UNITED STATES AIR FORCE
RESEARCH LABORATORY

A Personal Computer-Based
Head-Spine Model

David J. Pancratz
Linda J. Rogers
John B. Bomar, Jr.

Biodynamics Research Corporation
9901 IH 10 west Suite 1000
San Antonio TX 78230

September 1998

Final Report for the period May 1996 to September 1998

19991123 129

Human Effectiveness Directorate
Biodynamics and Protection Division
Biodynamics and Acceleration Branch
2800 Q Street BLDG 824 RM 206
Wright-Patterson AFB OH 45433-7947

DTIC QUALITY INSPECTED 4

NOTICES

When US Government drawings, specifications of other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from the Air Force Research Laboratory. Additional copies may be purchased from:

> National Technical Information Services
> 5285 Port Royal Road
> Springfield, Virginia 22161

Federal Government agencies registered with the Defense Technical Information Center should direct requests for copies of this report to:

> Defense Technical Information Center
> 8725 John J. Kingman Rd STE 0944
> Ft. Belvoir, VA 22060-6218

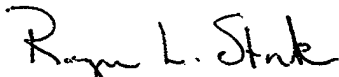## TECHNICAL REVIEW AND APPROVAL

DISCLAIMER

This Technical Report is published as received and has not been edited by the technical editing staff of the Air Force Research Laboratory.

AFRL-HE-WP-TR-1999-0175

This technical report has been reviewed and is approved for publication.

FOR THE DIRECTOR

ROGER L. STORK, Colonel, USAF, BSC
Chief Biodynamics and Protection Division
Human Effectiveness Directorate
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE September 1998	3. REPORT TYPE AND DATES COVERED Final Report May 1996 to September 1998	
4. TITLE AND SUBTITLE A Personal Computer-Based Head-Spine Model			**5. FUNDING NUMBERS** C: F41624-95-C-6010 PR: 3005 TA: 3005CB WU: 3005CB62 PE: 65502F
6. AUTHOR(S) David J. Pancratz, Linda J. Rogers, John B. Bomar, Jr.			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Biodynamics Research Corporation 9901 IH 10 West Suite 1000 San Antonio TX 78230			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory Human Effectiveness Directorate Crew Survivability and Logistics Division Air Force Materiel Command Wright-Patterson AFB OH 45433-7901			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER** AFRL-HE-WP-TR-1999-0175
11. SUPPLEMENTARY NOTES			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (Maximum 200 words) The Head-Spine Model (HSM) originally developed for the Unix environment was recoded for the personal computer. This recode included code improvements and bug fixes, as well as the development of a graphical interface for creating simulations. The HSM can be used to predict the forces and motions of a human spinal column. The vertebrae are modeled with rigid bodies while hydrodynamic elements model the intevertebral discs. A full set of ligamenteous and active muscular models is also incorporated. External accelerations can be applied to the system and the resulting response calculated. A novel method of calculating an initial equilibrium position was incorporated. Several simulations of frontal and vertical accelerations were conducted to demonstrate the utility of the software. Along with the HSM, a database of spinal element properties was created. This database can be separately used from the software to determine mass, material, and geometric properties of the spinal elements as derived from a variety of sources in the literature.

14. SUBJECT TERMS Spine, Cervical, Simulation, Muscle, Multi-body, Rigid-body			15. NUMBER OF PAGES 150
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	10. LIMITATION OF ABSTRACT UNLIMITED

NSN 7540-01-280-5500
Computer Generated 1996

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18
298-102

This page Intentionally Left Blank

ACKNOWLEDGMENTS

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

v

LIST OF FIGURES

FIGURES (continued)

LIST OF TABLES

1.0   Summary.

Biodynamic Research Corporation (BRC) of San Antonio, TX, completed an SBIR Phase II project to port the Air Force's Head-Spine Model (HSM) to a personal computer environment, improve certain features of the software, and add a user-friendly interface. The impetus for this project was the Air Force's desire to have a software tool capable of modeling the internal forces and motions of the human head and spine during impulsive acceleration events, particularly aircraft ejections. Although models exist to predict the gross motion of a human under acceleration loading, such as the Air Force Articulated Total Body (ATB) model, Dynaman, and MADYMO, the Head-Spine Model is the only tool able to provide estimates of internal forces. For this reason, the HSM could be valuable to the Air Force and other scientists for simulating acceleration environments.

In Phase I, BRC took the original archived FORTRAN 77 files for the Unix operating system and ported the software to the PC-DOS environment. Several simulations were conducted to confirm that the output from the original and ported software was the same. Several problems with the HSM software were described in Phase I, such as the necessity to recompile the software for different model configurations, the lack of comments and documentation for the software, the use of small angle approximations and possible miscalculations, and the storage of biomechanical data in the software itself instead of files that can be separately edited.

In Phase II, BRC rewrote the HSM software for the PC-environment and renamed it HSM-PC. The computational portion of the software was written using modern structured Fortran 90 code, while a user-friendly interface for the software was written in Visual Basic. All of the biomechanical data for the model is stored in Microsoft Access format, and can be viewed from the HSM-PC software or from Access directly. The user interface permits an operator to select different HSM models for a simulation; edit certain model, environment, or simulation parameters; and then visually review the results of a simulation through an animation or graphs of desired data. The software stores element, environment, simulation, and other data in separate text files, so that additional simulations with a different environment, different forcing functions, or different body elements can be quickly accomplished. The software runs on Microsoft Windows 95, 98, or NT, and requires a Pentium CPU or equivalent for reasonable operation.

It is BRC's belief that there are still sections of the HSM-PC that must be improved to be a validated biomechanical tool with commercial potential. For example, a significant amount of time was spent attempting to find a set of model parameters that were consistent in units and produced realistic output. Old input files for the HSM software were noted to have different element properties than published technical reports and journal articles. In many cases, the differences were several orders of magnitude. In addition, the original HSM software did not provide for a way to pre-load the elements of the model. BRC has introduced the concept of "settling" into HSM-PC, so that the elements of the model have realistic forces and moments acting on them at the start of the simulation. Finally, the element models can be improved significantly. The muscle model of the original HSM caused the muscles to become fully activated during each

simulation. Although the settling process improves the muscle response, research by BRC indicates that a muscle model employing a position feedback mechanism would be more appropriate. The intervertebral disc model was also noted to be difficult to implement and numerically unstable for large deflections. Finally, placing planes in the environment can be a difficult task. It is important to make sure the plane normal is pointing in the proper direction, and planes cannot start already engaging the model.

Future research in these areas will improve the model.

Figure 1-1 Summary of Results

Porting the HSM to a PC Environment
- Runs on Windows 95/98/NT.   - Features Fortran 90 computational module and Visual Basics user interface.   - Separate database for biomechanical data.
User Interface
- Operator can independently save model elements, environment elements, or simulation parameters.   - Model can be allowed to settle under gravity before simulations are conducted.   - Results can be animated or displayed in charts.   - The output parameters can be specified for each simulation.
Mathematical Model
- Permits control of error tolerance.   - Attempts to catch element errors before model is run.   - Requires no re-compiling for different model simulations   - Uses structured Fortran 90 code.   - Permits simple updating of element models when they become available.   - Computes dynamic motion in 3 dimensions.

2.0 Introduction.

This final report concludes an effort by Biodynamic Research Corporation (BRC) of San Antonio, Texas, to port the Air Force Head-Spine Model (HSM) to a personal computer environment to be a useful desktop tool for scientists. The effort was conducted under Contract F41624-95-C-6010 through the Armstrong Laboratory and is entitled "A Personal-Computer Based Head-Spine Model."

This effort emanates from the Air Force's desire to develop a mathematical model of the internal forces and motions that occur within the human head and spine during impulsive acceleration events and aircraft ejections in particular. Existing tools such as the Air Force Articulated Total Body (ATB) model, Dynaman, and MADYMO are able to simulate a human's gross motion to forces and accelerations, but are either unable or invalidated for computing the internal dynamic and kinematic quantities. These internal forces and stresses can be used as predictors for injury. The HSM-PC would be a useful tool for Air Force and commercial researchers and scientists in a variety of fields.

The specific objectives of the Phase II are listed below:

- Create an HSM geometry and materials properties database, for use by the model and as a stand-alone database.
- Re-code the HSM for the personal computer Windows 95/98/NT environment.
- Establish a simulation validation protocol and attempt to validate portions of the HSM-PC.
- Conduct parametric studies of the model using the HSM-PC.
- Document the operation and area of applicability of the HSM-PC.
- Demonstrate commercial potential of the software.

This final report documents and summarizes the work generally according to the objectives outlined above. The layout of the report is summarized in Figure 2-1.

Figure 2-1 The HSM Final Report

A PC-Based Head-Spine Model

Section 1.0 – Summary
• Successfully ported HSM to PC-environment. • Re-wrote HSM to have versatile computational module and user-friendly interface.

Section 2.0 - Introduction
• Define 6 objectives for Phase II.

Section 3.0 - Technical Background and Literature Review
• HSM created and improved during the 70's and 80's. • Program written in Fortran IV for a Unix mainframe.

Section 4.0 – The Head-Spine Model
• Mathematical models for vertebra, ligaments, muscles, discs, ribs, and articular facets. • Environment elements include planes and springs. • Different HSM models.

Section 5.0 – HSM-PC Software
• User-friendly graphical interface. • Computational model. • Biomechanics database.

Section 6.0 – HSM-PC Simulations, Verifications and Validations
• Typical HSM-PC simulations. • Verification of model equations. • Validation plan and results. • Suggestions for conducting simulations

Section 7.0 – Recommendations for the HSM-PC
• Commercialization potential. • Future improvements.

4

3.0   Technical Background and Literature Review.

The Air Force Head-Spine Model (HSM) was initially developed for the purpose of modeling pilot ejections from aircraft. In 1973-5, a three-dimensional model of the human spine and head was created at the University of Illinois, sponsored by the Air Force Aerospace Medical Research Laboratory.[1] The model consists of rigid bodies, which represent skeletal segments, and deformable elements, which represent ligaments, cartilaginous joints, viscera, and connective tissues. The model can be separated into sub-models to study the response of different body segments to accelerations and loads. The original HSM is theoretically sufficiently general to be applicable to other acceleration environments. A technique for modeling other aspects of the environment, such as seat geometry and restraint harness, are also considered in the HSM. Output of the model includes the forces and moments acting on the elements and the kinematics of the model elements.

The HSM was refined in 1976-7 during a second Air Force project.[2] During this project, four different versions of the model were created ranging from 32 to 252 degrees of freedom. The researchers also attempted to refine and validate the model using two methods -- by comparing the model frequency response to the experimentally determined frequency response of humans to vertical excitation and by creating head-spine models for other primates and comparing the model output to experimental data. The work with the primate model also provided a methodology for scaling dynamic response and injury data between the primates and humans. Finally, spinal injury criteria were developed that use the computed stress in vertebral bodies to predict the likelihood of vertebral body failure.

A database of head and cervical spine geometry and stiffness data was collected in a third project from 1978-80.[3] The geometry of the vertebrae and points of attachment of muscles, discs, and ligaments were obtained. Stiffness data were obtained or estimated from the literature. Inertia values were developed by approximating the geometry of the vertebrae and applying a uniform density. Simulations with the model were in relatively good agreement with experimental data for the first 150 milliseconds of response. After this time, the model output showed significant discrepancies that the authors attributed to poor modeling of the major muscles. Although this project focused on the head and cervical spine, it used portions of code from the entire HSM model.

One final revision to the Head-Spine Model was an effort from 1980-4.[4] During this project, a model of the diaphragm was incorporated into the HSM that better represented the vertical load path through the viscera, abdomen, and rib-cage region. Additional work was done to create a proposed injury criterion for the cervical spine. Simulations with the head and cervical spine model in the fore-aft and side-to-side directions showed a good agreement to experimental data, which led the authors to believe the model was a viable tool for prediction of head-cervical spine kinematics in three dimensions.

Two other comprehensive biomechanical model of the spine, head and upper torso have been reported in the literature.[5,6] The first model, a Finite Element Model (FEM), did not

5

include the head, however it did include descriptions of all the relevant tissues below the head. The paper reported static stress analysis of the model in the normal upright posture, however no dynamic analysis was mentioned or reported.[5] Generally speaking, FEM modeling capabilities are available only to specialists in structural analysis and not widely available to the entire biomechanics community. The second model[6] was a completely lumped model of the loading of the spine in the upright posture with the whole system modeled as a relatively simple mass-spring-damper system. Models with a high degree of lumping are useful for investigating a single posture, but are not flexible and are not capable of fully representing the effects of a dynamically changing posture.

There were several subsegment models reported.[7,8,9, 10,11, 12, 13] These models fall into two categories: (1) models of the human head and cervical spine[7-9]; and (2) models of subsegments of the vertebral column[10-13].

The first head-cervical spine model described in 1978 by Huston, et al[7] was a lumped parameter head-cervical spine model. That model included viscoelastic models for the soft tissues of the cervical spine (C-spine), while the head and bony structures of the C-spine and head are modeled as rigid bodies. There is no provision for active muscular reflex contraction in the model. Huston, et. al.[7] presented comparisons with that of a seated cadaver struck in the head by a mass suspended at the end of a pendulum. They concluded that the model's response compared favorably with that of the cadaver.[7] No independent evaluations of the Huston model were found in the literature by the authors.

In 1984, Merrill, et. al.[8] described a three-dimensional model of the head and C-spine, which was an extension of a previously developed two-dimensional saggital plane model. Although the authors claimed that their model was "three-dimensional" only flexion extension motion of the head and neck were modeled. No rotation was permitted at the intervertebral joints in the model. The viscoelastic properties of the intervertebral joints were modeled as lumped viscoelastic mechanical elements. Comparisons of the response of the model to impulsive acceleration with motions recorded in humans showed that the model's response did not agree well with the responses of human volunteers subjected to similar acceleration.[8]

The third C-spine model was described by de Jager, et. al.[9] in 1996. The de Jager model was built within the MADYMO[14] multi-body mechanical modeling environment. That model includes separate viscoelastic models for the soft tissues of the C-spine, as well as a contractile muscle model. The head and bony structures of the de Jager model are represented by rigid bodies. De Jager, et. al.[9] reported the model was too flexible when compared with "normal" human response to impulsive acceleration. However, they noted that when the soft tissue model parameters could be "scaled" to match the response of human volunteers, fairly good agreement was obtained between the model and the human subjects even though no comparisons were presented.[9]

Two of the subsegment models were more concerned with motion control in the spine than modeling the motion *per se*.[9,12] The others were FEM models of subsegments in

which detailed anatomical models of the vertebrae and intervertebral discs were modeled and subjected to a stress-strain analysis in FEM computer analyses.[10,11,13]

4.0   The Head-Spine Model.

The Head-Spine Model refers generally to a group of mathematical models that simulate the response of the head, neck, spine, and pelvis to accelerations and forces. There are actually several different versions of the model, each of which includes different anatomical elements. Although some of the models are subsections of one another, they can contain different elements or different properties for the elements. The different versions of the model are described in Section 4.4.

The elements of the HSM are separated into two types: model elements and environment elements. Model elements represent specific portions of the human anatomy and model the internal forces of the body. Environment elements are used to apply external forces to the HSM.

The model elements of the HSM consist of five types: rigid body elements, spring elements, muscle elements, hydrodynamic elements, and disc (beam) elements. These elements are described in detail in Section 4.2.

The environment elements of the HSM are described in Section 4.3 and consist of three types: spring elements, planes, and constraints.

To understand the HSM model and its components, the coordinate frames employed in the model are described in Section 4.1.

Figure 4-1 illustrates graphically how this section is organized.

Figure 4-1   Organization of Section Four

• Section 4.1	HSM Coordinate Frames and Units
• Section 4.2	HSM Model Elements
– 4.2.1	Rigid Body elements
– 4.2.2	Spring Elements
– 4.2.3	Muscle Elements
– 4.2.4	Hydrodynamic Elements
– 4.2.5	Beam Elements
• Section 4.3	HSM Environment Elements
– 4.3.1	Plane Elements
• Section 4.4	Different Versions of the HSM Model

4.1   HSM Coordinate Frames and Units.

Dimensions and motion of the Head-Spine Model are referenced to two main types of coordinate frames. There is a single "global" coordinate frame that is fixed to the earth and is used for positioning the HSM and associated environment elements. This coordinate frame uses the same directions for axes as most aerospace and automotive references. "Body" coordinate frames are attached to rigid body elements (elements with mass) of the model and are used to simplify the representation of inertial properties and the application of Newton's Laws to each body during a simulation. As in the original version of the HSM, the body coordinates frames of the HSM-PC are aligned so that the z-axis points to the center of the earth surface and the x-axis and y-axis forms a plane that is parallel to the earth surface when the model is in the upright position. The origin of each body coordinate system is at the center of gravity for that rigid body element. Figure 4-2 shows a representation of the HSM with global coordinate frame (labeled with the G subscript) and a typical body coordinate frame (labeled with the B subscript).

Other coordinate frames are used within the HSM-PC to compute forces and motions of individual elements. For example, a coordinate frame is assigned to each beam element to compute the forces and moments of an intervertebral disc that result from motion of adjacent vertebra. This coordinate frame is necessary to measure the three-dimensional change in orientation of the disc over time. The z-axis of the beam coordinate frame stretches from the inferior endplate of one vertebra to the superior endplate of the adjacent vertebra. The y-axis is found by averaging the x-axes of the body coordinate systems of the two vertebra and taking the vector cross-product of this axis with the z-axis. The x-axis can then be found with a second cross-product.

The graphical user interface for the HSM-PC software has its own screen display coordinate system. Motions of the model in global coordinates have to be expressed in the screen coordinates before they can be animated.

Calculations in the HSM are performed in the SI system and are similar to the units used in the original HSM. The HSM-PC interface permits English units as an option for output. The two significant changes in the HSM-PC from the original HSM are:

- The g-cm-dyne-sec system of the original HSM was replaced with kg-m-N-sec in the HSM-PC.
- The biomechanical database was converted to kg-N-sec.

9

Figure 4-2 Global and Body Coordinate Systems

+ Denotes locations of vertebra center of mass
• P Denotes Primary Node
• S Denotes Secondary Node
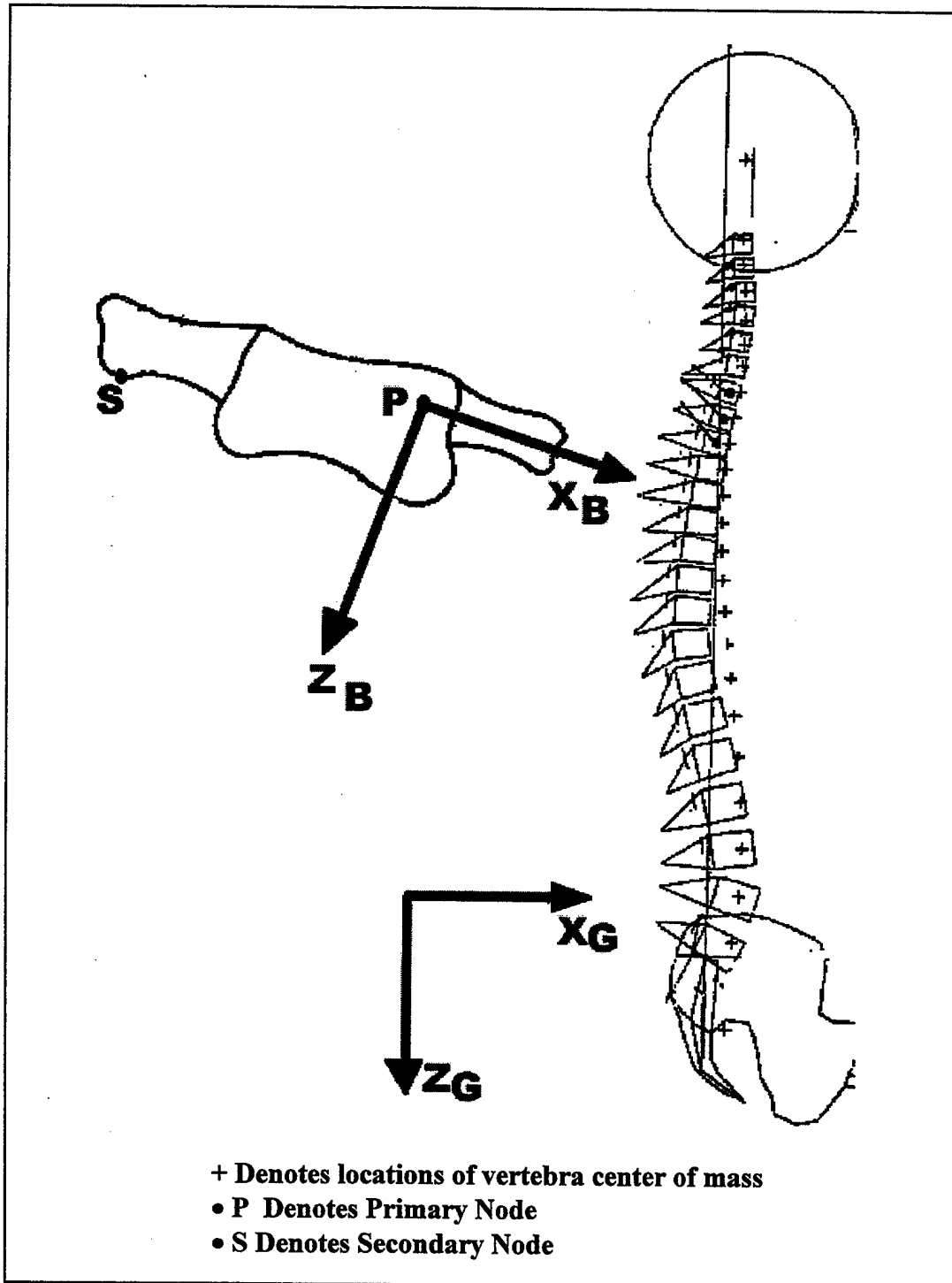
4.2   HSM Model Elements.

The anatomical model of the HSM-PC consists of five types of elements. A summary of the different model elements is displayed in Figure 4-3.

Rigid body elements model the vertebra, pelvis, ribs, and viscera of the HSM-PC. They are the only elements assigned mass properties and are capable of motion in six dimensions (translation and rotation). Rigid bodies have nodes that serve as attachment points for other elements, called secondary nodes.

Spring elements are used to represent ligaments, some articular facets, and some abdominal viscera elements. They are one-dimensional elements that produce a cubic force from deflection and a damping force from rate of deflection. All of the ligaments are represented by tension-only springs. All of the abdominal viscera are represented by dual acting springs. It is possible to also add compression-only springs to the model.

Muscle elements are appropriately used to model the muscles of the cervical spine. Muscles produce a force proportional to strain and the concentration of an activator molecule that governs their behavior. The relatively simple model of the original HSM was employed, although more advanced models could be added in the future. There are no muscles from the thoracic or lumbar spine in the HSM.

Fluid-filled hydrodynamic elements are used to model the cervical articular facets and the forces they produce between adjacent vertebra. The change in volume of the element and a bulk modulus parameter determines the internal pressure and the subsequent force applied to adjacent vertebra.

Finally, beam elements are used to model intervertebral discs and the connections from the ribs to the spine and to one another. The beam elements used as discs can produce axial and shear forces as well as torsional and bending moments between vertebra. The beam elements of the ribs do not produce shear forces.

11

Figure 4-3 Head-Spine Model Elements

Model Elements

Rigid Body Elements
(Vertebra, pelvis, head, ribs, abdominal elements)
- have primary and secondary nodes
- have six degrees of freedom
- have mass properties

Muscle Elements
- have no mass properties
- produce tension only
- connect to rigid body nodes
- can be active or passive
- are stimulated by strain

Spring Elements
(Ligaments, articular facets, abdominal viscera, rib connections)
- have no mass properties
- can be damped
- have a cubic stiffness
- can be tension-only, compression-only, or both
- connect to rigid body nodes

Hydrodynamic Elements
(Cervical articular facets)
- have no mass properties
- represented as a fluid-filled pentahedron
- produce three-dimensional forces and moments
- connect to rigid body nodes

Beam Elements
(Intervertebral discs and rib connections)
- have no mass properties
- produce axial and shear forces, and bending and torsional moments

4.2.1 Rigid Body Elements.

Rigid body elements in the HSM-PC software represent the vertebra, pelvis, ribs, head, and some visceral elements. The dimensions of most these elements were obtained from the HSM software and are considered anatomically accurate. In the case of the vertebra, however, the mass properties of the rigid body elements represents the properties of a horizontal slice of the body, rather than just the mass and inertia of the bone itself. The underlying assumption behind this approach is that horizontal slices of the body move together and can be associated with a corresponding vertebra.

Rigid body elements are the only elements in the Head-Spine Model that have mass properties. Thus, to simulate the response of the HSM to a force or acceleration input requires Newton's Laws to be applied only to the rigid body elements. All other elements in the model produce forces and moments that act directly on a rigid body.

Each rigid body element has a primary node, representing the center of gravity for that body, and multiple secondary nodes. Since the rigid body elements represent horizontal slices of the torso, it is possible for the center of gravity of a body to lie outside of the physical dimensions of the vertebra. The principal moments of inertia for each body are assumed to be aligned with its body coordinate frame.

Each rigid body element also has multiple secondary nodes that represent attachment points for other elements. The secondary nodes can lie outside the body, just as the primary node. Figure 4-2 shows an example of the location of the primary node and a secondary node on a rigid body element. The secondary nodes on a rigid body are fixed in space relative to the primary node -- they represent a rigid link.

The rigid body elements in the HSM-PC reflect the dimensions and mass properties from the original HSM database. A medical illustrator created anatomical representations of the vertebra and ribs for use in the graphical interface.

The equations of motion for a rigid body element are computed in a combination of body and global coordinates. Body coordinates are used to compute the acceleration and angular acceleration of the element due to forces from gravity and other elements. The body coordinates are used for this calculation to avoid the numerical difficulty of working with products of inertia for the body. The position and angular orientation of the body is then integrated from velocities in global coordinates. Global coordinates are used for these kinematic calculations because the animation of the HSM is only meaningful in a fixed coordinate system. Figure 4-4 describes the process for computing and integrating the equations of motion for each rigid body element.

BRC chose to use a set of four quaternion parameters to express the orientation of each rigid body internally. The quaternions are converted to conventional yaw-pitch-roll Euler angles for the HSM-PC output. The use of quaternions provides two benefits: there are no singularities in the expression of any orientation, and there is a redundant parameter

13

that can be used as a Lagrange multiplier to speed the solution of the equations of motion.[15]

Figure 4-4 Solution for the Equations of Motion of Rigid Body Elements

(1) $\sum F - ma - m\omega \times v = 0$

The force vector equation permits the computation of acceleration in body coordinates, which can be integrated to obtain velocity in body coordinates. The total force on each body is calculated by computing the force in each element, rotating to the proper coordinate frame, and adding to the existing sum.

(2) $\sum M - I\alpha - \omega \times I\omega = 0$

The moment vector equation permits computation of angular acceleration in body coordinates, which can be integrated to obtain angular velocity in body coordinates. The total moment acting on each body is calculated by computing the forces and moments in each element that attaches to a body, rotating to the proper coordinate frame, and then summing.

(3) $v_G = R_{BG} v_B$

An algebraic expression permits the rotation of body velocity into global coordinates, which can then be integrated to obtain position in global coordinates.

(4) $\omega_G = R_{BG}\omega_B$

An algebraic expression permits the rotation of body angular velocity into global coordinates.

$$\dot{e}_0 = -.5(e_1 p + e_2 q + e_3 r)$$
$$\dot{e}_1 = +.5(e_0 p + e_2 r - e_3 q)$$
(5) $\dot{e}_2 = +.5(e_0 q + e_3 p - e_1 r)$
$$\dot{e}_3 = +.5(e_0 r + e_1 q - e_2 p)$$
$$\lambda = 1 - (e_0^2 + e_1^2 + e_2^2 + e_3^2)$$

Algebraic equations permit the computation of four quaternion parameter rates from the angular velocity in global coordinates, which can then be integrated to obtain quaternion parameters.

4.2.2 Spring Elements.

Springs elements represent ligaments and visceral elements in the HSM, as well as restraints and cushions in the environment. There are three different types of spring elements: tension-only, compression-only, and dual-acting. All of the ligaments in the Head-Spine Model are tension only. All of the visceral elements are dual acting springs. The articular facets of the thoracic and lumbar vertebra are also represented as dual acting springs. Compression springs are most likely to be used as an environment element -- for example, to represent a cushion.

All of the spring elements can be assigned a proportional stiffness term and cubic stiffness term. The original HSM software did not add damping to the spring elements. Damping was added to the springs in the HSM-PC because even a small damping factor helped speed model computations and prevent numerical instability -- high frequency oscillations were damped in the early part of a simulation.

The force in a spring element, shown in Figure 4-5, can be computed from the following equation:

$$\mathbf{F} = \mathbf{k1} \cdot \delta + \mathbf{k2} \cdot \delta^3 + 2 \cdot \sqrt{\frac{k1}{m}} \cdot \zeta \cdot \dot{\delta}$$

and where k1 is a proportional stiffness term, k2 is a cubic stiffness term, δ is the deflection from the unstretched state, is $\dot{\delta}$ the rate of deflection, and ζ is the damping ratio.

One of the major difficulties BRC had in developing the spring element was the lack of unstretched spring element lengths – the biomechanical data from the original HSM software did not contain unstretched lengths for ligaments. Of course, in order to compute the tension in a ligament it is necessary to know its original length. Thus, in the HSM-PC, it was assumed that the initial endpoints of the ligament defined their unstretched length.

Springs can be connected to either rigid body elements or plane elements in the model. Connections to planes are useful for simulating restraints.

15

Figure 4-5 The Damped Cubic Spring Model
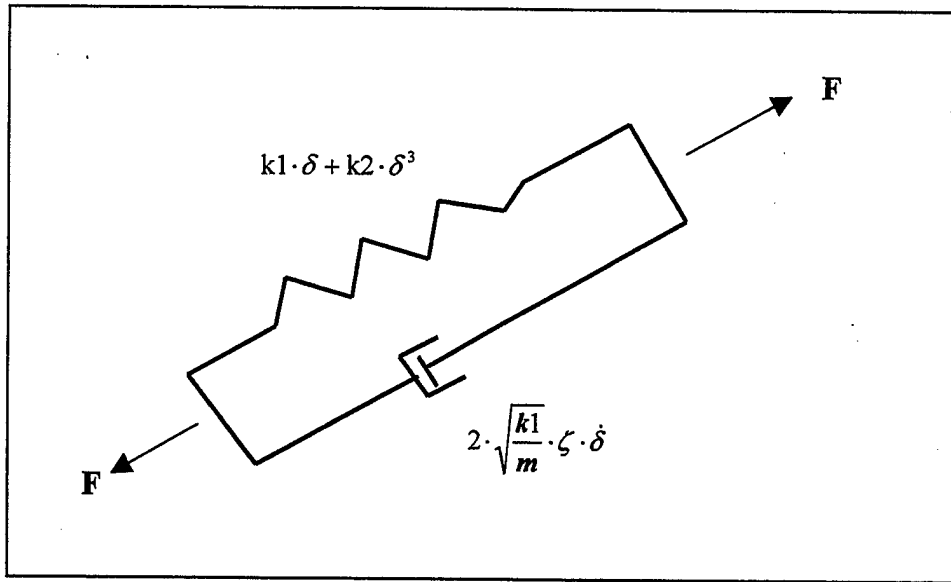
4.2.3 Muscle Model.

To model the effects of reflex muscle contraction in HSM-PC, we adopted the muscle model employed in the original HSM development[1], although it is implemented in a different manner. That model, originally proposed by Apter and Graessley[16], appears to have the requisite characteristics without undue complexity. As illustrated in the schematic below, the model comprises a Maxwell element in parallel with a spring. All elements have variable parameters which depend on the concentration of an "activator molecule" released within the myofilaments in response to stretch or stimulation.

Figure 4-6 Muscle Model Schematic
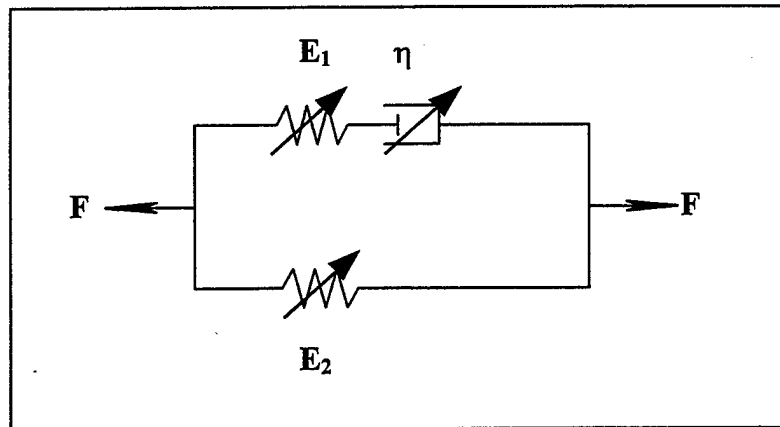
The muscle stress-strain relationships are described by two differential equations whose parameters are variables. The first equation describes the relationship between stress and strain and their rates of change:

$$\sigma + \frac{\eta}{E_2}\dot{\sigma} = E_1\varepsilon + \left(E_1 + E_2\right)\frac{\eta}{E_2}\dot{\varepsilon}$$

Where, $\sigma = \dfrac{F}{A}$ is stress with units of $F \cdot L^{-2}$, ε is strain, which is unitless and is defined by,

$$\varepsilon = \frac{\left(l - l_0\right)}{l_0}.$$

Where, l_0 is the "no tension" length [L] of the muscle, E_1 and E_2 are elastic moduli [$F \cdot L^{-2}$], and η is viscosity [$F \cdot T \cdot L^{-4}$].

The second equation describes the relationships between the concentration of "activator molecule" [$M \cdot L^{-3}$], c, and its rate of change, the applied stimulus S(t) [$M \cdot L^{-3} \cdot T^{-1}$], and strain, ε.

$$\dot{c} = k_2 \varepsilon - k_3 c + S(t)$$

Where, k_3 is a rate constant, and k_2 relates the increase in strain to the rate of increase in activator molecule. The stress-strain model parameters vary as simple functions of the concentration of activator molecule. The relationships are:

$$l_0 = l_0^\infty + \frac{l_0^0 - l_0^\infty}{1 + k_1 c}$$

$$E_1 = E_1^\infty - \frac{E_1^\infty - E_1^0}{1 + k_4 c}$$

$$E_2 = E_2^\infty - \frac{E_2^\infty - E_2^0}{1 + k_5 c}$$

$$\eta = \eta^\infty - \frac{\eta^\infty - \eta^0}{1 + k_6 c}$$

Where, the superscripts "∞" and "0" refer to the completely contracted ($c = \infty$) and completely relaxed ($c = 0$) states, respectively.

Many muscles in the spine actually follow curved paths across multiple vertebra and are attached at multiple locations along their length. The intermediary connections are called sliding nodes. The HSM-PC uses the same method as the original HSM for modeling muscles that have multiple connections. Separate lines of action are computed for each segment of the muscle, and forces are applied to each vertebra that is contacted by a muscle segment.

The HSM-PC muscle model was tested independently to verify its function. Two versions were programmed, the first simulated skeletal muscle response to stimulation during various degrees of stretch. (See Figure 4-7.)

Figure 4-7 Muscle Response to Stretch and Stimulation
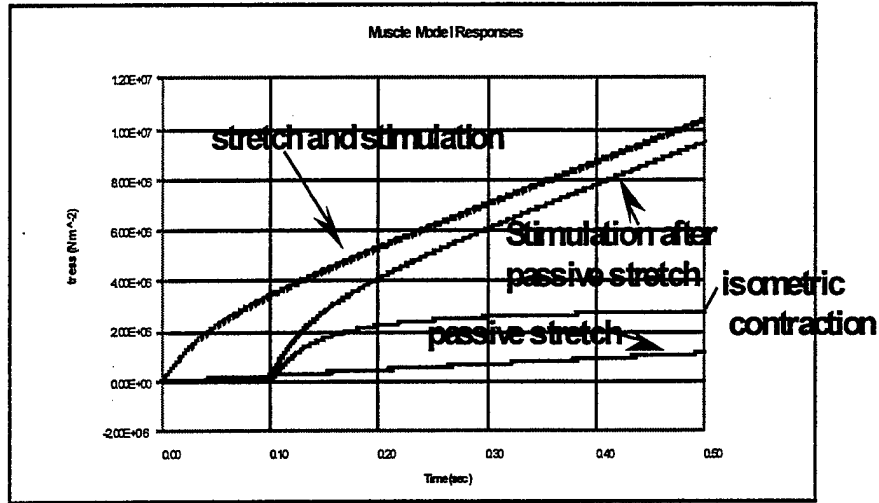
In the first case, the responses match the original model as tested and illustrated in Williams and Belyscho[17] Figure 4-8. The second simulation agrees with original work of Wilke[18] in form but does not match quantitatively. We are satisfied that the muscle model is behaving properly and the differences are simply the result of scaling differences between Wilke's[7] and Apter's[5] models.

