

MULTIBODY DYNAMICS MODEL BUILDING USING GRAPHICAL INTERFACES

Glenn A. Macala
Member of Technical Staff
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

ABSTRACT

In recent years, the extremely laborious task of manually deriving equations of motion for the simulation of multibody spacecraft dynamics has largely been eliminated. Instead, the dynamicist now works with commonly available general purpose dynamics simulation programs which generate the equations of motion either explicitly or implicitly via computer codes. The user interface to these programs has predominantly been via input data files, each with its own required format and peculiarities, causing errors and frustrations during program setup. This paper describes recent progress in a more natural method of data input for dynamics programs: the graphical interface.

INTRODUCTION

User interfaces to the variety of commonly used engineering software typically consist of input data files, each with its own required format and peculiarities. This situation inevitably causes errors and frustrations during program setup. Experience has shown that the development of graphical interfaces, full-screen, protected-field editors, and input error detection and correction schemes for commonly used programs not only alleviates most of these problems, but also increases the engineer's productivity. Building these interfaces is not a trivial matter, however, and more often than not the available resources are spent in creating program capabilities rather than easy to use interfaces to the programs.

At the Jet Propulsion Laboratory, a project known as CASCADE (Computer Aided System Control And Dynamics Environment) has been charged with the task of producing graphical interfaces for a set of programs covering the design, analysis, and simulation of control systems for spacecraft. The CASCADE system has made significant progress in the area of interfaces for multibody dynamics. Prototype interfaces now exist for three programs under the general category of dynamics: a mass property calculator, an equations of motion generator for rigid multibody systems, and an equations of motion generator for elastic multibody systems. An additional fallout of this work has been the development of a PC based program to depict three-dimensional animations of rigid multibody spacecraft. The following sections describe the development of these interfaces, in chronological fashion, and will provide an introduction to the existing prototype interfaces and animation program.

MASS PROPERTY FACILITY

Complex spacecraft typically consist of hundreds of pieces of structure. The mass properties of the spacecraft are determined by weights and balances engineers via combining the individual mass properties of these elements according to their assembled locations and orientations on the spacecraft. Dynamics and control engineers need to know mass properties for major portions of the spacecraft in order to perform multibody dynamics simulations. Therefore, comprehensive mass properties lists of spacecraft subassemblies are generated for them by the weights and balances engineers. Invariably, however, there is some combination of subassemblies that is not on a list and is subsequently found to be needed.

A locally developed program that computes composite mass properties for a collection of spacecraft subassemblies, commonly referred to as bodies, had been used for years to calculate the mass properties for combinations of subassemblies missing from the master list. This program used the parallel axis theorem to calculate the mass properties of collections of bodies whose individual mass properties were known. The original user interface to this program was a "question and answer" type but had subsequently been changed to an input data file type. This program was a natural for development of a graphical interface. It was simple, yet had elements common to the more complex dynamics programs in use. Therefore, the first interface was developed using this mass property calculator and was named the MPF - Mass Property Facility.

The first development decision to be made was a hardware issue. What computers and terminals would comprise the system and where would the software be based? The engineers who would be using the program had IBM PC's with Hercules graphics cards. However, the majority of the dynamics software was hosted on a VAX 11-780 and could not be ported to the PC's because of sheer size. Therefore, the decision was made to use the PC's as terminal emulators and to host all of the interface software on the VAX. The PC based terminal emulator was able to emulate a VT100 alpha-numeric terminal and a Tektronix 4010 graphics terminal and was able to switch between emulations via a software command (receipt of certain byte sequences). Therefore, detailed line drawings and cursor position reporting could be done in the Tektronix mode. Full-screen editor style data entry could be done in the VT100 mode.

The resulting features of the MPF can best be conveyed by the following commentary on how an engineer would use the program to calculate the composite mass properties of a set of bodies. The engineer starts the terminal emulator on his PC, connects to the VAX, and logs on to his account. He gives the command to run the MPF. Figure 1 depicts the opening graphics menu of the MPF. The area which takes up most of the left half of the screen contains labeled boxes which will activate a host of functions from file manipulation to operations on the bodies of the system. The upper right portion of the screen contains small pictures of common objects: a cylinder, a rectangle, and a frustrum. A scale is also located there to indicate relative size in pixels. The lower right corner of the screen will contain a 5/12 scaled down version of the current full-screen graphics page on which the system of bodies is drawn. A two body system is shown in this figure.

