N87-16332

PSEUDO-PROTOTYPING OF AEROSPACE MECHANICAL DYNAMIC SYSTEMS WITH A GENERALIZED COMPUTER PROGRAM

Vikram N. Sohoni
and
Milton A. Chace

1. INTRODUCTION

Mechanical system CAE is a distinct, relatively new field of computer-aided mechanical engineering. It is complimentary to neighboring fields such as geometrical or solid modelling, finite element stress analysis and vibration analysis. The functional distinction of the field is that it determines the time-dependent behavior of entire interconnected systems of parts and other elements, ranging through angular displacements which may be sufficiently large to require non-linear solution.

Engineers responsible for mechanical design are particularly assisted by mechanical system CAE, since this technology enables them, accurately and early, to predict the behavior of machinery or vehicles for many variations in design. Behavior assessment can be done even before the first prototype exists, or in compliment to prototype or product testing. The assistance for aerospace design is especially compelling due to extreme requirements for reliable performance, and the difficulty of providing a zero gravity environment for physical testing.

For many years machine designers appraised the performance of devices such as four-bars, slider-cranks and cam-follower mechanisms, utilizing the assumption of kinematic behavior. Solutions were essentially geometric and were usually performed graphically. The first computer implementations were limited to kinematics. Two early programs were KAM [1] (Kinematic Analysis Method, 1964) and COMMEND [2] (Computer Oriented Mechanical Engineering Design, 1967). Both programs were created by IBM. KAM solved for displacement, velocity, acceleration and reaction force of a limited number class of spatial linkages, notably vehicle suspensions. COMMEND was a planar program particularly intended for computer-aided engineering of IBM's mechanical products.

The original version of DRAM was completed in 1969, at The University of Michigan [3, 4, 5] through the efforts of Professor Milton Chace and Michael

*Mechanical Dynamics, Inc., Ann Arbor, Michigan.

Korybalski. At that time it was named DAMN (Dynamic Analysis of Mechanical Networks). It was historically the first generalized (type-variant) program to provide time response of multifreedom, constrained, mechanical machinery undergoing large-displacement behavior. Major improvements and additions were made to the program by D.A. Smith in his doctoral thesis work over the period of 1968 to 1971 [6]. Since then, DRAM (Dynamic Response of Articulated Machinery) has undergone continuous improvement particularly through the efforts of John C. Angell [7, 8].

The ADAMS (Automated Dynamic Analysis of Mechanical Systems) program was originally completed in 1973 as doctoral thesis work by Nicolae Orlandea [9, 10]. ADAMS was designed as a three-dimensional, large-displacement dynamic program, without however some of the capabilities for impact and surface-to-surface contact possessed by DRAM. ADAMS also utilized a different coordinate scheme than DRAM and involved sparse matrix methods in the equation solutions. Again, major improvements and additions have been made to the original ADAMS code; most of them by J. Angell, R. Rampalli, and T. Wielenga. An important adjunct to ADAMS, ADAMS/MODAL, has recently been completed by V.N. Sohoni and J. Whitesell [11]. ADAMS/MODAL performs automatic linearization of mechanical systems (this of course requires circumstances of small-displacement), then proceeds to determine the system modal characteristics and time dependent response.

In this paper the scope and analytical methods involved in ADAMS are reviewed, followed by a discussion of some aerospace examples. ADAMS and DRAM are intended for direct use by engineers and senior designers. For this reason much effort has been devoted to facilitating ease of use. The programs self-formulate all of the relations described in the following sections of the mechanical system. Computer graphics is utilized to provide output in a flexible, comprehensive form.

ADAMS and DRAM are provided as proprietary software by Mechanical Dynamics, Inc., 3055 Plymouth Road, Ann Arbor, Michigan.

2. MODELLING OF MECHANICAL SYSTEMS

ADAMS is a general, fully three-dimensional code. For a given mechanical system, each rigid part is represented by six coordinates. A local part reference frame is attached to each part. The translational displacements of each part are measured as displacements of the local part reference frame origin along the three global coordinate axes. To orient the part in space, three Euler angles are employed.

Interactions between parts in a mechanical system can generally be classified into the following three categories.

1. Kinematic
2. Compliant
3. Elastic

2.1 Kinematic Connections

Two parts can be connected by a kinematic connection or joint. These connections are such that they only allow certain types of relative motions between the connected parts. The equations representing the relationships implied by the joint are formulated as non-linear algebraic equations in terms of the coordinates of the two parts connected and the geometry of the joint.