Figure 4-8 Stress-time Relation for Various Combinations of Stimulation and Stretching Muscle

∇ Stretching of passive muscles to twice its original length
   Isometric contraction
$\square$ Isometric contraction until t = 0.10 sec, then muscle is stretched while contraction continues
$\blacktriangledown$ Muscle is stretched while being stimulated to contract
--- Muscle is stretched, at first passively until t = 0.0 sec, then muscle is stimulated to contract
   while being
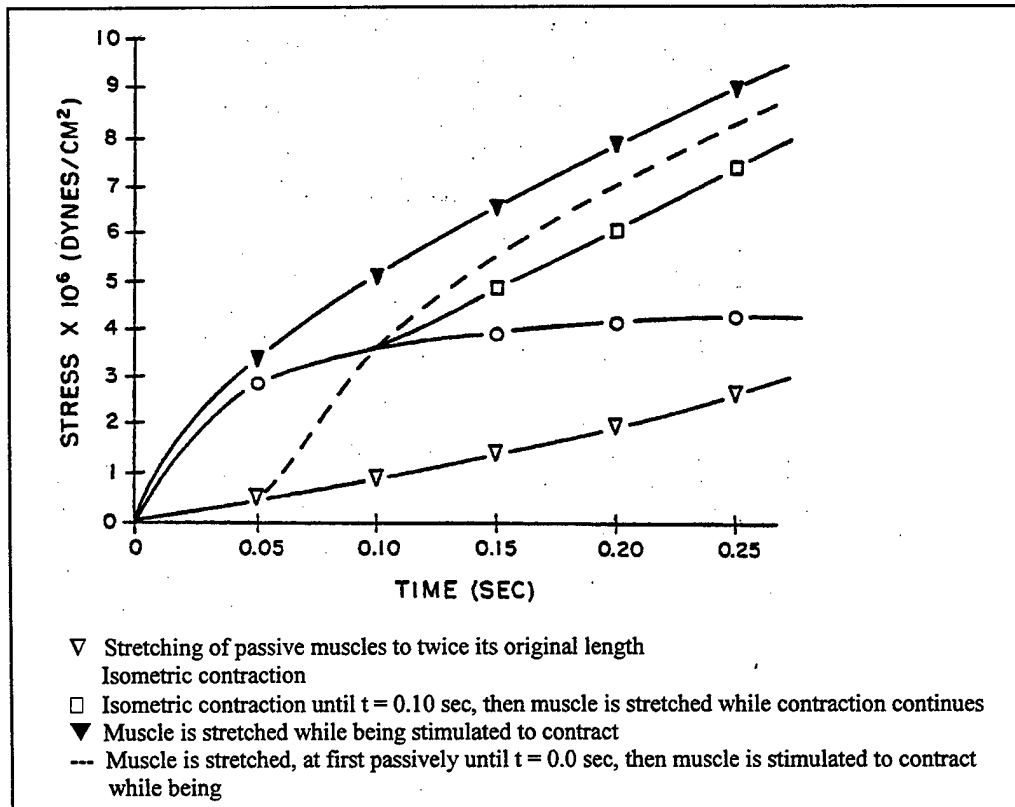
The second, simulated a series of isotonic contractions in which a mass was suspended from the muscle and its contraction velocity is measured in response to a single superthreshold stimulus. (See Figure 4-9.) Two different cross sectional areas were employed for the simulation, 0.33 cm^2 and 1.0 cm^2.

Figure 4-9 Force-Velocity Response of Muscle Model

HSM Muscle Model
Peak Contraction Velocity
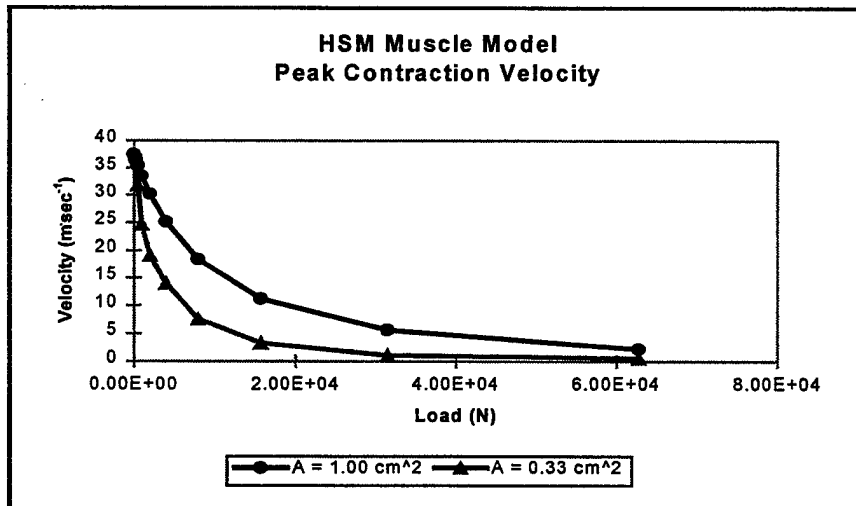
In the biomechanical property database, each muscle has specified origin and insertion location points (coordinates in the global frame) that determine the length of the muscle at any time. The total force exerted by the muscle at its endpoints is determined by solving the stress equation and multiplying the instantaneous stress times the effective cross sectional area of the muscle, which is also specified in the database. Thus, as the model is solved, strain and strain rate are known from the position of the rigid body elements and their instantaneous velocities. This information enables the solution of the muscle model for both "activation" state and stress.

This muscle model was the subject of concern at our first Confidence Assessment Review. The Confidence Assessment Team concluded that the "open loop" nature of the model was causing stability problems in the HSM. For example, unless the initial length of the muscle is set at exactly its unstretched length, it begins contracting immediately after the simulation begins causing gyrations in the model. This has been overcome by allowing the model to settle and capturing the "resting" positions of the rigid bodies as the model initial condition. Nevertheless, the current model would be improved by modifying it to be "closed loop" so that any starting length can be set as the "control" length for a muscle and only deviations from the control length would cause changes in the strength of muscle contraction. This technique is employed in postural muscle models and is suggested as a future improvement for the HSM-PC.

There are three significant differences between the implementation of the muscle model in the HSM-PC and the original HSM. First, in order to implement the model, it is necessary to know the initial unstretched lengths of the muscles (similar to the problem

20

with the ligaments described earlier). These unstretched lengths were not described in the original HSM documentation, and BRC was forced to assume that the muscles are in the unstretched position as they appear in the different HSM models.

A second difference is the effort to allow the HSM to settle under gravity before running simulations. When the HSM elements are first read into the software, all of the ligaments, muscles, and other elements have no strain, creating no forces on the vertebra. Since this is an unrealistic condition, it is desirable to allow the HSM elements to settle. In the case of the muscles, the process of settling is less straightforward than for other elements. Many muscles are stretched and put under tension during the settling phase. This activates the stretch reflex of the muscle so that activator molecule concentration begins increasing and the muscle starts contracting. As some muscles activate, they cause the spine to begin moving in a different direction such that other muscles activate, and so on, until virtually all the muscles of the cervical spine are activated and contracting. This condition is obviously not commensurate with a "settled" state for the model, so BRC implemented the settling phase in the following way. Activator molecule concentration was held at zero during settling, creating in effect a cadaver muscle. At the conclusion of the settling phase, a stimulus S(t) is computed that balances the activator molecule equation so that the muscle is in equilibrium and providing a constant tension. Then, when a simulation is conducted, the activator molecule is allowed to respond in the manner described above and the stretch reflex takes place as intended. While this method that was employed can certainly be improved in the future, the authors believe it represented a better approach to obtaining accurate simulations than starting with all elements of the model at zero force.

The final difference between the implementation of the muscle model in HSM-PC and the original HSM model is that the muscle model differential equations are solved by a sophisticated differential equation solver called DDASPG rather than by using difference equations as in the original implementation. This improvement improves the stability and accuracy of the solution.

4.2.4 Hydrodynamic Elements.

The HSM uses hydrodynamic elements to model the forces between articular facets in the cervical spine. There are two matching hydrodynamic elements acting on each vertebra in the HSM, matching the interacting articular facets of each set of vertebra.

A hydrodynamic element is a fluid-filled pentahedron element, with two triangular endplates that lie on adjacent vertebra. Three nodes, each as shown in Figure 4-10 define the triangular endplates. As the vertebra move relative to one another, the hydrodynamic element changes volume. By assigning a bulk modulus to the fluid in the element, the pressure within the hydrodynamic elements changes as the volume changes. A force proportional to the area of the triangular endplate is then produced on each of the adjacent vertebra. There is also a viscous damping term to reflect the highly damped nature of this interaction.

The pressure in the hydrodynamic element is a function of the volume and volume rate of change, according to the following relationship:

$$p = \mathbf{B}\Delta + \alpha \, \mathbf{B}\dot{\Delta}.$$

where, B is bulk modulus and Δ is defined by

$$\Delta = \frac{\mathbf{V} - \mathbf{V_0}}{\mathbf{V_0}}$$

and $\mathbf{V_0}$, is the original volume and $\mathbf{V}$ is the instantaneous volume.

The volume of the pentahedron is computed by separating it into four tetrahedrons and then computing the volume of each tetrahedron: $\mathbf{V} = \mathbf{V_{IJKL}} + \mathbf{V_{JLMN}} + \mathbf{V_{JKLN}}$, where $\mathbf{V_{pqrs}}$, is the volume of a tetrahedron defined by the global coordinates of nodes p, q, r, and s.

The force on each endplate is computed from the product of the internal element pressure and surface area of the endplate. The direction of this force is parallel to a line drawn from the centroid of the lower endplate to the centroid of the upper endplate. The force is then divided by three and applied evenly to the three nodes on each endplate.

The computation of tetrahedron volume in the original HSM was noted to be unintuitive, leaving BRC unsure about whether the forces in the element were being properly computed. To insure accuracy, BRC implemented the algorithm described below.

Consider a tetrahedron with the four vertices labeled P0, P1, P2, and P3. Starting at P0, the following three vectors can be created:

$$\mathbf{a} = P0\text{-}P1, \; \mathbf{b} = P0\text{-}P2, \text{ and } \mathbf{c} = P0\text{-}P3.$$

The volume of the tetrahedron can then be shown to equal:

$$V = \frac{1}{6} \det \begin{vmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{vmatrix}$$

The rate of change of volume equals:

$$\dot{V} = \frac{1}{6} \left(\det \begin{vmatrix} \dot{a}_x & \dot{a}_y & \dot{a}_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \end{vmatrix} + \det \begin{vmatrix} a_x & a_y & a_z \\ \dot{b}_x & \dot{b}_y & \dot{b}_z \\ c_x & c_y & c_z \end{vmatrix} + \det \begin{vmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ \dot{c}_x & \dot{c}_y & \dot{c}_z \end{vmatrix} \right)$$

Once the volume of the four tetrahedrons is known, they can be summed for the total volume of the hydrodynamic element and the internal pressure can be computed.

Figure 4-10 HSM Hydrodynamic Element

23
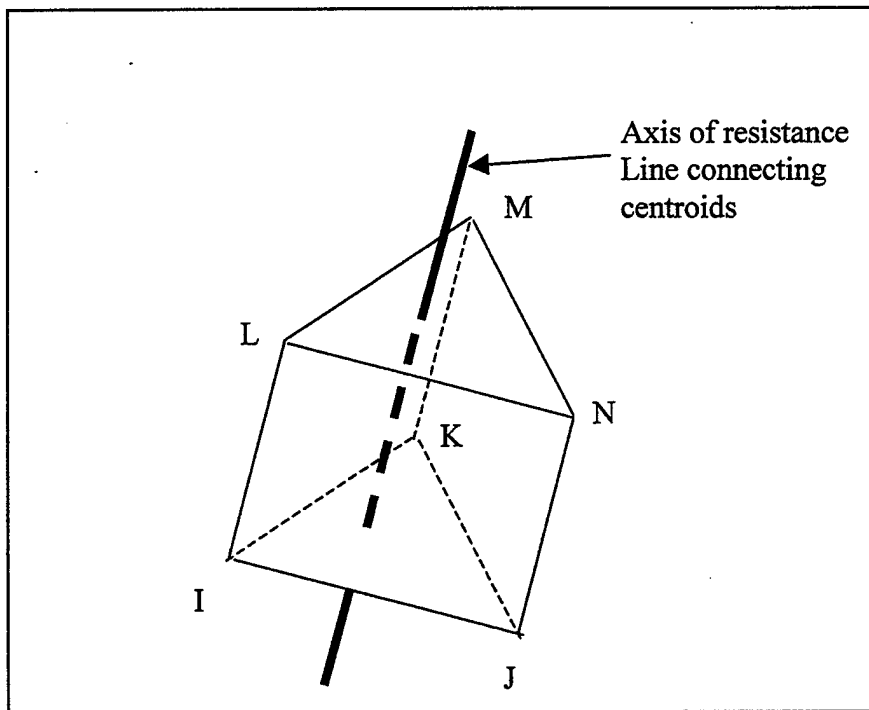
4.2.5 Beam Elements.

Intervertebral discs and some of the connections of the rib cage in the Head-Spine Model are modeled with beam elements (also referred to as disc elements in many locations). There is one beam element between each set of adjacent vertebra, except at the level of the first and second cervical vertebra. Although there is not an intervertebral disc at this location, the original HSM software used a pair of matching hypothetical beam elements to model the interactions between these two vertebra. HSM-PC uses the same approach and includes this set of beam elements.

Beam elements are capable of providing axial and shear forces as well as bending and torsional moments when displaced from their original position as shown in Figure 4-11. The beam elements in the rib cage do not have a shear force term. Each beam element has two endplates located on adjacent vertebra or ribs. The center of the endplate is attached to a secondary node of each vertebra or rib. The initial relative position and orientation between the two endplates is recorded before a simulation. Then, as the endplates move relative to one another during an HSM simulation, forces and moments are developed in the beam element that are transmitted to both endplates.

Because the beam elements can produce forces and moments in six dimensions, they were one of the most difficult elements to incorporate into the model. The beam elements of the original HSM software were implemented in a manner that resulted in instability in certain situations. An imaginary point was introduced to track the changing orientation of the beam endplates relative to one another. BRC modified the original implementation to be more robust in HSM-PC simulations. The new implementation does not require the imaginary tracking point -- orientation angles are computed for both endplates and the change in relative orientation is computed from these values.

To track the instantaneous position and orientation of a beam element, a new coordinate system for each beam is introduced, shown in Figure 4-12. The center point of the line connecting the two endpoints is the origin for the disc coordinate system. The z-axis of the beam coordinate system points along the same connecting line, in the direction from endpoint I to endpoint J. The x-axis for the disc coordinate system is defined relative to the average direction of the x-axes of the body I and body J coordinate frames. Figure 4-10 shows how the disc coordinate frame is computed.

Once the disc coordinate frame is defined, the linear and rotational strain can be computed throughout a simulation. The linear strain is computed similar to the original HSM formulation. Rotational strain is computed by comparing the original relative orientation of the body I and body J coordinate frames to the current relative orientation of these coordinate frames. BRC has added a shear damping term to the disc model to improve the response of the disc in long simulations. There was no shear damping in the original model, however the shear damping term does not have a large effect on the motion of the model and helps alleviate some of the computational difficulties created by the discs.

To compute the forces from the beam element acting on the attaching rigid bodies, first define a vector from endplate I to endplate J and compute the change in that vector from its original state:

$$\delta_{IJ} = r_{IJ} - r_{IJ}^0$$

where r_{IJ} is the vector from I to J and the superscript 0 denotes a value at time zero. All values are computed in the Body J coordinate frame for convenience.

Then the strain angles along the X and Y axes is computed from the relative position of the body endplates:

$$\gamma_X = \delta_{IJx} / r_{IJz}$$
$$\gamma_Y = \delta_{IJy} / r_{IJz}$$

The rate of change in the shear angles can be computed by differentiating these values.

The forces generated by the disc in coordinate frame J are then:

$$F_X = 12\gamma_X k_{bx} / (l^0 \phi_x) + 2\zeta\omega\dot{\gamma}_x$$
$$F_Y = 12\gamma_Y k_{by} / (l^0 \phi_y) + 2\zeta\omega\dot{\gamma}_y$$
$$F_Z = k_{az}\delta_{IJz} + 2\zeta\omega\dot{\delta}_{IJz}$$

where γ is the shear angle, ζ is the damping coefficient, ω is the natural frequency for the motion found by averaging the masses of both bodies, and ϕ is the shear stiffness term for the disc. The parameters k_{bx}, k_{by}, and k_{az} represent bending and axial stiffnesses respectively.

To find the moment generated by the disc on both bodies, it is first necessary to define a single axis rotation that represents the change in orientation of Body I with respect to Body J using an algorithm such as found in Craig.[19] Then, the change in angular orientation and the relative angular velocity are:

$$\Phi = \varphi_{IJ} - \varphi_{IJ}^0$$
$$\dot{\Phi} = \omega_I - \omega_J$$

where Φ is the change in relative angular orientation, ω is the relative angular velocity, and ψ is the single angle rotation that defines I relative to J. The 0 denotes the state at time zero.

Then, the moment acting on each body is:

$$M_x = -6k_{bx}\Phi_x/(1+\phi_x) - 2\zeta\omega\dot{\Phi}_x$$

$$M_y = -6k_{by}\Phi_y/(1+\phi_y) - 2\zeta\omega\dot{\Phi}_y$$

$$M_z = -k_{tz}\Phi_z - 2\zeta\omega\dot{\Phi}_z$$

where k_{tz} is the torsional stiffness.

Figure 4-11 Forces Produced by Beam Elements
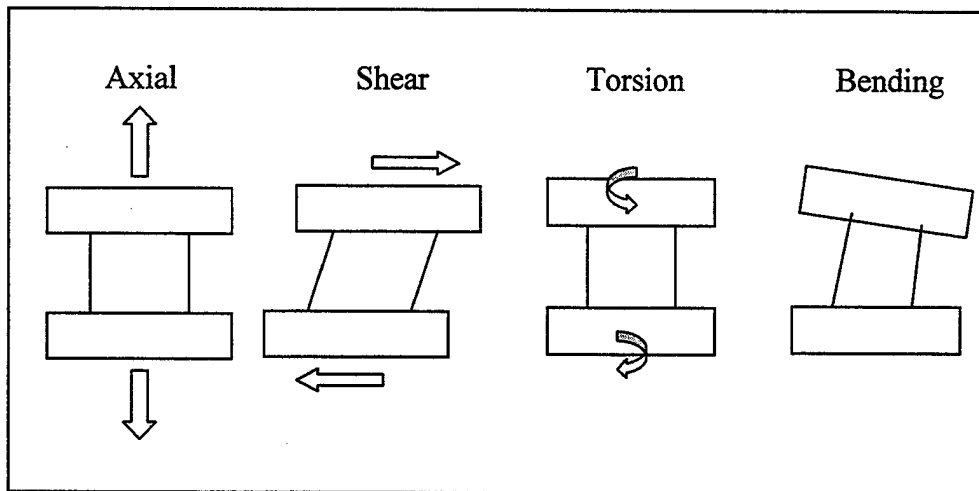
Figure 4-12 The Disc Element Coordinate System

$z_D = (P_J - P_I)/|(P_J - P_I)|$

$x_{ave} = (x_I + x_J)/2$

$y_D = z_D$ cross x_{ave}

$x_D = y_D$ cross z_D

26
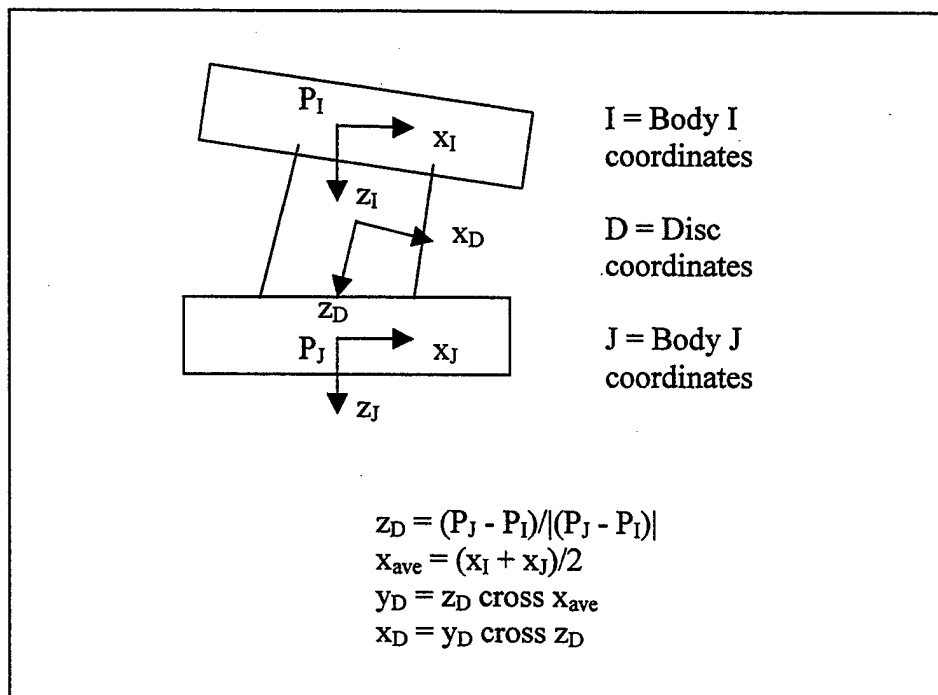
4.3.   Environment Elements.

The HSM software relies on elements of the environment for all of the forcing function inputs to the model. There are three types of elements in the HSM environment: springs for modeling restraints, elastic planes for providing contact surfaces, and motion constraints.

The spring elements are identical to those in the HSM anatomy and can be used to model restraints. The springs can be tension-only, compression-only, or dual acting.

Planes represent one of the sources for applying forces and accelerations to the HSM. Planes can be prescribed motion over time and produce a force proportional to deflection and rate of deflection when they interact with an element of the HSM. The HSM-PC software permits planes to be assigned position, velocity, or acceleration time histories to force the HSM model. Springs in the environment can be attached on one end to a plane. This permits the representation of restraint systems, which connect to a force plane on one side and a rigid body node on the other.

Constraints, although not a physical element like springs and planes, can be used to affect the motion of the model. Any or all of the translational or rotational degrees of freedom of rigid body elements can be restricted with the use of constraint elements.

Figure 4-13 summarizes the environment elements of the HSM-PC. More details about the plane elements are described in Section 4.3.1.

Figure 4-13 HSM Environment Elements

Springs

- identical to the springs that represent ligaments
- can be used to apply external forces and moments
- padding can be modeled as a compressive spring.
- restraints can be modeled as a tensile spring.

Planes

- apply forces to the model by striking rigid body nodes
- have stiffness and damping
- can have motions assigned through forcing functions
- have no mass properties
- provide attachment points for springs in the environment.

Constraints

- can be used to constrain a translational or rotation degree of freedom of a rigid body ·
- can improve accuracy and computation speed in a two-dimensional simulation.

4.3.1 Plane Elements.

Planes are used in the HSM-PC to load the Head-Spine Model. The operator can assign motions to planes by specifying a forcing function in the form of a displacement, velocity, or acceleration time history. Whenever a plane contacts any secondary node on a rigid body, a force is developed orthogonal to the plane and proportional to the depth of penetration and rate of penetration of the node into the plane. There are currently no tangential forces developed from friction or "pocketing" of a node into a plane.

HSM-PC is designed to store pre-defined forcing functions that can be assigned to planes. All of the forcing functions represent time histories of displacement, velocity, or acceleration. A number of shapes and parameters can be specified for the forcing functions. Since forcing functions and planes are defined separately in the HSM-PC, it is possible to create an environment of interaction surfaces and then change the forcing functions assigned to each for different simulations. The available forcing functions in HSM-PC are described later in this section.

The original HSM software required all planes to move in a straight line with the same forcing function motion. HSM-PC allows each plane to be assigned a different forcing function if desired, but planes must still move in a straight line with no rotation. The motion does not have to be along the normal to the plane however.

A plane is defined with three corner points in the global coordinate system in HSM-PC. The relationship between the three points is shown in Figure 4-14. The fourth corner point and the center of the plane can be found through geometry. The plane normal, **n**, is found as follows:

$$\vec{n} = \overrightarrow{p_1 p_2} \times \overrightarrow{p_1 p_3}$$

The direction of motion, **d**, is specified as part of the forcing function.

The process of computing a force between planes and nodes is conducted as follows. Before the simulation begins, the distance from the primary node (center of gravity) to each secondary node on a rigid body is computed. The maximum of these distances for each body is stored. At each time step of the simulation, the minimum distance from the plane to each rigid body primary node is compared to the maximum distance of the farthest secondary node from the same primary node. This determines whether it is possible if that plane could be contacting a node on this rigid body. If it is determined that a contact is possible, then the boundaries of the plane are checked to see which nodes on the body are inside or outside the boundaries. For each node that is determined to have penetrated a plane and is within the boundaries of the plane, then the following expression is used to compute the force exerted on that node:

$$F = k_1 \cdot x + k_2 \cdot \dot{x}$$

where x is the displacement and $\dot{x}$ is the rate of displacement into the plane.

29

One additional modification to the original plane model was to create a smooth transition when a plane first strikes a node. When a plane first strikes a node, a potentially large force is created very quickly. To help smooth this interaction, a decaying exponential term was added to the original HSM planes that reduces the plane stiffness for very small deflections and permits full stiffness as the plane deflection increases:

$$k_1 = k_{original} \cdot \left(1 - e^{-\alpha x}\right)$$

where α determines the shape of the response.

HSM-PC permits springs to be attached to planes in order to represent restraint systems for the model. Planes can have secondary nodes, similar to rigid bodies, which move with the plane. These nodes do not have to be on the plane and are specified as points in the global coordinate system before simulation is run.

Figure 4-14 Relationship between Points on a Plane Element
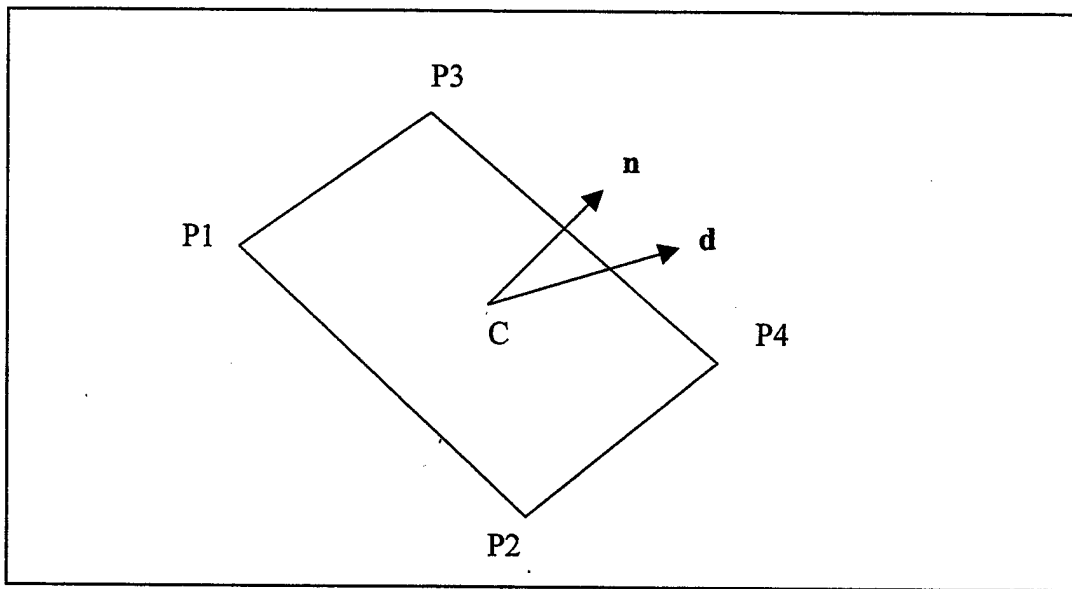
4.3.2 Forcing Functions.

Forcing functions are used in HSM-PC to apply motion to a plane or directly to the primary node of a rigid body. To specify a forcing function, the user must specify a beginning and ending time, a functional shape (Rectangular, Triangular, Sine, Haversine, or User Supplied), the type of forcing function (either Position, Velocity or Acceleration), and the direction of action with respect to the plane's normal vector. The matrix given below in Table 4-1 shows the data computed for the plane or node's motion during the simulation.

Table 4-1 Forcing Function Matrix

Supplied FF Shape	Supplied FF Type	Computed Quantities		
		Position	Velocity	Acceleration
Rectangular	Position	ZD	0	Error
Rectangular	Velocity	SI	ZD	0
Rectangular	Acceleration	DI	SI	ZD
Triangular	Position	ZD	FD	0
Triangular	Velocity	SI	ZD	FD
Triangular	Acceleration	DI	SI	ZD
Haversine	Position	ZD	FD	SD
Haversine	Velocity	SI	ZD	FD
Haversine	Acceleration	DI	SI	ZD
Sine	Position	ZD	FD	SD
Sine	Velocity	SI	ZD	FD
Sine	Acceleration	DI	SI	ZD
User	Position	ZD	FD	SD
User	Velocity	SI	ZD	FD
User	Acceleration	DI	SI	ZD

NOTE: ZD = Zeroth Derivative, FD = First Derivative, SD = Second Derivative, SI = Single Integral, DI = Double Integral

For example, suppose the user specifies that a plane's acceleration follows a triangular profile starting 0.25 sec after the start of the simulation, ending at 0.8 sec after the start of the simulation with a peak of 1.25 $m \cdot sec^{-2}$ at 0.55 sec. Then the Forcing Function Module will compute the curves shown in Figure 4-15. An exact value is computed at each millisecond over the entire duration of the simulation time and those values are stored in a table for look-up and interpolation during the model solution. Now, suppose the user specifies the function type as velocity, then the curves shown in Figure 4-16 are produced.

Figure 4-15 Triangular Acceleration Forcing Function
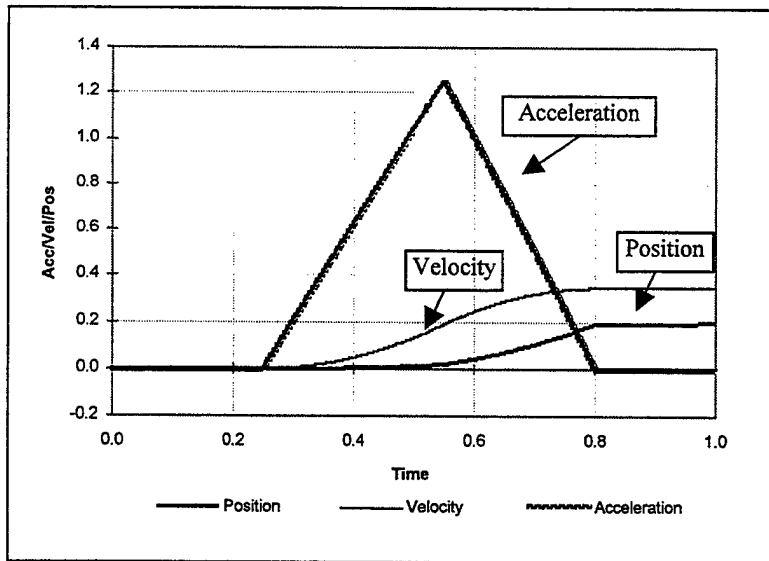
Figure 4-16 Triangular Velocity Forcing Function
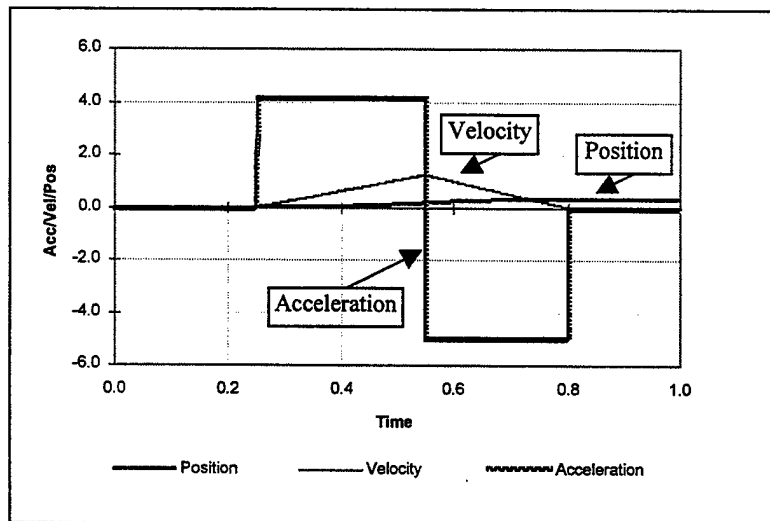
In some cases forcing functions produce discontinuous derivatives. HSM-PC relies on the user to use the forcing function specification properly to avoid this condition.

Similar results are obtained with the other functional shapes. The user can specify arbitrarily shaped functions via a User-Supplied forcing function specification. The user supplies at least three X-Y pairs that specify the amplitude of a forcing function during the simulation. These points are fit with splines and integrated or differentiated as necessary to produce the forcing function data for the specified plane. An example is shown in Figure 4-17, in which the user specified acceleration amplitude at seven times and the velocity and position for a plane were computed.

Figure 4-17 User-Supplied Acceleration Forcing Function

33
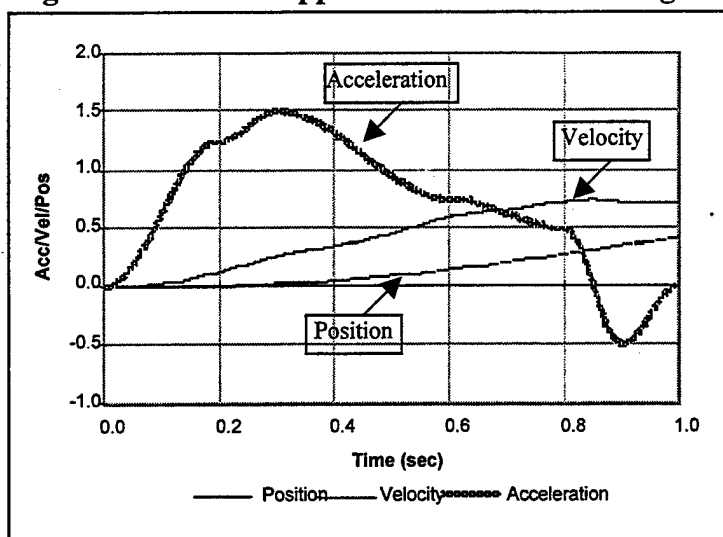
4.4 Versions of the HSM-PC.

The original HSM Development Team produced several versions of the HSM program, which were tailored to meet specific situations. A brief review of those models is presented and similar models developed by BRC are described.

4.4.1 The Preliminary Isolated Ligamentous Spine Model (PILSM)[20].

This model was intended to duplicate the response of the isolated ligamentous spine to impulsive acceleration. The PILSM contained elements representing the pelvis, the thoracolumbar spine, a single beam representing the cervical spine, the head. The model employed spring element representation of both ligaments and the facets. The PILSM was studied under upward Gz acceleration (aircraft ejection studies). It essentially buckled and became unstable shortly after the simulation started. The model was subsequently studied in attempt to stabilize it with planes providing support and alignment and restraints. BRC did not create a model equivalent to the PILSM because based on the reported studies, it did not seem to have applications in studying problems of interest. Moreover, the PILSM was superceded by subsequent developments in the original HSM research and development program.

4.4.2 The Complete Head-Spine Model (CHSM)[1].

In later work, the CHSM referred to as the Complex Spine Model (CSM), but the later version was improved over the CHSM. CHSM was comprised of the pelvis, the thoracolumbar spine, a single beam representation of the cervical spine, the head, the rib cage, and viscera represented by hydrodynamic elements. That model was studied under Gz and Gx, and in modal analysis. With somewhat more favorable kinematic results as compared to the ILSM. BRC has no equivalent model in the HSM-PC. Our most complete model has a cervical spine represented by individual elements and viscera represented by damped springs. It should also be noted that BRC had no properties or geometric data for the CHSM model because no input or output files from that model could be found in the HSM archived files.

4.4.3 The Preliminary Cervical Spine Model (PCSM).

The original development team created a preliminary model of the cervical spine using rigid body elements representing vertebrae and modified disc elements. The disc elements were modified to increase their stiffness to stabilize the model because it had no elements representing ligaments or muscles. Nevertheless, during the original development, the PCSM was added to the ILSM and that version of the model was employed in simulations of Gz with added asymmetrical head mass to study bending of the cervical spine under Gz loads. BRC has no equivalent of the PCSM. BRC's Head-Cervical Spine Model (HCSM) includes elements representing the musculature of the cervical spine.

4.4.4 The Isolated Ligamentous Model (ILS)[21].

In subsequent developments, the PILSM was modified to include ligaments, but it still contained no elements representing muscles. That model was know as the ILS and it was used in additional ejection studies with the PILSM. To stabilize the ILS and PILSM, the disc element stiffness was increased and additional elements representing the viscera were added with limited success. BRC has no equivalent model of the ILS.

4.4.5 The Isolated Ligamentous Spine with Viscera (ILSV) Model[2].

This model was created by adding hydrodynamic elements representing the visceral load path. The visceral elements spanned the region between the pelvis and T10. To stabilize the lumbar and cervical spine, stiffened disc elements were employed. The ILSV was employed in additional ejection simulations. BRC has no equivalent model comparable to the ILSV nor were any ILSV input or output files found in the HSM archives.

4.4.6 The Simplified Spine Model (SSM)[2].

The SSM was created to simplify and stabilize the ILSV for further simulation studies. In the SSM, the thoracolumbar spine was represented by with three beam elements spanning T1-T10, T10-L3, and L3-S1. The viscera were represented with spring and point mass elements between the pelvis and T10. The SSM was employed for comparison studies with the ILSV in ejection simulations. A large series of impedance and modal studies (z-axis) were done with the ILSV and SSM and compared with human data from vibration table studies. While fundamental response of the models agreed with human data the higher frequency components were not well duplicated. BRC has no equivalent of the SSM, but we employed the SSM visceral representations in our most complete HSM-PC model because we had no archived descriptions of the hydrodynamic visceral models.

4.4.7 The Complex Spine Model (CSM).

The CSM was essentially an updated version of the PHCSM described in Section 4.4.3, above. There were no archived input or output files. Therefore, the geometric and properties data for neither the PHCSM nor the CSM were available to BRC's development team.

