP 412
GSP 413
GSP 414
GSP 415
GSP 416
GSP 417
GSP 418
GSP 419
GSP 420
GSP 421
GSP 422
GSP 423
GSP 424
GSP 425
GSP 426
GSP 427
GSP 428
GSP 429
GSP 430
GSP 431
GSP 432
GSP 433
GSP 434
GSP 435
GSP 436
GSP 437
GSP 438
GSP 439
GSP 440
GSP 441
GSP 442
GSP 443
GSP 444
GSP 445
GSP 446
GSP 447
GSP 448
GSP 449
GSP 450
GSP 451
GSP 452
GSP 453
GSP 454
GSP 455
GSP 456
GSP 457
GSP 458
GSP 459
GSP 460
GSP 461
GSP 462
GSP 463
GSP 464
GSP 465

```



```

SUBROUTINE RMOVE (NTRY, IFILE)
COMMON NSET(1)
COMMON OSET(1)
      EDIPO4 /CCOM1/ ATRIP(25),JEVNF,MFB,MFE(180),MLE(180),HSTOP,HCRDR,HGCOM1
      1HAD,4HART,4HATP,4HBFIL,4H2(180),4HTRY,4HPNT,PPARM(50,4),THOW,TTEFGCCOM1
      2,TTCL4,TTFIN,TTTB(25),TTSET
      CDMPO4 /CCOM2/ M1(180),DOL(180),DIFUL,DTHON,ISEES,LFLAG(50),NFLPC,6COM2
      3HNGCD,4HFOS,4HBT,52(180),SSC(180),TTMEX
      4COMMON /CCOM3/ BAQR,DTAX,DTMIN,DTSAV,IITES,LLEPR,LLSAV,LLSEV,RREGCOM3
      5ER,TTLAS,TTSAV
      CDMPO4 /CCOM4/ EENO(180),IINH(180),KKRN(180),HMAX(180),OQTIM(180)GCCOM6
      6,SS09V(75,5),SSTPV(75,6),VVMQ(180)
      FOJVALENCE (NSET(1),OSET(1))
      CIVLNSION NTRY(1)
      PTR Y=NTRY(1)
      IF (NTRY) 191,181,187
      181 CALL ERROR (711)
C
C*****PUT VALUES OF NTRY IN ATRIP
C
      182 DO 193 I=1,4H4TR
      M1(54)=NTRY+I
      183 ATRIP(I)=NSET(NISIA)
      IF ILG=IFILE
      IF (IFILE) 186,181,185
      184 IF ILG=IFILE
C
C*****REMOVAL OF AN ENTRY FROM IFILE.
C*****IPTR ARE POINTERS TO ACCOUNT FOR REMOVAL OF NTRY.
C*****LET JL EQUAL SUCCESSOR OF NTRY AND JK EQUAL PREDECESSOR OF NTRY.
C*****IF JL=0, NTRY WAS LAST ENTRY. IF JK=0, NTRY WAS FIRST ENTRY.
C
      185 M1(58)=HMAX+HTRYC
      JL=MSET(NISIA)
      JK=MSET(NTRYC)
      M2(THISIA)=MFA
      MFA=HTRYC
      M1(54)=JK
      IF (JL) 187,187,188
      186 IF (JK) 189,189,188
      187 IF (JK) 111,111,110
C
C*****NTRY WAS NOT FIRST OR LAST ENTRY. UPDATE POINTERS SO THAT
C*****JL IS SUCCESSOR OF JK AND JK IS PREDECESSOR OF JL.
C
      188 M2(THISIA)=JL
      MSET(JL)=JK
      GO TO 117
C
C*****NTRY WAS FIRST ENTRY BUT NOT LAST ENTRY. UPDATE POINTERS.
C
      189 MSET(JL)=8
      MFE(IFILG)=JL
      GO TO 117
C
C*****NTRY WAS LAST ENTRY BUT NOT FIRST ENTRY. UPDATE POINTERS.
C
      190 M2(THISIA)=8
      MFE(IFILG)=JK
      GO TO 112
C
C*****NTRY WAS ONLY ENTRY. UPDATE POINTERS.
C
      191 MFE(IFILG)=8
      MLE(IFILG)=8
C
C*****IF IFILG=1 UPDATE TTMAX.
C
      192 IF (IFILG=1) 114,113,117
      113 MEXT=MFCE(1)
      IF (MEXT) 114,115,119
      114 MEXT=CMPTN,118)
      CALL ERROR (317)
      115 TTFX=NSET(NEXTF+1)
      IF (TTMAX-TTFIN) 117,117,116
      116 TT4EX=TTFIN
C
C*****UPDATE FILE STATISTICS.