The engineer selects bodies from the upper right portion of the screen and

places them on the full-screen graphics page. The bodies are to be used as sketches of the bodies in the system under study. Their sizes, shapes, and positions on the graphic screen have no effect on the subsequent mass property calculations. They are used by the engineer as mnemonic aids to represent the bodies of the system under study.

The engineer proceeds by using the PC's arrow keys to move the on-screen cursor until it points to one of the bodies in the upper right portion of the screen. He then pushes the space bar to select the body. He is asked the screen dimension of the body and may accept a default by pushing the "Enter" key. The screen clears to the graphics page and a cursor appears. The arrow keys are used to move the cursor until the desired position for the body is located. Pushing the space bar causes the body to be drawn at that screen location. He may continue to place bodies of this type around the screen, or return to the main menu by pushing the "R" key. At the main menu, he may similarly select other types of bodies to be placed in the system.

After having selected and placed the bodies of the system on the graphics page, mass properties for each body must be specified. The engineer selects the "Values" function from the main menu, either by using the cursor keys to "point and shoot" as with the bodies, or alternatively, by pushing the "V" key. Note that each function on the main menu has one capital letter. If the PC key corresponding to this letter is pushed, that function will be activated.

Once "Values" has been selected, the full-screen graphics page consisting of the previously placed bodies is presented to the engineer. He selects a body, using the cursor method, and a full-screen menu appears for that body as shown in Figure 2. This menu is actually a full-screen, protected-field editor that lets the engineer type in a name for the body, its mass, mass center location, and inertia tensor. These fields are shown in reverse video and the cursor cannot move outside of them. Therefore, the engineer need not worry about where he needs to type data. The bottom line of the menu contains some help information, but on-line help is available by pushing the F10 function key. The VAX's help facility is then activated and the appropriate help text and subjects for this menu are available for perusal by the user. When he is done reading the help text, pushing the "Enter" key will return the full-screen menu to the display.

When the engineer has completed filling in the data for this menu, he pushes the Alt-F1 function key combination, and the data can be saved. After similarly entering data for each of the other bodies in the system, he is ready to calculate the resulting system mass properties. He does this by returning to the main menu and selecting "System". An asterisk appears in the box next to "System" indicating that the items enclosed by the larger box, "Move", "Delete", and "Values", now will effect the entire system rather than individual bodies of the system. He then selects "Values" and the program proceeds to read all of the data that has been entered for all bodies in the system. If an error is detected during the reading of the data, say the engineer forgot to enter the mass of a body called Antenna, the program would display the following message on the screen: "An error has been detected. The mass for the body named Antenna must be a floating point number. Push the Enter key and the cursor will be positioned at the field where the error occurred. You may then correct the error, save the data, and try again." When the engineer pushes the "Enter" key, the full-screen field-

protected mass property editor appears for the Antenna body. The cursor appears in the "Mass" field and the engineer would see that he forgot to enter the mass. He would enter the mass, save the data, and return to the main menu where he would again select "Values". If no errors were detected this time, a file would be automatically written in the manner required by the original mass property program.

The MPF then runs the original mass property program and feeds it this file for its required input. The old program calculates the mass properties of the system and then writes the answer to another file. The MPF then reads this file and displays it to the engineer using the full-screen mass property menu format.

Note that the engineer may save his work at any time by selecting "Save picture file" from the main menu and specifying a name for the file. He may copy previously created mass property data into currently existing bodies by selecting "Get mass property file" from the main menu. A full-screen file selection menu appears, as shown in Figure 3, and the user selects a file by using the cursor keys to highlight the file name of interest and then pushing the "Enter" key. The main menu returns and the user then selects the "Deposit" function. The graphics screen appears and the user points at the body he wishes to deposit the data into, using the cursor, and then presses the space bar. If he returns to the main menu and selects "Values", he can then inspect the data just deposited into this body via the full-screen mass property editor.