To illustrate the formulation of these algebraic relationships consider parts i and j as shown in Figure 1. These two parts are connected by a spherical joint, for example, at markers[1] i_k and j_k on parts i and j, respectively. The spherical joint constraint requires that these two markers be coincident at all times. Writing the vector equation around the loop 0, 0_i, 0_j, 0:

$$\underline{R}_i + \underline{T}_i \; \underline{r}_{i_k} + \underline{p} - \underline{T}_j \; \underline{r}_{j_k} - \underline{R}_j = 0 \qquad (2.1)$$

where

- $\underline{R}$ = Position vector to the origin of the local part reference frame from the global origin, relative to the global frame.
- $\underline{T}$ = Transformation matrix from local part reference frame to global reference frame.
- $\underline{r}$ = Position vector between points in a part relative to the local part reference frame.
- $\underline{p}$ = Vector from marker j_k to marker i_k, relative to global reference frame.
- i,j = Part numbers being connected by the joint.
- i_k, j_k = Indices of the markers being connected by the joint

Since markers i_k and j_k are always coincident

$$\underline{p} = 0$$

From equation 2.1

$$\underline{R}_i + \underline{T}_i \; \underline{r}_{ik} - \underline{T}_j \; \underline{r}_{jk} - \underline{R}_j = 0 \qquad (2.2)$$

Equation 2.2 is a vector equation, equivalent to three scalar equations. Parts i and j have twelve degrees of freedom. However, the presence of three scalar algebraic constraint equations reduces the degrees of freedom to nine. In a similar manner, using vector equations, the constraint equations for all other possible physical joint types have been derived and are automatically invoked by ADAMS, depending on the mechanism example input.

[1] The term "marker" denotes the combination of a point (indicating translational position) and a triad of unit vectors (indicating orientation).

In general the algebraic equations representing joints can be written as

$$\underline{\phi}(\underline{q}, \underline{\dot{q}}, t) = 0 \qquad (2.3)$$

where

$\underline{\phi}$ = Vector of constraint equations
$\underline{q}$ = Vector of n coordinates
$\underline{t}$ = Time
$\underline{\dot{q}}$ = Vector of Velocities

Because of the generality of this form, user specified constraints can also be included, such as the interaction between variables due to controllers. ADAMS has a large library of kinematic joints. Some of these are:

1. Spherical
2. Rotational
3. Translational
4. Universal
5. Cylindrical
6. Gear

2.2 Compliant Elements

The second type of interaction between parts is through compliant elements. These do not reduce degrees of freedom. However, the forces developed in compliant elements are functions of the displacement and velocities of the parts on which these compliant elements act. Consider two parts i and j as shown in Figure 2. These parts are connected by a compliant element C at markers i_ℓ and j_ℓ in parts i and j, respectively. Force $\underline{f}$ developed in the compliant element acts with equal magnitude but opposite direction on markers i_ℓ and j_ℓ.

The simplest compliant element is a linear spring-damper. The force developed in such an element can be written as

$$\underline{f} = [k(l - l_0) + cv]\underline{p} \qquad (2.4)$$

where

$\underline{f}$ = force vector due to compliant element
$\underline{k}$ = spring constant
l_0 = free length of spring
l = distance between points pi and pj
c = damping coefficient
V = velocity of marker j_ℓ with respect to marker i_ℓ along the line connecting them
$\underline{p}$ = unit vector along the line from marker i_ℓ to j_ℓ

Force f acts at marker i_ℓ and an equal and opposite force -f at marker j_ℓ. The resulting moments on the two parts are

$$\underline{M}_i = \underline{r}_i \times \underline{f} \tag{2.5}$$

and $\underline{M}_j = \underline{r}_j \times -\underline{f}$

where

M = moment acting on respective part
$\underline{r}$ = position vector of marker in local part reference frame
i,j = parts being connected by compliant element

In a similar manner equations for other compliant elements can be developed. Some of the standard compliant elements that are available in ADAMS are:

1. Translational spring-damper element (three directional force)
2. Rotational spring damper element (one torque)
3. Bushing element (three forces and three torques)
4. Action only forces
5. Bistop (impact)

The characteristics of elements can be specified as linear or can be invoked from an extensive library of standard non-linear functions. These functions can be combined using arithmetic operators to conveniently formulate more specialized affects.