C
      117 THQ=MFO(IFILG)
      FENQ(IFILG)=EENO(IFILG)+XN2*(THOW-OQTIM(IFILG))
      VVO(IFILG)=VVO(IFILG)+XN2*XHO*(THON-OQTIM(IFILG))
      DOF IM(IFILG)=THON
      MM2(IFILG)=MM3(IFILG)-1
      RETURN
C
      118 FORMAT (//5X,64HPOSSIBLE CAUSE OF ERROR IS USER BLANK COMMON)
C
      END

```


APPENDIX E
INPUT ECHO CHECK

Format / Block N											
1	NNHARZ	NNPNTI	NNUN	NNDAY	NNYR	NNRNS	LLSUP	LLSUP	LLSUP	LLSUP	LLSUP
2	NNCLT	NNSTA	NNHIS	NNPEM	NNPLT	NNSTR	NNTR	NNATR	NNFIL	NNSET	NNEOD
3	1	LLABC()									
4	1	LLAAT(1)					Initial Value				
5	1	LLABH(1)					NNCCL(1)	HHLOW(1)	HHHIGH(1)		
6	1	LLABP(1)					TTAP(1)	NNVAA(1)	LLPLT	DTPLT(1)	
7	1	LLABP(1)					LLPLO(1)	LLPHI(1)	PPLO(1)	PPHI(1)	
8	1	LLABP(1)									
9	1	LLABR(1)									
10	1	LLABT(1)									
11	1	LLABU(1)									
12	1	LLABV(1)									
13	1	LLABW(1)									
14	1	LLABX(1)									
15	1	LLABY(1)									
16	1	LLABZ(1)									
17	1	LLABD(1)									
18	1	LLABE(1)									
19	1	LLABF(1)									
20	1	LLABG(1)									
21	1	LLABH(1)									
22	1	LLABI(1)									
23	1	LLABJ(1)									
24	1	LLABK(1)									
25	1	LLABL(1)									
26	1	LLABM(1)									
27	1	LLABN(1)									
28	1	LLABO(1)									
29	1	LLABP(1)									
30	1	LLABQ(1)									
31	1	LLABR(1)									
32	1	LLABS(1)									
33	1	LLABT(1)									
34	1	LLABU(1)									
35	1	LLABV(1)									
36	1	LLABW(1)									
37	1	LLABX(1)									
38	1	LLABY(1)									
39	1	LLABZ(1)									
40	1	LLABD(1)									
41	1	LLABE(1)									
42	1	LLABF(1)									
43	1	LLABG(1)									
44	1	LLABH(1)									
45	1	LLABI(1)									
46	1	LLABJ(1)									
47	1	LLABK(1)									
48	1	LLABL(1)									
49	1	LLABM(1)									
50	1	LLABN(1)									
51	1	LLABO(1)									
52	1	LLABP(1)									
53	1	LLABQ(1)									
54	1	LLABR(1)									
55	1	LLABS(1)									
56	1	LLABT(1)									
57	1	LLABU(1)									
58	1	LLABV(1)									
59	1	LLABW(1)									
60	1	LLABX(1)									
61	1	LLABY(1)									
62	1	LLABZ(1)									
63	1	LLABD(1)									
64	1	LLABE(1)									
65	1	LLABF(1)									
66	1	LLABG(1)									
67	1	LLABH(1)									
68	1	LLABI(1)									
69	1	LLABJ(1)									
70	1	LLABK(1)									
71	1	LLABL(1)									
72	1	LLABM(1)									
73	1	LLABN(1)									
74	1	LLABO(1)									
75	1	LLABP(1)									
76	1	LLABQ(1)									
77	1	LLABR(1)									
78	1	LLABS(1)									
79	1	LLABT(1)									
80	1	LLABU(1)									
81	1	LLABV(1)									
82	1	LLABW(1)									
83	1	LLABX(1)									
84	1	LLABY(1)									
85	1	LLABZ(1)									
86	1	LLABD(1)									
87	1	LLABE(1)									
88	1	LLABF(1)									
89	1	LLABG(1)									
90	1	LLABH(1)									
91	1	LLABI(1)									
92	1	LLABJ(1)									
93	1	LLABK(1)									
94	1	LLABL(1)									
95	1	LLABM(1)									
96	1	LLABN(1)									
97	1	LLABO(1)									
98	1	LLABP(1)									
99	1	LLABQ(1)									
100	1	LLABR(1)									
101	1	LLABS(1)									
102	1	LLABT(1)									
103	1	LLABU(1)									
104	1	LLABV(1)									
105	1	LLABW(1)									