He may delete bodies from the system by selecting "Delete" from the main menu. He may "pretty up" the graphics page by moving the bodies of the system around using "Move". All of these functions work similarly to the methods described earlier: the graphics page appears, the user selects the body he is manipulating using the cursor, and so forth.

SYMBOLIC DYNAMICS FACILITY

Once the MPF had been created and exercised, improvements were made and bugs were removed. It was then time to move on to the next interface in the series: the SDF - Symbolic Dynamics Facility. The basis for this interface was the SD/Exact [1] program. This program can compute the symbolic equations of motion for a system of hinge-connected rigid bodies. This is typically how the dynamics and control engineer represents articulated portions of a spacecraft for simulation purposes. For example, an antenna may be attached to the main bus of a spacecraft by a motor/gimbal combination that allows the antenna to be moved relative to the bus. This antenna may be required to point at a receiver on the Earth. It would be of interest to the dynamics and control engineer to simulate this two-body system to confirm that it met its pointing requirements.

The SD/Exact program requires that a file be created for input to the program. This file contains a description of each body in the system to be simulated. The description for each body consists of specifying its mass properties and also how it connects to other bodies in the system. Note that the MPF allows each body's mass properties to be specified. Accordingly, we had only to augment the MPF interface with a scheme to record body connection information in order to create the SDF interface.

Figure 4 presents the opening menu of the SDF. Note the similarity to the menu of the MPF (Figure 1). The most notable difference is in the upper right hand corner of the menu. Now in addition to bodies, there are "hinge" or "joint" icons. These icons allow the user to select 1, 2, or 3 degree-of-freedom rotational hinges or a 1 degree-of-freedom translational hinge to connect between bodies of the system. These were chosen because they are the hinge types supported by the SD/Exact program.

The engineer proceeds to set up the system of bodies as he did with the MPF. Once he has finished placing the bodies of the system on the full-screen graphics page and specifying their mass properties via the "Values" function, he concentrates on placing the hinges. This proceeds in a manner analogous to selecting and placing the bodies. For example, if the engineer wished to allow 1 degree of rotational freedom between two bodies of his system, he would select the 1 degree-of-freedom joint and place it between these bodies on the full-screen graphics page. He would then return to the main menu and select the "Connect" function. The graphics screen would re-appear and he would select the hinge he just placed using the cursor. He would then be asked to select the first body the hinge connects. He would use the cursor to do this. He would then be asked to select the second body the hinge connects. Again, he would "point and shoot" with the cursor. The SDF would then record this connection information.

One last major step must be taken before the system is completely specified. Just as "Values" is used to enter mass properties for bodies in the system, it is also used to enter information about the hinges in the system. Figure 5 shows the full-screen hinge menu that appears when "Values" is used on a 1 degree-of-freedom rotational hinge. This menu works like the mass properties menu: protected fields allow the user to enter data only in certain fields. As with a body, the user fills in a name for the hinge and the location of the hinge. Now, however, he must specify the direction cosines of the hinge: along what spatial vector it turns. Also note that the names of the bodies connected by this hinge are displayed at the top of the menu. This is the result of using the "Connect" function.

Once all hinges have been specified via "Values", the system is completely specified and the user can select "Generate equations" from the main menu. The SDF then reads all the data and looks for errors. If an error is found, as with the MPF, the user is informed and automatically placed in the menu of interest, mass properties or hinge properties, to correct the error. Once all errors have been corrected, the SDF automatically writes the required input file for SD/Exact and runs SD/Exact. SD/Exact generates equations of motion for the system, in terms of FORTRAN subroutines, and stops. The SDF then takes these subroutines and appends them to a bare-bones shell of an ACSL [2] simulation program it creates based on the problem. At this point, the user may exit the SDF and use this simulation program as he normally would have had he written it himself by hand.

FLEXIBLE DYNAMICS FACILITY

After the SDF had been completed, it was time to move on to the last interface in the dynamics series: the FDF - Flexible Dynamics Facility. The basis for this interface was the DISCOS [3] program. This program can compute the equations of motion for a system of hinge-connected rigid and elastic bodies and is also a self-contained simulation program. Typically,

the dynamics and control engineer uses DISCOS as a final, full-fidelity check on his control system design. For example, the Earth pointing antenna example mentioned in the SDF section was modeled there as a rigid antenna connected to a rigid bus. The antenna may have actually been elastic in nature but, for simplicity, may have been modeled as a rigid body during the design phase. Now the engineer wants to use DISCOS to simulate it as elastic to make sure that his design assumptions were valid and that his design still meets requirements.