4.4.8 Primate Models.

The original development teams attempted to create spine-visceral models of a Chimpanzee, Baboon, and Rhesus Monkey. Data from a limited number of studies were employed[2]. It was necessary to extrapolate and interpolate geometry and properties data to create the models. The Primate Models were employed in modal and impedance studies, but there were no comparisons with actual Primate data presented in the Technical Report.[2]

4.4.9 Pilot Alignment Models (PAM).

A series of PAM's were created from the ILS models and one PAM model employed the CSM. The models differed in their incorporation of restraint models attached to primary nodes through "beam element models" which are classically related to the disc element models. The models were employed in pre-ejection retraction studies. The beam model stiffnesses were adjusted to match actual data from restraint retraction experiments. The simulated restraint system forces matched the actual forces quite well.   The restrained ILS model was also employed for ejection studies, as was the restrained CSM model. The model results were reported, but there were apparently no actual data with which to compare the results.   BRC has created no equivalent models to the PAM models. There were no PAM models in the HSM archives so the geometry and properties data given in the Technical Report, which was reasonably complete, was available. BRC intends to add restraint system models to the HSM-PC in later versions.

4.4.10 The Cervical Spine and Head Model (CSHM)[22].

This model, the result of later development work was the apparently first previously developed model to contain a complete compliment of elements. None of the previously discussed models had elements representing muscles. The CSHM contains a total of 209 elements representing vertebrae, discs, facet joints (spring models), muscles, and ligaments. It requires the simultaneous solution of 362 first-degree differential equations to solve in three dimensions.   The original CSHM model was employed in Gx and Gy simulations with passive muscles to study its response to impulsive acceleration. Comparisons of model responses to published experimental data showed large differences in head acceleration in all three axes.   That result prompted later studies aimed at understanding the discrepancy between the simulation and actual data. It was concluded that active muscles were required so those were incorporated. The resulting comparisons were qualitatively similar but there were both phase and amplitude differences in the actual and simulated head accelerations.[23] There was complete but conflicting data for the CSHM model in the archives.

BRC has an equivalent model, but it has hydrodynamic elements for the facets. That version is known as the Head-Cervical Spine Model or HCSM. The best element data for this model came from a CSHM input file from the original HSM software. However, ligament stiffnesses from this input file were deemed incorrect after reviewing numerous literature sources. Ligament stiffnesses for the HSCM model were obtained from Air Force Technical Report TR-81-5.

Other models were created that were specific to a particular task. For example there were versions created to simulate ACES II ejection seat ejections and other experiments. These models were essentially custom models created by modifying the above models. Each version required modification of both data and source code files. The necessity to modify the HSM source code was a major drawback.

HSM Models Developed by the Original Development Team

- PILSM – Preliminary Isolated Ligamentous Spine Model

- CHSM – Complete Head-Spine Model

- PCSM – Preliminary Cervical Spine Model

- ILS – Isolated Ligamentous Spine Model

- ILSV – Isolated Ligamentous Spine with Viscera Model

- SSM – Simplified Spine Model

- CSM – Complex Spine Model

- Primate Models – Chimpanzee, Baboon, and Rhesus

- CSHM – Cervical Spine and Head Model

- PAM – Pilot Alignment Models (six separate models)

- Miscellaneous – Ejection, Gz drop, etc

PC-based Models Developed by BRC

- HCSM – Head Cervical Spine Model

- Simplified Spine Model with HCSM (HCSM + SSM)

- Isolated Ligamentous Spine with HCSM (HCSM + ILSM)

- HSM-PC – PC version of the Head-Spine Model

In contrast, BRC has created a Graphical User Interface (GUI), which allows the user to access model geometry and data from previously defined models. Therefore, any of the original models could be created, but in most cases they have not because we felt they were of little practical use. The models that BRC has created are described briefly below. They all include the HCSM representing the head and cervical spine.

4.4.11 The Head-Cervical Spine Model version 2 (SSM + HCSM).

This model contains the HCSM supported by lumped rigid body elements representing the lower spine and pelvis and a visceral model comprised of springs. There are fifteen rigid bodies in this model. It is basically equivalent to a combination of the SSM and HCSM models.

4.4.12 The Head-Cervical Spine Model version 3 (ILSM + HCSM).

This model is roughly equivalent to a combination of the ILSV and HCSM. It contains 26 rigid bodies representing the entire spine and pelvis with a spring model representation of the viscera. For this model, element data from Technical Report TR-78-7 was included because it represented a consistent set of parameters for the entire spine.

4.4.13 The PC-Based Head-Spine Model (HSM-PC).

This model is BRC's most complete model it contains rigid bodies representing the head, complete spine, pelvis, viscera, and rib cage (53 rigid bodies). Its solution requires the simultaneous solution of 860 first-degree differential equations

BRC has employed selected data from the original HSM researchers' reports[1-4] which document the 12 year evolution of the HSM leading up to this effort to create a PC-based version of the HSM. We have eliminated errors in the original data and documentation and modernized the computer code and accuracy of the numerical solution of the models. In the PC version, users can create new models, store the geometry and element properties data in an easy to use database, and then use the GUI to solve and animate the model as well as manipulate geometry and properties data for iterative studies. In addition kinematic data and force-moment data are available from the GUI to aid the research. Thus, the final HSM-PC can be used to speed development and refine new models and parametrically iterate the previously created ones.

References

* Belytschko, T., Schwer, L. and Schultz, A. A Model for Analytic Investigation of Three-Dimensional Head-Spine Dynamics. AMRL-TR-76-10. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1976).
* Belytscho, T. and Privitzer, E. Refinement and Validation of a Three-Dimensional Head-Spine Model. AMRL-TR-78-7. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1978).
* Williams, J. and Belytschko, T. A Dynamic Model of the Cervical Spine and Head. AMRL-TR-81-5. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1981).
* Belytschko, T. and Williams, M.R. AAMRL-TR-85-019. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1985).

5.0   HSM-PC Software.

The HSM-PC consists of three modules: a user-friendly graphical user interface, a computational module, and a biomechanical database.

The graphical interface is a feature new to the HSM-PC. The original HSM software required users to make changes to simulations either by modifying code and re-compiling or by working directly in text files. Users of HSM-PC can perform all of the operations related to creating and running a simulation entirely from the user interface. For example, an operator can select and display a model for simulation, viewing it from any angle and selecting certain elements to be displayed or hidden. The environment for a simulation can be specified and viewed. Parameters related to a simulation such as duration and desired output variables can be specified. A simulation can be executed from within the interface, and then the results can be viewed through charts or an animation of motion. The interface also displays the messages generated during the solution of the model.

The HSM-PC computational module is an executable program written in Fortran 90 that acquires simulation data from the graphical interface, applies Newton's Laws to the model and integrates the resulting motion over time, and writes the specified output into a file for use by the graphical interface. Information from the user interface is communicated via a series of ASCII text files. Although these files are not optimal for conserving space, they allow an experienced user of the software to run simulations without the graphical interface. The program automatically adjusts the solution of the model to account for the number of elements -- re-compiling the HSM-PC is not necessary for different simulations. The computational module also performs some simplistic element checking before attempting to solve model. For instance, the model will detect and report a ligament that is attached to a non-existent vertebra.

Finally, data for the HSM-PC is stored in the form of a Microsoft Access database. The database can be edited in a standalone fashion from the rest of the HSM-PC software. Although the graphical interface also allows parameters to be changed, those changes are temporary. Only changes to the database itself are permanent. A separate interface was written to help users acquire data from the database.

Figure 5-1 summarizes the functions of the three modules and their interactions with on another.

40

Figure 5-1 Schematic of the Operation of the HSM-PC Software

User
Interface

User Interface
- select model
- select simulation parameters
- specify environment
- view results
- view animation

Model data
Environment data
Simulation data

Simulation output

Biomechanical data

Computational module
- reads model, environment, and simulation data
- simulates model response
- writes output

Database
- stores biomechanical data
- has separate interface for changes

Computational
Module

Database

5.1   Graphical User Interface.

The Graphical User Interface facilitates the assembly of the initial model geometry, setting of initial conditions, design of forcing function time history, and point(s) of application. In addition, it enables creation and positioning of external surfaces (e.g., restraints, seating, surrounding objects). The interface was developed using Microsoft Visual Basic version 5.0, Microsoft Access 97 and several third-party packages. The third-party packages include a three-dimensional graphics package for modeling and control of the graphical representation of the model; a charting package for generating X-Y plots of model variables; and a menu/toolbar package for a professional looking interface.

The Graphical User Interface helps the user to set up a simulation by specifying all the pieces that are required. The design of the simulation is modular so that future simulations can reuse parts that have already been designed in a previous simulation. Text files are used to communicate between the interface and the computational program. The sophisticated user can edit these text files directly using the built-in text editor that is a tool in the program. Once a simulation is specified, the interface can initiate the computational program and monitor its progress. When the computations are complete, the interface reads the output data and provides the tools to review the results through animation or plotting.

The graphical representation of the model was developed using a combination of geometric values from the original HSM model, design suggestions from a medical illustrator, and a commercially available three-dimensional model. Rigid bodies in the model representing the head, the vertebrae, the pelvis, and the ribs, were modeled using AutoCAD by Autodesk and then transferred to the three-dimensional graphics package used by the application. This resulted in a representation where each body can be translated and rotated independently during animation.

All of the initial data values for each of the HSM models are stored in an Access database. The graphical user interface reads these values and displays them on the properties screens. These data values include geometric coordinates for nodes as well as parameter values such as mass or stiffness coefficients. The values can be displayed in English or metric units. The units preference is selected when the simulation is defined.

5.1.1   HSM-PC Interface.

The graphical user interface for HSM-PC includes a menu, a toolbar, and tabs that allow the user to view either the three-dimensional representation of the model or X-Y plots of output values. Other features of the interface include a start-up screen for selecting or designing a simulation, a simulation wizard, and a text editor. Each of these features are discussed in greater detail below (see Figure 5-2).

5.1.2 Menu.

The **File** menu contains New, Open, Close, Save, Save As and Exit options.

The **Edit** menu allows the user to display dialogs to modify properties of the Model, Environment, Forcing Functions, Simulation, and Output. These dialogs are described in the Dialog Boxes section in 5.1.8.

The **Run** menu allows the user to initiate the computational portion of the program.

The **View** menu contains options for viewing animation or plotting.

The **Tools** menu allows the user to use the Text Editor or the Calculator.

The **Help** menu contains options for the Contents or Index of the Help File or information About HSM-PC.

5.1.3 Toolbar.

The toolbar buttons are provided for quick access to menu options such as file handling, running a simulation, and plotting. Toolbar buttons are enabled, as they are relevant to the current environment of the program. For example, the **Run** button is not enabled unless there is a simulation currently open. The basic toolbar buttons are New, Open, Save, Properties, Run, Animate, Text Editor, and Help. When a three-dimensional model is visible, other tools are displayed on the toolbar that permit manipulation of the model. These include setting the view of the model, changing the magnification, and changing the visible features.

5.1.4 Tabs.

Model

When the Model Tab is selected, the left panel displays the three-dimensional model. The right panel will display the animation toolbar when it is selected.

Plotting

When the Plotting Tab is selected, the left panel displays a data plot and the right panel gives options for plotting values from the simulation run.

Figure 5-2 HSM-PC Graphical User Interface
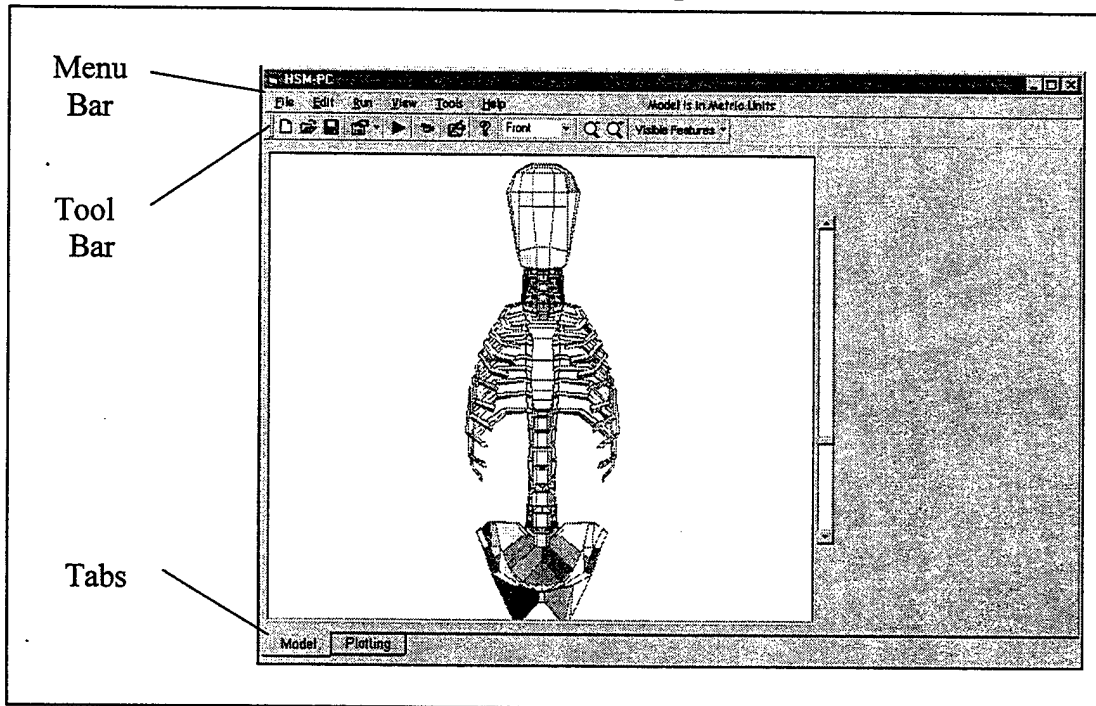
5.1.5 Startup Screen.

The HSM-PC Startup Screen allows the user to locate and open a new or existing simulation when the application is first loaded (see Figure 5-3)

New

The first tab displays the icon for the simulation wizard. When this icon is selected, the simulation wizard becomes active and the user is aided in specifying a new simulation.

Existing

The second tab has an explorer-like interface, which facilitates the navigation of the file directory hierarchy for the selection of an existing simulation file. The file path in initially set to the application directory, but the user can reposition to any local or network drive for file selection.

Recent

The third tab on the startup screen displays a list of the last four files that have been accessed by the user of the HSM-PC application. The user can select any of the four files and quickly load all of the relevant data into the program.

Figure 5–3 Startup Screen
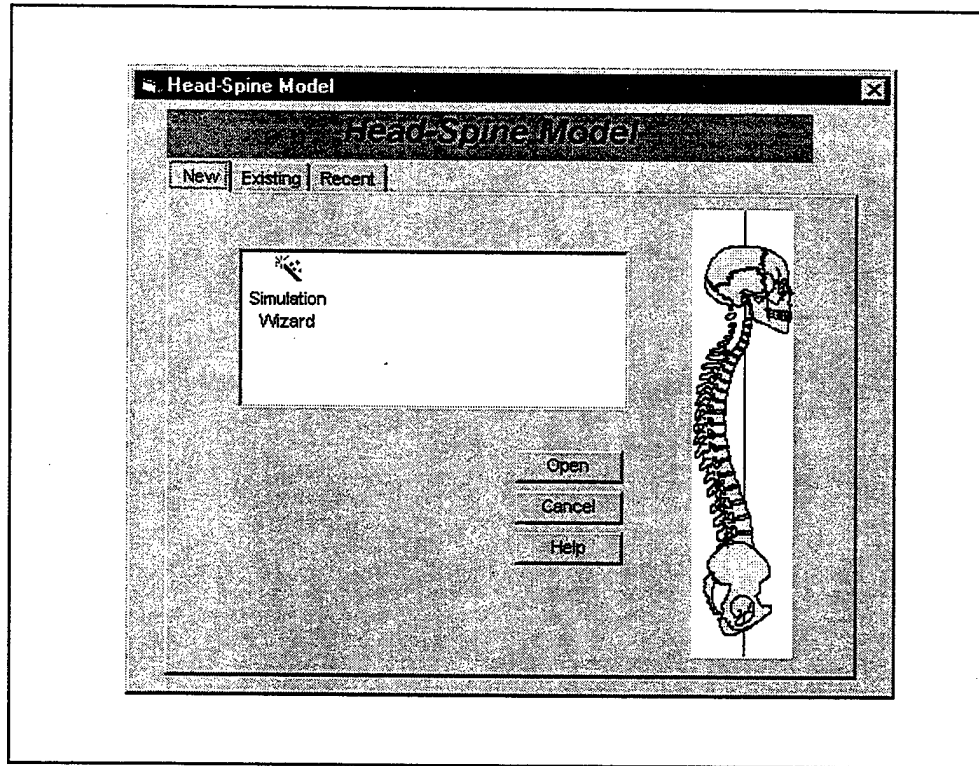
5.1.6 Simulation Wizard.

The Simulation Wizard was designed to aid the user in defining a simulation. The wizard consists of five screens which walk the user through all the necessary steps: model selection, file naming, simulation specification, output data specification, simulation naming and data modification. The wizard screen includes color coding to help the user identify the different steps involved in the definition process. The introductory screen is shown in Figure 5-4 and the five wizard screens are shown in Figures 5-5 through 5-9.

Introductory Wizard Screen

The initial wizard screen introduces the user to the configuration process by defining the five steps and displaying the colors that go with each step.

Wizard Screen 1

For the first step the user is asked to specify the timing, units, and the print style of the simulation. These data are entered on the Simulation Parameters screen, which is identified with a red folder icon.

Wizard Screen 2

The second step is to identify the model to be used in the simulation. The user has the choice of three models: Selected Vertebrae, the Head-Cervical Spine Model, and the Head-Spine Model. Once the model is selected, the database containing the geometric data for primary and secondary nodes can be read. In addition, the parameters for the elements that are included with that model (muscles, ligaments, discs, etc.) are also read. The model selection screen is identified with a yellow folder icon.

Wizard Screen 3

The third step in specifying a simulation involves defining various environmental features. These may include springs, planes, fixed points and constraints. The green folder icon is associated with this data.

Wizard Screen 4

The fourth step involves specifying the forcing functions that are a part of this simulation. There can be only one forcing function for each plane defined in the environment. This screen has all the options necessary for defining multiple forcing functions.

Wizard Screen 5

With step five, the output parameters for the simulation are specified. The Output Parameter dialog prompts the user to select that parameters are to be recorded

Figure 5-4 Introductory Screen
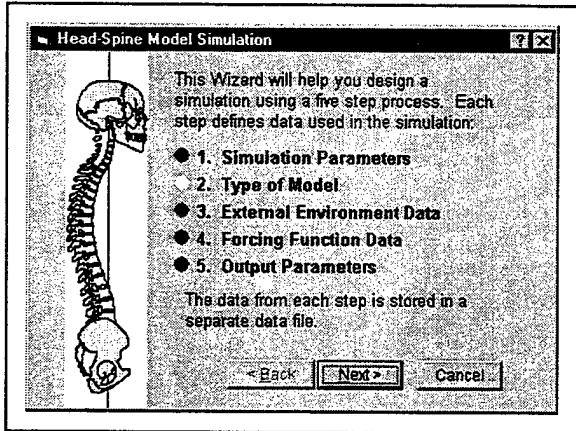
Figure 5-5 Wizard Screen 2
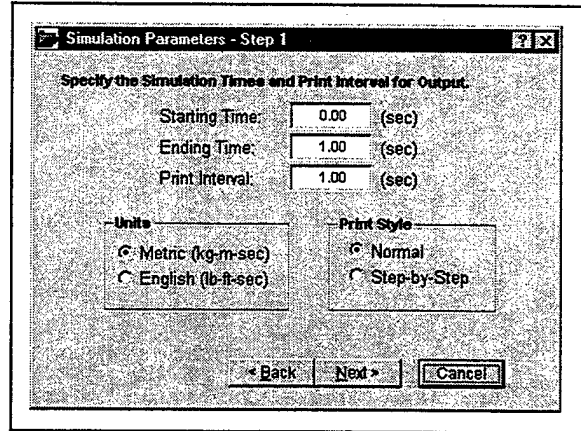
Figure 5-6 Wizard Screen 3
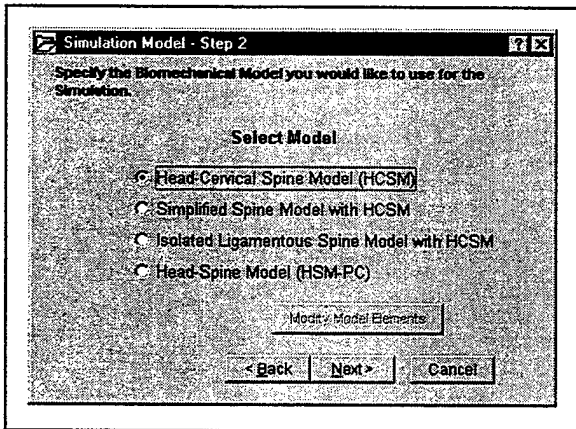
Figure 5-7 Wizard Screen 4
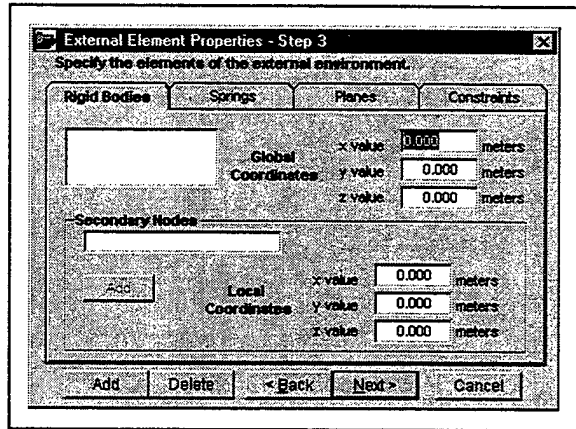
Figure 5-8 Wizard Screen 5
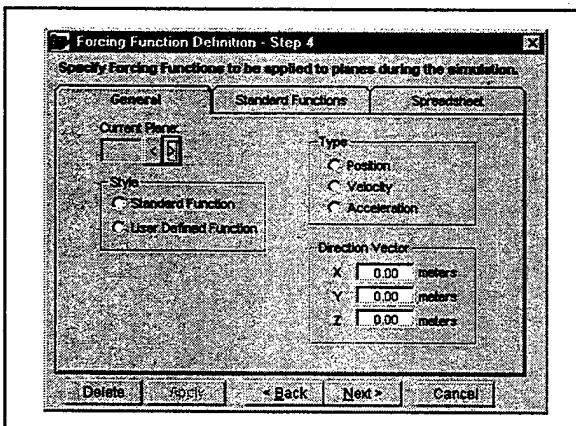
Figure 5-9 Wizard Screen 6

48
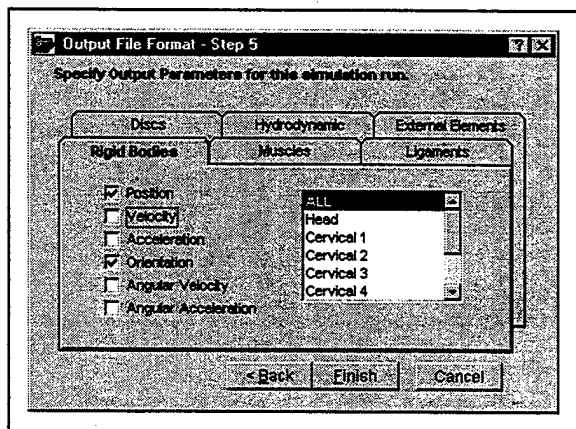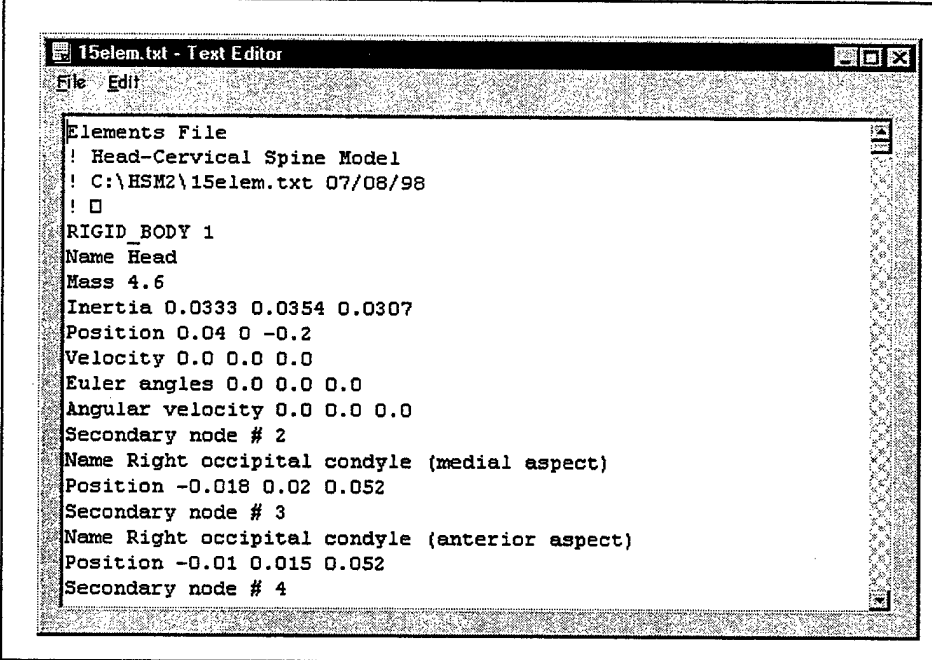
in the output data. The user should consider any information that he may wish to see in a plot after the simulation run and make sure that the data is available in the output file. Positional data for the primary nodes will be necessary in the output file to enable the animation option. The Output File Format screen has a magenta file folder icon.

Once the five dialogs are completed, all the text files are written and the simulation is ready to run. The user will then be asked whether they are ready to run the simulation. The Edit option on the main menu is available for changes to any of the properties of the simulation and the simulation can be triggered at any time using the Run option on the main menu. Once the simulation is completely specified, the screen will display the three-dimensional graphical representation of the model selected.

5.1.7 Text Editor.

The Tools Menu has options for loading a Text Editor. The Text Editor can be used to view or modify any text file. It can be used to view the output file that contains the raw data. It can also be used to modify any of the simulation-input files. The menu that is available with the Text Editor includes options to print text files under the File menu option.

Figure 5-10 HSM-PC Text Editor

```
15elem.txt - Text Editor
File   Edit

Elements File
! Head-Cervical Spine Model
! C:\HSM2\15elem.txt 07/08/98
! □
RIGID_BODY 1
Name Head
Mass 4.6
Inertia 0.0333 0.0354 0.0307
Position 0.04 0 -0.2
Velocity 0.0 0.0 0.0
Euler angles 0.0 0.0 0.0
Angular velocity 0.0 0.0 0.0
Secondary node # 2
Name Right occipital condyle (medial aspect)
Position -0.018 0.02 0.052
Secondary node # 3
Name Right occipital condyle (anterior aspect)
Position -0.01 0.015 0.052
Secondary node # 4
```

5.1.8 Dialog Boxes.

Model Properties Dialog

The Model Properties dialog box includes tabs for each of the types of elements in the model. Each tab includes the names of the elements and any pertinent parameters for that element. The first tab can be used to view or modify properties of the primary and secondary nodes. The second tab can be used to specify the muscles. The third tab contains information on the ligaments that are included in the model. The fourth tab has information on the discs that are defined for the model. The fifth tab has information on the facets and the sixth tab has information on the viscera. (See facing page for screen).

External Elements Properties Dialog

The External Element Properties dialog box includes tabs for fixed points, springs, planes, and constraints. The New button allows the user to add a new element and specify its properties. The Delete button will delete the currently selected element (see screen for step 3 of Wizard).

Forcing Function Dialog

The Forcing Function dialog enables the user to specify several parameters associated with the forcing functions. The software includes a variety of ways to specify forcing functions. If the user wants to use a standard function, they can select from rectangular, triangular, haversine, or sinusoidal. If the user does not wish to use a standard function, then they can design their own. A spreadsheet is made available to assist the user in entering a series of values to define the forcing function. Using the spreadsheet, time and data values can be entered in two columns of up to 50 values. The spreadsheet also includes an option to graph the values once they have been entered. The spreadsheet values are saved in an ASCII file that is later processed by the computational portion of the program. In addition, the user can specify a file that already contains a series of values defining a forcing function. There are several parameters that need to be defined in addition to the function such as the direction vector and the timing for a standard function (see screen for step 4 of Wizard).

Output File Format

The Output File Format dialog allows the user to select the elements for which parameters will be recorded in the output file. Where there are multiple elements of the same type available, the first option is ALL, i.e. print output values for all elements. This dialog also allows the selection of which parametric values to include in the output file (see screen for step 5 of Wizard).

Figure 5–11 Model Properties Dialog
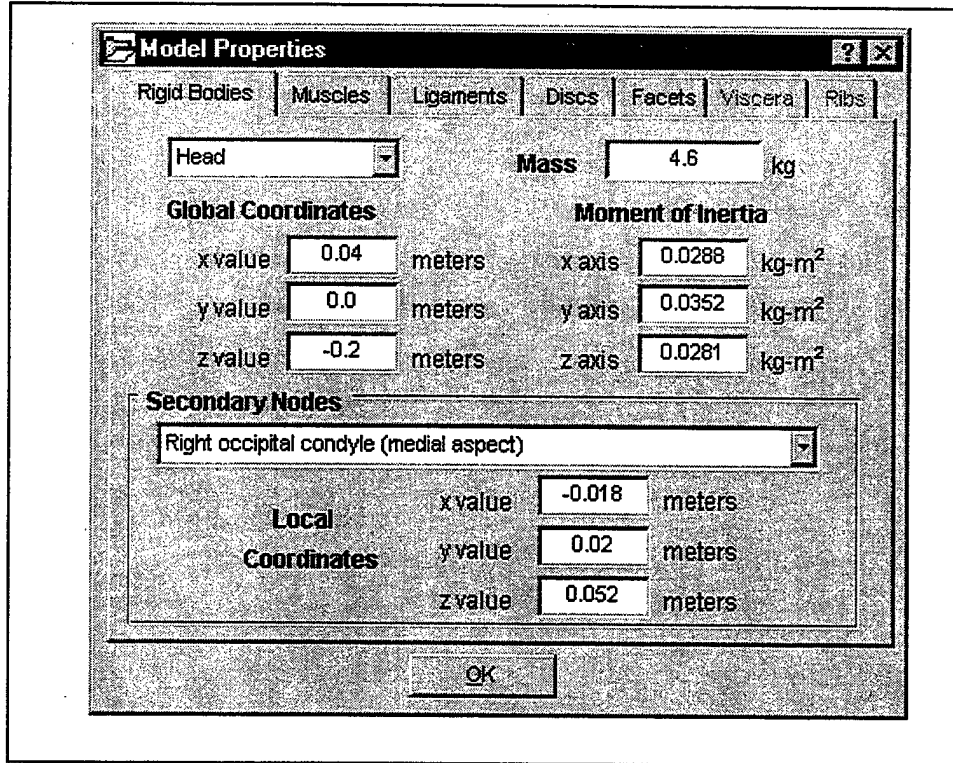
5.1.9 Animation.

Once an output file has been generated, a menu option for animation becomes available. A series of snapshots of the model's position can be displayed in real-time with a smooth animation effect. This animation has controls to Play, Stop, or Rewind the snapshot series. The time step of each snapshot is displayed on the animation toolbar. The snapshots can also be viewed in step mode going both forward and backward using the step mode controls. The animation can be viewed from several vantage points using the View options on the main tool bar. The speed option controls the sampling rate of the snapshots. The animation tool bar also includes screen capture and printing capabilities. In order to capture a screen image the user must first select an image and then download it. The step mode control enables the user to step through the animation a frame at a time. When the desired image is displayed the user can download that image by selecting the Save Image button on the animation toolbar. The program will prompt the user for a file name and then a bitmap image of the model will be saved in that file. The printing function is selected by using the Print Image button on the animation toolbar. When this selection is made, the program will display the dialog box that gives page layout options for model images. The print options include one half-page image, four medium size images, or four smaller images in a single row. After one of the print options has been selected, the user will be prompted for the names of the previously saved bitmap files to be positioned on the appropriate page layout.

5.1.10 Plotting.

When the **Plotting** tab is selected, the left panel displays a data plot and the left panel has tools for setting plotting options. Under Plotting Options, the upper box has a tree-type control which allows the user to expand an element type to display specific elements. Once an element has been selected, the lower box lists the available parameters and the value to be plotted can be selected. Once a selection is made from the lower box, the plot is generated.

Figure 5–12 Animation
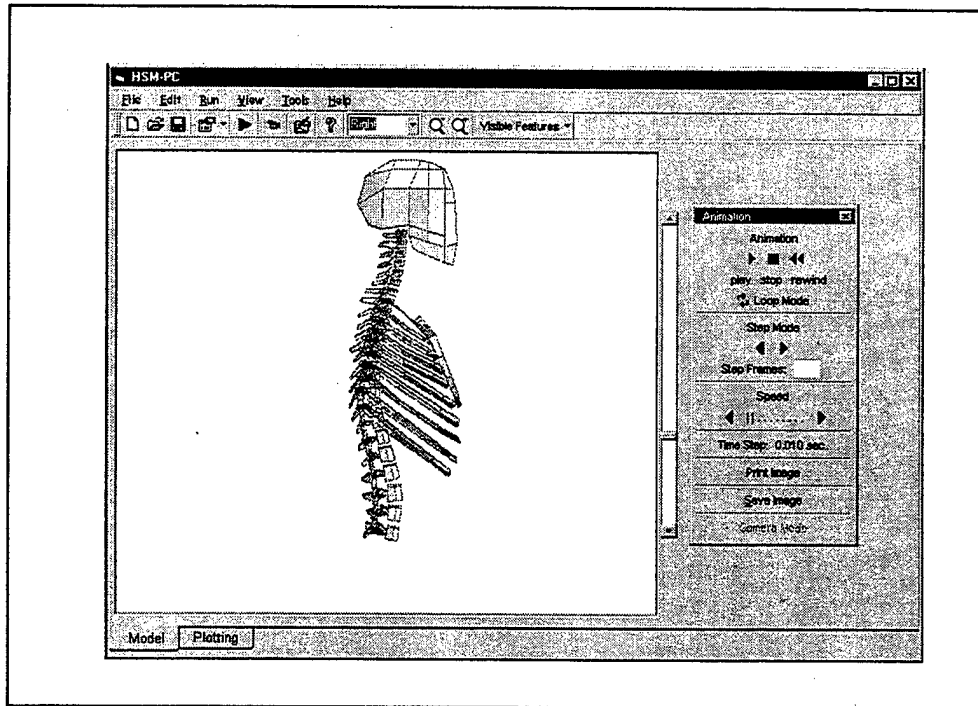
Figure 5– 13 Plotting
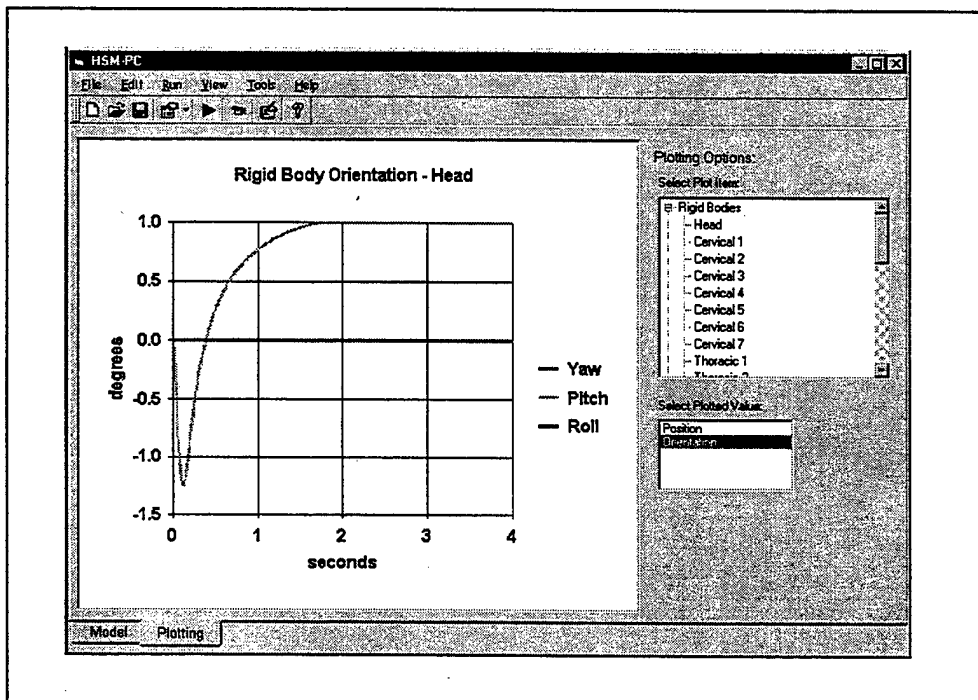
5.2   Computational Module.

The HSM-PC computational module is an executable program written in Fortran 90 that solves the equations of motion of the HSM and makes the results available to the user interface. The module is automatically executed by the user interface. All of the input data used by the computational module come from a series of ASCII text files that are created by the user interface. All of the status messages and output from the computational module are written to two additional ASCII text files for viewing by the user interface. The use of ASCII text files for input and output allows the computational module to be used in a standalone mode if necessary.

The computational module simulates the motion of the HSM by creating and solving a set of simultaneous differential equations for each rigid body and muscle element in the HSM. The motion of each rigid body element is described by six second-order differential equations, three each of translation and rotation. Each second-order equation is re-written as two first-order equations, resulting in twelve first-order equations per rigid body element. The force in each muscle element is determined by solving two simultaneous first-order differential equations. Thus, for any given version of the HSM, there are (12*number of rigid body elements) + (2*number of muscle elements) differential equations that must be solved simultaneously.