106	1	LLABX(1)									
107	1	LLABY(1)									
108	1	LLABZ(1)									
109	1	LLABD(1)									
110	1	LLABE(1)									
111	1	LLABF(1)									
112	1	LLABG(1)									
113	1	LLABH(1)									
114	1	LLABI(1)									
115	1	LLABJ(1)									
116	1	LLABK(1)									
117	1	LLABL(1)									
118	1	LLABM(1)									
119	1	LLABN(1)									
120	1	LLABO(1)									
121	1	LLABP(1)									
122	1	LLABQ(1)									
123	1	LLABR(1)									
124	1	LLABS(1)									
125	1	LLABT(1)									
126	1	LLABU(1)									
127	1	LLABV(1)									
128	1	LLABW(1)									
129	1	LLABX(1)									
130	1	LLABY(1)									
131	1	LLABZ(1)									
132	1	LLABD(1)									
133	1	LLABE(1)									
134	1	LLABF(1)									
135	1	LLABG(1)									
136	1	LLABH(1)									
137	1	LLABI(1)									
138	1	LLABJ(1)									
139	1	LLABK(1)									
140	1	LLABL(1)									
141	1	LLABM(1)									
142	1	LLABN(1)									
143	1	LLABO(1)									
144	1	LLABP(1)									
145	1	LLABQ(1)									
146	1	LLABR(1)									
147	1	LLABS(1)									
148	1	LLABT(1)									
149	1	LLABU(1)									
150	1	LLABV(1)									
151	1	LLABW(1)									
152	1	LLABX(1)									
153	1	LLABY(1)									
154	1	LLABZ(1)									
155	1	LLABD(1)									
156	1	LLABE(1)									
157	1	LLABF(1)									
158	1	LLABG(1)									
159	1	LLABH(1)									
160	1	LLABI(1)									
161	1	LLABJ(1)									
162	1	LLABK(1)									
163	1	LLABL(1)									
164	1	LLABM(1)									
165	1	LLABN(1)									
166	1	LLABO(1)									
167	1	LLABP(1)									
168	1	LLABQ(1)									
169	1	LLABR(1)									
170	1	LLABS(1)									
171	1	LLABT(1)									
172	1	LLABU(1)									
173	1	LLABV(1)									
174	1	LLABW(1)									
175	1	LLABX(1)									
176	1	LLABY(1)									
177	1	LLABZ(1)									
178	1	LLABD(1)									
179	1	LLABE(1)									
180	1	LLABF(1)									
181	1	LLABG(1)									
182	1	LLABH(1)									
183	1	LLABI(1)									
184	1	LLABJ(1)					</th				

SIMULATION PROJECT NUMBER 10 BY T. G. PAL
DATE 1/22/1978 RUN NUMBER 1 OF
LTSUP=06222/00028000 GASP IV VERSION 1 25JAN75

	B(I,1)	B(I,2)	B(I,3)	B(I,4)	B(I,5)
1	12250.00000	13683.00000	2780.70000	3900.00000	1880.00000
2	0.00000	0.07000	0.02394	0.00000	0.00000
3	0.00000	0.14000	0.07000	0.00000	0.00000
4	3.63271	2.37000	3.84300	2.31000	2.15200
5	670.30000	3711.10000	875.50000	47.04000	2970.90000
6	0.00000	-0.00204	0.00350	0.00000	0.00000
7	3391.40000	3172.70000	717.70000	1976.40000	168.12000
8	170.10000	53.23000	27.72000	31.90000	250.00000
9	0.00000	0.14000	0.07000	0.00000	0.00000
10	0.00000	0.00490	0.00057	0.00000	0.00000
11	0.27500	0.19000	0.42900	0.48700	0.08700
12	0.00000	0.50000	0.50000	0.32000	0.00000
13	1.00000	0.91400	0.54400	0.71000	1.00000
14	846.90000	0.32670	33.14300	3.68200	5000.00000
15	0.00000	0.74840	0.00000	0.00000	0.00000
16	1.00000	123.32000	1.00000	1.00000	1.00000
17	0.00000	93.82400	47.14000	12.32500	0.00000
18	1.57000	0.67000	1.12200	1.28000	1.57000
19	437.50000	2422.10000	571.40000	30.70000	1939.10000