The DISCOS program requires that a file be created for input to the program. This file contains a description of each body in the system to be simulated. The description for each rigid body consists of specifying its mass properties and also how it connects to other bodies in the system. In addition, each elastic body in the system must have its elasticity properties described in terms of its eigenvalues and eigenvectors. Note that the SDF provides for the specification of mass properties and connection properties. Accordingly, we had only to augment the SDF interface with a scheme to record a body's elasticity information in order to create the FDF interface.

Figure 6 presents the opening menu of the FDF. Note the similarity to the menu of the SDF (Figure 4). The most notable change is in the upper right hand corner of the menu. The 4 types of hinges of the SDF have now been replaced by a general 6 degree-of-freedom hinge: 3 translations and 3 rotations.

The engineer proceeds to set up the system of bodies and hinges as he did with the SDF. Once he has finished placing the bodies and hinges of the system on the full-screen graphics page, he then uses "Values" to specify their mass properties and hinge properties and "Connect" to specify hinge connections between bodies. If a body is rigid, this procedure is identical to that for the SDF with minor variations in the content of the hinge menus to accommodate the DISCOS type of hinge. If a body in the system is elastic, however, and the engineer uses "Values" to select that body, the menu of Figure 7 is displayed.

For DISCOS, an elastic body consists of a set of joints where lumped masses and their self-inertias reside. The locations and mass properties of joints are available from finite element analysis of the elastic body. The menu shown in Figure 7 requires the engineer to place the "joints" or "grid points" on the elastic body in question. Once this has been done, in the usual "point and shoot" manner, the engineer may specify the mass properties of each joint by selecting the "Mass properties" function on this menu. He is then able to select the joint of interest with the cursor and fill in the usual mass properties menu. The modes (eigenvalues) and their damping are specified by selecting the "Modes" function from the menu. Again, a full-screen editor type menu opens up to allow the engineer to type in the required information. The mode shapes of each joint are specified by using the "Mode shapes" function from this menu. Once again, the engineer "points and shoots" to select a joint. A full-screen menu appears and can then be filled in with mode shape information.

Once the engineer has placed all joints for each flexible body, and has entered all elastic data, the system is completely specified and the user can select "Setup Discos file" from the main menu. The FDF then reads all of the data and looks for errors. If an error is found, as with the SDF and MPF,

the user is informed and automatically placed in the menu of interest to correct the error. Once all errors have been corrected, the FDF automatically writes the required input file for DISCOS. At this point, the user may exit the FDF and execute the DISCOS program as he normally would have had he written the input data file himself by hand.

WRITEANI

A PC based program, WRITEANI, has been a by-product of the graphical interface work done on the SDF. The program itself was written for amusement by the author on his home computer, but was subsequently integrated with the SDF. WRITEANI reads an "animation" file created by the SDF and then automatically writes a FORTRAN animation program which can be run on a PC equipped with a Hercules graphics card. The resulting animation program then reads the file containing the results (hinge positions at each time step) of an SDF rigid multibody simulation run on the VAX. One 3-D picture of the multibody system is drawn for each simulation time step. Hidden-line removal is performed on each body, but not on overlapping bodies. Perspective is also provided. Figure 8 shows one frame of such a spacecraft animation.

Although the animation program was originally written for amusement purposes, there have been several occasions when engineering insight has been obtained by observing an animation of the results of a simulation. The program has also been found useful for explaining complex dynamic effects to interested parties.

SUMMARY

The summaries of the various dynamics interfaces presented in this paper have only touched on their basic operation. These interfaces present the engineer with a logical, consistent format for visually assembling dynamic systems and for specifying their properties along with on-line help and extensive error checking. Although some of these dynamics programs are very complex in nature and require lengthy study and use in order to completely master their capabilities, it has been estimated that setup time for these programs has been reduced by an order of magnitude through the use of these interfaces. These interfaces are considered to be working prototypes. It would be desirable to port them to workstation class computers in the future for improvements in graphics drawing speed and also for incorporating windowing techniques into the interfaces.