The computational module is written in Fortran 90 and was compiled with Digital Visual Fortran. All of the code in the program was written by BRC for the HSM-PC, with the exception of a differential equation solver called the Petzold-Gear method that was part of the IMSL library, a set of numerical routines for Fortran. The solver is referred to as DDASPG (Double precision Differential Algebraic Solver Petzold-Gear method)[†] within the software. The source code files for this portion of the HSM-PC are either *logically* related or *functionally* related. Logically related files refer to the time at which they are executed during a simulation. Functionally related files refer to a collection of subroutines that perform a similar function; for example, all the subroutines that perform operations on spring elements are located in the spring.f90 file. The subroutines that are related logically usually call subroutines that are part of a functional area of the model.

The computational section of the HSM-PC is designed to allocate memory and create the model equations of motion based on the elements that are specified by the user interface. It is not necessary to recompile the program for new versions of the HSM. Since it is possible to create a model that is physically impossible to achieve, the computational module performs some limited element validation before attempting a simulation. For example, the endpoints of most elements are checked to insure they connect to valid secondary nodes.

The software automatically records the status of progress through the computational module with a log file. There are three different types of messages recorded in this file:

[†] DDASPG is the IMSL version of DASSL from the IMSL Math Library, Visual Numerics, Inc., Houston TX. (1994).

status messages, warning messages, and error messages. These messages are described further in Section 5.2.3.

Because of the time it takes to run the HSM model, the software features the capability to save a settled version of the model for use in simulations. All of the data necessary to start the HSM from a settled state is stored in a single file.

5.2.1 Logical Flow of HSM-PC Computations.

Figure 5-14 summarizes how this section is organized.

Figure 5-14 Organization of Computation Module Discussion

Section	Title	Description
5.2.1	Logical flow of computations	A discussion of the sequential progression of HSM computations.
5.2.2	Functional organization of the computational module	A discussion of the division of the computational module code into functional areas of related subroutines.
5.2.3	The use of data files by the computational module	This section describes the data contained in each of the input and output files used by the computational module.
5.2.4	Fortran 90 in the computational module	A discussion of Fortran 90 constructs that were used in the program to make it easier to understand and modify.

To understand the flow of computations in the HSM-PC, consider the four-step process shown in Figure 5-15. The first step of any simulation is to read input data from a series of ASCII files (see Section 5.2.3 for information about the files). Simulation information such as the desired output variables, length of a simulation, desired print interval, and the names of other files used in the simulation are all read first. The software then reads the element file, environment file, and forcing file information.

The second step of the model is to initialize values for the simulation. The software computes the degrees of freedom for the model, allocates the necessary space for the different elements, and performs a check of the elements. If the simulation is starting from the settled position, the model reads in settled rigid body positions. Files for output and status messages are opened during this step. The parameters necessary for the differential equation solver are also initialized.

The third step is the solution of the model over the specified time interval. The DDASPG solver automatically iterates through the interval at the necessary time steps to achieve the specified error tolerances. If the operator has chosen the standard output method, the software writes output into memory arrays to be printed into a file when the model is complete. If the operator has chosen the step-by-step printing method, the software prints a select set of data into the output file after each successful integration step. For the step-by-step method, position and orientation of the first and last rigid bodies is printed.

The final step for the computational module is to print data into the output file, and is only necessary if the operator is using the default print mode. The software copies the data of output arrays in memory to the specified output file.

There are additional files and subroutines in the computational module, but all are affiliated with one of the steps above.

Figure 5-15 Logical Flow of the HSM Computations

program main (main.f90)

subroutine GetInputData (input.f90)
- read simulation file
- read element data
- read environment data
- read forcing function data

subroutine Initial (intial.f90)
- initialize internal arrays
- validate element connections
- read settled position data if necessary
- initialize HSM elements
- open output and status file

subroutine RunSimulation (model.f90)
- loop through time interval
- call DDASPG solver
- compute forces in each element
- form equations of motion
- form muscle differential equations
- apply constraints to motion
- solve for motion of rigid bodies
- write data to internal arrays or directly to output file

subroutine PrintToFile (output.f90)
- copy output data from internal arrays to output file
- close output files

5.2.2 Functional Organization of HSM Computations.

The operation of the computational module can be interpreted by the function of its components. Figure 5-16 summarizes the functions of different parts of the program.

A user-defined Fortran 90 TYPE was created for each of the elements in the HSM software. This allows each element to be referenced in a standard format with equivalent variables. All of the actions that are related to a certain element type are organized into a single source code file and a single Fortran 90 MODULE. For example, the subroutines that perform operations on rigid body elements are all located in the rigidbody.f90 file. Some of the subroutines in this file are used in the initialization portion of the program, like InitializeRigidBody, while others are used in the simulation portion of the program, like UpdateRigidBody. Information about the Fortran 90 TYPE and MODULE constructs can be found in Section 5.2.4.

The software has several files that perform utility operations. A global.f90 file specifies variables and subroutines that are used in many subroutines. All of the elements and simulation variables are defined in the global.f90 file. The file also contains subroutines for writing status messages to a file and parsing input from a file.

Matops3.f90 is a collection of linear algebra routines for three-element vectors and nine-element matrices. Routines for vector and matrix operator overloading are contained in this file (and described further in Section 5.2.4).

The only subroutine in the HSM-PC software that was not included in a Fortran 90 MODULE is the subroutine RES in the file IMSL_routines.f90. This subroutine is required by the IMSL DDASPG solver, and cannot be in a MODULE.

Figure 5-16 Description of HSM Files by Function

File name:	Description
rigidbody.f90	Subroutines associated with the rigid body elements, including printing the element data and updating velocities and positions of nodes at each time step.
spring.f90	Subroutines associated with the spring elements, including printing the element data and updating the endpoints and force in the spring.
hydrodynamic.f90	Subroutines associated with the hydrodynamic elements, including printing the element data and updating the endpoints, volume, and pressure.
disc.f90	Subroutines associated with the beam elements, including printing the element data and updating the endpoints and deflection from the original position.
muscle.f90	Subroutines associated with the muscle elements, including printing the element data and updating the endpoints, activator concentration, and other muscle parameters.
plane.f90	Subroutines associated with the environment planes, including printing the plane positions, computing the force exerted by the plane, and updating the plane position according to the forcing functions.
forcingfunction.f90	Subroutines associated with the forcing functions, including creating an array of displacements, velocities, or accelerations that are prescribed for a plane.
initial.f90	A single subroutine that initializes all of the HSM elements and establishes simulation parameters.
input.f90	Subroutines that read input files and assign the element, environment, and simulation data to the proper variables.
output.f90	Subroutines that write data from internal arrays into output files.
main.f90	The main program for the computational module that reads input, initializes the model, runs a simulation, and writes output.
model.f90	Subroutines that loop that solve the HSM equations of motion and update variables used by the IMSL differential equation solver.
imsl_routines.f90	A single subroutine used by the IMSL differential equation solver that computes the *residual* of the system of equations at each time step.
matops3.f90	Subroutines and functions that define the matrix and vector type and perform matrix and vector operations.

5.2.3 Data Files in the Computational Module.

The HSM-PC uses ASCII data files to communicate input and output between the graphical user interface and computational module. BRC chose ASCII files so that the computational module could be run in a standalone mode if necessary.

There are five input files used in the HSM-PC. The simulation file is the first file required by the computational module. It contains the names of the other files that will be required, as well as simulation data such as desired output variables, the time of the simulation, and error tolerance information. The element file contains the biomechanical data needed to create the mathematical model of the HSM, including all of the rigid body, spring, beam, muscle, and hydrodynamic element information. The environment file describes the planes and springs in the environment, as well as any motion constraints imposed on the model. If planes are specified in the environment file, then a forcing function file is required to impart motion to the planes. Finally, an initialization file is used to store data from the model in a settled posture after it is exposed to gravity. Since the HSM-PC can take a long time to execute, it is useful to save the settled model position before forcing the model with environment planes.

There are three output files used in the HSM-PC. The first is the output file for simulation results. This file can be in two forms, depending on a flag specified in the simulation input file. One form calls for a limited set of data to be printed at each successful step in the simulation (described in Section 5.2.1), while the second form allows for data to be stored internally in memory and then printed organized by element type after the simulation is complete. The second output file is the status file, which contains status, warning, and error messages. The computational program automatically logs messages to this file, and the user interface can display these messages. Status messages reflect progress of the software through different parts of the simulation process. Warning messages represent conditions that might be a problem in a simulation, but the simulation is able to continue. Error messages represent incorrect element connections, invalid simulation conditions, or errors during the solution and the program is terminated. Examples of the three types of messages are shown below:

- "Status: HSM-PC has successfully read the input files."
- "Warning: No forcing functions were read. The model is forced only by gravity."
- "Error: The environment file can not have muscle elements - these should appear in the model element file."

It should be noted that all of the input and output files can be accessed directly from the user interface. It is recommended that operators of the HSM-PC use the interface to create input files.

Figure 5-17 summarizes the contents of the different input and output files used by HSM-PC.

61

Figure 5-17 Summary of Files.

There are five input files used in the HSM-PC:

1. Element File -- all of the rigid body, spring, hydrodynamic, muscle, and disc elements that make up the HSM-PC.

2. Environment File -- all of the planes, springs, and constraints that make up the environment for the HSM-PC.

3. Simulation File -- a file that specifies simulation parameters such as stop time, print interval, desired output variables, and the names of the other files.

4. Forcing Function File -- all of the forcing functions that are applied to the environment planes.

5. Initialization File -- a file that contains all of data necessary to re-start an HSM-PC simulation after it has settled under gravity but before forcing functions were applied.

There are three output files used in the HSM-PC:

1. Output File -- a time history of the specified output data.

2. Status File -- a list of status messages, warnings, and errors encountered by the computational module during execution.

3. Initialization File (also an input file) -- this file can be created by the computational module when it is desired to settle the model under gravity.

5.2.4 Fortran 90 in the Computational Module.

The computational module of the HSM-PC is coded exclusively in the Fortran 90 programming language. Fortran provides several useful features that make the HSM-PC version of the HSM easier to modify, verify, and understand than the original version. Examples of the constructs are shown in Figure 5-18.

The HSM-PC uses the Fortran 90 TYPE extensively to create user-defined types of variables. The TYPE feature permits elements, such as rigid body elements, to have the same set of data available automatically. For example, each rigid body has a velocity vector for the center of mass. This vector is referenced RB(1)%v for the first rigid body element. This representation is much more readable than multi-dimensional arrays.

The use of the MODULE construct allows data variables and subroutines that are functionally related to be grouped into a module. The variables can only be operated on by member subroutines of that MODULE. This protects variables from being changed by other subroutines and permits future changes to the software with less risk of unintentional repercussions.

BRC made extensive use of the operator overloading features of Fortran 90. Using this feature, the standard mathematical operators such as +, -, *, and / can be taught how to operate on new data types. For example, the HSM-PC uses overloaded operators to perform vector and matrix operations in a single step. With overloading, Newton's Second Law, F = m*a, can be applied to a three-dimensional vector of accelerations and forces with a single line of code.

Fortran 90 also permits dynamic memory allocation, which allows the HSM-PC to size arrays of elements (rigid bodies, springs, etc.) according to how many elements are found in the input file. It should never be necessary to re-compile the HSM computational module to change the number of elements, and the memory requirements of the software are minimized because of this feature.

The free-form source code feature of Fortran 90 creates a much more user-friendly code. Free-form source code permits the following:

- Lowercase code is acceptable, but not distinguished from uppercase
- Code can start and end in any column
- The character "&" at end of line denotes continuation
- The character "!" denotes comment and can appear at any column in the code

Figure 5-18 Examples of Fortran 90 Constructs

- ## Defined TYPEs

  ```
  type matrix
     real(real*8) elt(3,3)
  end type matrix

  type rigid_body
     type(vector) a, v, s
     type(matrix) inertia
  end type rigid_body

  a = 0.d0
  ```

- ## Use of MODULEs

  ```
  module global
     type(rigid_body) rb(10)
     type(spring) s(10)

     subroutine writetofile
        write(*,*) time
     end subroutine writetofile
  end module global

  module rigid_body
     use global
     ! global variables and subroutines now
       accessible
  end module rigid_body
  ```

- ## Operator overloading

  ```
  function mattimesvec(a,b)
     type (matrix) a
     type(vector) b, mattimesvec
     ...
  end function mattimesvec

  F = m*a
  ```

- ## Dynamic memory allocation

  ```
  module global
     type(rigid_body), allocatable, dimension(:) :: rb
  end module global

  subroutine initialize
     use global
     allocate(rb(10))
  end subroutine initialize
  ```

- ## Free-form source code
  - lowercase acceptable, but not distinguished from uppercase
  - start in any column, end in any column
  - "&" at end of line denotes continuation
  - "!" denotes comment - can appear at any column

  ```
  F = m*a          ! comments can appear here

  X = m*a + F + (1-(10*velocity)) & ! continued
          + 10*acceleration
  ```

5.3 Biomechanical Database.

A biomechanical database has been developed to hold the geometry and materials properties for the spine. The coordinates and parameter values used in HSM are based on these data and we have added additional data from the more current literature. The database application is written using Microsoft Access 97 and required that version of Access to be executed.

The database application has a main menu that allows access to data from different areas of the spine. Once a selection has been made from the main menu, individual records from the database can be viewed. The buttons at the lower left corner on the screen allow navigation through the database. From this screen a multiple record view can be selected by pressing the button labeled Multiple Record View.

Figure 5–19 Head-Spine Database Main Menu
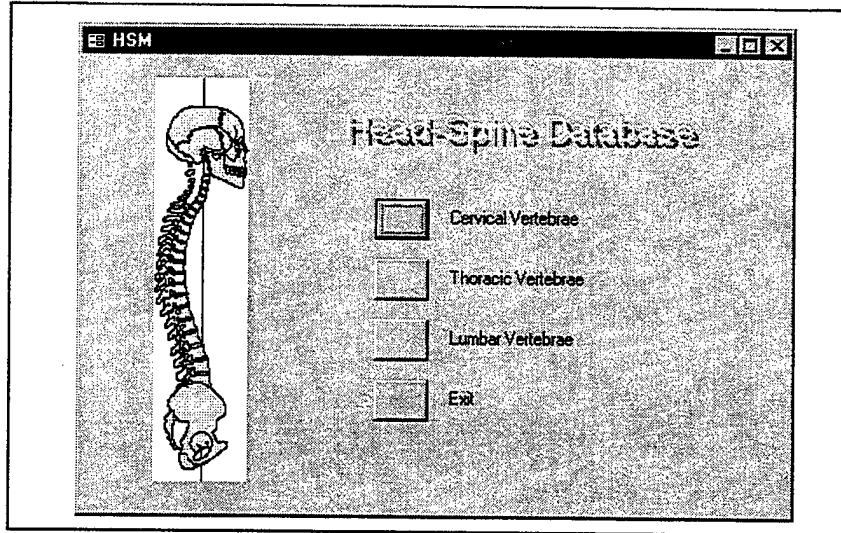
Figure 5–20 Single Record View
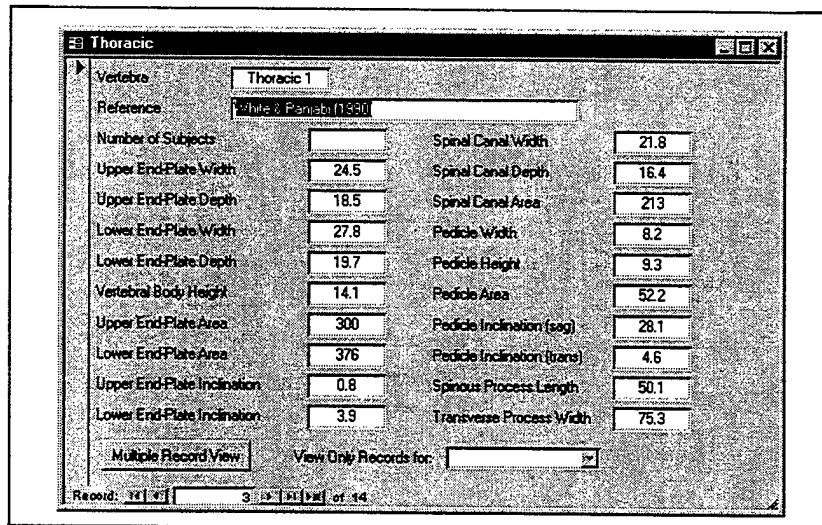
Figure 5-21 Multiple Record View

6.0   HSM Verification, Validation, and Simulations.

BRC addressed the verification and validation of the HSM-PC software during its development. Software verification is the process of confirming that computations and algorithms are coded correctly and yield the expected the result in all instances. BRC verified the algorithms for individual elements by creating simulations of two rigid bodies and a single element, whose response to input could be predicted with hand calculations. Simulations with several rigid bodies and spring elements were also conducted to verify the proper application of Newton's Laws in six degrees of freedom.

Because of the complexity of the fully integrated HSM-PC, no complete model validation was attempted. However, a limited operational validation was done. Operational validation is the classic validation of the model's output against the real-world entity it purports to simulate. Levels of testing from inspection to analytical testing are employed. The activities comprising operational validation are briefly described below. More complete descriptions of the requirements are contained in Simulation Validation[10]. See Figure 6-1 for a summary of the tasks required for operational validation.

Figure 6-1 Operational Validation

- Inspection Tests

    - Delphi Tests
    - Turing Tests
    - Input/Output Relationships Tests
    - Event-sequencing Tests

- Demonstration Tests

    - Animation Tests
    - Fixed Value Tests
    - Simplified Tests
    - Predictive Validation Tests
    - Internally Validity Tests
    - Extreme-condition Tests
    - Limited Standards Tests

- Analytical Tests

    - Predictive Validation
    - Comparison to Test Data
    - Sensitivity Analysis
    - Feedback Loop Anlysis

Inspection Tests

Inspection testing was completed by the Development Team and the Confidence Assessment Team. It involved reviewing the model in a non-rigorous manner. This was primarily done by reviewing the model element models individually and judging their reasonableness and applicability to the task for which they were intended. Output data from the model were evaluated and compared to high-speed film studies of human subjects. We also observed animations from the HSM to judge its responses in time to settling under G and to forcing functions. These observations were vital in assessing how "realistic" the model's behavior was. Several problems were detected at this stage and the model was investigated and debugged to get its responses aligned with the expectations of experienced experts.

Demonstration Tests

Demonstration test illustrating the model's responses as compared to the human head neck and human spine under simplified and easy to understand situations. Primarily these comprised experiments and demonstrations to demonstrate the model's stability under + 1 G and the fact that a normal static posture was achieved by the model. The HSM-PC has stable response under gravity and assumes a reasonable posture. The original HSM was unstable under +1 G. The demonstration tests included animation, static tests, and dynamic tests. The output was compared to hand calculations and simplified model outputs. This testing is continuing as more complex versions of the HSM-PC are completed.

Analytical Tests

In analytical testing comparisons of two and three body models were compared to output to simulations of identical models via MADYMO and ATB.

The following paragraphs describe the testing that has been accomplished to date.

BRC conducted several simulations with the HSM-PC software, described in Section 6.3, to show how the model can be used. A set of "settling" simulations was created that served two purposes: (1) to show the response of the model to gravity, and (2) to provide a stable configuration for other simulations. Because of computational limitations, simulations with the original HSM were not initiated from a settled and stable configuration. Other simulations of vertical and horizontal acceleration pulses were also conducted.

Finally, Section 6.4 describes practical guidelines for using the HSM-PC.

6.1   HSM Verification.

Confirming that calculations are performed correctly in the HSM is the process of software verification. BRC conducted several simulations using portions of the HSM to verify that the software was functioning properly. There were two basic types of verification:

- element verification -- a single element was simulated
- system verification -- multiple elements/bodies were simulated

For the verification of elements, BRC compared HSM equilibrium conditions to hand-calculated equilibrium conditions. The following pages illustrate several examples of element verification. The test system shown in Figure 6-2 was used for most of the simulations.

Verification of rigid body motion and a single spring element can be found in Figure 6-3. A spring with stiffness 10 N/m and damping of 0.0 or 1.0 was used to connect the two bodies, and the results are shown in Figure 6-3. Hand calculations indicate the equilibrium position for the suspended body should be 0.0981 cm below the starting position. Note that the damped system proceeds directly to the equilibrium position, while the system with no damping oscillates about the equilibrium position.

A similar configuration was created for a disc element. Figure 6-4 shows the response of a suspended body and a connecting disc element. The element had an axial stiffness of 10 N/m and damping of 0.1. The disc bending and torsional stiffness were 50 Nm/degree, and the shear value was 10. Figure 6-5 shows the bending and torsional deflection in the suspended rigid body in response to a 0.1 Nm bending and torsional moment on the lower rigid body. Using the equations from Section 4, it can be confirmed that the expected equilibrium torsional deflection is 0.115 degrees, and the expected bending deflection is 0.207 degrees.

Simulations in Figure 6-6 shows the response of the suspended body with an intervening hydrodynamic element. The element has a bulk modulus of 40.0 and damping factor of 0.01. Equations from Section 4 verify that the correct deflection of the suspended body is 0.01 m.

To verify the operation of a plane, the system shown in Figure 6-7 with a single connecting spring was modified to include an upward-moving plane with stiffness of 50 N/m moving at constant velocity of 1.0 m/s (recall that up is the negative direction for HSM!). The two rigid bodies were dropped onto the plane. Figure 6-7 shows the response of both bodies from this simulation. As expected, the bottom rigid body (#2) assumes the velocity of the plane (with a small deflection into the plane), while the upper body (#1) displays a damped elastic response.

Figure 6-2 Model for Verification Tests
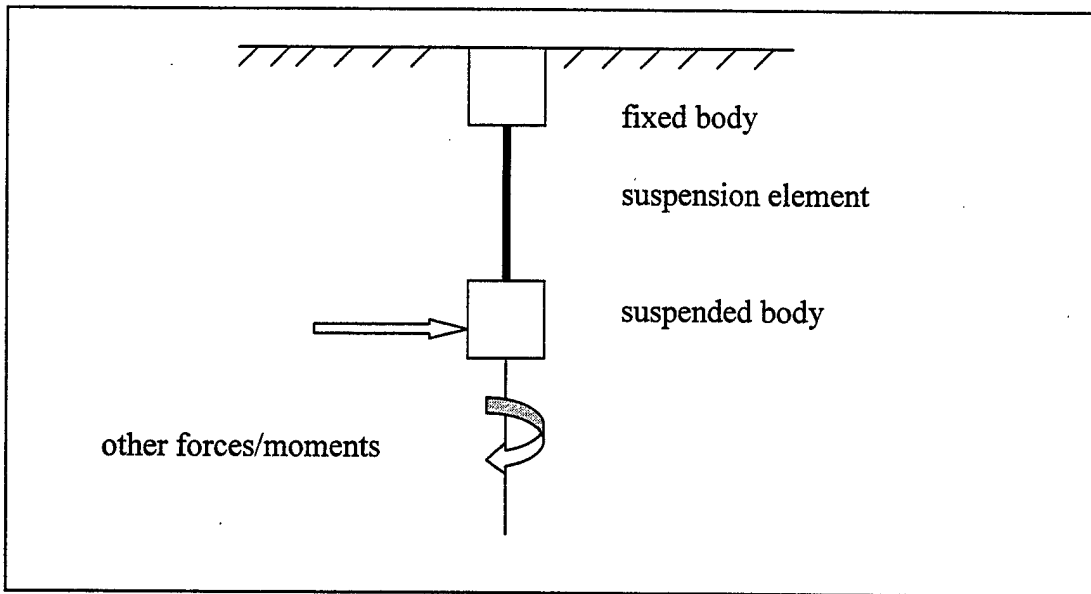
Figure 6-3 Verification of Spring Element
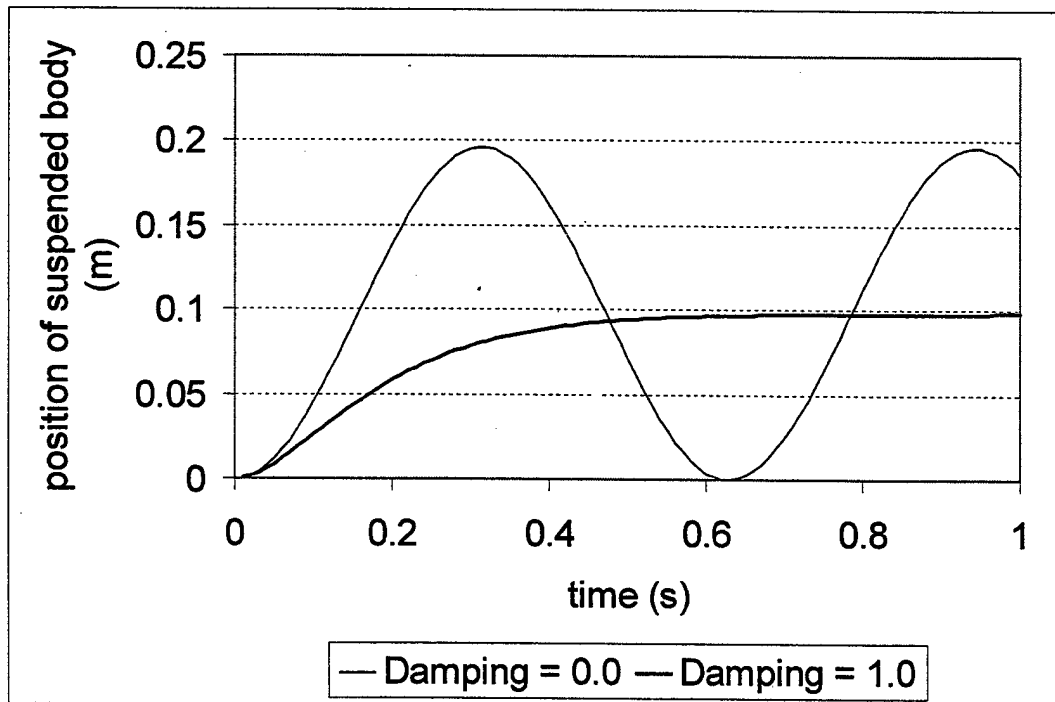
Figure 6-4 Verification of Disc in Tension
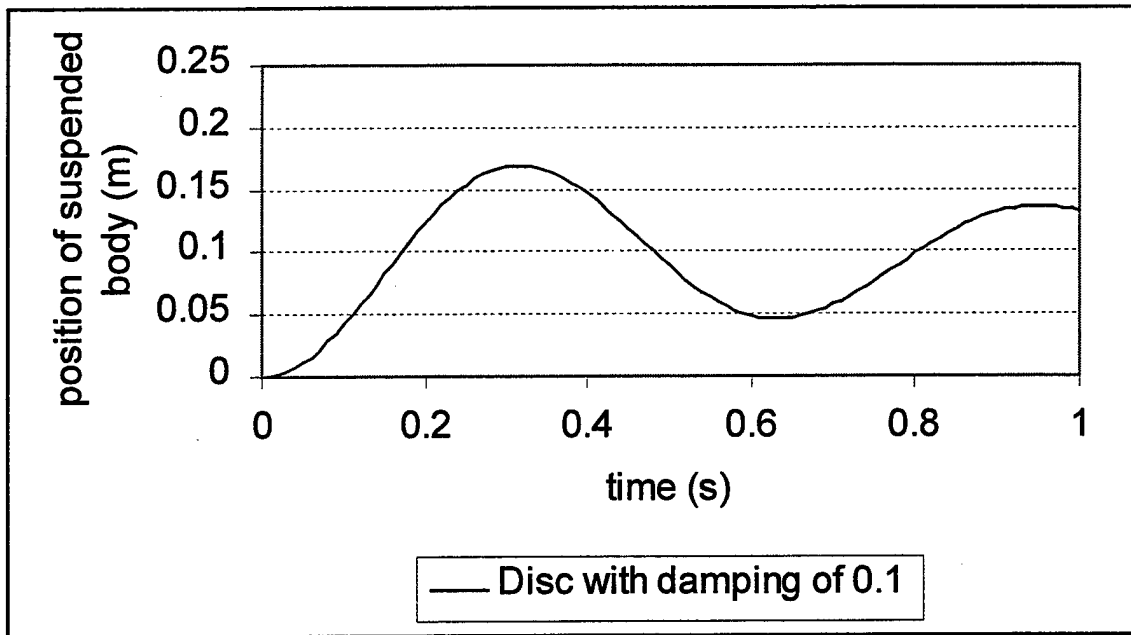
Figure 6-5 Verification of Disc in Bending/Torsion
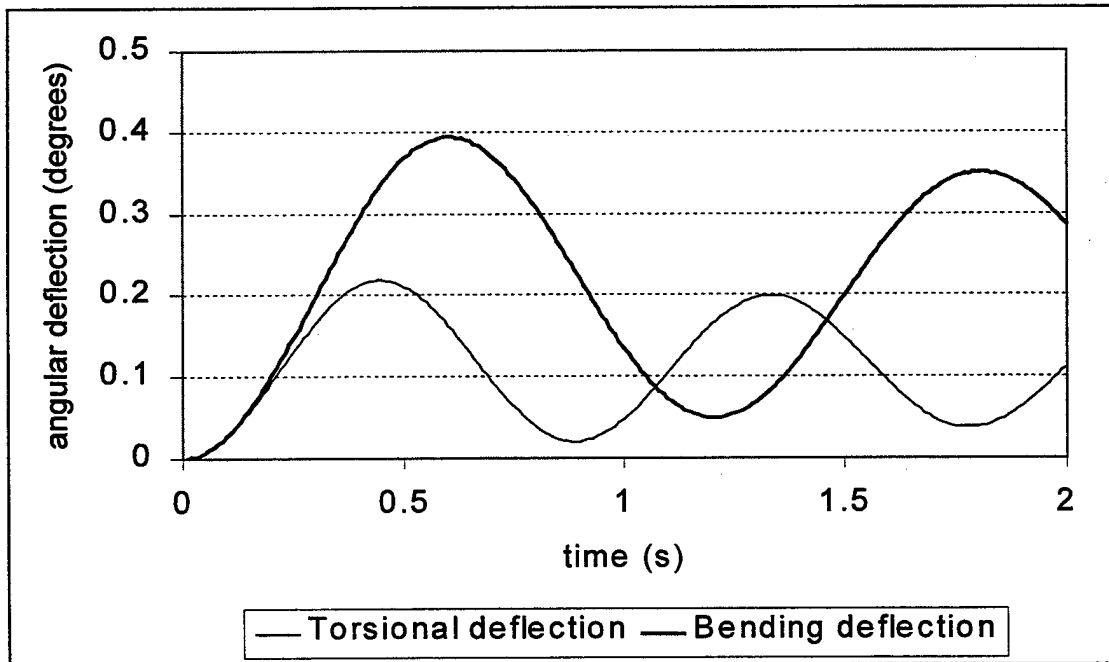
Figure 6-6 Verification of a Hydrodynamic Element
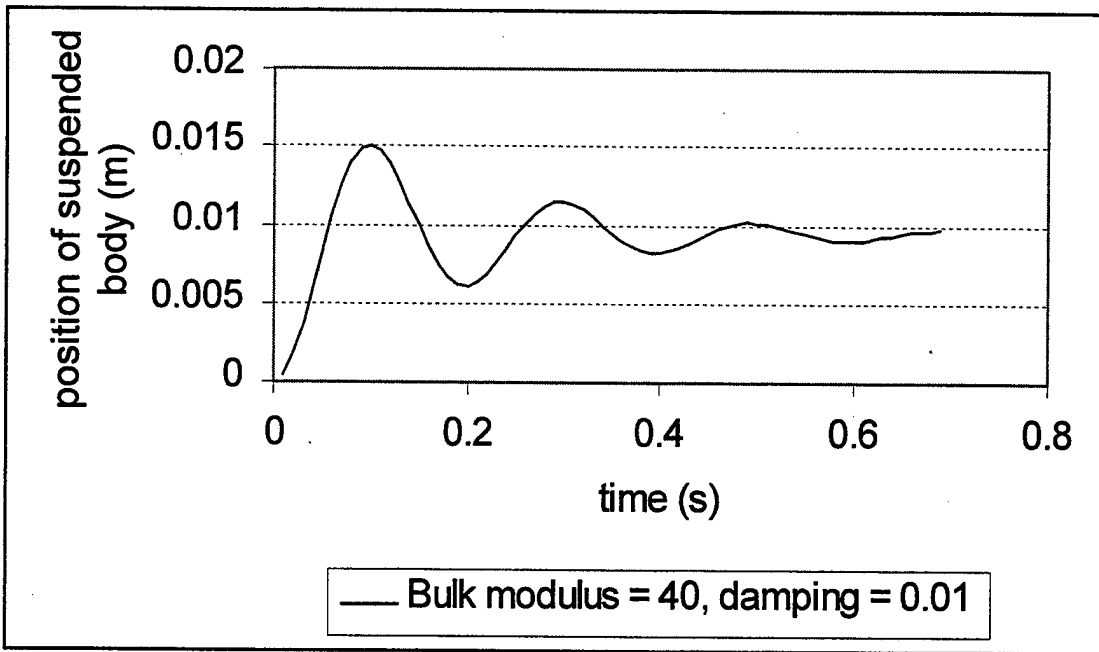
Figure 6-7 Verification of Plane and Body Motion
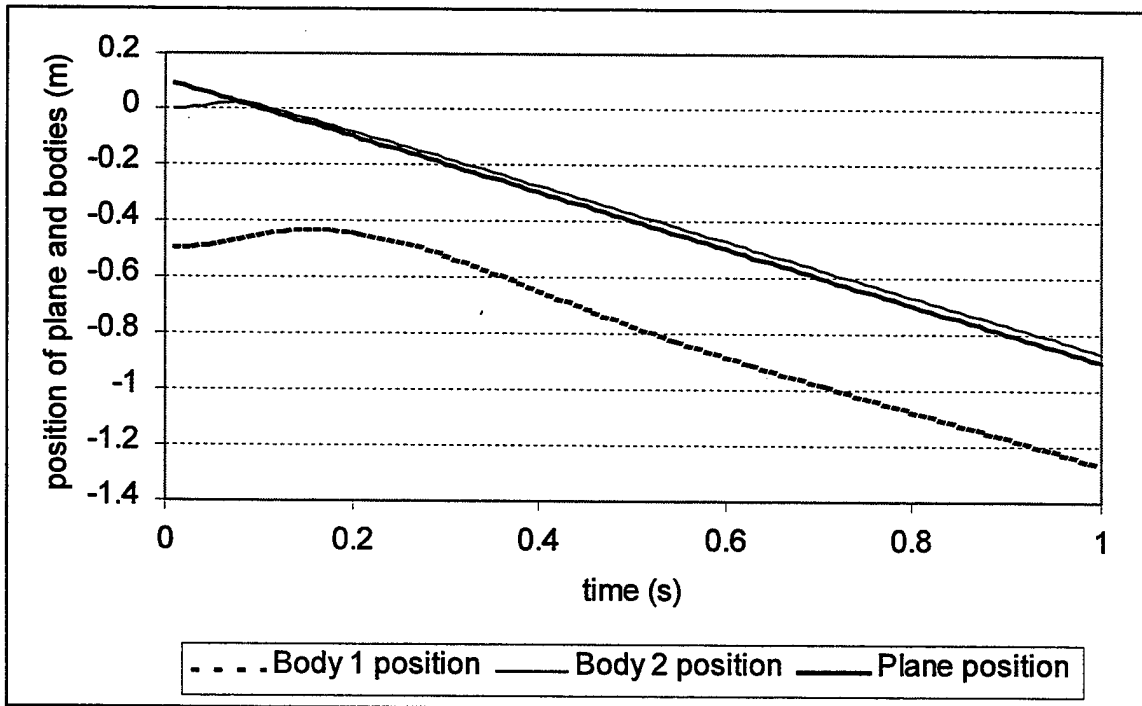
BRC also verified proper operation of the HSM by considering a multi-body, multi-element model. A three-body subsection of the cervical spine was selected for simulation and included spring elements representing ligaments of the neck. This same segment was modeled with a software package called the Articulated Total Body (ATB) software to verify operation of the HSM. The ATB program allows a user to simulate the dynamic behavior of rigid bodies that have been linked together with spring and damper components.

The model represented vertebra C5-C7 of the neck and connecting ligaments. The segment was fixed at C7, with C5 and C6 allowed settling under gravity. The segments were in a non-equilibrium position at the start of the ATB simulation. Throughout the simulation, the segments were allowed to oscillate in a damped free vibration state. As usual in the HSM, the gravity vector was in the vertical direction, along the z-axis, with positive directed downward.

The verification model was constructed using the ATB software and a pre-processing package called Dynaman version 4.0. To construct the model using ATB, the mass and moment of inertia for each of the segments was used. To connect the segments, seven spring/damper combinations were used between each segment, with the appropriate spring constants. The spring force was calculated using a non-linear cubic model. The viscous damping coefficient associated with each spring had to be calculated assuming a damping ration of 0.5, which represents an underdamped condition. In addition, the initial spring displacement at time zero had to be calculated for ATB.

To verify the HSM, the position and orientation of the two free vertebra were computed using both programs. The time interval was from 0 to 1 second, with a time step of 0.01 seconds. Results from the HSM and ATB simulations are shown in Figure 6-8 through 6-12.

Figure 6-8 shows the x, y, and z positions calculated by both models for C6. Figure 6-9 shows the pitch angle for the same vertebra. Note the close agreement of results between the two models. The absolute difference in displacements for this segment did not exceed 0.6 mm and the difference in pitch angles did not exceed 0.25 degrees.

Figure 6-10 shows the x, y, and z positions calculated by both models for C5. The pitch angle for both models is shown in Figure 6-11. Again, there is close agreement of results between the two models. The difference in displacements varied by 0.7 mm and the difference in orientation differed by up to 0.5 degrees. Since the motion of C5 is dependent on C6, errors accumulated in the displacement and pitch of C6 exacerbate the error present in C5.