In general equations representing compliant elements can be written as

$$\underline{F}(\underline{q}, \underline{\dot{q}}, \underline{f}, t) = 0 \tag{2.6}$$

where

$\underline{f}$ = vector of force in compliant elements
$\underline{F}$ = vector of equations defining the compliant forces

2.3 Elastic Elements

Elastic elements are a further generalization of compliant elements. While with the compliant element, force in the element is defined to be along the line defined by markers between which the element is connected, this is not generally required for an elastic element. An example of an elastic element is a beam element. The forces applied on the two parts connected are functions of the beam stiffness and damping matrices and the relative displacement and velocity of the two parts. The standard stiffness and damping matrices are the 6x6 matrices as for a beam element with clamped-clamped boundary conditions. Non-standard stiffness and damping matrices can be specified. Under this category, multi-dimensional elements such as tires can also be considered. The equations for representing elastic elements are also given by Equation (2.6).

2.4 Equations of Motion

In ADAMS the equations of motion for parts in the system are written as second order Lagrange's equations of motion in the constrained form [12].

$$d/dt\{\partial T/\partial \underline{\dot{q}}\} - \partial T/\partial \underline{q} - [\partial \underline{\phi}/\partial \underline{q}]^T \underline{\lambda} = \underline{f} \tag{2.7}$$

where

- T = System kinetic energy
- $\underline{\lambda}$ = Vector of Lagrange multipliers corresponding to the equations of constraint
- $\underline{f}$ = Vector of conservative and non-conservative "generalized" forces

3. ANALYSIS PROCEDURES

Mechanical systems can be modelled in ADAMS using the various entities described in the preceding section. These models can then be analyzed in any one of the following modes:

1. Static
2. Quasi-static
3. Kinematic
4. Transient dynamic
5. Modal

The first three modes of analysis are described only briefly. In the static mode, starting from an initial estimate of position, ADAMS computes the position of static equilbrium. The quasi-static mode allows the system to be stepped through time while computing static equilibrium at output time steps. The system velocities and accelerations are ignored in this analysis. The kinematic mode, works from only the constraint conditions to determine position and orientation of all parts in the mechanical system. The velocity and acceleration of all parts, if requested, can also be computed. Forces in compliant and elastic elements and joint reaction forces can also be obtained.

3.1 Transient Non-linear dynamics

In the transient dynamic mode the mechanical system is presumed to be multifreedom and its transient performance is to be determined by numerical integration of the governing system equations (2.3), (2.6), and (2.7). In general the governing equations can be written as a mixed system of second order differential and algebraic equations as:

$$\underline{H}(\ddot{\underline{q}},\dot{\underline{q}}, \underline{q},\underline{\lambda},\underline{f},t) = 0 \qquad (3.1)$$

In order to utilize a standard numerical integrator, the second-order differential equations have to be reduced to the first-order form by introducing velocities as solution variables. In the first order form the governing equation is given as

$$\underline{g}(\dot{\underline{y}},\underline{y},\underline{f},t) = 0 \qquad (3.2)$$

where

$$\underline{y} = \begin{bmatrix} \underline{q} \\ \dot{\underline{q}} \\ \underline{\lambda} \end{bmatrix}$$

There are two integrators available in ADAMS at present.

1. Non-Stiff Integrator (Adams-Moulton)
2. Gear's multi-step Stiff Integrator

The non-stiff integrator is only used for systems considered to be "non-stiff" [13]. However, since most mechanical systems are considered to be 'stiff' [13], (i.e. have widely separated eigenvalues) the Gear multi-step stiff integrator is generally applicable.