ACKNOWLEDGMENT

The research described in this paper was performed by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

REFERENCES

1. Rosenthal, D. E., and Sherman, M. A., "High Performance Multibody Simulations via Symbolic Equation Manipulation and Kane's Method," Journal of the Astronautical Sciences, Vol. 34, No. 3, July-September 1986, pp. 223-239.

2. Mitchell and Gauthier, "Advanced Continuous Simulation Language (ACSL) Reference Manual," 1987, Mitchell and Gauthier Associates, Concord, Mass.
3. Bodley, C. A., et al., "A Digital Computer Program for the Dynamic Interaction simulation of Controls and Structure (DISCOS)," NASA Technical Paper 1219, 1978.

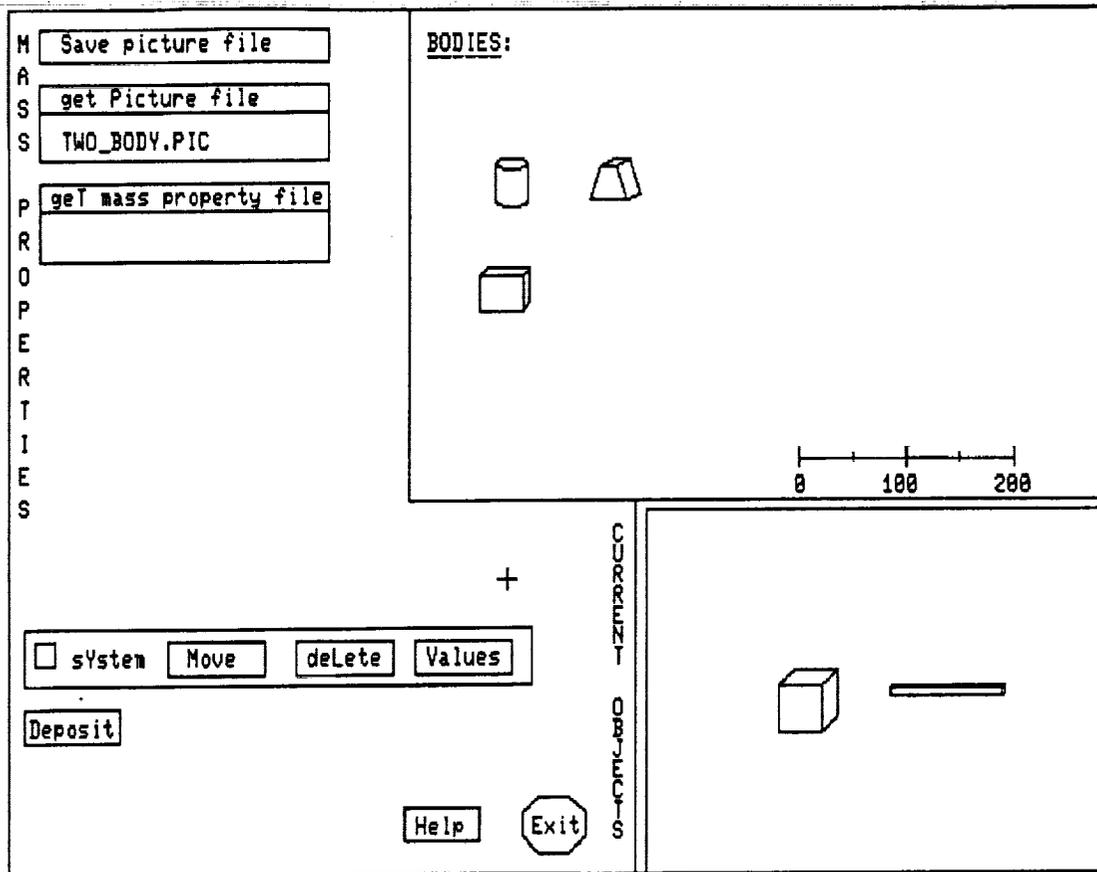


Figure 1. MPF Main Menu

ORIGINAL PAGE IS
OF POOR QUALITY

MASS PROPERTIES		
NAME OF BODY Rotor	MASS 1724.686	UNITS: Length - m Mass - Kg Inertia- Kg m**2
X -0.6959E-02	CENTER OF MASS Y -0.1686	Z -0.6359
IXX 1217.0	INERTIA TENSOR IXY -55.08	IXZ -4.020
IYY -55.08	IYY 2894.0	IYZ -85.08
IZX -4.020	IZY -85.08	IZZ 3389.0
Overstrike Mode Active		
Help: F10 Field: F1/F2 Mode: F3 Units: F4 Exit: Alt-F1		

Figure 2. Full-Screen, Protected-Field Mass Properties Editor

MASS PROPERTY FILES		
Select an item using the ARROW KEYS: the item will appear in reverse video. Complete the selection by hitting ENTER.		
BODY01	BODY02	COMBINED_SYSTEM
DEPOSIT	MAG_BOOM	ROTOR
USR1:[GMCASCADE.CASCADE.TWO_BODY]*.MPD		
Help: F10 Type Filename: F1 Directory: F2 Quit: Ctrl-Z		