BRC also conducted six degree-of-freedom verification simulations using the multi-body dynamics software MADYMO early in the Phase II program. As with the ATB, the results of MADYMO verified proper motion of HSM bodies when subjected to gravity or inertial accelerations.

Figure 6-8 Comparison of C6 Motion

Figure 6-8 Comparison of C6 Motion — plot of Position (m) vs Time (sec), with legend: ATB X, ATB Y, ATB Z, HSM X, HSM Y, HSM Z
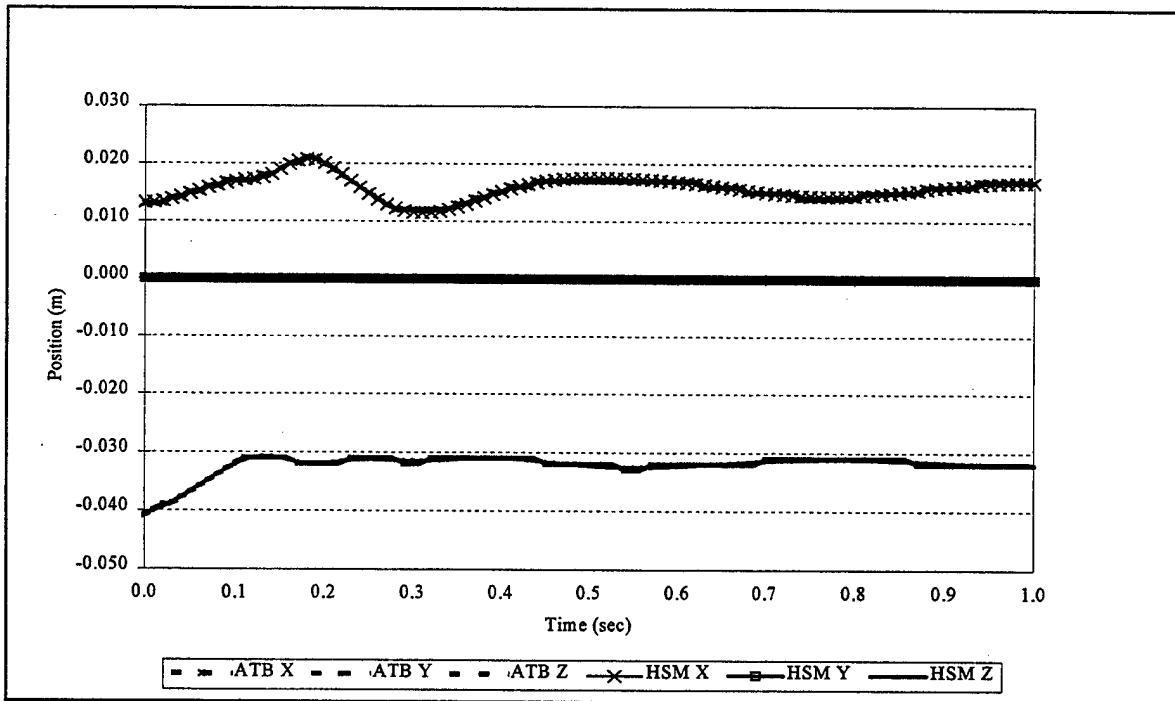
Figure 6-9 Comparison of C6 Orientation

Figure 6-9 Comparison of C6 Orientation — plot of Relative Pitch (deg) vs Time (sec), with legend: ATB, Fortran
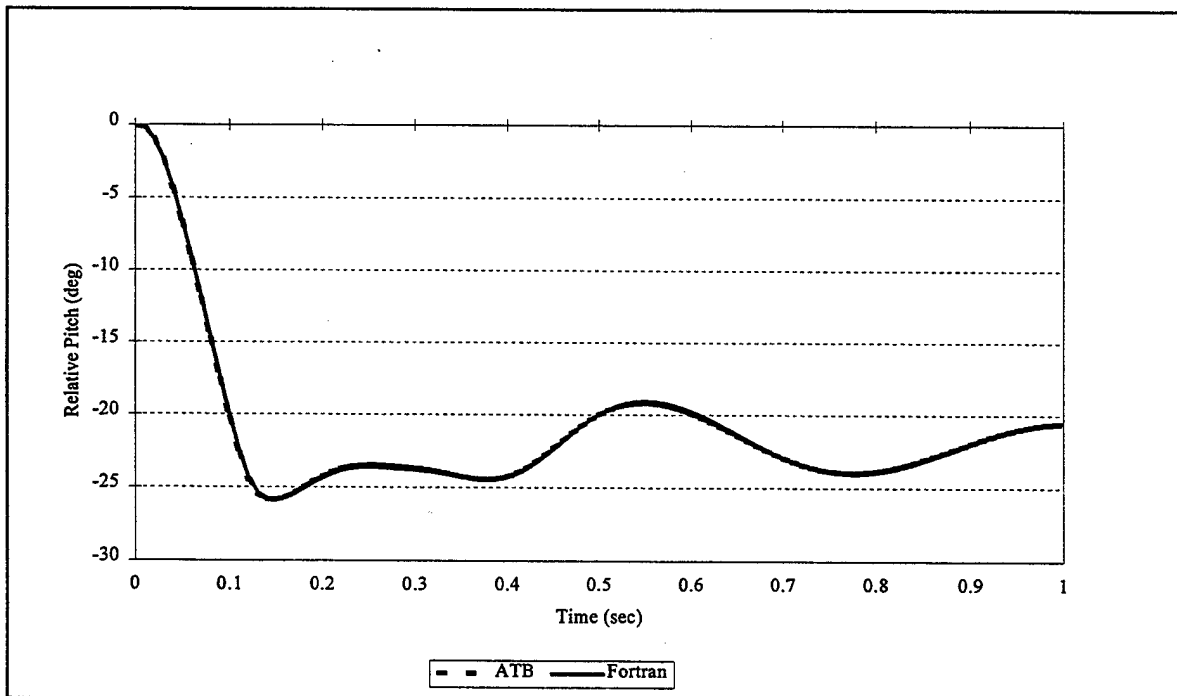
Figure 6-10 Comparison of C5 Motion
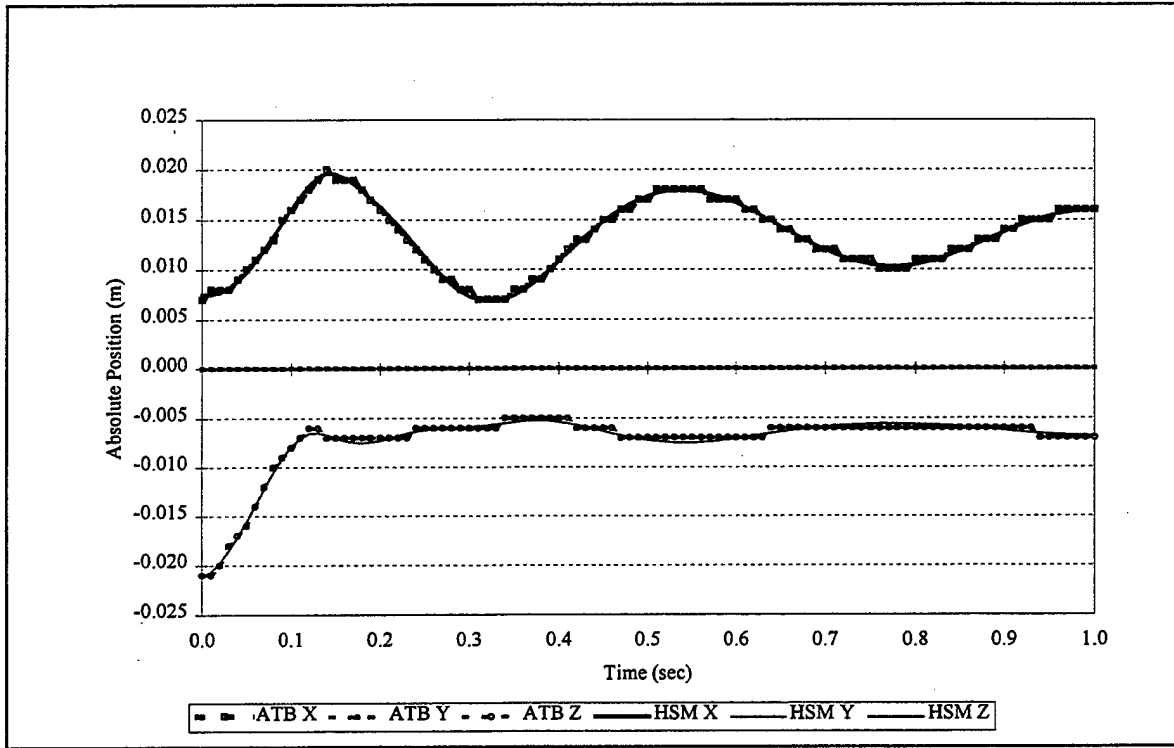
Figure 6-11 Comparison of C5 Orientation
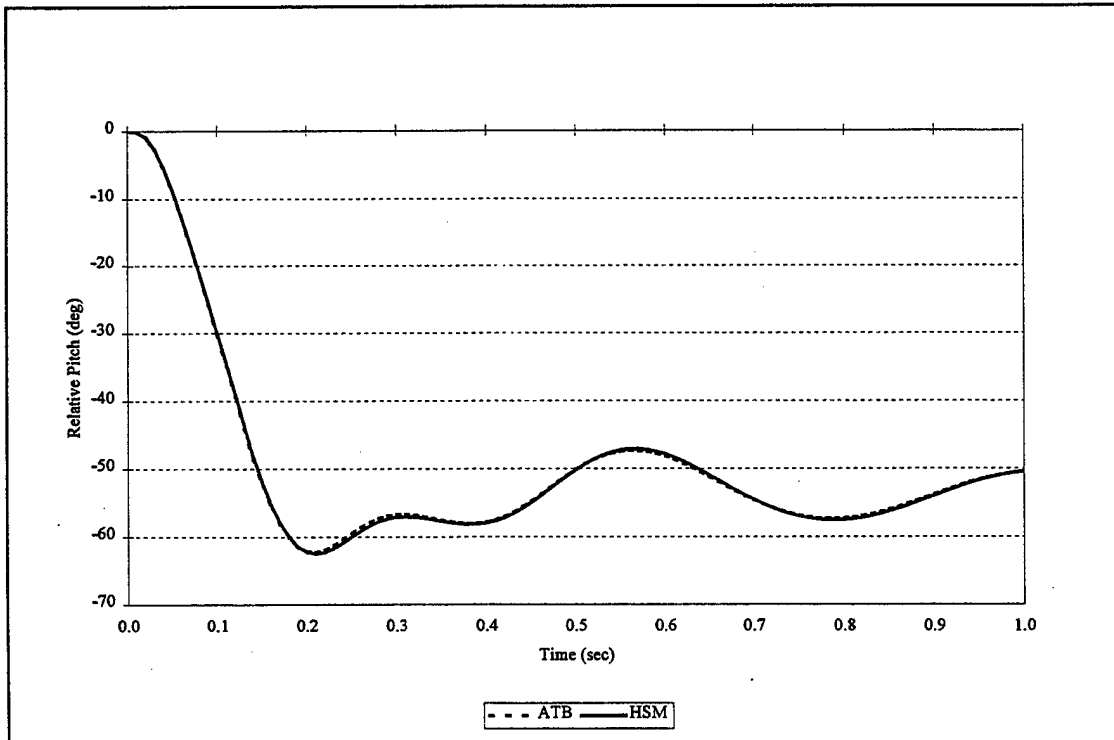
6.2   Validation.

BRC employed a DOD developed methodology described by Knepell and Arango[8] to assist in the validation of the HSM-PC. Their *confidence assessment* methodology provides the necessary framework for controlling the development of a complex simulation, while ensuring the validation of the modeling concept, code verification, documentation, and, most importantly, a systematic evaluation of the model's credibility. Figure 6-12 shows a diagram of the overall scheme (adapted from Figure 2-3 of Reference 10). A full discussion of the confidence assessment methodology is given in the Confidence Assessment Plan in Appendix A.

As shown in Figure 6-12, the entire process of *confidence assessment* is designed to ensure the "real-world problem entity," in this case, the human head and spine, are represented and simulated in the computer model as logically and accurately as possible.

Figure 6-12 Confidence Assessment Methodology

76
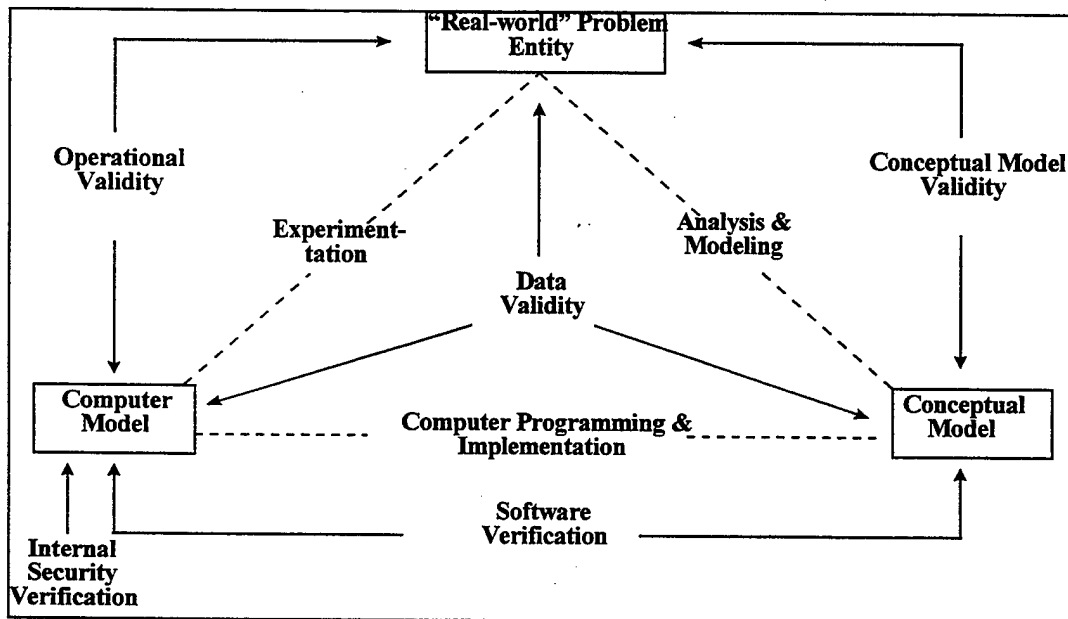
6.3   Simulations.

BRC conducted simulations with the 4 HSM models described in Section 4, containing 9, 15, 26, and 54 rigid bodies. Before running the simulations, the elements of each model were thoroughly reviewed to prevent unexpected rotations or instability. For example, a mistyped connection for a pair of otherwise symmetrical muscles could cause the model to become unstable.

The HCSM and SSM were determined to be stable under gravity. Thus, BRC allowed these models to reach "settled" positions before loading the models. Figures 6-13 and 6-14 shows the HCSM and SSM models in their original and settled positions. The ILSM and HSM-PC models did not appear to be stable under gravity. The ILSM, with no lower spine musculature and no supportive viscera or rib cage, is expected to be unstable under gravity. The benefits of the viscera and rib cage are evident in the HSM-PC, because it appears more stable. Figures 6-15 and 6-16 show 350 ms simulations of both models settling under gravity. The SSM, ILSM, and HSM-PC models were all constrained from motion at the pelvis, while the HCSM was constrained at T1.

Since the HCSM model was the most stable, BRC conducted a series of 2 simulations with the model forced at T1 along the X and Z axes. Both of the simulations were started at the settled position described above. The response of the model to a 5.1 G vertical acceleration is shown in Figure 6-17. Figure 6-18 shows the response of the model to a 5.1 G forward acceleration. Springs were used to attach the environment plane to T1 to avoid having the model loaded at multiple loads by a plane. Note that the vertical and forward acceleration simulations result in sagittal plane motion of the head because the model is symmetric across the sagittal plane.

The stiffnesses of the elements of the SSM model appear to be improperly scaled when considered relative to one another. Figure 6-19 shows a 5.1 G vertical acceleration of the model loaded at the pelvis. Although there is deflection of the cervical spine with respect to the thoracic spine, the individual vertebra of the cervical spine do not exhibit significant relative motion. A plan for addressing the parameters of the model is described in Section 7.

Finally, three simulations were conducted with the SSM, ILSM, and HSM-PC models with a 3 G forward acceleration to compare the motions of the head. All three were started from the unsettled state, and a plane was used to tow the model from the front. Figures 6-20 through 6-22 show the motion of the models. Figure 6-23 shows a comparison of the angular motion of the head over time. As expected, the head on the ILSM and HSM-PC models initially begins to nod forward, while the more stable SSM stays upright. The effect of the HSM-PC ribs and viscera on the orientation of the head can be seen by comparing to the ILSM motion -- the ILSM appears to be subject to high head angular accelerations than the HSM-PC. The downward motion of the rib cage in the HSM-PC simulation is due to the viscera, modeled as springs, pulling downward on the lowest rib as the spine moves rearward relative to the pelvis. Obviously, this behavior is unnatural and must be changed to make the HSM-PC suitable for fore-aft simulations.

Figure 6-13 Head Cervical Spine Model (original and settled positions)
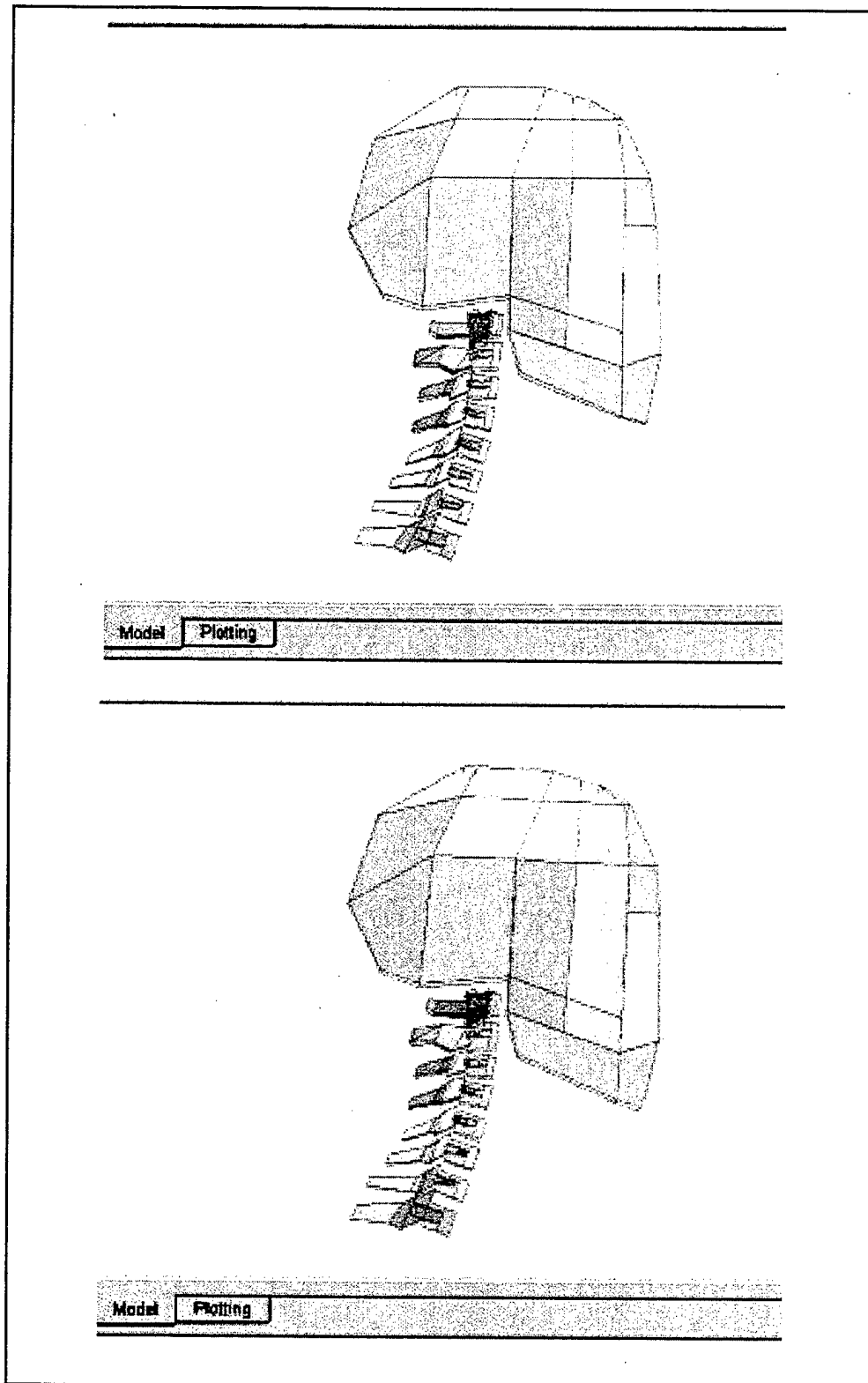
Figure 6-14 SSM Model (original and settled positions)
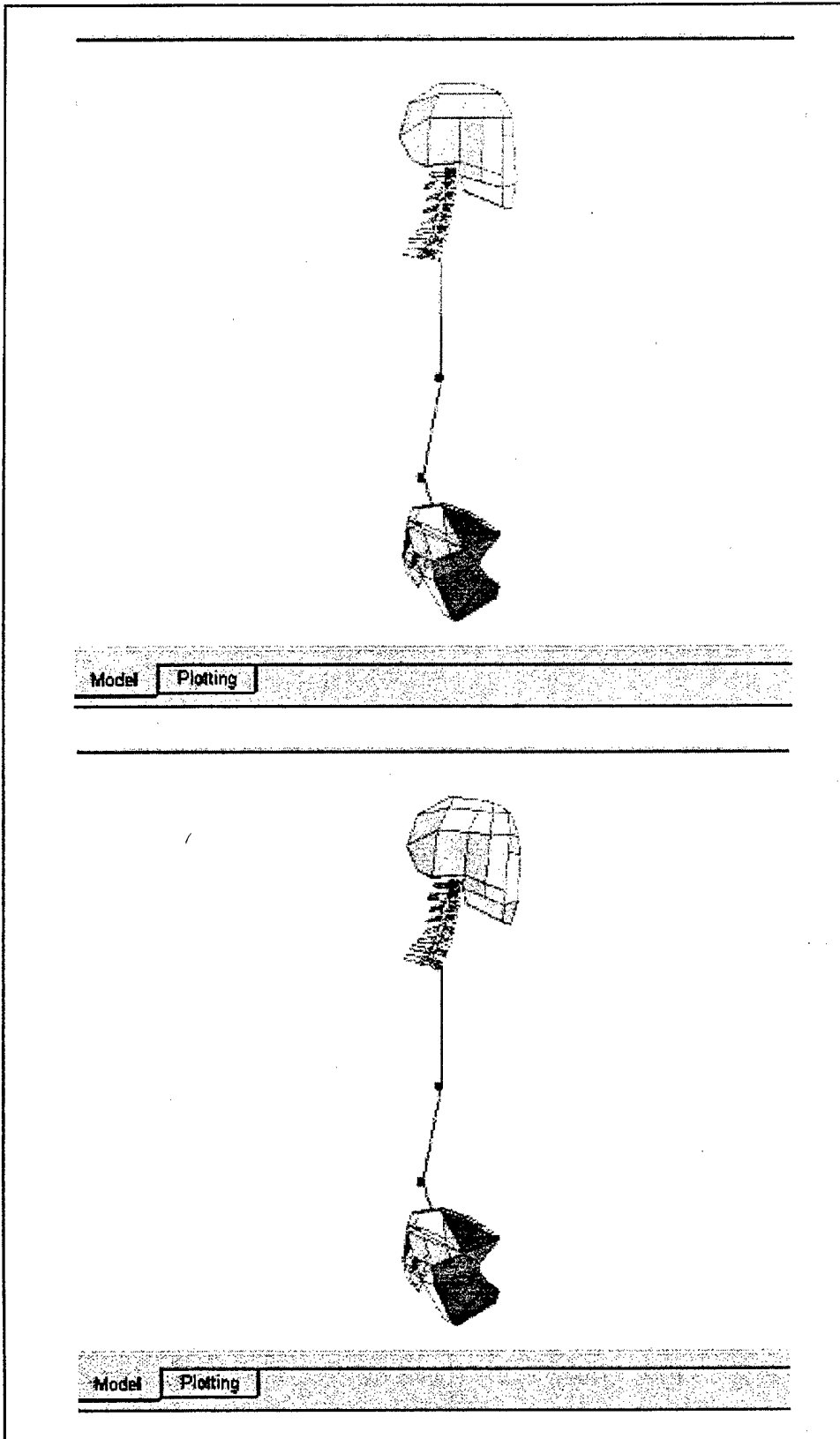
Figure 6-15 ILSM Model Settling Under 1 G
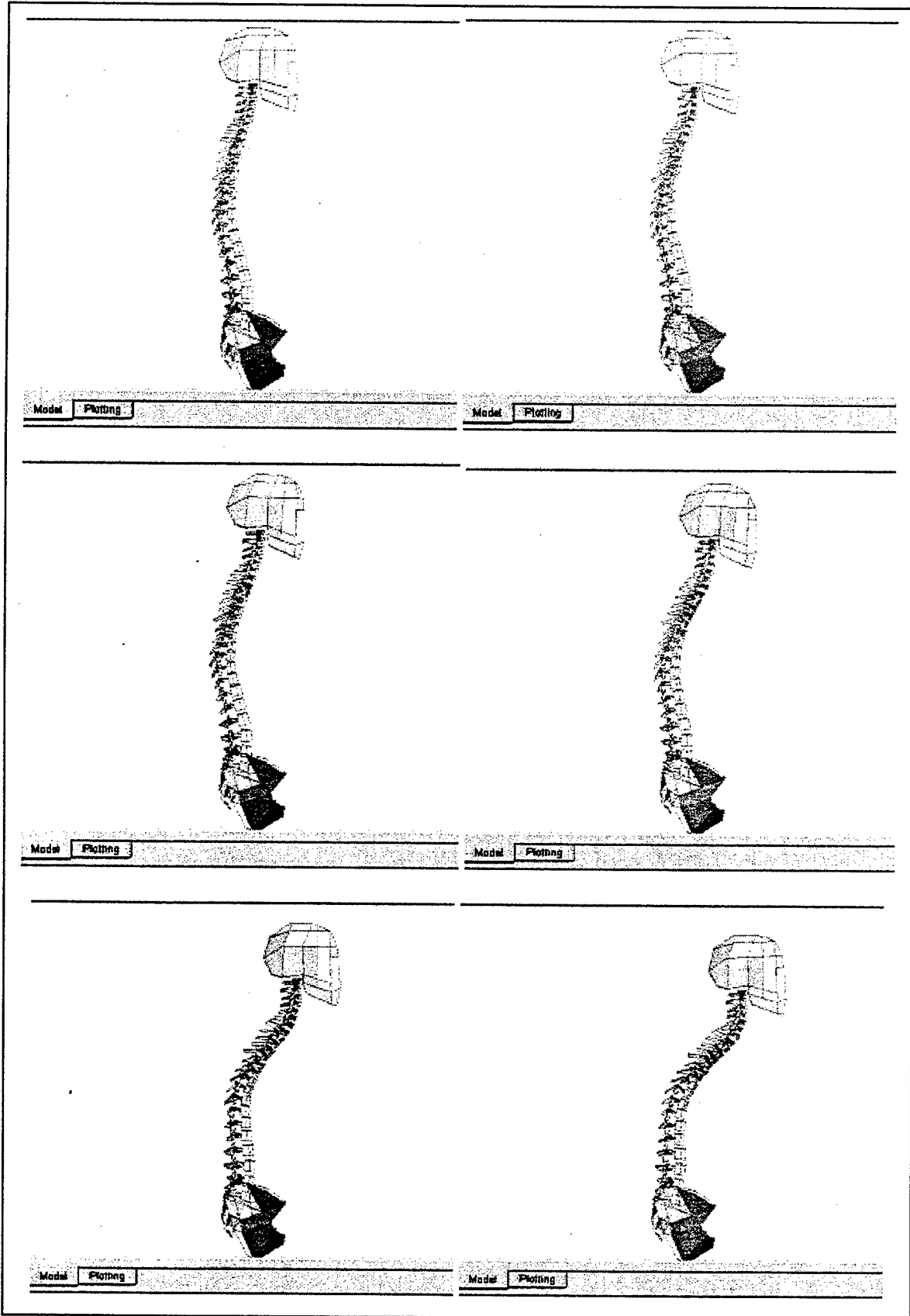
Figure 6-16 HSM-PC Settling Under 1 G
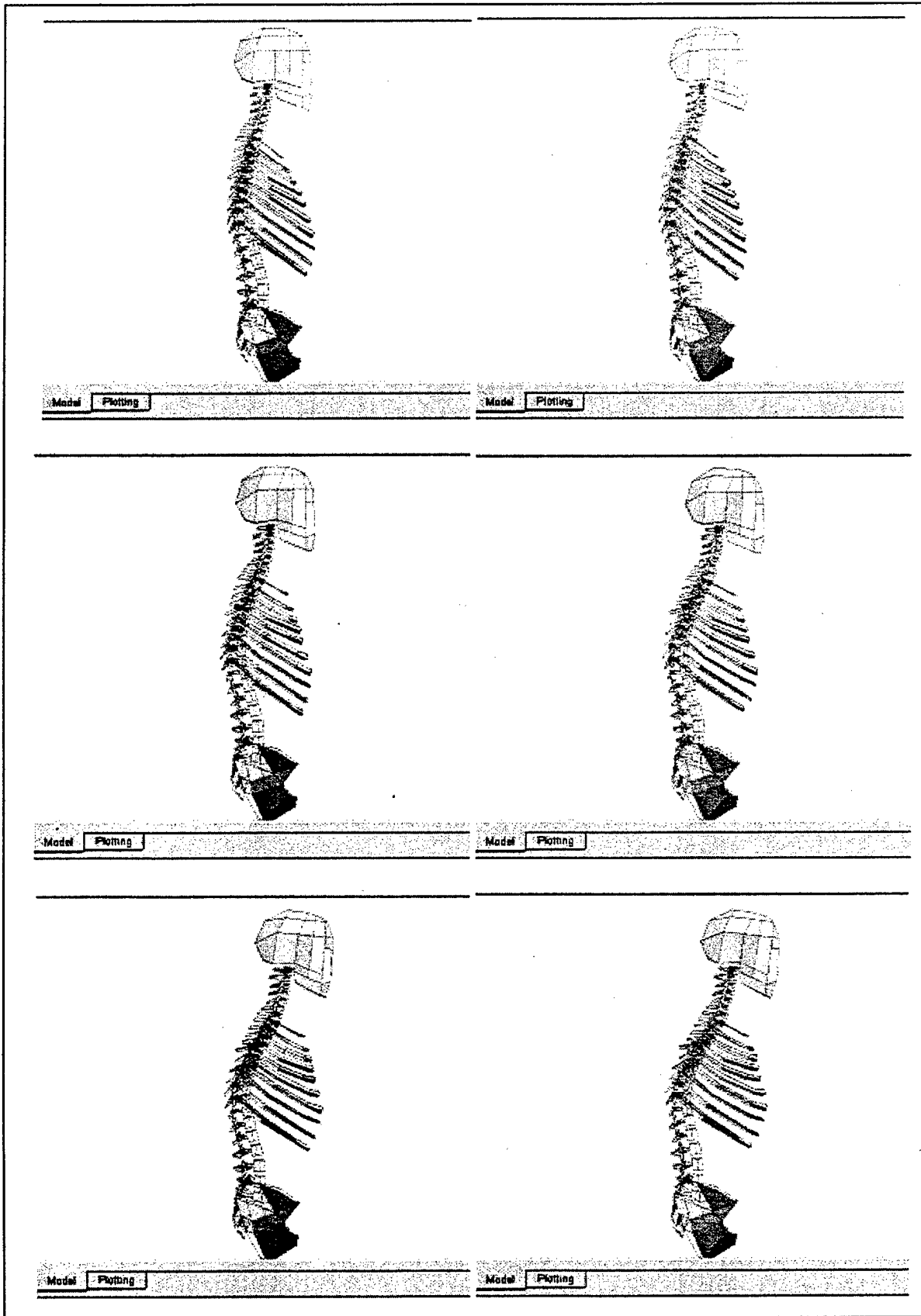
Figure 6-17 HCSM with 5.1 G Vertical Acceleration
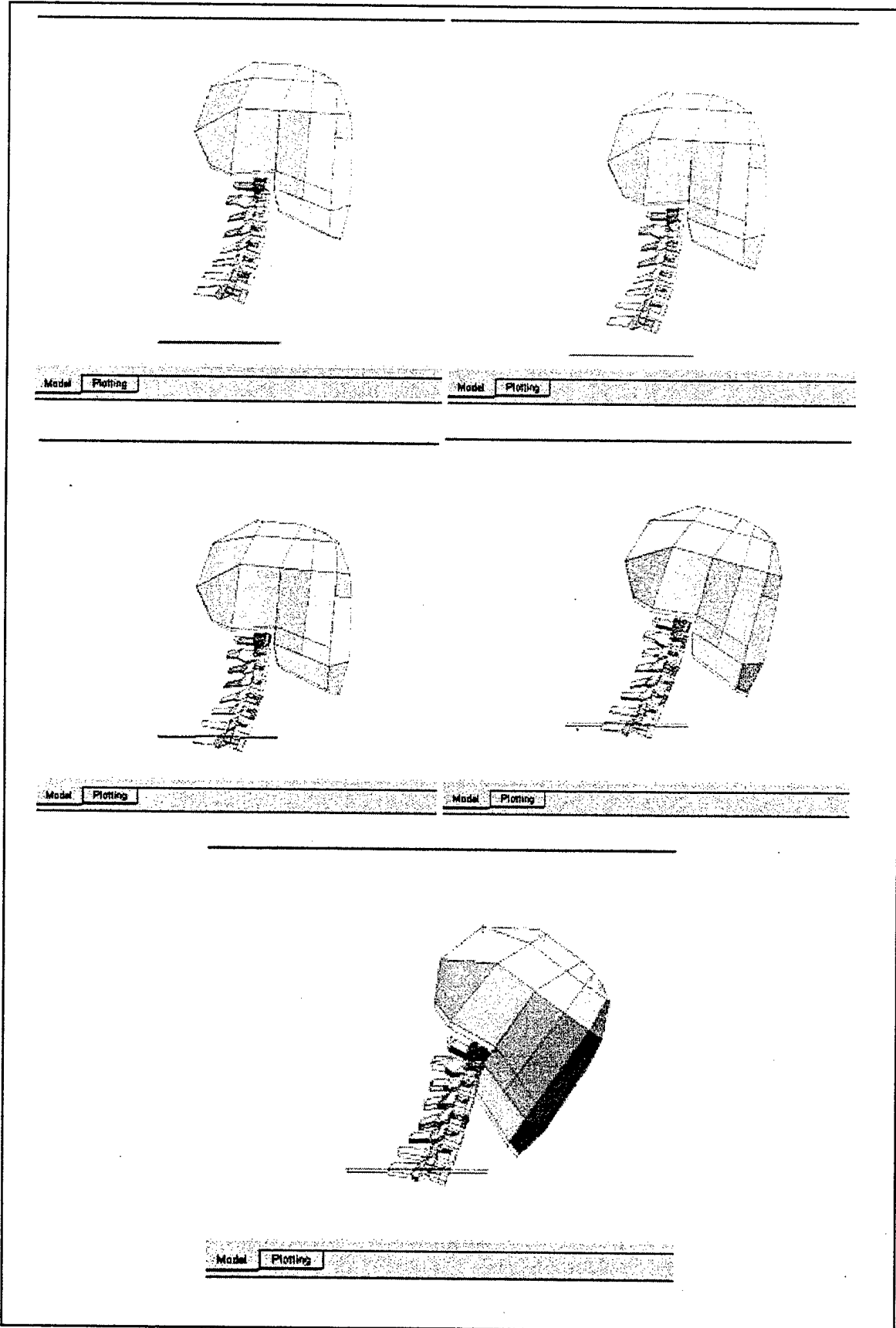
Figure 6-18 HCSM with Horizontal 5.1 G Acceleration
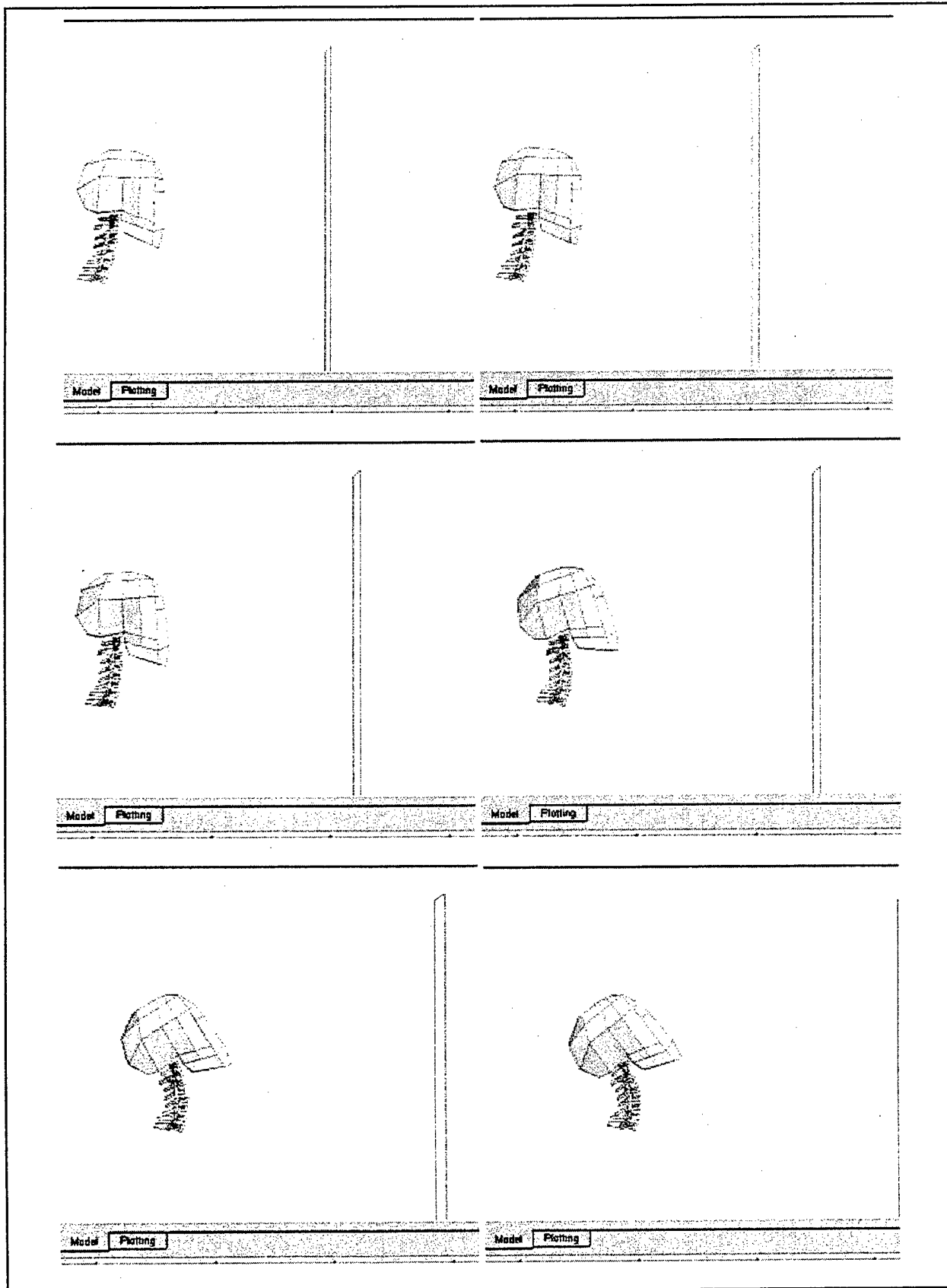
Figure 6-19 SSM with 5.1 G Vertical Acceleration
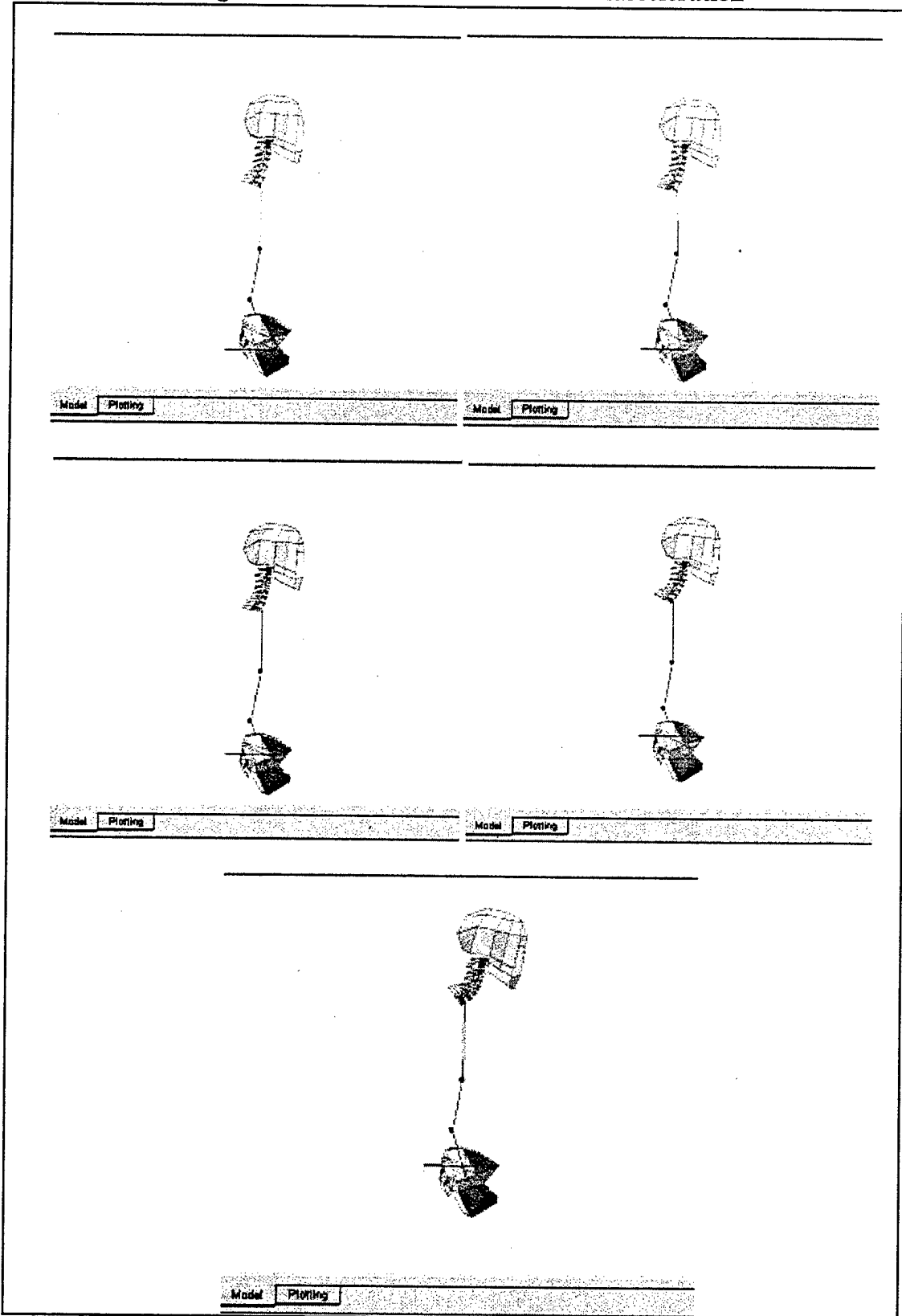
Figure 6-20 SSM with 3 G Forward Acceleration
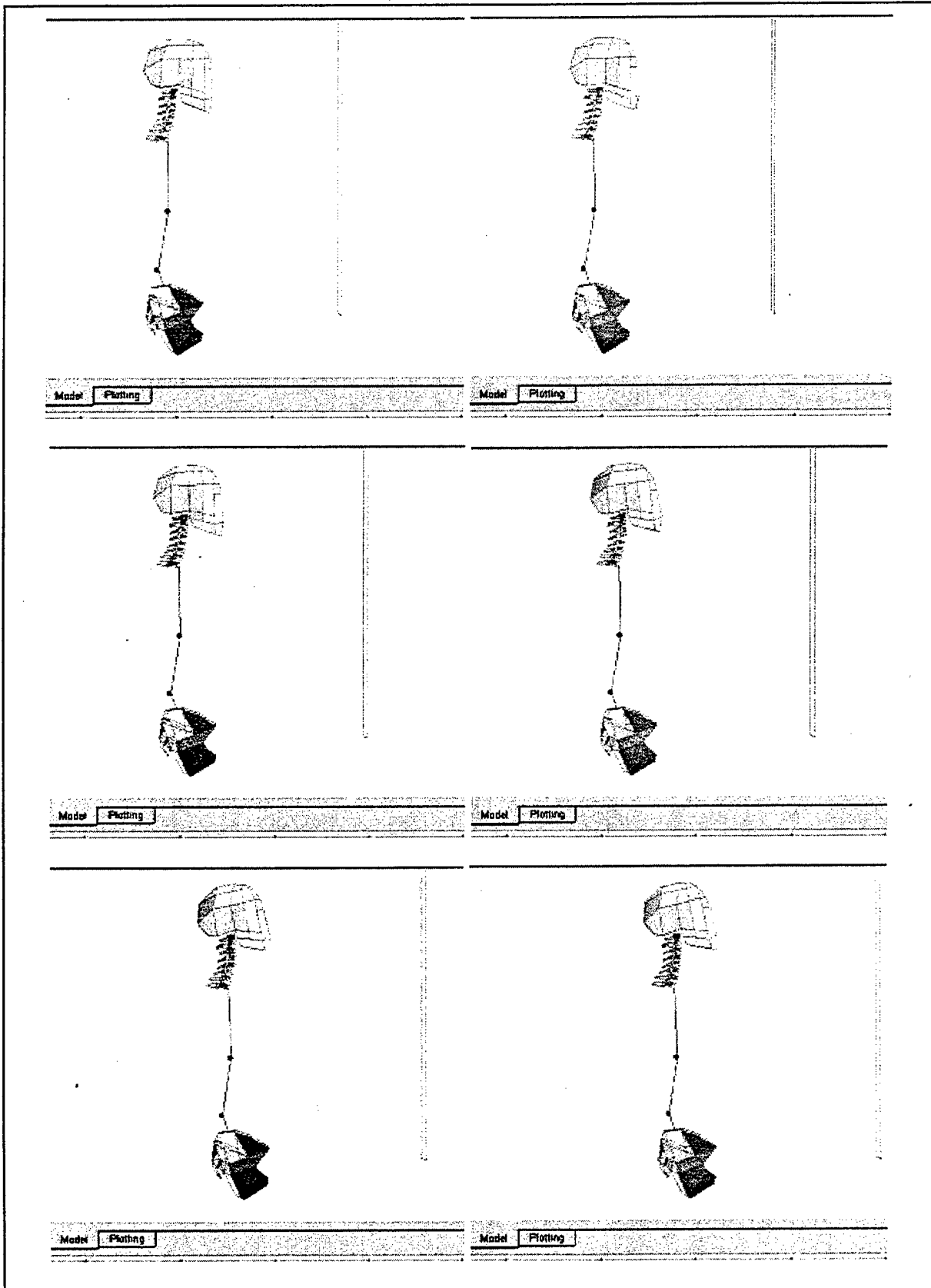
Figure 6-21 ILSM with 3 G Forward Acceleration
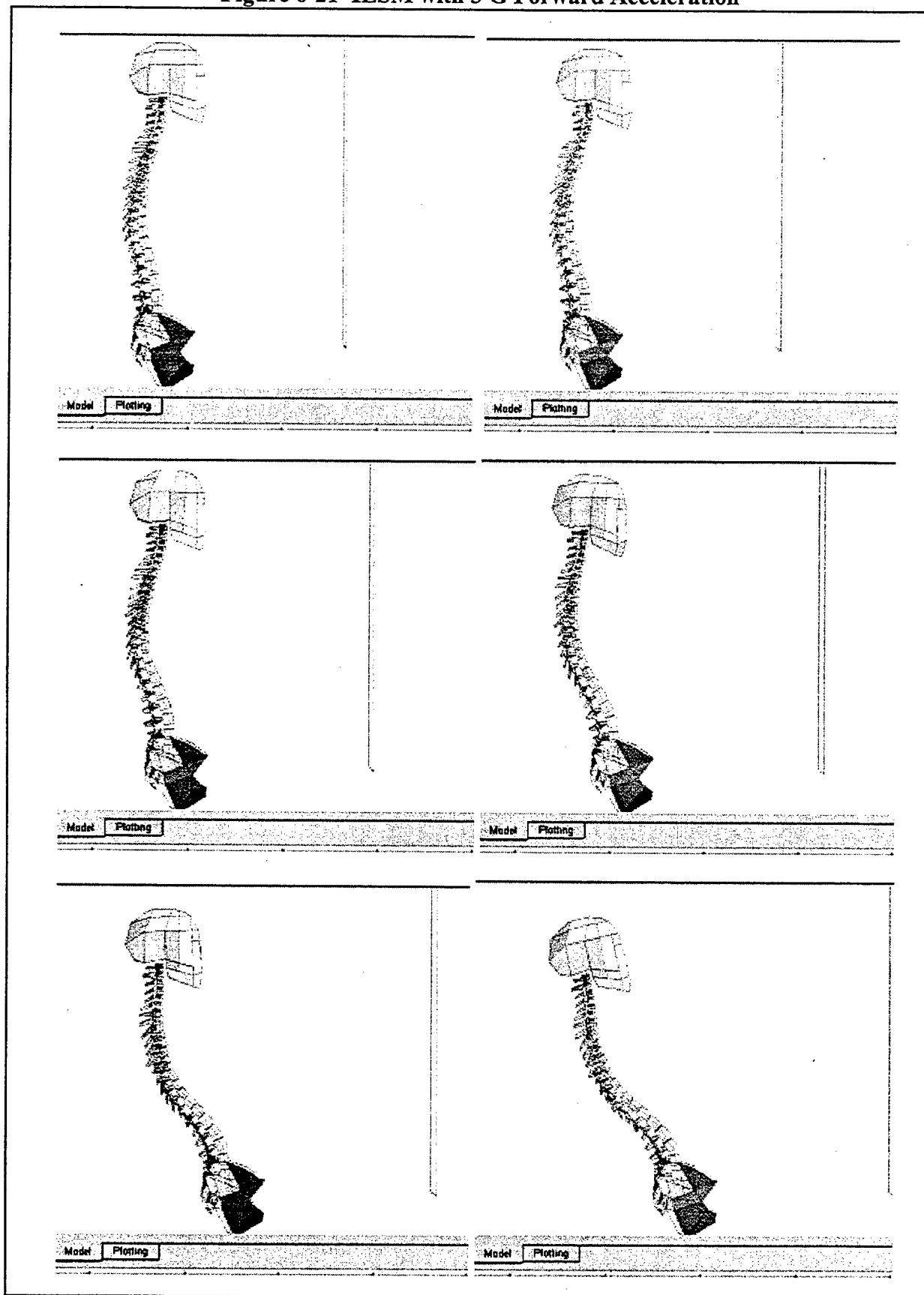
Figure 6-22 HSM-PC with 3 G Forward Acceleration
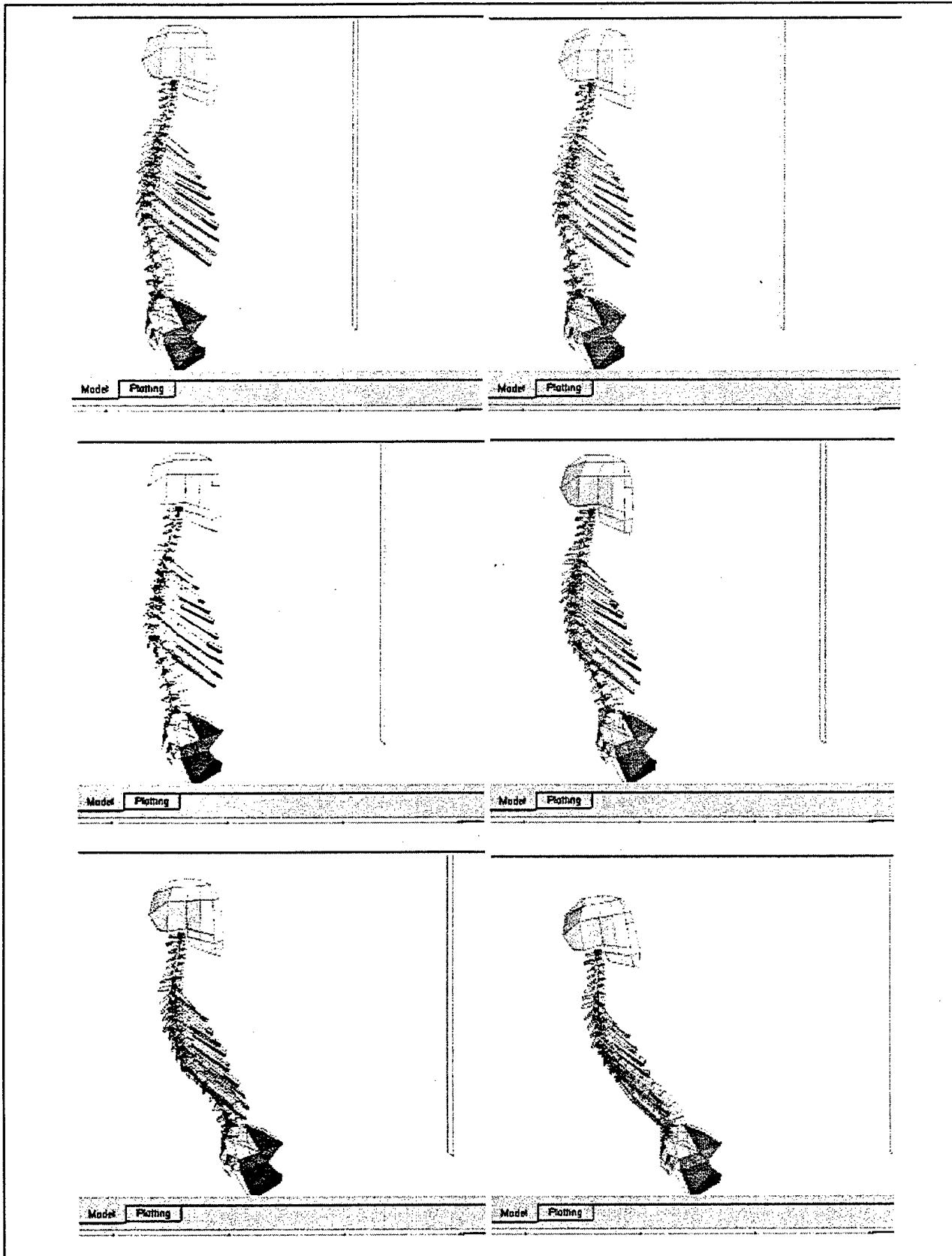
Figure 6-23 Comparison of Head Pitch For 3 Models in Forward Acceleration
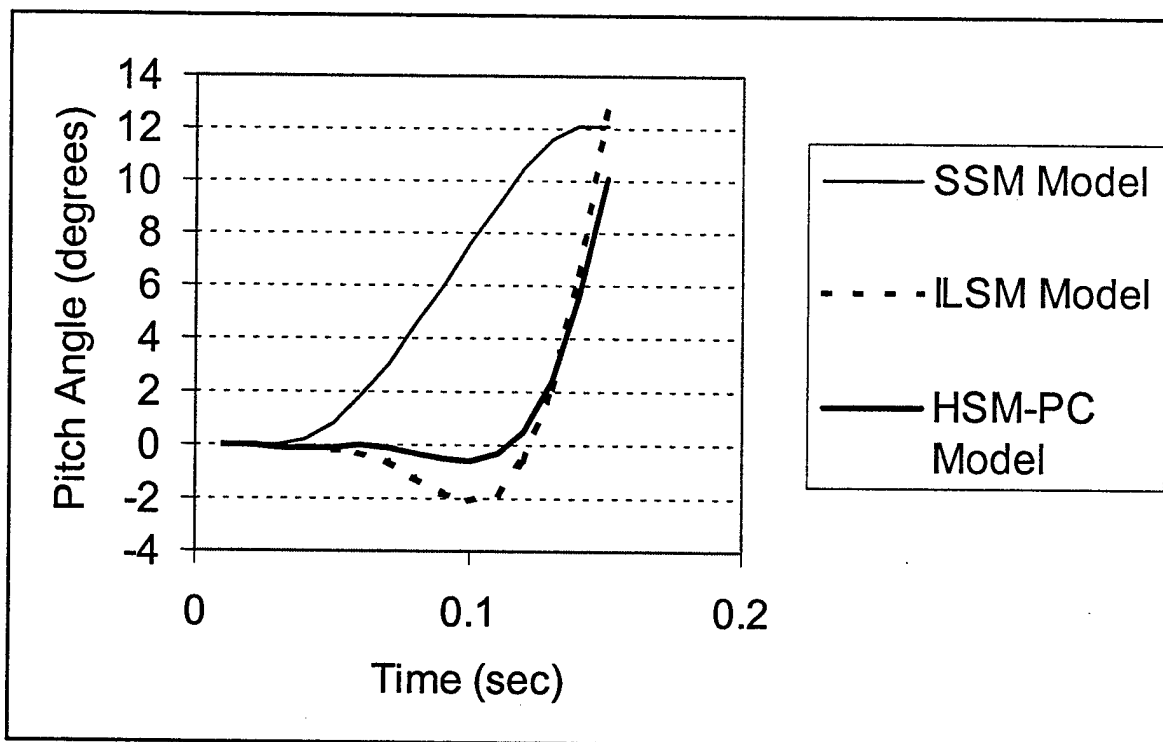
6.4   Creating HSM-PC Simulations.