The Gear stiff integrator formula is a predictor-corrector formula. The prediction for the system state at a point ahead in time is made by an explicit predictor formula not presented here. The corrector for the system is given by the following implicit formula (orders of the dependent variable y (n+1) occur in different terms).

$$\underline{y}^{n+1} = -h\beta_0\dot{\underline{y}}^{n+1} + \sum_{j=1}^{k} (\alpha_j \underline{y}^{n-j+1} + h\beta_j\dot{\underline{y}}^{n-j+1}) \qquad (3.3)$$

where

h = Integration step size
α,B = Gear integration constants

As can be seen, this formula is of the implicit type. Repeated application of this formula about a fixed point in time can reduce the integration error further. This, however, is not a numerically stable procedure. A numerical stable procedure is to solve the non-linear governing equations by employing the Newton-Raphson interative procedure. This procedure requires the initial corrected state of the system to be computed by substituting predicted values on the right hand side of equation (3.3). Sucessive corrections to state vector can then be made by the following Newton-Raphson equation.

$$\{\partial \underline{g}/\partial \underline{y} + (-1/h\beta_0)\partial \underline{g}/\partial \dot{\underline{y}}\}\ \Delta \underline{y} = -\underline{g} \tag{3.4}$$

In a compact form

$$\underline{J}\Delta \underline{y} = -\underline{g} \tag{3.5}$$

where the Jacobian matrix

$J = \{\partial \underline{g}/\partial \underline{y} + (-1/h\beta_0)\partial \underline{g}/\partial \dot{\underline{y}}\}$

$\Delta \underline{y}$ = Correction in $\underline{y}$

The numerical integration procedure starts by computing y from equation (3.3) on the basis of the history of y and $\dot{y}$ over the preceding k time steps. The residual of the governing equations, obtained after evaluation, using predicted values of y is reduced by repeated application of the Newton-Raphson formula of equation (3.4). The iterative procedure is stopped when the convergence criterion is satisfied. An important observation to be made about the Jacobian is, that while the governing equation for the system may consist of a large number of equations, the Jacobian matrix is extremely sparse (less than ten percent non-zero entries). This permits use of sparse matrix algorithms for the rapid repetitive solution of Equation (3.4).

3.2 Linearized Analysis of Mechanical Systems

Recent developments in the ADAMS software now allows determination of natural frequency and mode shapes for linear circumstances of systems which are normally non-linear. The governing equations of the mechanical system, equation (3.2), can now be linearized about an operating point

$$\underline{y}^* = (\dot{\underline{y}}_0, \underline{y}_0, \underline{f}_0, t_0) \tag{3.6}$$

to give

$$\delta \underline{g} = \underline{A}|_{\underline{y}*}\delta \underline{y} - \underline{B}|_{\underline{y}*}\delta \dot{\underline{y}} + \partial \underline{g}/\partial f|_{\underline{y}*}\delta f + \partial \underline{g}/\partial t|_{\underline{y}*}\delta t = 0$$

where $\underline{A} = \partial \underline{g}/\partial \underline{y}$ and $\underline{B} = \partial \underline{g}/\partial \dot{\underline{y}}$

If we assume that the mechanical system represented by equation (3.2) is in a state of equilibrium (or other state such that matrices A and B are time invarient) then

$$\partial \underline{g}/\partial t = 0$$

Furthermore since the modal characteristics are independent of system applied forces,

$$\delta \underline{f} = 0 \tag{3.7}$$

In this case we may express

$$\delta \underline{y} = e^{\sigma t}\underline{z} \tag{3.8}$$

Equation (3.8) may be differentiated with respect to time to give

$$\delta\dot{\underline{y}} = \sigma e^{\sigma t}\underline{z} \qquad (3.9)$$

The resulting eigenvalue problem is

$$\underline{A}\ \underline{z} = \sigma\underline{B}\ \underline{z} \qquad (3.10)$$

To construct the eigenvalue problem of equation (3.10) requires that matrices A and B be constructed. From equation (3.4) it can be seen that these are the very same matrices constructed for the corrector formula of the integration procedure. Therefore, at a given operating point, the Jacobian matrix computed in ADAMS is sufficient information to construct the eigenvalue problem. However, the presence of algebraic equations in the governing equations causes the eigenvalue problem to take on non-standard characteristics. Matrix B is inherently singular due to the absence of any derivatives of Lagrange multipliers in the governing equations. The large eigenvalue problem of equation (3.10) is not well posed.

It is possible to reduce this large ill-conditioned problem to a well-conditioned standard problem. The procedure involves recognizing that the form of the algebraic equations allows us to represent a set of variables as being a linear function of another independent set of variables. This fact can be used to reduce this large eigenvalue problem to one that has a size of 2 x the number of degrees of freedom. That is the smallest size to which a first order problem can be reduced. This procedure is embodied in the ADAMS/MODAL linear analysis software. The details of this procedure are described in reference [11].