20	0.02400	0.01224	0.01750	0.09060	0.01140
21	0.16370	0.19300	0.15850	0.18840	0.18940
22	7.25000	13.90000	8.29000	1.56600	12.45000
23	0.10880	0.12560	0.10980	0.12560	0.12560
24	32.00000	125.00000	32.00000	125.00000	125.00000
25	2.00000	2.00000	1.00000	1.00000	2.00000
26	5.93000	15.00000	10.84770	3.30000	14.70000

A1= 1.9020 A2= .3270 C1= 4.6980 C2= 1.1520 C3= .5870
 ALL1= .5100 ALL2= .4750
 GRAV= 9.8070 W= 653.2100

I	ABAR(1,I)	ABAR(2,I)	C(I)	ETA(I)	GG(I)
1	106.8000	.3200	574.0000	.0800	.2500
2	177.7000	.1900	980.3000	.0340	.9000
3	106.8000	.3200	126.8000	.0700	1.1000
4	177.7000	.1900	410.7000	.3020	1.5000
5	177.7000	.1900	292.4000	.0780	1.4000

TINIT= 0.000 TPRIM= 1.000 T2PRIM= 1.0200 TEND= 2.0000

BETA1= .03710 BETA2= .52500 TAC= 2.00000 VMEAN= .99500
 XHQP= .85700 XNOT= -.39560

DALPH1= .207 DALPH2= .245

SS(1)= .5711 SS(2)= .6886 SS(3)= -.6984 SS(4)= .2971 DD(3)= -.5.8867 DD(4)= -10.1920

TSCALE = .10463 XSCALE = .09862

TSCALE = .00463 HSCALE = .0113E

VSCALE = .01774

TSCALE = .10463

The image shows a decorative border consisting of a repeating geometric pattern. The pattern is composed of small circles and squares arranged in a grid-like structure. The border is composed of four rows of this pattern, creating a decorative frame around the central text area.

မြတ်စွာလောက်မှုများဖြစ်သည့်အကြောင်းရှိခဲ့သူများ၏အနေဖြင့်

କାନ୍ତିର ପାଦରେ ମହାଶୁଣ୍ଡର ପାଦରେ
କାନ୍ତିର ପାଦରେ ମହାଶୁଣ୍ଡର ପାଦରେ

I	PITHETA(I)	PPI(I)	SS(I+9)
1	.28061000	.34160000	.99960316
2	.28824000	.44564000	.81359324
3	.06253000	.12133000	.99444334
4	.14147000	.36674000	.24691297
5	.00010000	.00601000	.99987160

I	FSE(I)	FSE(I)/FBAR	SS(4+I)
1	1225.012682	.100001	.016489
2	2616.042012	.205006	.01604
3	26.000816	.009350	.049693
4	311.966322	.079991	.016073
5	9.588000	.005100	.005094

APPENDIX F

TYPICAL OUTPUT

T= .025 MODE= 1

(X,Y)= (-.3754, .9805) (-.1212, .5378) (.0112, .0757)

PHI(I)= .1662 .2331 .4398 .5619 .0000

THETA(I)= .1241 .0279 .0663 .2147 .0000

FORCE(I)= 68.9755 98.0435 57.3047 785.2939 22.9360

FSE(I)= 68.9755 98.0435 53.6278 639.4610 22.9360

Z(I)= .5237 .2342 .4800 1.7800

T= .450 MODE= 1

(X,Y)= (-.0122, .9982) (.1954, .5298) (.0990, .0642)

PHI(I)= .2629 .2829 .0241 .0542 0.0000

THETA(I)= .1043 .1653 .0040 .0140 .0000

FORCE(I)= 919.9799 1904.3460 18.1356 102.9200 27.0720

FSE(I)= 919.9799 1904.3460 14.5242 60.9755 27.0720

Z(I)= .4151 .6400 -1.3400 -.9400

T= .975 MODE= 1

(X,Y)= (.4419, .9960) (.3843, .4869) (.1098, .0396)

PHI(I)= .4997 .1833 .0663 .0241 .3518

THETA(I)= .4074 .0936 .0281 0.0000 .2361

FORCE(I)= 2258.1325 266.1192 29.9453 14.6358 121.4480

FSE(I)= 2258.1325 266.1182 15.6414 51.5947 121.4480

Z(I)= -.1070 .5107 .0800 2.1300

T= 1.290 MODE= 4

(X,Y)= (.6391, .9650) (.6677, .4537) (.2577, .2080)

PHI(I)= .2067 .1554 .1707 .2508 .0145

THETA(I)= .1023 .1474 .1345 .0623 .0145

FORCE(I)= 305.0201 903.4012 64.7151 -5.9439 11.2800

FSE(I)= 305.0201 903.4012 47.4829 89.1180 11.2800

Z(I)= .6361 1.1000 1.4300 3.0300