Because of the complexity of the HSM-PC model and its variants, creating a successful simulation can sometimes be difficult. Following are suggestions for simulations:

- Always start simulations using the "settled" version of that model. If a model is modified, then a new settling simulation must be conducted first, usually of 0.5 seconds duration.
- Simulations with forcing functions should last approximately 100 ms on average. Longer times generally result in the model becoming unstable or unable to find a solution that matches the specified error tolerance.
- Error tolerances should generally be 1e-3 or 1e-4 for optimal results. Lower values slow the simulations down significantly. These values set the allowable single step error for all equations in the numerical integrator as it compares higher and lower precision solution methods to estimate the solution to the model. Higher values result in instability.
- Environment planes should be situated very near the model to begin a simulation. The process for situating the planes can be tedious, because the model does not permit any nodes to start the simulation already penetrating the plane surface. There are two methods for locating a plane: studying the node positions of the model in the biomechanics database, or positioning a plane in a realistic manner in the GUI and then seeing if an error condition is detected by the model. The offending rigid body and node will be specified in the log file in this case.
- Planes should generally have a stiffness of 5000 to 75,000 N/m. Planes stiffer than that may prevent the solution of model motion. Springs in the environment (restraints) can have stiffness of 10,000 N/m with damping 0.25. The best model to use with environment planes and restraints is the HCSM-2 (15-body) model since it models the thoracic and lumbar spine but has fewer elements than the full HSM-PC model.
- Forcing functions specifying high accelerations may prevent the solution of model motion. Try decreasing the accelerations of the planes in the environment.
- Nodes can be created for planes for use with restraint springs. These nodes automatically move with a plane, whether they are on the plane or not.
- The debugging print mode can be used for a simulation so that some output can be obtained in a model that does not solve properly. This mode prints the positions and orientations of the first and last rigid body elements as a function of time.

7.0 Applicability and Recommendations.

In its current state of development the HSM-PC has a largely untested domain of applicability. The HSM-PC currently model presently exists in four variants summarized in Figure 7-1.

Figure 7-1 Constituent Models in Prototype HSM-PC

The Head-Spine Model for the PC
(HSM-PC)

- HCSM – Consists of the head and the complete cervical spine plus T1.
  - 9 Rigid Body Elements
  - 9 Disc Elements
  - 59 Ligament Elements
  - 118 Muscle Elements
  - 14 Articular Facet Elements
- Simplified Spine Model with HCSM (HCSM + SSM)
  - 15 Rigid Body Elements
  - 12 Disc Elements
  - 59 Ligament Elements
  - 7 Visceral Elements
  - 118 Muscle Elements
  - 14 Articular Facet Elements
- Isolated Ligamentous Spine with HCSM (HCSM + ILSM)
  - 26 Rigid Body Elements
  - 41 Disc Elements
  - 125 Ligament Element
  - 7 Visceral Elements
  - 118 Muscle Elements
  - 46 Articular Facet Elements
- HSM-PC – Consists of HCSM-3 plus 10 pairs of rib elements, the sternum, and 7 viscera
  - 53 Rigid Body Elements
  - 41 Disc Elements
  - 125 Ligament Element
  - 7 Visceral Elements
  - 118 Muscle Elements
  - 46 Articular Facet Elements
  - 100 Costovertebral, Intercostal, and Costochondral Elements

Additional research and development work is required to produce a commercially viable version of the HSM-PC. This section presents possible applications for a commercialized version of the HSM-PC together with recommendations for additional development work.

Using the GUI as a development environment a commercialized version of the HSM-PC model can be employed simulate the head neck response to impulsive acceleration such capability could have applications such as those listed below. For each application, a baseline acceleration pulse would be created and the parameters of the HSM-PC adjusted to match the baseline response of human subjects. Subsequent simulations with changed conditions.

Head-mounted Equipment

The military services have and will continue to employ head-mounted displays and night vision aids. The mass and location of added head weight can cause high forces and bending moments to develop in the cervical spine. HCSM simulations could aid in discovering trends in forces and moments could be related to changes in the size and location of head-mounted devices.

Modern Ejection Seats

The HSM-PC could simulate the effects of ejection seat acceleration maneuvers on the posture of the occupant and the loads applied to his spine. This would aid the design of restraint systems, which would minimize the neck and lower spinal loads imposed by maneuvering ejection seats.

Restraint System Design

The HSM-PC could simulate the biomechanics of active positioning of different size occupants and simulate how rapidly a given active restraint system can position an occupant during ejection.

High Agility Cockpit

Future tactical aircraft will be designed to rapidly change directions in flight. Moreover, there is a trend to move more displays and vision aids onto the helmet. The extra head weight imposes higher loads on the neck and upper torso in both high agility maneuvering and in ejection. The HSM-PC could play a role in the design of aircraft seats, restraint systems, and cockpit displays by simulating the head and spine responses to rapidly changing acceleration vectors.

Crash Worthy Seating for Rotary Wing Aircraft

The HSM-PC could be employed to simulate the effects of seating design options on the spinal loads produced by typical crash forces. Also, the effects of seating restraint systems can be simulated to enable design engineers to understand the interaction between restraint system design features and seat design features.

91

Automotive Seating Design

Automobile seating designers are faced with a number of difficult choices seating designs for modern automobiles and light trucks. The HSM-PC could play a key role in simulation of the human occupant's responses and loads during the hazardous high acceleration events.

Seating and Restraint in Other Transportation Systems

The seating and restraint systems provided for occupants of boats, aircraft, heavy trucks, heavy construction vehicles, and buses is often implicated in accidental injuries. The applications of the HSM-PC to recreational and commercial transportation, and construction industries would be similar to the automotive applications noted above.

Improved Simulation of Human Biomechanics

Presently, there are several "human occupant simulator" software programs on the market. The three most popular systems are ATB/CVS, Dynaman™, and MADYMO™. The HSM-PC can extend the domain of the simulation of human biomechanics into the body and give researchers a method of estimating the internal stresses that must be borne by the skeleton during high acceleration and force events. The HSM-PC could give insight into the effect of external loads simulated by the occupant simulator models presently in use.

Amusement Park Ride Design

Some mechanical rides do produce injuries in susceptible passengers. The designers who seek to create ever-increasing "thrills" build rides, which are overly aggressive in the acceleration or rate of change in acceleration imposed on the riders. The HSM-PC could be employed to simulate occupant biomechanical response to acceleration imposed by mechanical rides.

Figure 7-2   Possible HSM-PC Applications

Possible Applications of a Commercialized HSM-PC

- Military Ejection Seat Design
- Restraint System Design
- High Agility Cockpit Design
- Seating Design
- Studies of Helmet/Head-Mounted Devices
- Crashworthiness Studies
- Amusement Park Ride Design

7.1   Business Plan.

This section presents BRC's preliminary Business Plan for commercialization of the HSM-PC technology. There is some repetition of the applications made in Section 7.0, but the Business Plan was drafted as a stand-alone section to be followed as BRC proceeds with commercial development of the HSM-PC. In its present state, the HSM-PC requires further development and validation against known human response to impulsive acceleration. A conservative estimate of the total time for commercialization has been tentatively placed at five years. Parts of the HSM-PC code such as the Head-Cervical Spine Model will be ready much sooner than that. So, it may be possible for BRC to begin commercialization of the HSM-PC well before the current estimate. Progress will, of course, depend on the availability of BRC resources for internal development or alternatively the availability of outside funding to support further development. BRC has plans for internal development, but that route will be much slower than sponsored development simply because of the scarcity of non-revenue resources within BRC.

7.1.1   Company Mission, Vision, and Commercialization Strategy.

<u>Mission Statement</u>

Biodynamic Research Corporation (BRC) will develop, test, and license a computer model of the human head and spine for the PC compatible Windows™ environment. The product will be known as the Head-Spine Model-PC (HSM-PC). The HSM-PC will simulate the biomechanical response of the head and spine to potentially traumatic impact events. The HSM-PC will be the only PC model of its kind in existence and will become an additional tool in the design and analysis of vehicular restraint and seating systems aimed at improving occupant acceleration protection and vehicle crash worthiness. The HSM-PC technology will enable BRC to market software and wider consulting services to crash protection and safety specialists in government, academia, and the commercial automotive, aviation, and amusement park industries.

<u>Vision·Statement</u>

Within five years, the HSM-PC will be available to commercial and government crash protection and safety specialists as a stand-alone software product. For the non-specialist, an occupant modeling and injury prediction service will be available from Injury Sciences Incorporated (ISI), a wholly owned subsidiary licensee of BRC. ISI consultants will employ the HSM-PC as one of its principal tools. The anticipated sources of revenue related to HSM-PC will include consulting fees, licensing fees, software maintenance and upgrade fees, and user training. Customers will exist in U.S. and foreign governments and in the commercial transportation industries. The expected annual revenue stream for HSM-PC activities will be $750,000 generated from sales and fees produced by three full-time professional staff with appropriate administrative and marketing support.

Commercialization Strategy

The PC-based Head Spine Model concept will be developed and its feasibility was established in a Phase I SBIR. The actual model coding and prototype development occurred in a Phase II SBIR sponsored by the USAF. The final development and further validation will be sponsored in a Phase III SBIR by BRC. The completed HSM PC software will be sold under license to ISI. ISI will market the HSM-PC to commercial customers and provide a commercial consulting service based on HSM-PC. ISI will also be able to provide training, maintenance, and software support under contract to the USAF and other users in the Government.

7.1.2 Technology.

Possible Technology Applications (Military)

Conventional Ejection Seat Design. USAF and USN are attempting to integrate women flyers into the combat tactical aircraft pilot career field. Because ejection seats were designed to safely eject men under high speed-low altitude conditions, the acceleration applied to the occupant is near the limit of an adult man's tolerance. Women, being somewhat smaller boned and lighter in weight than men, do not tolerate acceleration at the same levels as men. Therefore, placing women in current tactical aircraft may put them at risk for injury should an ejection be necessary.

The HSM-PC could model the biomechanical responses of various size occupants to ejection accelerations. This information would be useful in redesign or modification of existing ejection systems or in establishing selection criteria for occupants of high acceleration ejection systems.

Maneuvering Ejection Seats. Ejection seats are aerodynamically unstable in the free air stream. Ejection seat designers are currently working on stabilization aids and "self orienting " designs which will "fly the ejection seat and its occupant" to an attitude and airspeed compatible with parachute deployment.

The HSM-PC could simulate the effects of ejection seat acceleration maneuvers on the posture of the occupant and the loads applied to his spine. This would aid the design of restraint systems that would minimize the neck and lower spinal loads imposed by maneuvering ejection seats.

Restraint System Design. Active restraint systems in military aircraft -- they actively position the seat occupant during an ejection sequence. The coupling of the occupant to the seat and the posture imposed by the restraint system critically affect the tolerance of the occupant to a given ejection event.

The HSM-PC could simulate the biomechanics of active positioning of different size occupants and simulate how rapidly a given active restraint system can position an occupant during ejection.

High Agility Cockpit. Future tactical aircraft will be designed to rapidly change directions in flight. New technologies such as: inherently unstable airframes, adaptive control surfaces, thrust vectoring, and stable off-axis attitude control, will shape the future of tactical military aircraft. The cockpit designs necessary to protect the aircrew against rapid onset acceleration and rapid changes in aircraft velocity are being studied in the government and in the aircraft industry. Moreover, there is a trend to move more displays and vision aids onto the helmet. The extra head weight imposes higher loads on the neck and upper torso in both high agility maneuvering and in ejection.

The HSM-PC can play a role in the design of aircraft seats, restraint systems, and cockpit displays by simulating the head and spine responses to rapidly changing acceleration vectors. The nature of the acceleration environment and the response of the head and neck to it may seriously circumscribe design features of head-mounted displays and other devices.

Crash Worthy Seating for Rotary Wing Aircraft. Modern rotary wing aircraft provide "energy management" within the seating structure to dissipate inertial forces during aircraft crashes. The size and stature of potential seat occupants affects the design of the seat and its energy management components.

The HSM-PC could be employed to simulate the effects of seating design options on the spinal loads produced by typical crash forces. Also, the effects of seating restraint systems can be simulated to enable design engineers to understand the interaction between restraint system design features and seat design features.

7.1.3 Possible Technology Applications (Commercial).

Automotive Seating Design. Automobile seating designers are faced with a number of difficult choices in the seat design of modern automobiles and light trucks. The seats must: (1) be comfortable and functional for the driver and passengers during long duration travel; (2) contribute to the restraint of occupants during normal or emergency maneuvering; (3) aid in occupant protection and restraint during crash events; and (4) finally, and significantly, accommodate the entire driving population. For example, seats must damp out vibration and provide soft low friction surfaces for comfort, while providing support and head restraint in rear-end collisions. They must cushion and stabilize the occupants in violent off-road maneuvers while suppressing the effects of large movements in the vehicle's suspension system. These characteristics are difficult to design and optimize since optimal performance of one requirement often means large compromises in achieving good performance. Presently, seats and restraint designs are tested using humans in benign acceleration environments and using manikins in hazardous acceleration or impact tests.

The HSM-PC could play a key role in simulation of the human occupant's responses and loads during the hazardous high acceleration events. Differences in the responses of different size people to the various seat designs could be simulated in representative situations.

Seating and Restraint in Other Transportation Systems. The seating and restraint systems provided for occupants of boats, aircraft, heavy trucks, heavy construction vehicles, and buses is

often implicated in accidental injuries. Most of the compromises in the design of seating and restraint systems for automobiles also must be made by designers of vehicles for these modes of transportation. The applications of the HSM-PC to recreational and commercial transportation, and construction industries would be identical to the automotive applications noted above.

Improved Simulation of Human Biomechanics. Presently, there are several "human occupant simulator" software programs on the market. The three most popular systems are ATB/CVS, Dynaman™, and MADYMO™. All of these systems simulate a humanoid manikin in a dynamic environment by representing the manikin by "ellipsoid" segments connected by user-defined joints. These models can depict the initial responses and loads imposed on the joints and *outside* surfaces of the body.

The HSM-PC can extend the domain of the simulation of human biomechanics into the body and give researchers a method of estimating the internal stresses that must be borne by the skeleton during high acceleration and force events. The HSM-PC could give insight into the effect of external loads simulated by the occupant simulator models presently in use.

Amusement Park Ride Design. Presently, the amusement and theme park industry is a target of litigation related to alleged personal injuries sustained by riders of mechanical rides. In general, the designers of such systems have little or no background in occupant tolerance, occupant protection, or the biomechanics of acceleration-related injuries. In fact, some mechanical rides do produce injuries in susceptible passengers. The designers who seek to create ever-increasing "thrills" build rides that are overly aggressive in the acceleration or rate of change in acceleration imposed on the riders. Despite warnings and disclaimers, suits are filed and won against the owner-operators of the rides.

The HSM-PC could be employed to simulate occupant biomechanical response to acceleration imposed by mechanical rides. The design of the ride itself, passenger compartments and seating, as will as restraint systems could be optimized to create the desired entertainment effects without exposing the "customers" to unduly harsh vibration or acceleration environments. The effects of the ride on unusually small or large riders could be more carefully studied. Moreover, the known deterioration of skeletal and muscular strength with age could be studied as a risk factor for injury. Improved rider screening methods could be employed to reduce the risk of injuries to susceptible prospective rider. This is an underlined market.

7.1.4 Customer Feedback on Selected Applications.

Conventional Ejection Seat Design. The HSM-PC could model the biomechanical responses of various size occupants to ejection accelerations. This information would be useful in redesign or modification of existing ejection systems or in establishing selection criteria for occupants of high acceleration ejection systems. This is an USAF application suggested by Dr. Louise Obergefell, which we believe would also have application in the USN. Their escape system design problems are similar, but the markets are independent because there are different escape systems in Air Force and Navy aircraft.

96

Name and Address of Potential Customer Contacted

Dr. Louise Obergefell
AFRL/HEPA
Wright-Patterson AFB, OH 45433

How Customer Presently Fills this Need

The need is partially unfilled at present. External load measurements but at present there is no non-invasive method of estimating the *internal* motions and loading of the spine. Dr. Obergefell presently employs the ATB/CVS computer simulation to estimate the acceleration profile of a conventional ejection seat containing a small human occupant sized to represent a female aircrew member. The ATB/CVS model yields the acceleration profile and estimates of the *external* loading applied to the seat occupant. The program also provides information on the whole body response to the applied acceleration. The deficiency in this approach is that limited *internal* biomechanical forces are calculated by ATB/CVS nor can the injury potential of the acceleration profile be easily assessed.

Degree of Interest in the New Solution

Dr. Obergefell has expressed a keen interest in the development of the HSM-PC model. The predecessor of the Human Effectiveness Directorate of the USAF Research Laboratory (AFRL/HE) formerly, The Armstrong Laboratory (AL), developed the mainframe version of HSM. Unfortunately, there are presently no experts at HE who are familiar with how to use or apply the HSM. Its potential utility is recognized and the HE desires a replacement tool written in modern computer languages for the PC environment.

Basis for Purchase Decision.

The USAF and USN require research and development studies to support the safety of integrating women into the tactical fighter combat pilot career field. Despite the social desirability for such a change, there are genuine concerns for the safety of women pilots flying in aircraft designed for an exclusively male pilot population. Studies are presently underway to try to quantify the risks. However, there are limited tools available to model the biomechanical effects and injury potential of short duration accelerations produced by existing ejection systems. The expense of manikin testing is very high, upwards of $1 million per ejection test. Live human testing cannot be done. Every effort is being made to make the best-informed decision related to the safety of future women pilots.