4. EXAMPLES

Two examples are now described illustrating the application of ADAMS to aerospace mechanical system problems.

4.1 Example 1 - Boom Docking

The first example is that of an ADAMS simulation of a boom docking maneuver to couple two satellites. As shown in Figure 3, the target and chaser satellite are maneuvered to within one meter of one another. The target vehicle is equipped with a funnel that has a latching mechanism at its base. The chaser vehicle has a telescoping boom that can be extended or retracted as desired. The object is to extend the boom so that its tip reaches into the base of the funnel on the target satellite. Once this is accomplished the latching mechanism at the base of the funnel is tripped and latches onto the tip of the boom. The chaser satellite then begins to retract the boom, thus pulling the two vehicles together. In the ADAMS model the boom is represented by a number of parts that slide with respect to one another. The entire boom is elastically connected to the chaser satellite.

Figures 3 to Figure 7 show a sequence of snapshots of the docking maneuver. In the first snapshot, Figure 3 the two vehicles are separated by about one meter. The chaser satellite now begins to extend the boom as shown in Figure 4. The tip of the boom makes contact with the funnel, Figure 5, and is guided towards the base of the funnel. Impact and surface geometry of the tip and the funnel is modelled by user supplied subroutines in ADAMS. The next snapshot, Figure 6, shows the tip of the boom extended beyond the base of the funnel. The latching mechanism at the base of the funnel attaches onto the tip of the boom. The chaser satellite now begins to retract the boom, causing the two vehicles to move closer. The final snapshot, Figure 7, shows the two satellites coupled together.

Output can be requested from ADAMS in a tabular or graphical form. The output could consist of displacements, velocities, and accelerations at any point on any of the parts. The forces acting in various elements of the model can be obtained. The forces acting on the tip of the boom when it comes in contact with the funnel can be obtained.

Since all the parameters necessary to perform this simulation were not available, parametric studies had to be performed to obtain acceptable values for certain parameters in order to produce the desired docking maneuver. Initially it was found that the velocity with which the chaser satellite approached the target was too high. This caused the target to spin away when the funnel was impacted by the boom. The parameters related to the latching mechanism at the base of the funnel had to be adjusted to achieve a rapid latching of the boom. Initially the latching mechanism was not quick enough, thus the boom tip that ran into the latching mechanism was retracted by the chaser before the mechanism latched.

4.2 Example 2 - Satellite Docking Using Clamp Mechanisms

A second example is a satellite docking maneuver using a clamp mechanism. In this simulation it is assumed that the chaser satellite can be steered to within fifteen centimeters in front of the target. As shown in Figure 8 the chaser has four locking handles. Correspondingly the target vehicle has four claws with spring loaded levers. When the handles come to within seventy millimeters of the base of the claws the locking levers on the claws are triggered to cause the handles to be pulled into the claws.

In the ADAMS model the levers are connected to the claws by means of revolute joints. The clamping action of the lever is caused by a torsional spring of linear characteristics. The claws are themselves mounted on the target by an elastic connection.

Figure 8-12 shows a sequence of snapshots of the two satellites during simulated docking. In the first snapshot, Figure 8, the two vehicles are at some distance from one another and have some angular misalignment. As can be seen all the claws on the target are open. As the two vehicles approach, one of the handles on the chaser vehicle gets close enough to the claw to