Figure 3. Full-Screen File Selection Menu

ORIGINAL PAGE IS
OF POOR QUALITY

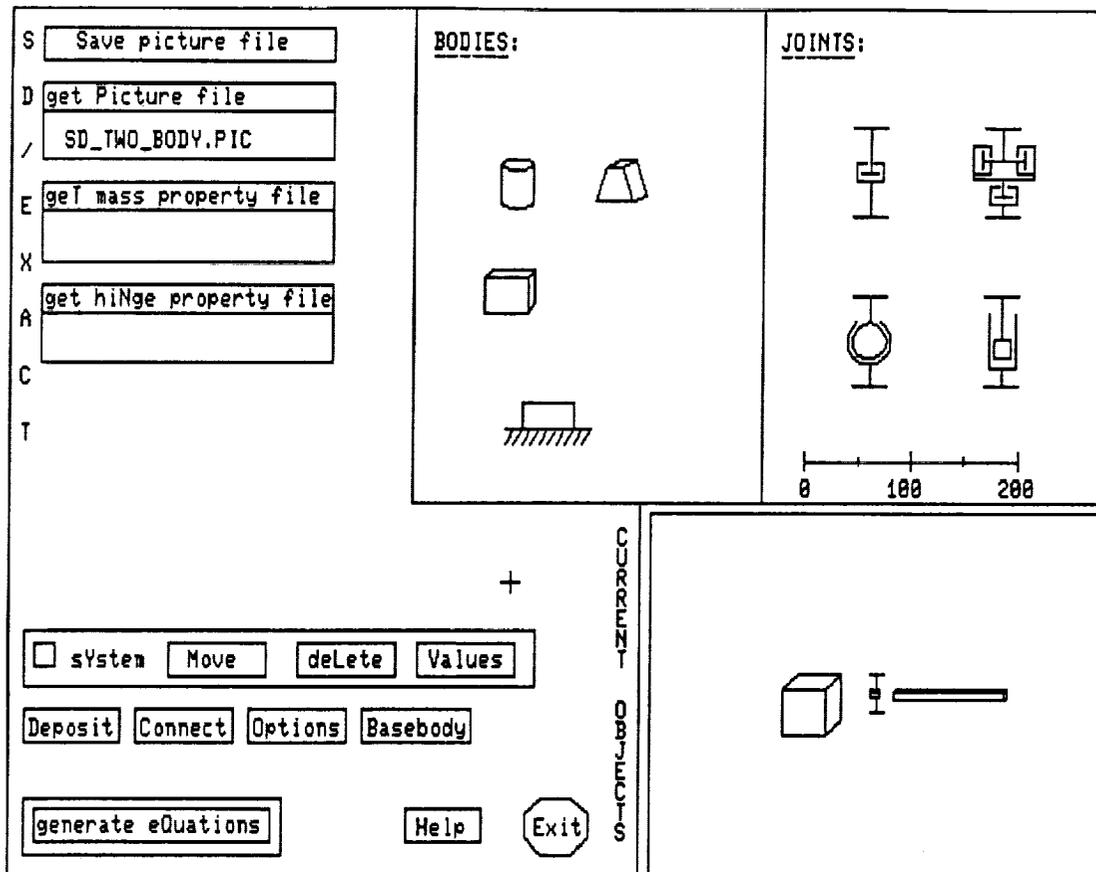


Figure 4. SDF Main Menu

HINGE NAME			PIN JOINT		
Mutation Damper			Connects:		
			Mag Boom		
			Rotor		
			Motion: Free		
			Length: n		
HINGE POINT LOCATION					
X	Y	Z			
0.0	1.0	-0.3			
DIRECTION COSINES FOR HINGE PIN					
X	Y	Z			
1.0	0.0	0.0			
Overstrike Mode Active					
Help:F10 Field:F1/F2 Mode:F3 Motion:F4 Lgth:F5 Exit:Alt-F1					