The HSM-PC would be a valuable addition to the existing research and development tools. Even a partially validated HSM-PC model would be valuable to the R&D community in making such decisions now and into the future. Because the development of the HSM-PC was sponsored by the USAF, US Government agencies would have an unlimited right to use or modify the HSM-PC code without payment of royalties. Therefore the decision to use the model would be based simply on its ability to contribute positively to the research information available. Key features are highlighted below.

Figure 7-3 Feature, Advantage, and Benefit of this Technology

Feature	Advantage (supplier)	Benefit (customer)
PC Hosted Software with Graphical User Interface (GUI) eases problem setup and analysis of results	large numbers of potential users, GUI makes a complex problem easier for non-experts to understand and use	Inexpensive platform reduces expense of ownership non-experts can employ in problem setup and iterative solution of complex problems previously requiring experienced experts
Dynamic 3-D Force and Motion Analysis with well defined Domain of Applicability	simulation verification and validation and confidence methodologies employed during development	user understands which scenarios have been validated against actual human data
Modern, Portable Language	portability of code	ease of re-hosting software on faster machines or in client server environments
Anatomic representation of internal structures	model estimates internal motion and skeletal loads	injury risks of high acceleration/force environments can be more fully understood
Synergism with existing occupant simulation tools	works with existing occupant models to extend their domain to internal structures	can leverage results of previous and new modeling experiments to yield more information related to injury potential of modeled events.
Similar applications in both civilian and military environments	potential market includes both US government and non-US government agencies, academia and industry.	application by many users to common and different scenarios builds confidence speeds development of new features and validation data

7.1.5 Market.

Conceptual Definition, Market Segmentation, and Market Niche: In concept, the HSM-PC provides a third leg to BRC's existing consulting service. A validated HSM-PC code would be of value to practitioners in the accident reconstruction, transportation, and amusement park industries. Because of the long lead-time to commercialization BRC has not undertaken extensive market analysis.

7.1.6 Time Line for Bringing Technology from R&D to:

a. **Required Military Standard.** The original HSM software development program has essentially defined the Military Standard for the HSM-PC. Thus, the time line for bringing the HSM-PC to the required military standard is somewhat shorter than the commercialization time line, say in the order of two or three years.

b. **Standard for Private-Sector Customer.** The time line for developing the private sector HSM-PC project follows the Phase II program Work Plan, but continues into a Phase III SBIR program internally funded by BRC. The work required to extend the Phase II product to the private-sector is work devoted to further model development work and partial validation of the HSM-PC code in applications of interest to the private-sector. Emphasis will be placed on automotive applications related to seating and restraint system safety and on amusement and theme park applications related to mechanical ride safety.

7.1.7 Intellectual Property Strategy.

The US Government and all its agencies will have royalty free use of the HSM-PC source code. Therefore, it will be impossible to protect the source from use, modification, and reproduction by the US Government and its agencies. However, as the actual code will be created, owned, maintained, and improved independently by BRC, it will be protected by software copyright for applications outside the US Government.

Consulting service based on HSM-PC will be marketed to amusement and theme parks and roller coaster designers to ensure that their roller coasters and mechanical rides are safe. By providing exciting and safe rides to their customers, amusement and theme parks can increase their attendance, profits and decrease injury related litigation and insurance expenses. Presently, there are no similar products or services existing in the market.

7.2   Recommendations.

BRC has created a development roadmap that is summarized in Figure 7-4. The plan calls for the HCSM to be improved and validated first, before proceeding to more complex models. The HCSM was the most stable of the models, probably because of the inclusion of neck muscles. It represents a good starting point for HSM-PC improvements. The first step will be to carefully audit the biomechanical and geometric data in the model. A muscle model with position feedback would assist in positioning the model before simulations. Constraining the relative motion between vertebra and other elements would insure that motion was realistic.

With a validated HCSM, lumped lower spine bodies and elements will be added to obtain a simple complete model. The general response of this model will be validated before proceeding to the full HSM-PC.

Figure 7-5 presents a summary of the recommended improvements that will be required to produce a commercially viable HSM-PC model. It is estimated that the work will require approximately 3 man-years of technical and computer programmer effort and 2 man-years of marketing and sales manpower to make the first commercial placements. Beyond that there will be requirements for technical support and additional technical and market development.

Figure 7-4. BRC Development Roadmap

HCSM

1. Audit biomechanical and geometric data for HCSM model.
2. Incorporate position feedback loop into muscle model.
3. Improve intervertebral disc model.
4. Add relative motion constraints.
5. Validate response of HCSM.

SSM

6. Audit lumped thoracic, lumbar, and sacral vertebra.
7. Audit lumped lower spine connective elements.
8. Add relative motion constraints.
9. Validate response of HCSM-2.

HSM-PC

10. Audit thoracic, lumbar, sacral biomechanical and anatomical data.
11. Audit rib biomechanical and anatomical data.
12. Model lower spine musculature.
13. Create lower spine hydrodynamic facets, viscera, and muscles.
14. Add relative motion constraints.
15. Validate response of HSM-PC.

Figure 7-5 Recommendations

Model Configurations

- Add full complement of muscles to the HSM. The original model and HSM-PC only include muscles of the cervical spine.
- Find element properties that can be used in all HSM configurations. The original model and the prototype HSM-PC use different properties for some elements in different configurations. Confirm that the units of element parameters are consistent.

Model Elements

- Improve the muscle model to allow the model to maintain different starting positions or postures.
- Add injury prediction models to elements to record likely injury conditions in the model output.
- Create a fluid-filled annulus model of intervertebral discs that will have a more realistic response to relative vertebra motion.
- Impose relative motion limits between elements. For example, the model should detect when an injurious displacement occurs and signal a failure event.

Environment Elements

- Create a restraint system model instead of using damped springs.
- Create models of different surfaces instead of using only elastic planes. Account for tangential forces in addition to normal forces.
- Allow forces and moments to be used as forcing functions instead of just kinematic forcing functions.

HSM-PC Software

- Allow the user to replace sections of the model with simplified models to speed computations when reasonable. For example, allow the user to replace the cervical spine elements with a single lumped parameter segment for ejection seat simulations.
- Add injury post-processing capabilities, such as a graphical display of possible injury and high strain locations.

8.0 References.

[1]Belytschko T., Schwer L., and Schultz A. A Model for Analytic Investigation of Three-Dimensional Head-Spine Dynamics. AMRL-TR-76-10; 1976.

[2]Belytschko T. and Privitzer E. Refinement and Validation of a Three-Dimensional Head-Spine Model. AMRL-TR-78-7; 1978.

[3]Williams T. and Belytschko T. A Dynamic Model of the Cervical Spine and Head. AFAMRL-TR-81-5; 1981.

[4]Belytschko T., Williams J., and Rencis M. Head-Spine Structure Modeling: Enhancements to Secondary Loading Path Model and Validation of Head-Cervical Spine Model. AAMRL-TR-85-019; 1985.

[5] Dietrich, M., and Kedzior, T. A Biomechanical Model of the Human Spinal System. Proc. Instn. Mech. Engrs. Vol 205, Part H:19-26, 1991.

[6] Soechting, J.F. and Paslay, P.R. A Model for the Human Spine During Impact Including Musclature Influence. J. Biomech. 6:195-203, 1973.

[7] Huston, R.L., Huston, J.C., and Harlow, M.W. Comprehensive, Three-Dimensional Head-Neck Model for Impact and High-Acceleration Studies. Aviat., Space, and Environ. Med. 49(1):205-210, 1978.

[8] Merrill, T., Goldsmith, W., and Deng, Y.C. Three-Dimensional Response of a Lumped Parameter Head-Neck Model Due to Impact and Impulsive Loading. J. Biomech. 17(2):81-95, 1984.

[9] Farfan, H., Gracovetsky, S., and Helleur. Cervical Spine Analysis for Ejection Injury Prediction. AFOSR-TR-33, Air Force Office of Scientific Research, Bolling AFB, D.C. Nov, 1982.

[10] De Jager, M., Sauren, A., Thunnissen, J., and Wismans, J. A Global and a Detailed Mathematical Model for Head-Neck Dynamics. SAE 962430. Society of Automotive Engineers, Warrington, PA, 1996.

[11] Yoganandan, N., Kumaresan, S., Voo, L., and Pintar, F.A. Jou. Biomech. Engr. 119(2):87-92, 1997.

[12] Gracovvetsky, S., Farfan, H.F., and Lamy, C. A Mathematical Model of the Lumbar Spine Using and Optimized System to Control Muscles and Ligaments. Symposium on the Lumbar Spine II, Othop. Clinics of N. America, 8(1):135-153, 1977.

[13] Lavaste, F., Skalli, W. Robin, S., Roy-Camille, R., and Mazel, C. Three-Dimensional Geometric Modelling of the Lumbar Spine. J. Biomech. 25(1):1153-1164, 1992.

[14] TNO Crash Safety Centre, Delft, The Netherlands. MADYMO User's Manual, Version 5.1, 1994.

[15] Rolfe, J.M. and Staples, K.J. Flight Simulation. Cambridge University Press. New York, NY. (1990)

[16] Apter, J. and Graessley, W. A Physical Model for Muscle Behavior. Biophysical Journal, (10):539-555.(1970).

[17] Williams, J. and Belytchko, T. A Dynamic Model of the Cervical Spine and Head. AFAMRL -TR-81-5. Air Force Aeromedical Laboratory, Wright-Patterson AFB, OH. (November 1981).

[18] Wilke, D.R. The Relation Between Force and Velocity in Human Muscle. J. Physiol. (110):249-280. (1950).

[8] Knepell, P.L. and Arangno, D.C. Simulation Validation. IEEE Computer Society Monograph. IEEE Computer Society Press O.N. 3512-04. IEEE Computer Society Press, Los Alimitos, CA 90702-1264 (1992).

[19] Craig, John J. Introduction to Robotics Mechanics & Control. Addison-Wesley Publishing Company, Inc. Reading, MA. (1986).

[20] Belytschko, T., Schwer, L. and Schultz, A. A Model for Analytic Investigation of Three-Dimensional Head-Spine Dynamics. AMRL-TR-76-10. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1976).

[21] Belytscho, T. and Privitzer, E. Refinement and Validation of a Three-Dimensional Head-Spine Model. AMRL-TR-78-7. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1978).

[22] Williams, J. and Belytschko, T. A Dynamic Model of the Cervical Spine and Head. AMRL-TR-81-5. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1981).

[23] Belytschko, T. and Williams, M.R. AAMRL-TR-85-019. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45433. (1985).

APPENDIX A- Confidence Assessment Plan
Preface to the Final Confidence Assessment Plan for
The Personal Computer Version of the Head Spine Model (HSM-PC)

As this plan is published in final form BRC is approximately three months behind the original development schedule. The Confidence Assessment review of the three-link spinal subsegment is pending and should be complete by the end of 1997. Work has already begun on completing the head and neck portion of HSM-PC, which as described in the Confidence Assessment Plan (CA Plan) will be known as the Head Cervical Spine (HCS) Model. The HSM-PC "mid-term" review has already been completed, but that review did not include a review of either the three-link or the HCS models as stated in the CA Plan. Rather than change the plan or its sequence of events, BRC will attempt to speed the development of the HCS so that the Assessment Team review of the HCS will come just after the beginning of 1998. Work has already begun on extending the HCS to the complete HSM-PC and the Assessment Team review of the entire model should occur late in the first quarter of 1998.

November 14, 1997

CONFIDENCE ASSESSMENT PLAN FOR
THE PERSONAL-COMPUTER BASED HEAD-SPINE MODEL

1.0 Introduction and Background.

The USAF Aeromedical Research Laboratory originally developed the Head Spine Model (HSM) during the late 1970s[1-4]. The HSM was created to aid the solution of problems related to spinal loads resulting from ejection seat acceleration. Unfortunately, some of the original researchers who developed and used the HSM left the Government and the original code was not been maintained or used in research for several years.

In a 1996 Phase I SBIR[5] investigation sponsored by the Armstrong Laboratory, BRC's Research Development and Technical Support Division (RDTS) recovered the archived mainframe/workstation version of the HSM, ported the code to the Microsoft DOS® environment, and added user-friendly graphical user interfaces (GUI's) for managing the input and output files. Thus, the Phase I SBIR demonstrated the feasibility of creating a PC version of the HSM with user-friendly GUI's. Unfortunately, it was also discovered that the HSM existed in more than one version and that the code for each version was different. A different version of HSM was required for each of the various scenarios to which it was applied. For some versions, the archived code was incomplete—those versions would not compile because of missing subroutines. Further investigation revealed to the satisfaction of BRC that in some versions the equations of motion were not being solved correctly by the code.

For these and other reasons documented in the Phase I report[5], BRC recommended that the equations of motion be re-coded for solution on a PC under a Windows® environment. The Phase II SBIR aimed at creating a PC version of the HSM was awarded to BRC on May 1, 1996. The product of the effort will be known as the Head-Spine Model-PC (HSM-PC). The HSM-PC will be designed to create simulations of the biomechanical responses of the head and spine to potentially traumatic impulsive acceleration and impact events. It will be the only PC model of its kind in existence.

This plan describes the technical objectives and work plan for the research program and steps required to conduct a formal assessment of the validity of the HSM-PC model. The complete technical proposal, which describes the technical work plan for HSM-PC, is available in RDTS for your review. BRC envisions the HSM-PC to be a commercially viable product and to that end, BRC will prepare a Phase III Business Plan. BRC and its wholly owned subsidiary, Injury Sciences Incorporated (ISI), will sponsor the Phase III SBIR program required to commercialize and market the HSM-PC model to the private sector.

2.0   Programming Environment.

Figure 2-1 illustrates the programming environment. The HSM-PC will be developed
and hosted on IBM compatible PCs running the Windows 95™ operating system. The
Input/Output GUI's will be programmed in Microsoft Visual Basic™ and Microsoft
Visual C^{++}™ and the numerical analysis and solver modules will be written in
Microsoft's Fortran-90™. The geometry and materials properties database will be
implemented in Microsoft Access™. All of the software applications employ the Object
Oriented Programming (OOP) technology. The choice of a single software manufacturer
and the OOP programming constructs will ensure that there is mutual compatibility
between the operating system and the major programming applications employed to
create the HSM-PC.

Figure 2-1 Program Overview

```
                        ┌──────────────┐
                        │     User     │
                        └──────────────┘
                              │  ▲
                              ▼  │
┌──────────────┐      ┌──────────────────┐
│              │◄─────│ Pre-processor    │
│  Input File  │      │ Input GUI (Visual│────────┐
│              │─────►│ Basics)          │        │
└──────────────┘      └──────────────────┘        ▼
                              │          ┌──────────────────┐
                              ▼          │   Properties     │
                      ┌──────────────┐   │   Database       │
                      │ Numeric Solver│◄─│   (MS Access)    │
                      │ (FORTRAN 90) │──►│                  │
                      └──────────────┘   └──────────────────┘
                              │
                              ▼
┌──────────────┐      ┌──────────────────┐
│              │◄─────│ Post-processor   │
│  Output File │      │ Output GUI       │
│              │─────►│ (Visual Basics)  │
└──────────────┘      └──────────────────┘
     │   ▲                │      ▲
     ▼   │                ▼      │
┌──────────────┐      ┌──────────────┐
│   Injury     │◄─────│              │
│Post-processor│      │     User     │
│  (FEM Code)  │─────►│              │
└──────────────┘      └──────────────┘
```
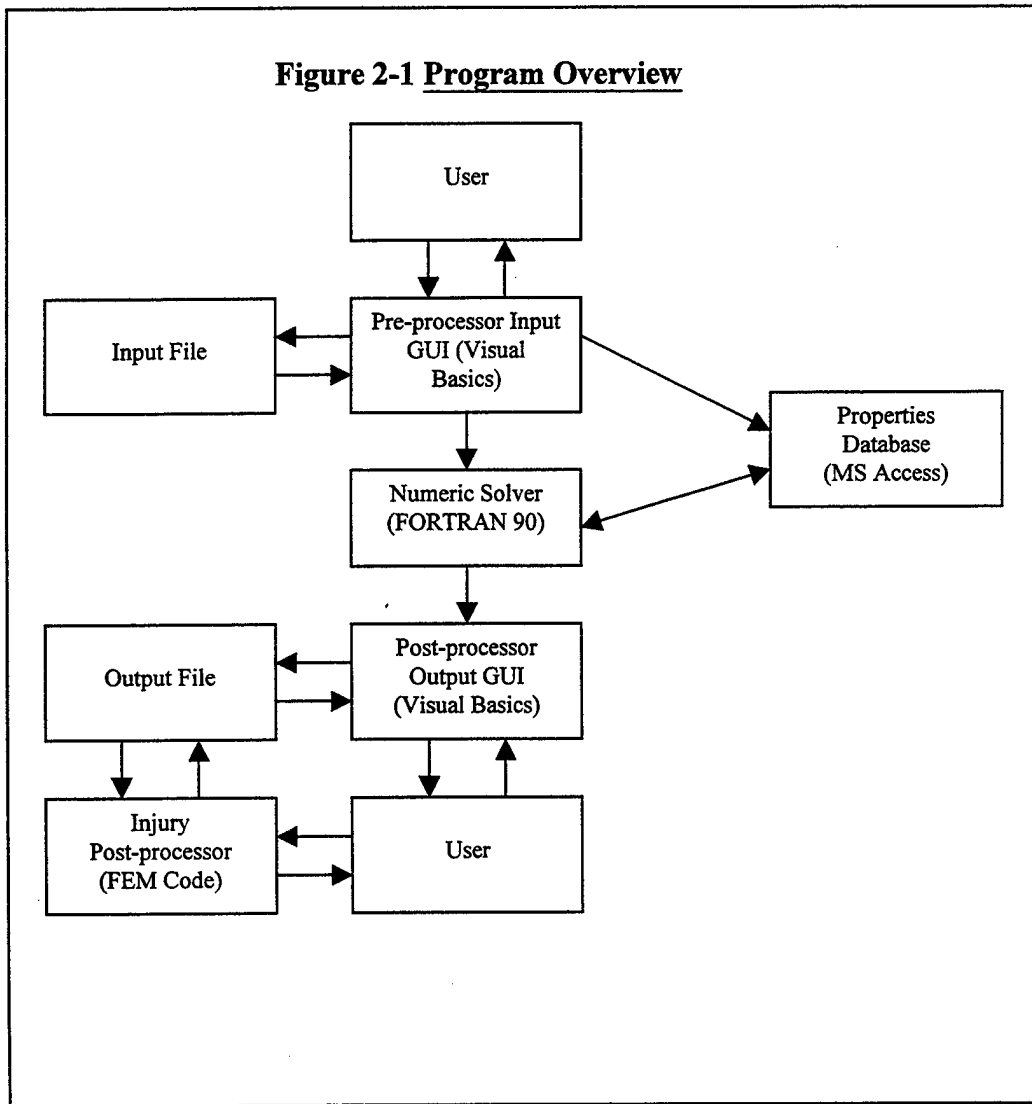
3.0 Technical Objectives.

The Technical Objectives listed in Figure 3-1 comprise the technical steps necessary to create a partially validated, user-friendly HSM model for the Windows 95™ operating system. For the purposes of this plan, the proposed re-coded HSM will be called "HSM-PC." The work plan to accomplish the steps listed in Figure 3-1 is explained more fully in the following paragraphs.

Figure 3-1 <u>Phase II Technical Objectives</u>

- **Plan and Establish the Simulation Validation Protocol**

- **Create HSM Geometry and Materials Properties Database**

- **Re-Code the HSM for the Personal Computer Environment**

- **Partially Validate the HSM-PC**

- **Conduct Parametric Studies Using the HSM-PC Simulation**

- **Document the HSM-PC and Its Applicability**

- **Reduce Commercial Risk and Demonstrate Commercial Potential of the HSM-PC**

4.0 Work Plan Summary.

Figure 3-1 outlined the Technical Objectives of the proposed Phase II research program. Section 4.0 presents the details of the work necessary to achieve the program objectives. The Work Plan is designed to support achievement of the Technical Objectives and to ensure that necessary program controls are instituted to ensure the necessary confidence assessment (See Section 6.0 for a description of Confidence Assessment activities.). The major Work Plan tasks are presented in Figure 4-1. The details of the activities necessary to carry out the tasks noted in Figure 4-1 are presented in the HSM-PC proposal a copy of which has been provided to each of the Assessment Team members. Figure 4-2 presents the roadmap of the HSM-PC development process that shows the dependencies between the major tasks.

Figure 4-1 <u>Summary of Major Work Plan Tasks</u>

- **Write the Confidence Assessment Plan**
- **Literature Review**
- **Extract and Audit Data from Archived AAMRL HSM Code**
- **Create User Friendly Interfaces for HSM-PC**
  - Program Single-Body Geometric Description Modules
  - Program Forcing Function Modules
  - Extend Single-Body Geometry to Multiple Body Assembly and Input Modules.
  - Create, Verify and Validate Models for Inter- and Para-Vertebral Elements
  - Create the Head-Cervical Spine Model
    - Create a Three-Link Cervical Subsegment Model
    - Conduct Parametric Studies of Three-Link Model
    - Assessment Team Evaluation of Tree-Link Model
    - Extend Three-Link Model to Full Head-Cervical Spine Model
    - Conduct Parametric Studies of the Head-Cervical Spine Model
    - Conduct a Partial Validation of the Head-Cervical Spine Model
    - Assessment Team Evaluation of the Head-Cervical Spine Model
  - Extend the Head-Cervical Spine Model to Include the Entire Spine
  - Create and Evaluate the Hydrodynamic Elements Simulating the Thoracic-Abdominal Load Path
  - Create and Evaluate the Restraint System and Elastic Plane Models
  - Create a Prototype Injury Assessment Post-Processor
- **Conduct Parametric Studies of the Head-Spine Model (HSM-PC)**
- **Conduct Partial Validation of the HSM-PC**
- **Assessment Team Evaluation of the HSM-PC**
- **Document the HSM-PC Development Process**
- **Develop and Write the Phase III Business Plan for HSM-PC**
- **Write the Final Report**

Figure 4-2 Roadmap to Development Process

Write Confidence Assessment Plan

Literature Review

Extract and Audit Data from Archived AAMRL

Create User Friendly Interfaces for HSM-PC

Create HSM-PC Biomechanical Model

Single Body Model

Multi Body Assembly Modules

Forcing Function Modules

Head Cervical Spine Model

Extend HCS Models to HSM-PC

Create Hydrodynamic Elements

Create Restraint System & Elastic Planes Models

Create Prototype Injury Assessment Processor

Parametric Studies of HSM-PC

Partial Validation of HSM-PC

Assessment Team Evaluation of HSM-PC

Business Plan Development

Write Final

1 Year

4.5 Summary of Overall Work Plan for Creating the HSM-PC Biomechanical Model.

a. Program Single Body Geometric Description Modules. The re-coding of the HSM model will begin with producing the code necessary to specify the geometry and the materials properties of a single vertebra. A vertebra from the central cervical spine (C5) will be employed. The code necessary to visualize and view the vertebra will be designed and tested during this initial programming project. At this stage, only the bony structures of the spine will be specified and modeled.

b. Program the Forcing Function Modules. The programming of the forcing function modules will be done in parallel with the steps in the previous section. The programming constructs of the original HSM code will be preserved and modernized. The original HSM code allowed the forcing functions to be specified as values of position, velocity, or acceleration versus time. The actual forcing function applied to the model was always position versus time, but the forcing function modules in the HSM performed interpolation and integration as necessary. These characteristic features will be preserved in the HSM-PC. The HSM-PC addition forcing function modules will: (1) fit interpolation polynomials to the user-specified forcing function; (2) integrate the user-input function, if necessary; and (3) transform the user-specified time step to model-solution time step.

c. Extend Single-Body Geometry to Multiple Body Assembly and Input Modules. After successful creation of the modules for creating, displaying, and specifying the single vertebral object, two, and then three vertebrae will be specified, modeled, and displayed. The final three-segment model will consist of models of C5-C7 as shown in Figure 4-3. The assembly, display, and manipulation of a three-segment model will be a rigorous test of the Input GUI and will aid in its refinement.

The two- and three-body models will be subjected to testing with simple forcing functions to ensure proper simulation dynamics are computed. BRC has already programmed simple two- and three-link models of the cervical spine and is familiar with their dynamics.

These models will be further verified by programming their geometry in multibody simulation packages such as ATB*. The dynamic response of the three-link model will be validated by comparison to the multibody modeling package results. This will be an important confidence building event in the HSM-PC development program. The equations of motion in BRC's model will be solved with a general purpose Differential-Algebraic Equation (DAE) solver such as DASSL.[7] BRC owns the source code for DASSL and several other DAE solvers.

---

* Articulated Total Body Model Users Guide, AAMRL-TR-88-043, Armstrong Laboratory, Vulnerability Assessment Branch (AL/CFBV) Wright-Patterson AFB OH 45433 (1988).

Figure 4-3 Three-segment Cervical Subsegment
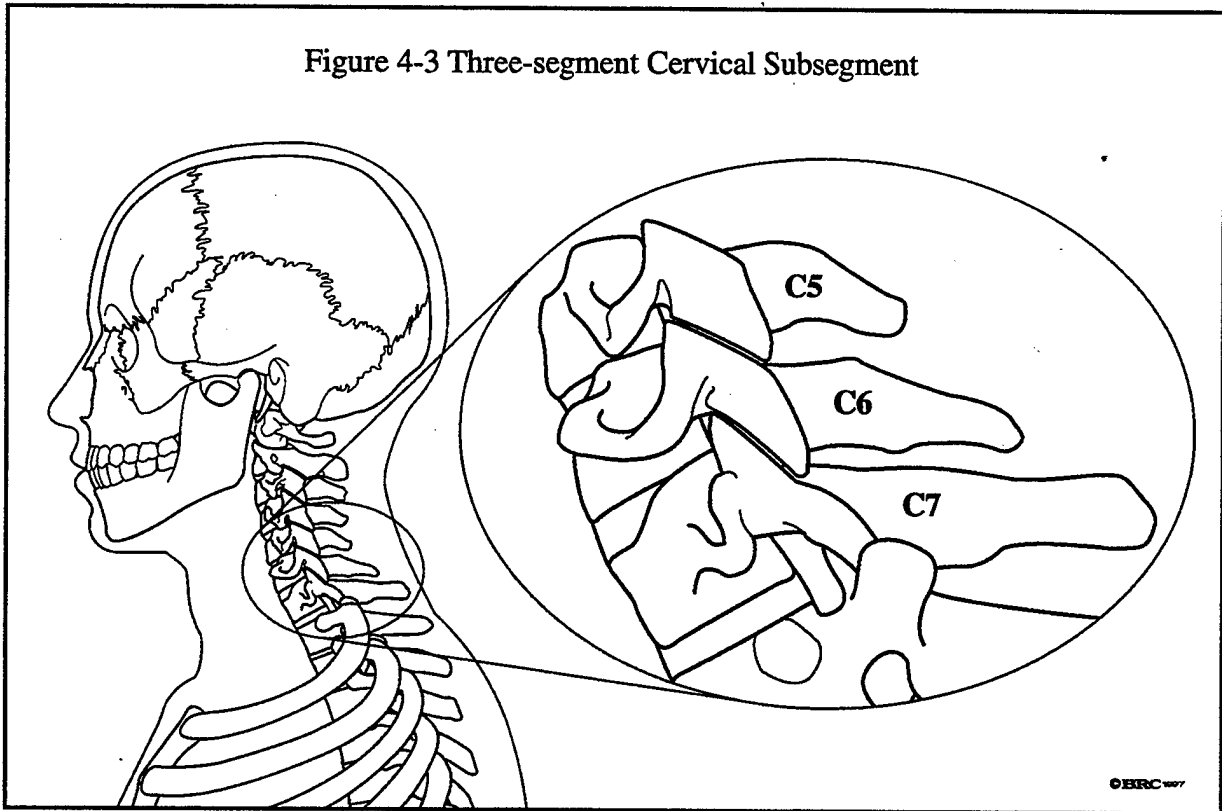
d. Create, Verify and Validate Models for Inter- and Para-Vertebral Elements. In parallel with the work described above, programming of the other model elements will be completed. These elements will be created to model the intervertebral and paravertebral tissues associated with the spine. Tissues such as tendon, ligament, muscle, intervertebral discs and other connective tissues will be modeled. The models for each of these elements will be created and tested individually in large part by reproducing the models employed in the original HSM model. Unique symbology will be developed to depict and distinguish the various tissues in the GUI displays. This will facilitate "assembly" of the vertebral segments by the user. Choices of attachment sites for the various tissues will be appropriately limited as dictated by the actual anatomy and the coordinate system constructs created by the original HSM development team.

e. Create the Head-Cervical Spine Model. Once the subsegment models of the bony parts of the spine have been created and tested, the three-segment model of C5-C7 will be combined with the appropriate models of the inter- and para- spinal tissues and evaluated parametrically and against known physical principles. Following the successful completion of the cervical subsegment model, that model will be extended to include the spine from C1 to T1 and the head. This model will be called the Head-Cervical Spine Model and it will be evaluated parametrically and against available human data.

1. Create a Three-Link Cervical Subsegment Model. The first step in creating the subsegment model will be creating the assembly and display code which allow the attachment and specification of element model properties to the bony skeleton models. We expect this phase to be a difficult one and we will

A-8

closely follow the experience of the original HSM development team as reported in the HSM Technical Reports.[1-4]

2.. Conduct Parametric Studies of Three-Link Model. Simple forcing functions will be applied to models produced by both methods and their responses will be compared output from other multibody simulation codes such as ATB. Discrepancies will be resolved before BRC proceeds with further development programming.

3. Assessment Team Evaluation of Three-Link Model. The Assessment Team review process will be employed to formally assess and document the verification and validity of the three-link cervical subsegment model. Technical issues arising from the confidence assessment process will be communicated to the Government and resolved before development proceeds to more complicated models.

4. Extend Three-Link Model to Full Head-Cervical Spine Model. Once technical issues related to either the development process or the verification and validation of the three-segment model have been successfully resolved, work will proceed to extend the three-segment model to the complete cervical spine, the first thoracic vertebra (T1) and the head.

5. Conduct Parametric Studies of the Head-Cervical Spine Model. Parametric studies similar to those described previously will be employed to study the behavior of the HCS Model. Because of the HCS model's increased complexity, these studies will be more difficult and time consuming.

6. Conduct a Partial Validation of the Head-Cervical Spine Model. Once the parametric studies have been completed, BRC will conduct a partial validation of the HCS model. BRC has conducted two series of low-speed automobile crash tests in which drivers and passengers were human volunteers. The data collected included high-speed sagittal plane film and accelerometer recordings from a tri-axial accelerometer package mounted on a bite block worn by the subject drivers.[8,9] These data have been analyzed and their kinematics characterized and reported.[8,9] BRC will employ these data to estimate the parameters necessary to reproduce the sagittal plane kinematics in the low-speed crash test data. These validations will be necessarily limited by the data content. Therefore, BRC will employ data documented in the HSM TR's[1-4] and data supplied by the Armstrong Laboratory to extend the domain of the HCS model.

7. Assessment Team Evaluation of the Head-Cervical Spine Model. The Assessment Team review process will be employed to formally assess and document the verification and validity of the HCS model. The results of the confidence assessment of the HCS model will be communicated to the Government in a formal project review hosted by BRC. During this review, the entire development program will be reviewed with Government and invited outside experts.

Successful resolution of technical issues arising from the confidence assessment of the HCS model will be a major project milestone. However, it should not be

expected that the entire domain of possible application of the HCS model will be validated before work begins on completing the remainder of the HSM-PC.

f.   Extend the Head-Cervical Spine Model to the Lower Spine and Rib Cage. The techniques developed during the development of the HCS model will be employed to extend the model to the lower spine, rib cage, and pelvis. Iterative testing of the spinal subsections will be employed to conduct parameter identification and covariance as discussed above.

g.   Create and Evaluate the Hydrodynamic Elements Simulating the Thoracic-Abdominal Load Path. The hydrodynamic elements developed in the original HSM development will be re-coded, tested and added to the HSM-PC following the methods outlined in the HSM TR's.[1-4]

h.   Create and Evaluate HSM-PC Restraint System and Elastic Plane Models. The restraint system models and elastic plane models defined in the original HSM TR's[1-4] will be coded and tested to verify and validate their behavior. The provisions for specifying the geometry and properties of these model elements will be added to the GUI's and properties database.

i.   Create a Prototype Injury Assessment Post-Processor. BRC will investigate, create, and code a prototype Injury Assessment Post-Processor (IAPP) for the HSM-PC. The IAPP will be largely based on ideas developed in the original HSM development which were based on finite element modeling of the HSM structures. The general scheme will be to create a simulation from which the force and moment data related to the vertebral segments can be extracted. Once the force and moment data is available, points of greatest stress in the model structure will be identified and modeled using the finite element method to estimate strain within the spinal skeleton.

4.6   Conduct Parametric Studies of the Head-Spine Model (HSM-PC).

Considerable effort will be devoted to studying the parameters and their interdependence, interaction, and sensitivities. Documentation of the relationships between the model parameters and the various model responses will be an important addition to the HSM-PC's overall documentation.

The effect of the reduction of the number of degrees of freedom will be systematically investigated as part of this task. Models with "lumped" vertebral segments will be employed to simulate experimental data and the effect of changing the number of degrees of freedom (adding and removing vertebra) will be studied. It is anticipated that a model such as the HSM-PC will be somewhat over determined for many problems of interest and that considerable parsimony can be achieved by eliminating redundant degrees of freedom and unnecessary parameters. Moreover, understanding of the effect of particular parameters and their effect on the dynamic responses of the HSM-PC will guide the use of the model.

4.7   Conduct Partial Validation of the HSM-PC.

BRC and Government data will be employed to conduct a partial validation of the complete HSM-PC. This activity will define the domain of applicability for the HSM-PC and identify "gaps" in the knowledge base. Researchers can employ the knowledge of the relative value (cost-benefit) of improving the HSM-PC performance to logically assess the need for future research.

4.8   Assessment Team Evaluation of the HSM-PC.

The Assessment Team will conduct a complete retrospective review of the development process, documentation, and data produced by the HSM-PC. Emphasis will be directed to those activities conducted after the Head-Cervical Spine milestone review. However, the entire program will be reviewed to generate a confidence assessment for the complete HSM-PC. Technical issues will be reviewed with the Assessment Team, the Government and, if required, outside subject matter experts. Technical issues, which cannot be resolved within the scope of the development program, will be documented for future investigation.

4.9   Document the HSM-PC Development Process. The documentation generated by the Simulation Validation confidence assessment process will be compiled, indexed, and archived in appendices to the Final Report.

5.0   The Simulation Validation Protocol.

The development of a complex computer simulation such as the HSM-PC demands that a logical and rigorous protocol be employed to manage the risks of creating a software simulation that is difficult to employ and maintain in the "operational environment." Therefore, BRC will employ a DOD developed methodology described by Knepell and Arango.[10] Their *confidence assessment* (CA) methodology provides the necessary procedural framework for controlling the development of a complex simulation, while ensuring the validation of the modeling concept, code verification, documentation, and, most importantly, a systematic evaluation of the model's credibility. Figure 5-1, which was adapted from Figure 2-3 of Reference 1, shows a diagram of the overall scheme of CA.

Figure 5-1 Confidence Assessment

┌──────────────────┐
│ "Real-world"   │
│ Problem Entity   │
└──────────────────┘

Operational
Validity

Experimentation

Analysis
&
Modeling

Conceptual
Model

┌──────────────┐   ┌──────────────┐
│ Computer   │   Computer   │ Conceptual   │
│ Model   │   Programming &   │ Model   │
└──────────────┘   └──────────────┘

Software
Verification

Internal
Security
Verification
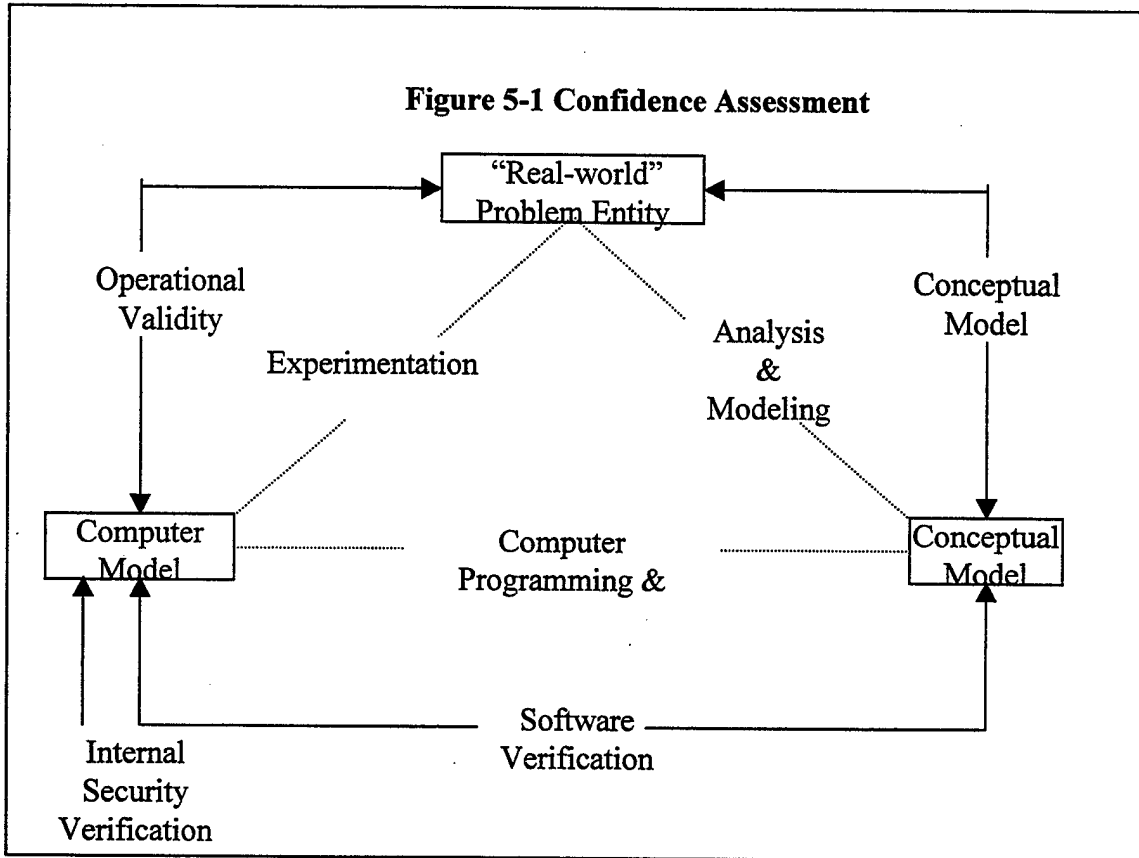
To explain the method, a few definitions are required

a. Conceptual Model Validation. The independent review of the concept of the model and its implementation to ensure the model addresses the relevant features of the "real-world" entity being simulated--in this case, the head and spine of the human being.

b. Software Verification. A process by which it is ensured that the software code computes the desired results. Essentially, this process verifies that the coded software implements the desired algorithms correctly and that it performs as desired.

c. Operational Validation. This is the classic validation process necessary of any computer modeling program. Operational validation ensures that the computer simulation has a satisfactory range of accuracy within its intended domain of applicability and that the scope of its domain is defined and documented. In relation to the HSM-PC, operational validation will ensure that the HSM-PC predictions and their errors are understood and that its domain of applicability is known.

d. Data Validation. The process of documenting the data employed in generating model parameters and equations. For the HSM-PC this will consist of creating, updating, and auditing a database of geometry and materials properties necessary to specify the physical structure and the properties of the constituent tissues of the spine and its associated structures.

e. Internal Security. Internal security establishes a configuration control procedure and protects the code against tampering or unauthorized modification. Within BRC, the HSM-PC source code will only be available to those staff programmers who work on the project. The Principal Investigator will strictly enforce this policy.