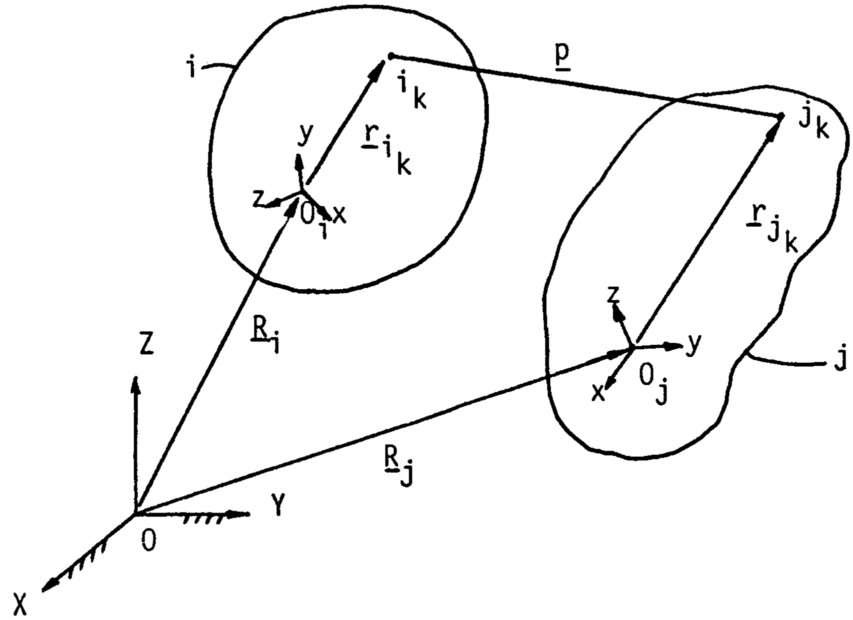

Figure 1. - Two parts connected by kinematic joint.

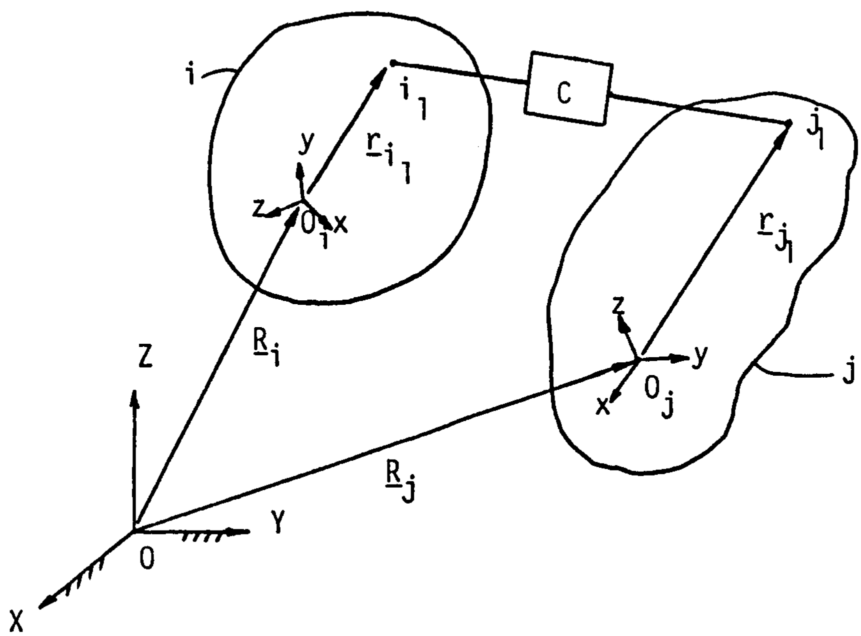

Figure 2. - Two parts connected by compliant element.