Figure 5. Full-Screen, Protected-Field Hinge Property Editor

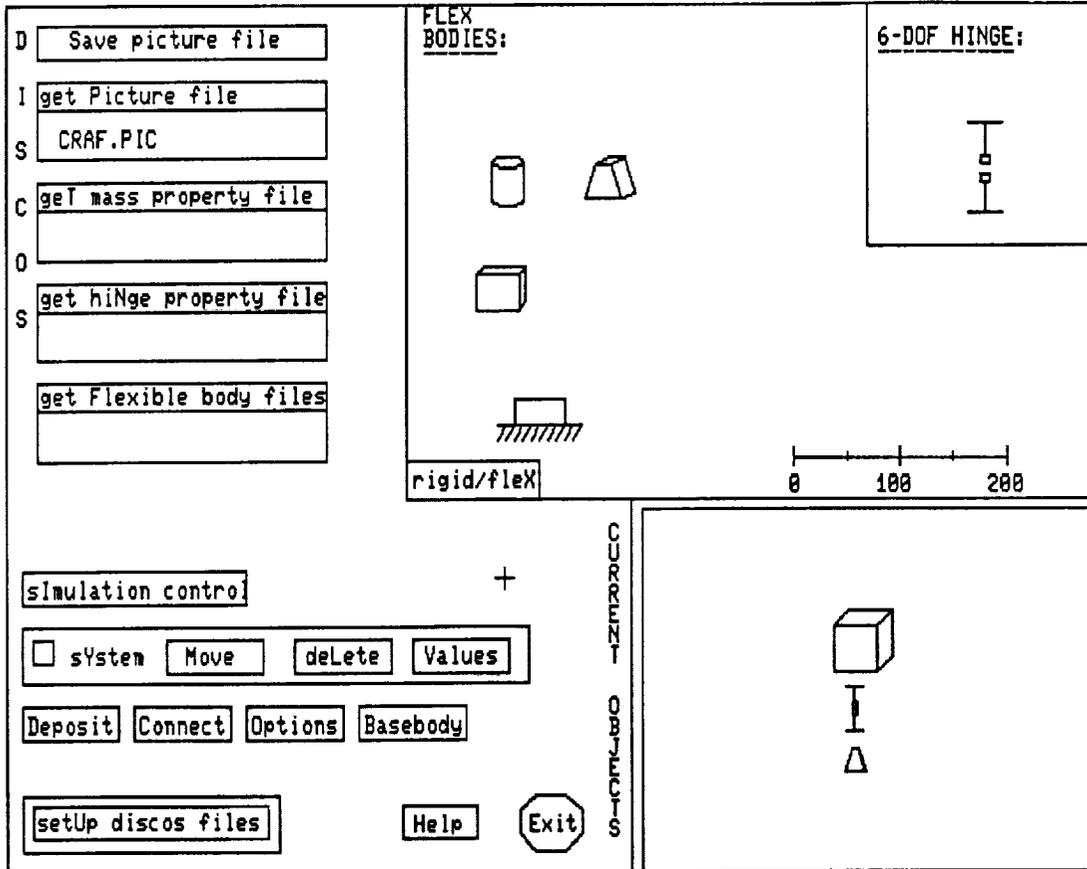


Figure 6. FDF Main Menu