As shown in Figure 5-1, the entire process of confidence assessment is designed to ensure the "real-world problem entity," in this case, the human head and spine, are represented and simulated in the computer model as logically and accurately as possible. An Assessment Team whose members are listed in Figure 5-2, below will manages the entire process.

Figure 5-2 <u>Assessment Team Representatives</u>

- Project Management
- Development Team
- Users
- Community Experts

An Assessment Team has been established to monitor, assess and control the HSM-PC development. Because of the necessary limits to the staff and resources, which can be devoted to the HSM-PC development, some members serve in dual roles. Members of the Assessment Team are identified below. Dr. James Raddin will chair the Assessment Team.

a. Management. Dr. John Bomar, the Principal Investigator, will serve as the BRC Management representative to the Assessment Team. The Government Project Manager,

A-13

Dr. Louise Obergefell, will also participate in this capacity. The role of management in the process is to provide support and direction as necessary to ensure sufficient resources are available to complete the development program on schedule and within budget.

b.   Development. Developers include the programmers, consultants, scientists, and computer infrastructure experts necessary to implement the Technical Objectives of the HSM-PC development.   Mr. Pancratz will serve as the Development representative on the Assessment Team. Within BRC, Dr. John Bomar, Mr. David Pancratz, Ms. Linda Rogers, and Mr. Darrin Smith will serve as the primary Developers. Dr. Richard Howard will serve on the development team as a consulting expert in the biomechanics of the spine.   Dr. Richard Harding will assist Dr. Howard as necessary.

c.   Users. The users group represents knowledgeable scientists and practitioners who possess the necessary skill and technical knowledge to employ the HSM-PC in an investigative or problem solving environment. Within BRC, Dr. Whitman McConnell, a Physician-Biomechanic, will serve as the User Group member to the Assessment Team. Dr. McConnell will call on others to aid in User evaluations of the HSM-PC. The Government has been invited to designate a User representative.

d.   Community Experts. To provide an objective review of the HSM-PC and its development process, subject matter experts (SME's) will be employed as members of the review team. Within BRC, Dr. James Raddin and Dr. Charles Hatsell, both Physician-Biomechanics, will serve as SME's on the Assessment Team. Dr. Hatsell will also serve as a Development Team expert, but Dr. Raddin will not be involved in the development process in any other capacity other than as a SME reviewer and Head of the Assessment Team. The Government has been invited to add SME's to the Assessment Team.

The model evaluation approach is detailed in Simulation Validation.[10] RDTS has copies of this text available for the Assessment Team Members' use. Simulation Validation will be the standard reference for the HSM-PC simulation validation activities. The confidence assessment (CA) process provides: (1) the detailed methodology and tools to continuously assess, verify, validate, and document a complex software simulation during its development; (2) an explicit method to assess the credibility of the model's simulations and its domain of applicability; and (3) an audit trail to support the confidence assessment process governed by the Assessment Team. Use of the CA process will ensure that the HSM-PC software is properly assessed and documented.

6.0   Assessment Activities.

This section outlines the details of the assessment processes in each of the assessment activities: Conceptual Model Validation, Software Verification, Operational Validation, Data Validation and Internal Security Verification.   The activities have been tailored to the HSM-PC and the size of the Assessment Team. As was mentioned earlier, the HSM-PC confidence assessment will include activities from both the *Formal Assessment* and *Limited Assessment* Methodologies as defined by Knepell and Arango in their text Simulation Validation[10]. However, because the size of the HSM-PC Assessment Team necessarily limits the scope of the assessment which can be performed, the scope of the HSM-PC assessment will be more like the Limited Assessment than

A-14

the Formal. Confidence assessments are divided into a *Planning Phase* and an *Application Phase*. Only Application Phase activities will be described in this plan.

6.1   Conceptual Model Validation.

Many of the tasks required for Conceptual Validation of the HSM-PC have already been carried out to the satisfaction of the sponsors of the project (AL/CFBS) in the SBIR Phase I program. Nevertheless, the Confidence Assessment Team will carry out an abbreviated Conceptual Validation. The Development Team members will compile the answers to the questions in Appendix A and review that data with the CA Team Chairman and SME's. The Conceptual Validation analysis will be documented for review by the entire CA Team and their comments will be collected an employed to guide development activities during the SBIR Phase II program. The activities and tools employed in Conceptual Model Validation are outlined in Figure 6-1. The primary purpose of the following description of Conceptual Validation is to familiarize the CA Team with the process of so that a Conceptual Validation of the present HSM-PC program can be completed successfully.

     a.   Face Validity Analysis. This activity comprises a subjective review of the information available on the model. The primary goal of Face Validity analysis is to discover areas where additional information is necessary to raise the confidence in the model. Detailed explanations of the activities required for each activity are contained in Simulation Validation10.

     b.   Historical Analysis. The history of the model development should be documented together with descriptions of the model's design requirements and specifications, which guide the development. Any Independent Verification and Validation (IV&V) Analysis by third parties or the sponsor should be documented. If derivative models are created their relationships to the original should be clearly defined so that verification and validation work done on the parent model can be employed for the derivative model. Uses of the model and the degree of its utility and success in a particular application should be documented as an aid to understanding the model's domain of applicability.

Figure 6-1 Conceptual Validity Assessment

- Historical Analysis
  - Development History Analysis
  - Independent Verification and Validation
  - Model Derivative Analysis
  - Previous Model Analysis

- Intended Use and Requirements Analysis
  - Critical Analysis
  - System Analysis

- Model Concepts and Fidelity Analysis
  - Modeling Concepts Analysis
  - Input/Output Analysis
  - Algorithm Analysis

- Logic Trace Analysis

c.   Intended Use and Requirements Analysis. The program's standards and requirements should be reviewed in light of its intended use. The criticality meeting a particular requirement should be evaluated in terms of the impact of the failure to meet the requirement on a particular intended use and the technical risk associated with meeting the requirement. Each requirement can understood in terms of its relative criticality and technical difficulty so that the appropriate resources can be devoted to meeting its goals. The intended use should include a traceability analysis, which should prove that the model's requirements are traceable through the model to its submodels and their requirements.

d.   Model Concept and Fidelity Analysis. The overall model concept should be reviewed for soundness in relation to the model's intended uses. Issues such as level of detail, accuracy, and consistency should be addressed. The level of detail should be sufficient to meet the requirements of the intended use. Mismatches should be addressed in terms of their criticality and technical risk as noted above. Algorithms should be examined and their appropriateness and limitations. The accuracy of a particular algorithm should be judged in light of both its input data and the requirement for use of its output. The CA Team will require consultation with the model programmers to examine the technical merits of various analytical and numerical approaches to subsystem models.

e.   Logic Trace Analysis. This activity relates to tracing particular entities, such as input data or a subsystem model, through the modeling process to ensure that the model uses the entity properly and produces the expected result when using the entity. Logic trace analysis is related to Software Verification as explained in the next section. Those entities selected for logic trace analysis can be selected by their level of criticality to the

A-16

model's intended use. The results of logic trace analysis aid in characterizing the model and produces information on critical entities and documents their behavior.

6.2 Software Verification.

As noted previously software verification is a process by which it is ensured that the software code computes the desired results. Essentially, this process verifies that the coded software implements the desired algorithms correctly and that it performs as desired. The Development Team will carry out routine software verification activities on subsystems and software modules. The results will be documented and reported to the sponsor and CA Team periodically. These activities will be noted in the SBIR Phase II Monthly Reports. Figure 6-2 lists the activities typical of Software Verification.

Figure 6-2 Software Verification

- CASE and Design Methodology Adherence Analysis

- Software Metric Analysis

    - Process Metric Analysis
    - Product Metrics Analysis

- Internal Software Testing Analysis

- Code Analysis

    - Program Logic Analysis
    - Program Data Analysis
    - Program Interface Analysis
    - Program Constraint Analysis

- Correctness Proofs

---

a. Computerized Analysis. Computerized software verification relates to the use of Computer Aided Software Engineering (CASE) analysis applications. Because of the resource limitations BRC will not employ computerized analysis techniques during the Phase II development of the HSM-PC.

b. Software Metrics Analysis. BRC will employ software metrics analysis to document the performance of subsystem modules in order to characterize the software. Software metrics fall into two broad categories: Process Metrics Analysis and Product Metrics Analysis.

A-17

model's intended use. The results of logic trace analysis aid in characterizing the model and produces information on critical entities and documents their behavior.

6.2   Software Verification.

As noted previously software verification is a process by which it is ensured that the software code computes the desired results. Essentially, this process verifies that the coded software implements the desired algorithms correctly and that it performs as desired. The Development Team will carry out routine software verification activities on subsystems and software modules. The results will be documented and reported to the sponsor and CA Team periodically. These activities will be noted in the SBIR Phase II Monthly Reports. Figure 6-2 lists the activities typical of Software Verification.

Figure 6-2 Software Verification

- CASE and Design Methodology Adherence Analysis

- Software Metric Analysis

    - Process Metric Analysis
    - Product Metrics Analysis

- Internal Software Testing Analysis

- Code Analysis

    - Program Logic Analysis
    - Program Data Analysis
    - Program Interface Analysis
    - Program Constraint Analysis

- Correctness Proofs

---

a.   <u>Computerized Analysis</u>. Computerized software verification relates to the use of Computer Aided Software Engineering (CASE) analysis applications. Because of the resource limitations BRC will not employ computerized analysis techniques during the Phase II development of the HSM-PC.

b.   <u>Software Metrics Analysis</u>. BRC will employ software metrics analysis to document the performance of subsystem modules in order to characterize the software. Software metrics fall into two broad categories: Process Metrics Analysis and Product Metrics Analysis.

1.   Process Metrics Analysis. Process metrics analysis requires documentation of the development process activities such as design documentation, module testing, and development test reports. This data enables the retrospective analysis of the development process for gleaning lessons learned and assessing the soundness of the model development activities. The monthly status and activities reports will serve as the primary source of this data for the CA Team.

2.   Product Metrics Analysis. Product metrics include documentation of references for algorithms and scientific data, data on the source code, model interface documents, compilers, computer resource utilization measures, and software metrics. These are data that are related to the model's reliability, efficiency, correctness, maintainability, etc. These data will be collected via internal software testing and application of the model to test cases.

c.   Code Analysis. Code analysis will be the primary software verification tool in the HSM-PC program. Code analysis involves a review of the coded representation of the model. The model's software modules and data are evaluated by the CA Team specialists to detect problems in model logic, data, interfaces, and constraints. This analysis also sheds light on the model's limitations and helps to define its domain of applicability. The software development environment will be employed to analyze program cross-references, proper use of global data, and discover coding errors, memory over/under flows and inefficient code. The numerical portions of the HSM-PC will be coded under Microsoft's Developer's Studio using Fortran 90. Both the environment and the language have modern error detection and data structures that will aid in detection of coding and numerical errors. The Graphic User Interface (GUI) will be coded in Microsoft Visual Basic, and the HSM-PC database will be accessed through a Microsoft Access relational database. Particular attention to the interfaces between these modules will be required to verify that communication between the modules is accurate and efficient.

d.   Correctness Proofs. Correctness proofs rely on assertion checking of input/output relationships. Essentially, correctness is verified by asserting the model's output to a given input and then checking to ensure that the expected output was actually produced. Correctness proofs are conducted and documented both on the entire model and its subsystem modules during their development.

6.3.   Operational Validation.

Operational validation is the classic validation of the model's output against the real-world entity it purports to simulate. Levels of testing from inspection to analytical testing are employed. The activities comprising operational validation are briefly described below. More complete descriptions of the requirements are contained in Simulation Validation[10]. See Figure 6-3 for a summary of the tasks required for operational validation.

Figure 6-3 Operational Validation

- Inspection Tests

    - Delphi Tests
    - Turning Tests
    - Input/Output Relationships Tests
    - Event-sequencing Tests

- Demonstration Tests

    - Animation Tests
    - Fixed Value Tests
    - Simplified Tests
    - Predictive Validation Tests
    - Internally Validity Tests
    - Extreme-condition Tests
    - Limited Standards Tests

- Analytical Tests

    - Predictive Validation
    - Comparison to Test Data
    - Sensitivity Analysis
    - Feedback Loop Anlysis

a. <u>Inspection Tests</u>. Inspection testing involves evaluating the model in its entirety in a non-rigorous manner. They are conducted by experts knowledgeable on the system who assess the model as to how well it behaves like the real world entity being simulated. Output data are evaluated and compared to actual test data collected from the actual entity. Inspection tests also examine the static and dynamic behavior of the model together with internal event sequencing analysis to judge how "real" the model's behavior seems. Problem areas are identified and prioritized for further analysis.

b. <u>Demonstration Tests</u>. Demonstration testing is more analytical than inspection testing in that it involves actual data comparisons between the model and the real-world entity, but under simplified and easy to understand situations. Demonstration tests include animation, static tests, internal validity, dynamic tests, extreme condition tests, etc. and comparison of model outputs to hand calculations and simplified model outputs. This testing is aimed at revealing patterns of model behavior to varying data input which can begin to reveal how sensitive the model is to input variations and the accuracy of its internal data. A series of standard tests can be developed to demonstrate the range of model behavior as well as its "average" response to typical input and parameter combinations.

A-19

c.   Analytical Tests. In analytical testing comparisons of the model's output is compared statistically to test data from the simulated entity. Such measures as predictive testing, comparison to test data, sensitivity analysis, and feedback loop analysis. The HSM-PC project will make extensive use of sensitivity analysis to determine parameter intercorrelation and to discover how the model's complexity might be reduced without significant effects on its predictive accuracy. The Development Team will conduct the analytical testing for presentation to and review by the Assessment Team.

6.4   Data Validation.

Data validation is the process employed to audit and validate the data employed by the model. The major repository of HSM-PC data will be contained in the database which will hold the geometric and materials properties data required for the model assembly and display. The major emphasis of the data validation will be audits to ensure the numerical accuracy of the data in the database as well as dimensional consistency and standardization of units. The basic unit system employed for the HSM-PC will be the SI system (mks).   Conversion to other unit systems will be incorporated in the user interface. Data embedded in the program modules must also be checked and verified to be consistent with those in the database. The programmers will create modularized code and global variables, which will simplify this task. If statistical distributions are employed in the model to simulate random phenomena the distributional form of those variables must be audited and verified.

6.5.   Internal Security Verification.

Internal security focuses on configuration control and physical security of the software. Its goals are to ensure that the state of the code is controlled and known to the development team members and that no unauthorized access to the source code is allowed. BRC will employ a simplified system of ensuring security of the code by creating a log of changes to modules and a configuration control committee which approve code changes and the alteration of the "master copy" of the code. The latest version of the code will be stored in a restricted directory on BRC's network, which is backed up daily.

7.0   Application of Confidence Assessment Methodology.

The day-to-day implementation of CA activities will be the primary responsibility of the HSM-PC development team. Each month the Development Team will report technical progress and CA results to the Chairman of the CA Team, Dr. Raddin. Dr. Raddin will call on other CA Team members as required to address specific technical issues as they arise. It is anticipated that most of the CA activities and their documentation can be carried out in this manner. However, the SBIR Phase II proposal calls for three formal CA Team reviews as noted below.

a.   Assessment Team Evaluation of Three-Link Model. The Assessment Team will conduct a review of the Three-Link Model to formally assess and document the verification and validity of the three-link cervical subsegment model. Technical issues arising from the confidence assessment process will be communicated to the Government

and resolved before development proceeds to more complicated models. It is anticipated that the review will be conducted in early 1997

b. Assessment Team Evaluation of the Head-Cervical Spine Model. The Assessment Team review process will be employed to formally assess and document the verification and validity of the HCS model. The results of the confidence assessment of the HCS model will be communicated to the Government in a formal project review hosted by BRC. During this review, the entire development program will be reviewed with Government and invited outside experts. The CA Team review of the HCS Model will take place in conjunction with a planned major Phase II Program Review. The final date for this meeting will be set at the conclusion of the review of the Three-Link Model.

c. Assessment Team Evaluation of the HSM-PC. At the conclusion of the development, the Assessment Team will conduct a complete retrospective review of the development process, documentation, and data produced by the HSM-PC. Emphasis will be directed to those activities conducted after the Head-Cervical Spine milestone review. However, the entire program will be reviewed to generate a confidence assessment for the complete HSM-PC. Technical issues will be reviewed with the Assessment Team, the Government and, if required, outside subject matter experts. Technical issues, which cannot be resolved within the scope of the development program, will be documented for future investigation. It is expected that the CA Team review of the HSM-PC will take place in combination with the Final Report presentation and program review that is scheduled to occur in July 1998.

7.1   Reporting

As Confidence Assessment proceeds, findings of the Confidence Assessment Team or Development Team will be recorded on a Model Assessment Finding Form and its disposition will be recorded on an Assessment Disposition Form. (See Appendix B for sample copies of the forms.). These forms serve to formally raise and document the resolution of issues for final review by the Chairman of the CA Team. The Chairman in turn may seed assistance from the Development Team and outside experts as required and may require further assessment or resolution action before the issue is considered to have been resolved. The documentation developed during the entire Confidence Assessment process will be reported in the Confidence Assessment Report. The outline of the Confidence Assessment Report is shown in Appendix C. The Confidence Assessment will be included in the Final Phase II report for the HSM-PC development program.

REFERENCES

1.   Belytschko, T., Schwer, L. and Schultz, A. A Model for Analytic Investigation of Three-Dimensional Head-Spine Dynamics. AMRL-TR-76-10. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45333 (1976).

2.   Williams, J. and Belytschko, T. A Dynamic Model of the Cervical Spine and Head. AFAMRL-TR-81-5. Air Force Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45333 (1981).

3.   Belytschko, T. and Privitzer, E. Refinement and Validation of a Three-Dimensional Head-Spine Model. AMRL-TR-78-7. Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45333 (1978).

4.   Belytschko, T., Rencis, M. and Williams, J. Head-Spine Structure Modeling: Enhancements to Secondary Loading Path Model and Validation of Head-Cervical Spine Model. Harry G. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson AFB, OH 45333 (1985).

5.   Bomar, J.B., Pancratz, D.J. and Rogers, L.J. Feasibility Study for a Personal-Computer Based Head-Spine Model. AL/CF-TR-1996-0014. Air Force Materiel Command, Wright-Patterson AFB, OH. (1996).

6.   Hollars, M.G., et al. SD/FAST User's Manual. Symbolic Dynamics, Inc. Mountain View, CA (1991).

7.   Hindmarsh, A.C. ODEPACK, A Systematized Collection of ODE Solvers. Lawrence Livermore National Laboratory, Livermore, CA (1987).

8.   McConnell, W.E., Howard, R.P., Van Poppel, J., Krause, R., Guzman, H.M., Bomar, J.B., Raddin, J.H., Benedict, J.V., and Hatsell, C.P., "Human Head and Neck Kinematics After Low Velocity Rear-End Impacts -Understanding 'Whiplash'." Presentation to Society of Automotive Engineers, Inc., 39th Annual Stapp Car Crash Conference Proceedings, Coronado, CA, November (1995).

9.   McConnell, W.E., Howard, R.P., Guzman, H.M., Bomar, J.B., Raddin, J.H., Jr., Benedict, J.V., Smith, H.L., and Hatsell, C.P. - "Analysis of Human Test Subject Kinematic Responses to Low Velocity Rear End Impacts." Presentation to Society of Automotive Engineers, Inc., 1993 SAE International Congress & Exposition, Detroit, MI, March (1993).

10.   Knepell, P.L. and Arangno, D.C. Simulation Validation. IEEE Computer Society Monograph. IEEE Computer Society Press O.N. 3512-04. IEEE Computer Society Press, Los Alimitos, CA 90702-1264 (1992).

APPENDIX A
ASSESSMENT QUESTION LIST

The questions provided in detail below are designed to guide the assessment process and aid in the determination of assessment procedures to apply during the application phase. The issues addressed by this formal assessment inquiry are summarized as follows.

1. *Review of intended use*

2. *Review of simulation developer's effort*

   - Problem entity

   - Conceptual model derivation

   - Conceptual model structure

   - Conceptual model validation

   - Software development

   - Computerized model

   - Software verification

   - Historical use

   - Operational validation

   - Data validation

   - Internal security validation

   - Configuration management/quality assurance

       - Other

3. *Comparison of simulation capability to intended use*

   - Problem entity versus intended use

   - Conceptual model versus intended use

   - Computerized model versus intended use

1. *Review of intended use*

   - Is the intended use to be supported documented?

A23

- Are the elements needing to be modeled documented?
- Is the level of detail and fidelity needed for each entity documented?
- Are the outputs and measures of effectiveness (MOEs) needed documented?
- Are the required key inputs documented?
- Are the ranges of values necessary for the key inputs documented?
- Were special requirements to be levied on the model documented?

2. Review of simulation developer's effort

Problem entity
- Was the original intended use to be supported documented?
- Were the elements needed to be modeled documented?
- Was the level of detail and fidelity needed for each entity documented?
- Were the outputs and MOEs needed documented?
- Were the required key inputs documented?
- Were the ranges of values necessary for the key inputs documented?

Conceptual model derivation
- Was a plan developed for deriving the conceptual model?
- Does the conceptual model provide the level of detail and fidelity for each element that was required for the problem entity?
- Does the conceptual model provide the outputs and MOEs needed for the problem entity?
- Does the conceptual model provide the key inputs that the problem entity required?
- Does the conceptual model provide the range of values for key inputs required to the problem entity?
- Were the assumptions made and theories used during the derivation of the conceptual model documented?
- Were the assumptions made and theories used during the derivation of the conceptual model supported?

Conceptual model structure
- Does the conceptual model contain the elements required by the problem entity?
- Are the limitations and restrictions of the conceptual model documented?
- Are the limitations and restrictions of the conceptual model supported?

Conceptual model validation
- Was there a plan developed for validating the conceptual model?
- Were credible methods used for validating the conceptual model?
- Did the validation of the conceptual model include:
    - validation of the assumptions made?
    - validation of the theories used?
    - validation of the range of inputs expected?
    - validation of the data used?
    - validation of the outputs?

Software development
- Was a plan created for developing the software?

A24

- Was a software development methodology employed?
- Were design standards employed during the development cycle?
- Were coding standards employed during the development cycle?
- Was CASE (computer-aided software engineering) technology employed?
- Was a plan developed for testing the software during the development?
- Were credible methods used for testing the software during the development?
- Did the testing of the software during the development cycle include testing at the following levels:
    - module level?
    - unit level?
    - integration level?
- Were the following aspects of software tested:
    - interfaces?
    - assumptions?
    - inputs?
    - error cases?
    - paths?
    - theories?
- Were external interfaces documented?
- Were internal interfaces documented?

Computerized model
- Were the following aspects of the model documented:
    - elements modeled?
    - level of detail and fidelity of the elements?
    - outputs and MOEs?
    - ranges of values for outputs and MOE's?
    - inputs?
    - ranges of values for inputs?
    - assumptions made during development?
    - theories used in the software?
    - restrictions and limitations of the software?
    - changes made since the code was baselined?
- Do the following aspects of the model match the conceptual model:
    - elements modeled?
    - level of detail and fidelity of the elements?
    - outputs and MOE's?
    - ranges of values for the outputs and MOEs?
    - inputs?
    - ranges of values for inputs?
    - assumptions made during development?
    - theories used in the software?
    - restrictions and limitations of the software?
- Were assumptions made and theories used in the software supported?
- Were the restrictions and limitations in the software supported?
- Does the software documentation include the following?
    - complete user-level information?
    - complete programmer-level information?

A25

- complete design specification information?
- complete requirements information?
- Does the code contain the following attributes to make it more maintainable:
  - amplifying comments?
  - organized structure?
  - module-level input/output specification?

Software verification

- Was there a plan developed for verification of the software?
- Were credible methods used during the verification of the software?
- Did the verification of the software include verification of the following:
  - modules?
  - module-level interfaces?
  - assumptions?
  - limitations?
  - restrictions?
  - paths?
  - equations?
  - global data use?

Historical use

- Was a plan developed for the previous use of the model?
- Was the model configuration for that use clearly defined?
- Is that model configuration similar to that proposed for the intended use?
- Are the inputs, outputs, and MOEs from the previous use relevant to the intended of the model?
- Was the previous use of the model accepted by that user of the model?
- Was the previous use of the model assessed by this confidence assessment team?

Operational validation

- Was there a plan developed for the operational validation of the model?
- Were credible methods used for operational validation of the software?
- Did the operational validation of the software include validation of the following?
  - input-output relationships?
  - event ordering?
  - event occurrence?
  - extreme conditions?
  - stochastic variability?
  - sensitivity to key parameters?

Data validation

- Was a plan developed for collection of data from external sources?
- Were the data collected from external sources supported?
- Were the data collected from external sources checked for the following:
  - consistency?
  - units?
  - range of values?
- Was a plan developed for the generation of data not collected from external sources?

A26

- Were the generated data supported?
- Were the generated data checked for the following:
    - consistency
    - units?
    - range of values?

Internal security validation
- Was there a plan developed for the validation of internal security?
- Were credible methods employed during the internal security validation effort?
- Did the internal security validation effort include validation of the following:
    - code-locking techniques?
    - configuration management (CM) procedures?
    - possible inputs and outputs?
    - code constructs used?
    - nonexistence of viruses?

Configuration management/quality assurance (CM/QA)
- Were CM/QA controls provided during the following phases of the a model's development and use:
    - conceptual model derivation?
    - conceptual model validation?
    - software development?
    - development cycle software testing?
    - software verification?
    - each instance of previous use?
    - operational validation?
    - external data collection?
    - data generation?
    - internal security validation?
- Did these CM/QA controls provide the following for each phase:
    - Was there A CM/QA plan produced for each phase?
    - Were there procedures for verifying compliance with CM/QA controls?
    - Were there procedures for the controlled incorporation of the results of various analyses and testing back into the design and code of the simulation?
    - Were there procedures for controlling changes to interfaces?

Other
- Were any external reviews or audits (such as independent verification and validation), conducted on this simulation?

(Repeat relevant portions of the above question list for each review)
3. Comparison of simulation capability to intended use

Problem entity versus intended use
- Do the elements needing to be modeled for the intended use match the entities that were needed for the problem entity?
- Does the level of detail and fidelity required for the problem entity match the level of delta and fidelity required for the intended use of the model?

A-27

- Do the outputs and MOEs required for the intended use match the outputs and/or MOEs needed for the problem entity?
- Do the key inputs required for the problem entity match the key inputs required for the intended use of the model?
- Does the problem entity support the special requirements of the intended use?

Conceptual model versus intended use
- Are the elements needed for the intended use developed in the software?
- Is the level of detail and fidelity of the elements modeled in the software sufficient for the intended use of the model?
- Are the output and MOEs needed for the intended use provided by the software?
- Are the key inputs needed for the intended use provided by the software?
- Is the range of values for the input needed for the intended use provided by the software?
- Are the assumptions made during the development compatible with the model's intended use?
- Are the theories implemented during the development compatible with the model's intended use?
- Do any limitations or restrictions of the computerized model conflict with the model's intended use?
- Does the computerized model support the special requirements of the intended use?
- Were interface standards used during the development?

APPENDIX B - REPORTING

1.0 Findings of the Confidence Assessment Team will be reported on the Model Assessment Form B-1 and the disposition of the Finding will be recorded on the Assessment Disposition Form B-2.

Form B-1 Model Assessment Finding

- Finding #:

  - Conceptual Model Validation:
  - Software Verification:
  - Operational Validation:
  - (Optional Additional Procedures)

- Risk:

Form B-2 Assessment Procedure Report Outline

1. Purpose

   1.1 Description of Problem Area
   1.2 Procedures to be Employed

2. Expected Results

3. Procedures

   3.X Procedures Implemented
   3.X.1 Description of Problem Area
   3X.2 Results

4. Summary of Results

   4.1 Findings and Limitations
   4.2 Open Risk Areas
   4.3 Further Assessment Action

APPENDIX C - CONFIDENCE ASSESSMENT REPORT

1.0   The outline shown in Figure C-1 will be employed for the Confidence Assessment Report which will be attached to the HSM-PC Final report.

Figure C-1 Confidence Assessment Report Outline

1.   Preliminary

    1.1   Document Purpose
    1.2   Organization of Document

2.   Assessment Introduction

    2.1   Assessment Background
    2.2   Model Description
    2.3   Assessment Scope
    2.4   Assessment Team

3.   Assessment Approach

    3.1   Overview of Assessment Process
    3.2   Conceptual Model Validation Approach
    3.3   Software Verification Approach
    3.4   Operational Validation Test Approach
            (1) Operational Validation Test Approach
            (2) Synopsis of Operational Validation Test Plan

4.   Model Characterization and Description

5.   Assessment Findings

6.   Summary of Major Risk Areas

APPENDIX D- Text Files
Description of ASCII Files for use with HSM-PC.

There are five input files used in the HSM-PC. The simulation file is the first file required by the computational module. It contains the names of the other files that will be required, as well as simulation data such as desired output variables, the time of the simulation, and error tolerance information. The element file contains the biomechanical data needed to create the mathematical model of the HSM, including all of the rigid body, spring, beam, muscle, and hydrodynamic element information. The environment file describes the planes and springs in the environment, as well as any motion constraints imposed on the model. If planes are specified in the environment file, then a forcing function file is required to impart motion to the planes. Finally, an initialization file is used to store data from the model in a settled posture after it is exposed to gravity. Since the HSM-PC can take a long time to execute, it is useful to save the settled model position before forcing the model with environment planes. The initialization file must be created by the HSM-PC software and not the user.

The format of these files is detailed in this Appendix. The computational portion of the HSM will automatically look for the file "input.txt" to run a simulation.

SIMULATION INPUT FILE

HSM uses the text descriptions on each line to determine which variable is being read. The parameters do not have to be in the order shown below. All spaces in the file should be actual spaces and not tabs. Exclamation points (!) indicate a comment line.

Format: (example values are shown)

```
! Beginning and ending times for simulation, and time print interval for output file.
Beginning Time (sec):          0.0
Ending Time (sec):             0.1
Print Interval (sec):          0.01


Tolerance for Integrator:      1.0e-4


! Read or write initialization data (read or write)
Status:                        read


! File names.
Element file name:             9elem.txt
Environment file name:         9env.txt
Forcing function file name:    ff9lat.txt
Output file name:              out9lat.txt
Error file name:               error.txt
Initialization file name:      initial9.txt
```

! Metric units are kg-m-sec, English units are lbf-ft-sec.
Units for Output File: Metric
! Print Styles are: 1 = Normal,   2 = Step-by-Step
Style of Printing: 1
! Gravity (gravity in G, vector in earth coordinates)
Gravity: 1.0
Direction Vector: 0.0, 0.0, -1.0

! Acceptable Rigid Body data: Position, Velocity, Acceleration, Orientation, AngVel, AngAcc
! Acceptable Spring data: Magnitude
! Acceptable Plane data: Position
! Acceptable Muscle data: Stimulus, Stress, Activation, Strain Rate
! Acceptable Disc data: Force, Moment
! Acceptable Hydrodynamic data: Volume, Pressure
! The variables for printing should be preceded by the number of variables that will be printed
for that variable. At the end of each line, a * indicates all elements of that type should be
printed, while individual numbers indicate only output for that specific element should be
printed. Example: the muscle output specified below is for three variables (stress, strain, and
stimulus), and 2 muscles, #4 and #6.

Rigid Body Output: 6, Position, Velocity, Acceleration, Orientation, AngVel,
AngAcc, *
Spring Output: 1, Magnitude, *
Plane Output: 1, Position, *
Muscle Output: 3, Stress, Strain, Stimulus, 2,4,6
Disc Output: 2, Force, Moment, *
Hydrodynamic Output: 2, Volume, Pressure, *

ELEMENT INPUT FILE

HSM uses the text descriptions on each line to determine which variable is being read. Since other elements connect to Rigid Bodies, it is important that all Rigid Body elements appear in the file first! All spaces in the file should be actual spaces and not tabs. Exclamation points (!) indicate a comment line.

Format: (example values are shown)

```
RIGID_BODY 1
Name Head
Mass 4.6
Inertia 0.0333 0.0354 0.0307          ! x, y, and z values
Position 0.04 0 -0.2
Velocity 0.0 0.0 0.0
Euler angles 0.0 0.0 0.0              ! yaw, pitch, roll at start
Angular velocity 0.0 0.0 0.0
Secondary node # 2
Name Right occipital condyle (medial aspect)
Position -0.018 0.02 0.052           ! secondary node coordinates are  relative to
primary node
Secondary node # 3
Name Right occipital condyle (anterior aspect)
Position -0.01 0.015 0.052
! so on for other secondary nodes


MUSCLE 1
Name Right Rectus Capitis Posterior Minor
Sliding Nodes 0
Endpoints 2,2 1,11
Cross Sectional Area 0.0001


MUSCLE 18
Name Left Longissimus Capitis C4-Head
Sliding Nodes 3                       ! two endpoints, and three intermediary points
Endpoints 5,3 4,31 3,33 2,24 1,22    ! body, node combinations
Cross Sectional Area 0.000017


SPRING 26
Name Right Ligamentum Flavum C4-C3
Class T                              ! T = tension-only, B = dual-acting, C =
compression only
Stiffness 15000 0                    ! linear and cubic stiffnesses
Endpoints RB,5,12 RB,4,12
Damping 0.1
```

DISC 3
Name C1-C2 Intervertebral Disc
Endpoints 2,5 3,6   ! upper endplate is listed first (body, node), then
lower endplate
Stiffness 5.00E+05 5 40 90   ! axial, bending x, bending y, and torsional
stiffness
Damping 0.002 0.002   ! axial and bending damping
Shear 17.7 13.4   ! x and y shear factors

HYDRODYNAMIC 6
Name C3-C4 Left Articular Facet
Bodies 5,4   ! lower and upper body connections
Endpoints 22,23,24 19,20,21   ! three endpoints on lower body, and 3 on upper
body
Bulk Modulus 4.00E+07
Damping 0.0001

ENVIRONMENT INPUT FILE

HSM uses the text descriptions on each line to determine which variable is being read. All spaces in the file should be actual spaces and not tabs. Exclamation points (!) indicate a comment line.

Format: (example values are shown)

Constraint 1
Name Constraint 1
Applied to Rigid Body: 9
Constrained Axes: 0 1 1 1 1 1   ! x, y, z translational, then x, y, z rotational. 0 means not constrained, 1 means constrained.

Plane 1
Name Plane 1
Pnt1 -.1 10 -10
Pnt2 -.1 10 10
Pnt3 -.1 -10   0
Stiffness 500.0 50.0

FORCING FUNCTION INPUT FILE

HSM uses the text descriptions on each line to determine which variable is being read. The parameters do not have to be in the order shown below. All spaces in the file should be actual spaces and not tabs. Exclamation points (!) indicate a comment line.

Format: (example values are shown)

```
PLANE                    ! if forcing function is applied to a plane
Element 1                ! plane number for this ff
Name FF1
Direction 1.0 0.0 0.0
! Direction is a unit vector which specifies the direction of action through the plane
! with respect to the plane's normal vector. Positive is assumed to be in the same
! direction as the normal to the plane. The motion resulting from the forcing function
! will be in the direction of the Direction unit vector.
Source Standard
! Type designates weather the function is a standard function or a user suppplied
! set of [t,amplitude] pairs in a user file
Function Triangular
! Function designates the type of standard function
FFType Acceleration
! FFType can be either position, velocity, or acceleration
Begin 0.0
End   0.1
Peak  0.05
Offset 70.0
Amplitude 0.0


NODE                     ! this forcing function is applied directly to a node
Element 9, 1             ! body, node for application
Name  Forcing Function 1
Direction 0.0 1.0 0.0
Source Standard
FFType Acceleration
Function Rectangular
Begin 0.0
End 0.09
Peak 0.0
Offset 0.0
Amplitude 50.0
```