ORIGINAL PAGE IS
OF POOR QUALITY

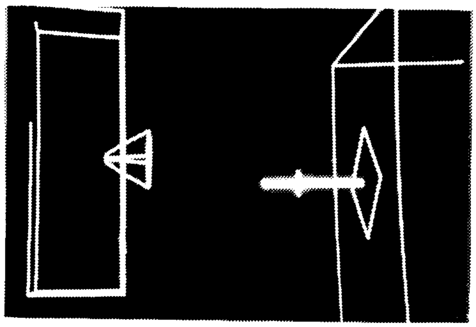

Figure 3

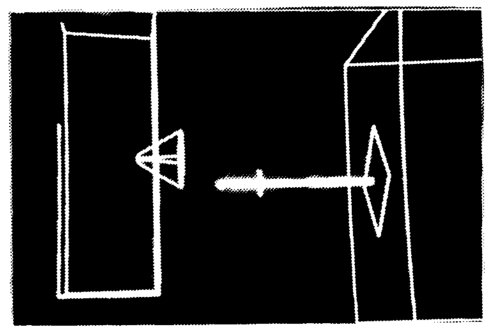

Figure 4

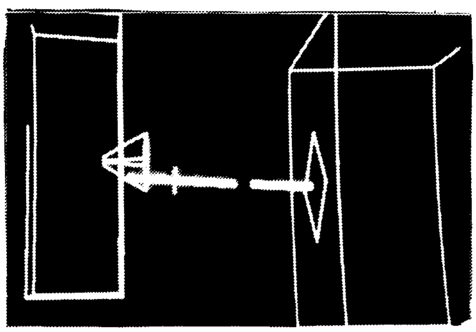

Figure 5

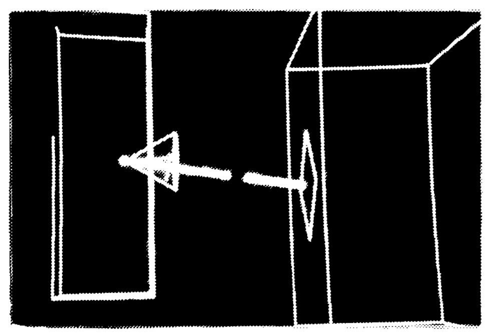

Figure 6

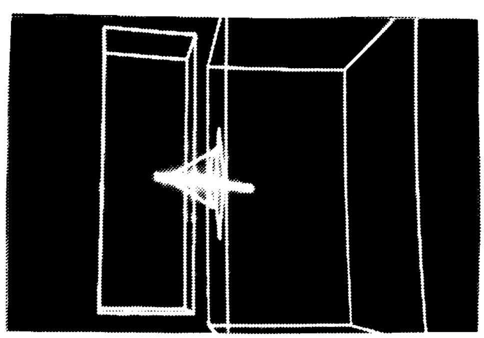

Figure 7

Figures 3 thru 7. - Snapshots of boom docking maneuver.

ORIGINAL PAGE IS
OF POOR QUALITY

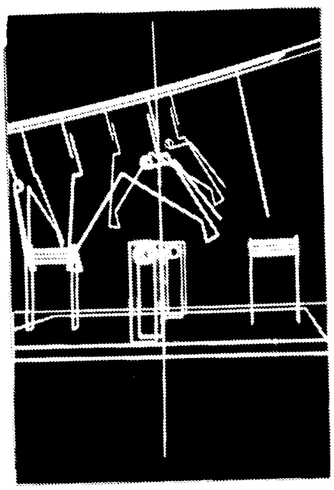
Figure 8

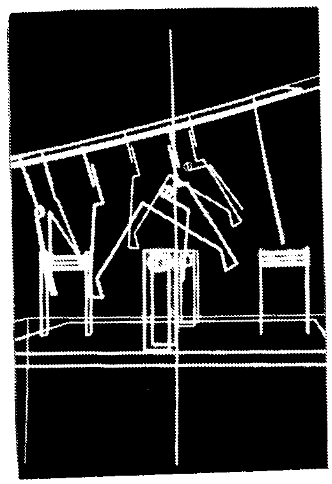
Figure 9

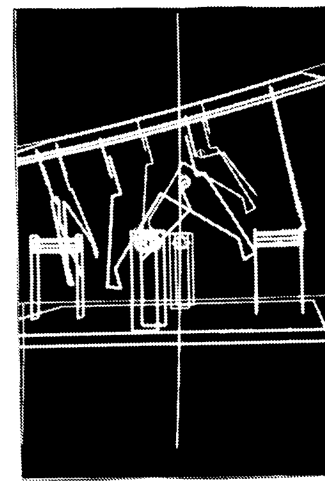
Figure 10

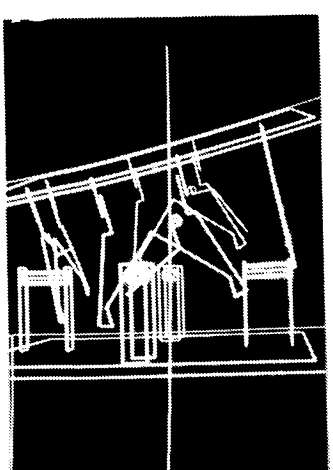
Figure 11

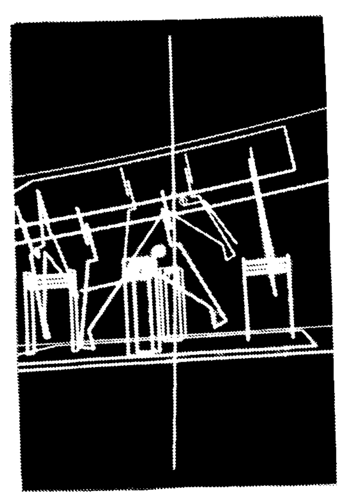
Figure 12

Figures 8 thru 12. - Snapshots of satellite docking with clamp mechanism.