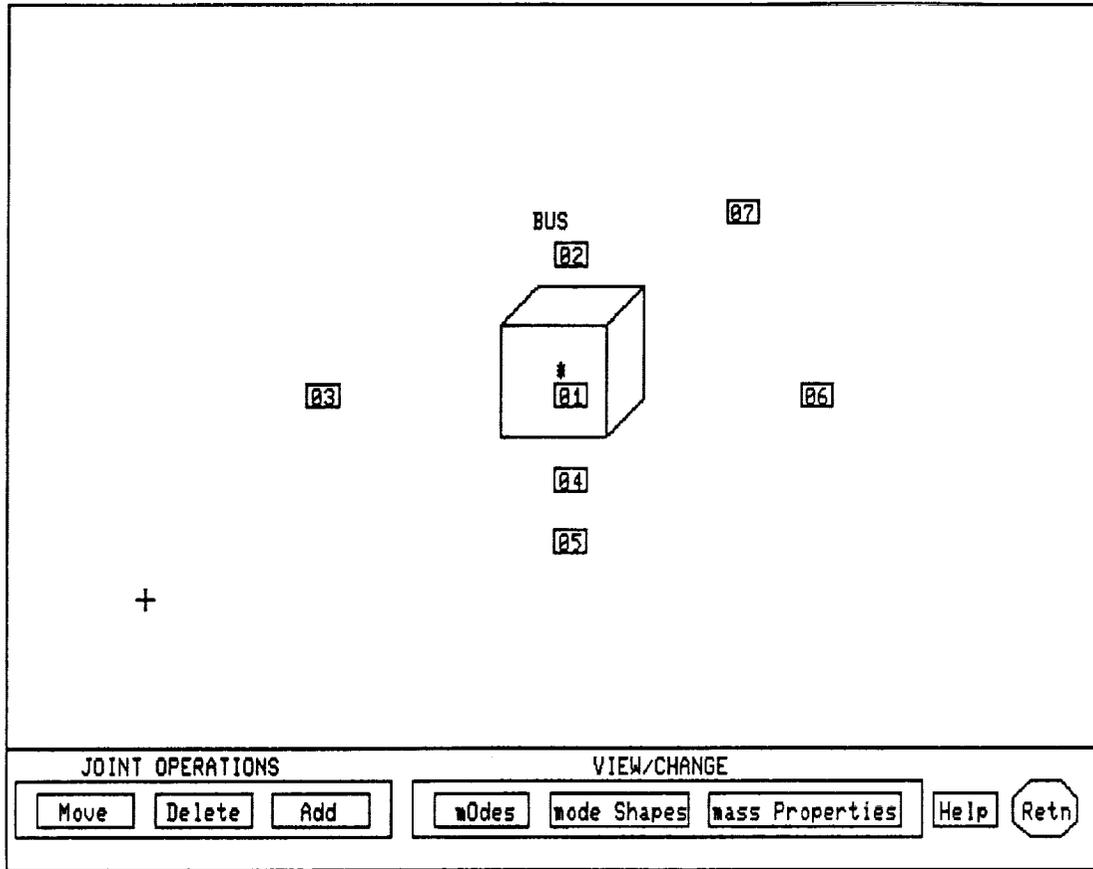


Figure 7. FDF Values Menu for Flexible Body

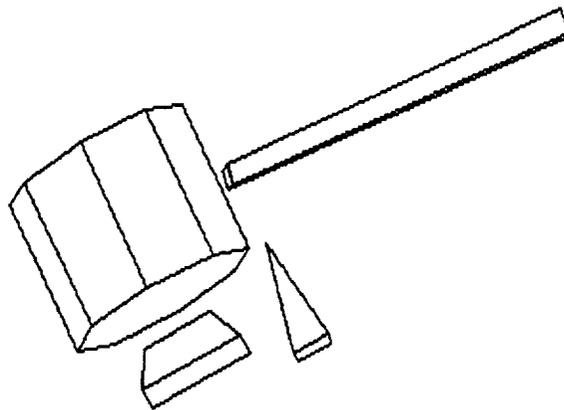


Figure 8. One Frame from Animation Display